

系统管理指南：命名和目录服务（**DNS**、**NIS**和**LDAP**）

版权所有 © 2002, 2010, Oracle 和/或其附属公司。保留所有权利。

本软件和相关文档是根据许可证协议提供的，该许可证协议中规定了关于使用和公开本软件和相关文档的各种限制，并受知识产权法的保护。除非在许可证协议中明确许可或适用法律明确授权，否则不得以任何形式、任何方式使用、拷贝、复制、翻译、广播、修改、授权、传播、分发、展示、执行、发布或显示本软件和相关文档的任何部分。除非法律要求实现互操作，否则严禁对本软件进行逆向工程设计、反汇编或反编译。

此文档所含信息可能随时被修改，恕不另行通知，我们不保证该信息没有错误。如果贵方发现任何问题，请书面通知我们。

如果将本软件或相关文档交付给美国政府，或者交付给以美国政府名义获得许可证的任何机构，必须符合以下规定：

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

本软件或硬件是为了在各种信息管理应用领域内的一般使用而开发的。它不应被应用于任何存在危险或潜在危险的应用领域，也不是为此而开发的，其中包括可能会产生人身伤害的应用领域。如果在危险应用领域内使用本软件或硬件，贵方应负责采取所有适当的防范措施，包括备份、冗余和其它确保安全使用本软件或硬件的措施。对于因在危险应用领域内使用本软件或硬件所造成的一切损失或损害，Oracle Corporation 及其附属公司概不负责。

Oracle 和 Java 是 Oracle 和/或其附属公司的注册商标。其他名称可能是各自所有者的商标。

Intel 和 Intel Xeon 是 Intel Corporation 的商标或注册商标。所有 SPARC 商标均是 SPARC International, Inc 的商标或注册商标，并应按照许可证的规定使用。AMD、Opteron、AMD 徽标以及 AMD Opteron 徽标是 Advanced Micro Devices 的商标或注册商标。UNIX 是 The Open Group 的注册商标。

本软件或硬件以及文档可能提供了访问第三方内容、产品和服务的方式或有关这些内容、产品和服务的信息。对于第三方内容、产品和服务，Oracle Corporation 及其附属公司明确表示不承担任何种类的担保，亦不对其承担任何责任。对于因访问或使用第三方内容、产品和服务所造成的任何损失、成本或损害，Oracle Corporation 及其附属公司概不负责。

目录

前言	15
第 1 部分 关于命名和目录服务	19
1 命名和目录服务（概述）	21
什么是命名服务？	21
Solaris 命名服务	26
DNS 命名服务的说明	27
/etc 文件命名服务的说明	27
NIS 命名服务的说明	27
NIS+ 命名服务的说明	27
LDAP 命名服务的说明	28
命名服务：简要比较	28
2 名称服务转换器（概述）	31
关于名称服务转换器	31
nsswitch.conf 文件的格式	32
nsswitch.conf 文件中的注释	35
密钥服务器和转换器文件中的 publickey 项	35
nsswitch.conf 模板文件	36
缺省的转换器模板文件	36
nsswitch.conf 文件	39
选择其他配置文件	40
▼ 如何修改名称服务转换器	40
DNS 和 Internet 访问	41
IPv6 和 Solaris 命名服务	41
确保与 +/- 语法兼容	42

转换器文件和口令信息	42
第 2 部分 DNS 设置和管理	45
3 DNS 设置和管理（参考）	47
相关材料	47
从 BIND 8 迁移到 BIND 9	48
DNS 和服务管理工具	49
实现 rndc	50
rndc.conf 配置文件	50
控制通道之间的差别	50
BIND 9 rndc 的命令	51
BIND 9 命令、文件、工具和选项	52
BIND 9 工具和配置文件	52
BIND 8 和 BIND 9 命令和文件比较	52
命令和选项更改的说明	53
named.conf 选项	54
BIND 9 中的语句	56
named.conf 选项摘要	58
第 3 部分 NIS 设置和管理	65
4 网络信息服务 (Network Information Service, NIS)（概述）	67
NIS 介绍	67
NIS 体系结构	68
NIS 计算机类型	68
NIS 服务器	69
NIS 客户机	69
NIS 元素	69
NIS 域	69
NIS 守护进程	70
NIS 实用程序	70
NIS 映射	71
与 NIS 相关的命令	74

NIS 绑定	75
服务器列表模式	76
广播模式	76
5 设置和配置 NIS 服务	79
配置 NIS 任务列表	79
配置 NIS 之前的准备工作	80
NIS 和服务管理工具	80
规划 NIS 域	81
标识 NIS 服务器和客户机	81
准备主服务器	82
源文件目录	82
Passwd 文件和名称空间安全	82
为向 NIS 映射的转换准备源文件	83
准备 Makefile	84
使用 ypinit 设置主服务器	85
支持多个 NIS 域的主服务器	86
在主服务器上启动或停止 NIS 服务	87
自动启动 NIS 服务	87
从命令行启动和停止 NIS	87
设置 NIS 从属服务器	88
准备从属服务器	88
设置从属服务器	88
设置 NIS 客户机	89
6 管理 NIS (任务)	91
口令文件和名称空间安全	91
管理 NIS 用户	92
▼ 如何向 NIS 域添加新 NIS 用户	92
设置用户口令	93
NIS 网络组	94
使用 NIS 映射	95
获取映射信息	95
更改映射的主服务器	96
修改配置文件	97

修改和使用 Makefile	98
修改 Makefile 项	99
更新和修改现有映射	100
▼ 如何更新随缺省集合提供的映射	101
维护已更新映射	101
修改非缺省映射	103
使用 makedbm 修改非缺省映射	104
从文本文件创建新映射	104
向基于文件的映射中添加项	104
通过标准输入创建映射	104
修改通过标准输入创建的映射	104
添加从属服务器	105
▼ 如何添加从属服务器	105
使用启用 C2 安全性的 NIS	106
绑定到特定 NIS 服务器	107
更改计算机的 NIS 域	107
▼ 如何更改计算机的 NIS 域名	107
将 NIS 与 DNS 结合使用	108
▼ 如何配置通过 NIS 和 DNS 执行计算机名和地址查找	108
处理混合的 NIS 域	108
禁用 NIS 服务	109
 7 NIS 疑难解答	111
NIS 绑定问题	111
症状	111
影响一台客户机的 NIS 问题	112
影响多台客户机的 NIS 问题	115
 第 4 部分 LDAP 命名服务的设置和管理	119
 8 LDAP 命名服务介绍（概述/参考）	121
目标用户	121
建议的背景读物	121
其他先决条件	122

LDAP 命名服务与其他命名服务的比较	122
LDAP 命名服务的优点	123
LDAP 命名服务的限制	123
设置 LDAP 命名服务（任务列表）	123
9 LDAP 基本组件和概念（概述）	125
LDAP 数据交换格式 (LDAP Data Interchange Format, LDIF)	125
随 LDAP 使用全限定域名	128
缺省目录信息树 (Directory Information Tree, DIT)	129
缺省的 LDAP 架构	129
服务搜索描述符 (Service Search Descriptor, SSD) 和架构映射	130
SSD 说明	130
LDAP 客户机配置文件	132
客户机的配置文件属性	132
本地客户机属性	133
ldap_cachemgr 守护进程	134
LDAP 命名服务安全模型	135
介绍	135
传输层安全性 (Transport Layer Security, TLS)	136
指定客户机凭证级别	137
选择验证方法	139
可插拔验证方法	142
帐户管理	146
10 LDAP 命名服务的规划要求（任务）	149
LDAP 规划概述	149
规划 LDAP 网络模型	149
规划目录信息树 (Directory Information Tree, DIT)	150
多台目录服务器	150
与其他应用程序共享数据	151
选择目录后缀	151
LDAP 和副本服务器	151
规划 LDAP 安全模型	152
规划 LDAP 的客户机配置文件和缺省属性值	154
规划 LDAP 数据置备	154

▼ 如何使用 ldapaddent 向服务器置备 host 项	155
11 为使用 LDAP 客户机设置 Sun Java System Directory Server (任务)	157
使用 idsconfig 配置 Sun Java System Directory Server	158
基于服务器安装创建核对表	158
架构定义	159
使用浏览索引	160
使用服务搜索描述符修改客户机对各个服务的访问	160
使用 idsconfig 设置 SSD	160
运行 idsconfig	161
▼ 如何使用 idsconfig 配置 Sun Java System Directory Server	162
idsconfig 设置示例	162
使用 ldapaddent 置备目录服务器	166
▼ 如何使用 ldapaddent 向 Sun Java System Directory Server 置备用户口令数据	166
管理打印机项	167
添加打印机	167
使用 lpget	167
向目录服务器置备其他配置文件	167
▼ 如何使用 ldapclient 向目录服务器置备其他配置文件	168
配置目录服务器以启用帐户管理	168
对于使用 pam_ldap 的客户机	168
对于使用 pam_unix 的客户机	170
迁移 Sun Java System Directory Server	172
12 设置 LDAP 客户机 (任务)	173
LDAP 客户机设置的先决条件	173
LDAP 和服务管理工具	174
初始化 LDAP 客户机	175
使用配置文件初始化客户机	175
使用每用户凭证	176
使用代理凭证	178
在 LDAP 中启用 shadow 更新	178
手动初始化客户机	179
修改手动客户机配置	180
取消客户机初始化	180

设置 TLS 安全性	181
配置 PAM	182
检索 LDAP 命名服务信息	183
列出所有 LDAP 容器	184
列出所有用户项属性	184
定制 LDAP 客户机环境	185
为 LDAP 修改 nsswitch.conf 文件	185
为 LDAP 启用 DNS	185
13 LDAP 疑难解答 (参考)	187
监视 LDAP 客户机状态	187
验证 ldap_cachemgr 是否正在运行	187
检查当前的配置文件信息	188
验证基本的客户机/服务器通信	189
从非客户机检查服务器数据	189
LDAP 配置问题及解决方案	189
无法解析主机名	189
无法远程访问 LDAP 域中的系统	190
登录功能不起作用	190
查找速度太慢	191
ldapclient 无法绑定到服务器	191
使用 ldap_cachemgr 进行调试	191
ldapclient 在设置过程中挂起	191
14 LDAP 一般参考 (参考)	193
空白核对表	193
LDAP 升级信息	194
兼容性	194
运行 ldap_cachemgr 守护进程	195
新的 automount 架构	195
pam_ldap 更改	195
LDAP 命令	196
常规 LDAP 工具	196
需要 LDAP 命名服务的 LDAP 工具	196
pam_ldap 的示例 pam.conf 文件	197

为帐户管理配置的 pam_ldap 的示例 pam_conf 文件	198
LDAP 的 IETF 架构	200
RFC 2307 网络信息服务架构	200
邮件别名架构	205
目录用户代理配置文件 (DUAProfile) 架构	206
Solaris 架构	207
Solaris 项目架构	208
基于角色的访问控制和执行配置文件架构	208
LDAP 的 Internet 打印协议信息	210
Internet 打印协议 (Internet Printing Protocol, IPP) 属性	210
Internet 打印协议 (Internet Printing Protocol, IPP) ObjectClasses	216
Sun 打印机属性	217
Sun 打印机 ObjectClasses	217
LDAP 的常规目录服务器要求	218
LDAP 命名服务使用的缺省过滤器	218
15 从 NIS 转换为 LDAP (概述/任务)	223
NIS 到 LDAP 转换服务概述	223
NIS 到 LDAP 转换工具和服务管理工具	224
NIS 到 LDAP 转换的目标用户	224
不应使用 NIS 到 LDAP 转换服务的情况	224
NIS 到 LDAP 转换服务对用户造成的影响	225
NIS 到 LDAP 转换术语	225
NIS 到 LDAP 转换的命令、文件和映射	226
支持的标准映射	227
从 NIS 转换为 LDAP (任务列表)	228
NIS 到 LDAP 转换的先决条件	228
设置 NIS 到 LDAP 转换服务	229
▼ 如何使用标准映射设置 N2L 服务	230
▼ 如何使用定制映射或非标准映射设置 N2L 服务	232
定制映射的示例	234
使用 Sun Java System Directory Server 进行 NIS 到 LDAP 转换的最佳做法	235
使用 Sun Java System Directory Server 创建虚拟列表视图索引	236
避免 Sun Java System Directory Server 服务器超时	237
避免 Sun Java System Directory Server 缓冲区溢出	237

NIS 到 LDAP 转换的限制	238
NIS 到 LDAP 转换的疑难解答	238
常见的 LDAP 错误消息	238
NIS 到 LDAP 转换的问题	239
恢复为 NIS	242
▼ 如何基于旧的源文件恢复到 NIS 映射	242
▼ 如何基于当前的 DIT 内容恢复为 NIS 映射	243
16 从 NIS+ 转换为 LDAP	245
NIS+ 到 LDAP 的转换概述	245
rpc.nisd 配置文件	246
NIS+ 到 LDAP 转换工具和服务管理工具	247
创建属性和对象类	249
NIS+ 到 LDAP 转换入门	249
/etc/default/rpc.nisd 文件	249
/var/nis/NIS+LDAPmapping 文件	252
NIS+ 到 LDAP 迁移方案	256
合并 NIS+ 数据和 LDAP 数据	258
主服务器和副本服务器（从 NIS+ 转换为 LDAP）	260
复制时间标记	260
目录服务器（从 NIS+ 转换为 LDAP）	261
配置 Sun Java System Directory Server	261
指定服务器地址和端口号	261
安全性和验证	261
性能和索引	263
映射表项以外的 NIS+ 对象	264
NIS+ 项的所有者、组、访问权限和 TTL	265
▼ 如何将其他项属性存储到 LDAP 中	266
主体名和网络名（从 NIS+ 转换为 LDAP）	268
client_info 和 timezone 表（从 NIS+ 转换为 LDAP）	270
client_info 属性和对象类	270
timezone 属性和对象类	271
添加新的对象映射（从 NIS+ 转换为 LDAP）	272
▼ 如何映射非项对象	272
添加项对象	274

将配置信息存储到 LDAP 中	277
A Solaris 10 软件中对 DNS、NIS 和 LDAP 的更新	281
服务管理工具的更改	281
DNS BIND	281
pam_ldap 更改	282
文档错误	282
 词汇表	 283
 索引	 289

示例

示例 2-1	NIS+ 转换器文件模板: nsswitch.nisplus	36
示例 2-2	NIS 转换器文件模板	37
示例 2-3	文件转换器文件模板	38
示例 2-4	LDAP 转换器文件模板	38
示例 3-1	rndc.conf 文件样例	50
示例 3-2	用于 rndc 的 named.conf 文件项样例	50
示例 6-1	ypxfr_1perday Shell 脚本	102
示例 11-1	为 Example, Inc. 网络运行 idsconfig	162

前言

《Solaris 管理指南：命名和目录服务（DNS、NIS 和 LDAP）》介绍了设置和管理 Solaris 操作系统 (Solaris OS) 的命名和目录服务：DNS、NIS 和 LDAP。本指南是当前 Solaris 发行版的系统和网络管理集合的一部分。

注 - 此 Solaris 发行版支持使用 SPARC 和 x86 系列处理器体系结构的系统。支持的系统可以在 [Solaris OS: Hardware Compatibility Lists \(http://www.sun.com/bigadmin/hcl\)](http://www.sun.com/bigadmin/hcl) (Oracle Solaris OS：硬件兼容性列表) 中找到。本文档列举了在不同类型的平台上进行实现时的所有差别。

在本文档中，这些与 x86 相关的术语表示以下含义：

- "x86" 泛指 64 位和 32 位的 x86 兼容产品系列。
- "x64" 专指 64 位 x86 兼容 CPU。
- “32 位 x86”指出了有关基于 x86 的系统的特定 32 位信息。

若想了解本发行版支持哪些系统，请参见 Solaris OS: Hardware Compatibility Lists。

目标读者

本指南是为经验丰富的系统管理员和网络管理员编写的。

本书介绍了与 Solaris 命名和目录服务相关的网络概念，但并未介绍网络基础知识，也未介绍 Solaris OS 中的管理工具。

本书的结构

本指南按照不同的命名服务分成四部分。

第 1 部分

第 2 部分

第 3 部分

第 4 部分

系统管理指南系列书籍的结构

下面是系统管理指南系列书籍中各本书包含的主题列表。

书名	主题
《系统管理指南：基本管理》	用户帐户和组、服务器和客户机支持、关闭和引导系统，以及管理服务
《系统管理指南：高级管理》	终端和调制解调器、系统资源（磁盘配额、记帐和 crontab）、系统进程以及 Oracle Solaris 软件问题故障排除
《系统管理指南：设备和文件系统》	可移除介质、磁盘和设备、文件系统以及备份和还原数据
《系统管理指南：IP 服务》	TCP/IP 网络管理、IPv4 和 IPv6 地址管理、DHCP（动态主机配置协议）、Ipsec（Internet 协议安全）、IKE（Internet 密钥交换）、Solaris IP 过滤器、移动 IP、IP 网络多路径 (IP network multipathing, IPMP) 以及 IPQoS
《系统管理指南：命名和目录服务（DNS、NIS 和 LDAP）》	DNS、NIS 和 LDAP 命名和目录服务，包括从 NIS 转换到 LDAP 以及从 NIS+ 转换到 LDAP
《System Administration Guide: Naming and Directory Services (NIS+)》	NIS+ 命名和目录服务
《系统管理指南：网络服务》	Web 高速缓存服务器、与时间相关的服务、网络文件系统（NFS 和 Autofs）、邮件、SLP（服务定位协议）和 PPP（点对点协议）
《系统管理指南：打印》	打印主题和任务，使用服务、工具、协议和技术来设置及管理打印服务和打印机
《系统管理指南：安全性服务》	审计、设备管理、文件安全、BART（基本审计和报告工具）、Kerberos 服务、PAM（可插拔验证模块）、Solaris 加密框架、特权、RBAC（基于角色的存取控制）、SASL（简单身份认证和安全层）和 Solaris 安全 Shell
《系统管理指南：Oracle Solaris Containers—资源管理和 Oracle Solaris Zones》	资源管理主题项目和任务、扩展记帐、资源控制、公平份额调度器 (fair share scheduler, FSS)、使用资源上限设置守护进程 (rcapd) 的物理内存控制，以及资源池；使用 Solaris Zones 软件分区技术和 lx 标记区域的虚拟功能
《Oracle Solaris ZFS 管理指南》	ZFS（Zettabyte 文件系统）存储池以及文件系统的创建和管理、快照、克隆、备份、使用访问控制列表 (Access Control List, ACL) 保护 ZFS 文件、在安装区域的 Solaris 系统中使用 ZFS、仿真卷以及故障排除和数据恢复
《Oracle Solaris Trusted Extensions 管理员规程》	特定于 Oracle Solaris Trusted Extensions 功能的系统管理
《Oracle Solaris Trusted Extensions 配置指南》	从 Solaris 10 5/08 发行版开始，介绍如何规划、启用及初始配置 Oracle Solaris Trusted Extensions 功能

相关书籍

- 随 Sun Java Enterprise System 文档提供的 Sun Java System Directory Server 部署指南
- 随 Sun Java Enterprise System 文档提供的 Sun Java System Directory Server 管理指南
- 《DNS and Bind》，由 Cricket Liu 和 Paul Albitz 编著，（第四版，O'Reilly 出版社，2001 年）
- 《Understanding and Deploying LDAP Directory Services》，由 Timothy A. Howes 博士和 Mark C. Smith 编著

文档、支持和培训

有关其他资源，请参见以下 Web 站点：

- 文档(<http://www.oracle.com/technetwork/indexes/documentation/index.html>)
- 支持(<http://www.oracle.com/us/support/systems/index.html>)
- 培训(<http://education.oracle.com>)—单击左侧导航栏中的 Sun 链接。

Oracle 欢迎您提出意见

Oracle 欢迎您针对其文档质量和实用性提出意见和建议。如果您发现任何错误或要提出其他改进建议，请转到 <http://www.oracle.com/technetwork/indexes/documentation/index.html>，然后单击 "Feedback"（反馈）。请提供文档的标题和文件号码，以及章节和页码（如果有）。如果您需要回复，请告知。

Oracle Technology Network (<http://www.oracle.com/technetwork/index.html>)（Oracle 技术网）提供一系列与 Oracle 软件相关的资源：

- 可在 Discussion Forums (<http://forums.oracle.com>)（讨论论坛）上讨论技术问题和解决方案。
- 通过 Oracle By Example (<http://www.oracle.com/technology/obe/start/index.html>)（Oracle 示例）获得逐步的实践教程。
- 下载 Sample Code (http://www.oracle.com/technology/sample_code/index.html)（示例代码）。

印刷约定

下表介绍了本书中的印刷约定。

表 P-1 印刷约定

字体或符号	含义	示例
AaBbCc123	命令、文件和目录的名称；计算机屏幕输出	编辑 <code>.login</code> 文件。 使用 <code>ls -a</code> 列出所有文件。 <code>machine_name% you have mail.</code>
AaBbCc123	用户键入的内容，与计算机屏幕输出的显示不同	<code>machine_name% su</code> <code>Password:</code>
<i>aabbcc123</i>	要使用实名或值替换的命令行占位符	删除文件的命令为 <code>rm filename</code> 。
<i>AaBbCc123</i>	保留未译的新词或术语以及要强调的词	这些称为 <i>Class</i> 选项。 注意： 有些强调的项目在联机时以粗体显示。
新词术语强调	新词或术语以及要强调的词	高速缓存 是存储在本地的副本。 请勿保存文件。
《书名》	书名	阅读《用户指南》的第 6 章。

命令中的 shell 提示符示例

下表显示了 Oracle Solaris OS 中包含的缺省 UNIX shell 系统提示符和超级用户提示符。请注意，在命令示例中显示的缺省系统提示符可能会有所不同，具体取决于 Oracle Solaris 发行版。

表 P-2 shell 提示符

shell	提示符
Bash shell、Korn shell 和 Bourne shell	\$
Bash shell、Korn shell 和 Bourne shell 超级用户	#
C shell	machine_name%
C shell 超级用户	machine_name#

第 1 部分

关于命名和目录服务

本部分介绍 Solaris OS 的命名和目录服务。此外还介绍了 `nsswitch.conf` 文件，该文件用于协调不同服务的使用。

命名和目录服务（概述）

本章概述了 Solaris 中使用的命名和目录服务。本章还简要介绍了 DNS、NIS 和 LDAP 命名服务。有关 NIS+ 的详细信息，请参见《[System Administration Guide: Naming and Directory Services \(NIS+\)](#)》。

什么是命名服务？

命名服务在一个集中位置存储信息，这样用户、计算机和应用程序便可通过网络进行通信。此信息包括：

- 计算机（主机）名和地址
- 用户名
- 口令
- 访问权限
- 组成员关系、打印机等

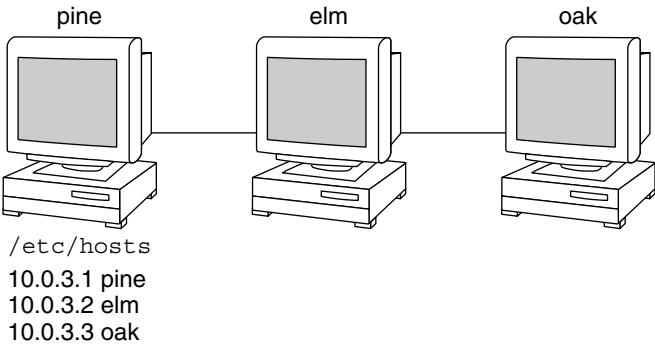
如果没有集中的命名服务，则每台计算机都必须单独保留一份此信息的副本。命名服务信息可以存储在文件、映射或数据库表中。如果集中存储所有数据，管理将变得更加轻松。

命名服务对任何计算网络都是至关重要的。除了其他特性以外，命名服务还提供能够执行以下操作的功能：

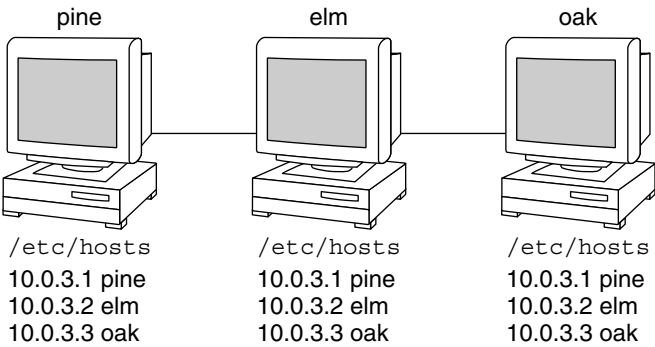
- 将名称与对象关联（**绑定**）
- 将名称解析为对象
- 删除绑定
- 列出名称
- 重命名

网络信息服务使计算机可由通用名称而非数字地址来标识。这样可以简化通信，因为用户不需要记住并尝试输入那些繁琐的地址（例如 192.168.0.0）。

例如，假设有一个网络具有三台计算机，名称分别为 pine、elm 和 oak。pine 必须先知道 elm 或 oak 的数字网络地址，才能向其发送消息。因此，pine 需要保留一个文件（/etc/hosts 或 /etc/inet/ipnodes），以存储网络中每台计算机（包括 pine 本身）的网络地址。



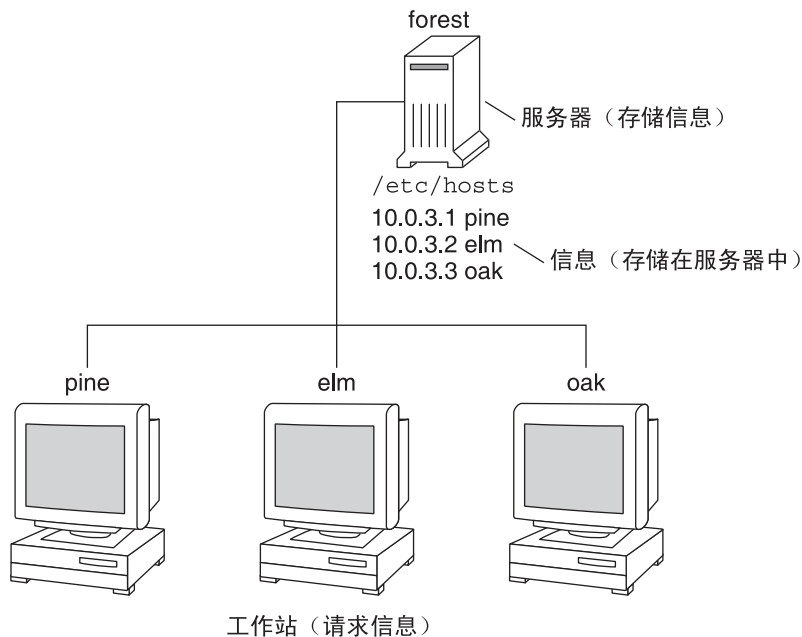
类似地，elm 和 oak 要与 pine 通信或彼此进行通信，也必须保留类似的文件。



除了存储地址外，计算机还存储安全信息、邮件数据、网络服务信息等。随着网络提供的服务越来越多，存储信息的列表会不断增大。因此，每台计算机都需要保留一整套类似于 /etc/hosts 或 /etc/inet/ipnodes 的文件。

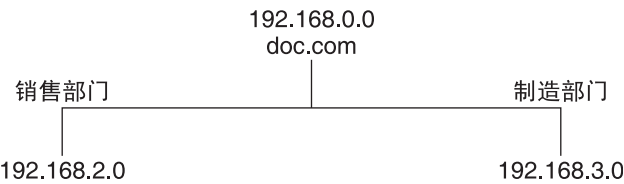
网络信息服务在服务器中存储网络信息，任何计算机都可以查询该信息。

这些计算机称为服务器的**客户机**。下图显示客户机/服务器布局。每次网络信息发生变化时，管理员将只更新网络信息服务存储的信息，而不更新每个客户机的本地文件。这样做可以减少错误、客户机之间的不一致性以及任务的绝对工作量。

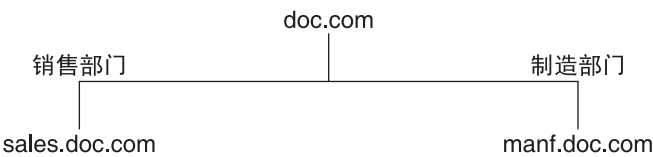


这种由服务器向网络中的客户机提供集中式服务的布局称为**客户机/服务器计算**。

尽管网络信息服务的主要用途是集中存储信息，但它也可以简化网络名称。例如，假设您的公司设置了一个与 Internet 连接的网络。Internet 为您的网络指定了网络号 192.168.0.0 和域名 doc.com。公司有两个部门：销售和制造 (Manf)，因此，其网络将划分为一个主网和两个子网（每个部门对应一个子网）。每个网络有自己的地址。



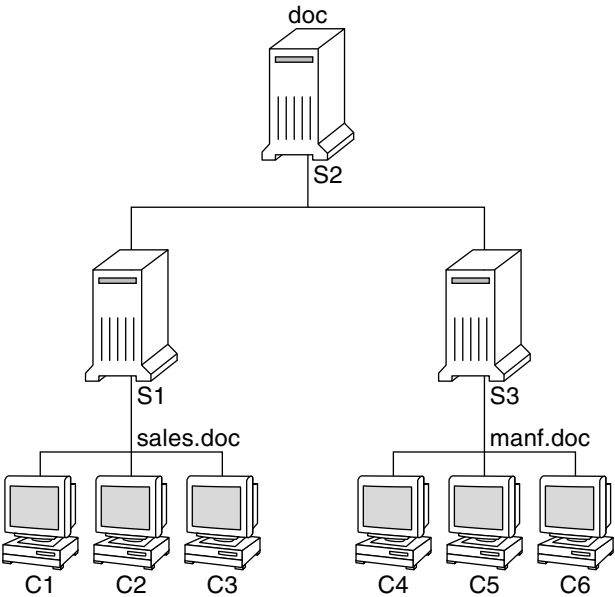
如上所述，每个部门可由网络地址来标识，但用命名服务提供的描述性名称来标识会更好。



这样，可以无需输入 198.168.0.0 物理地址作为邮件或其他网络通信的地址，而是输入 doc 作为地址。可以无需输入 192.168.2.0 或 192.168.3.0 物理地址作为邮件或其他网络通信的地址，而是输入 sales.doc 或 manf.doc 作为地址。

名称还比物理地址更灵活。因为物理网络通常不会改变，而公司组织可能会发生变动。

例如，假定有三台服务器（S1、S2 和 S3）支持 doc.com 网络。且其中两台服务器（S1 和 S3）支持客户机。

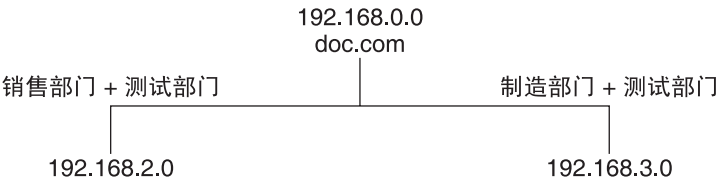


客户机 C1、C2 和 C3 将从服务器 S1 获取网络信息。客户机 C4、C5 和 C6 将从服务器 S3 获取信息。下表对生成的网络进行了汇总。该表是该网络的大致说明，与实际的网络信息映射并不相似。

表 1-1 docs.com 网络说明

网络地址	网络名称	服务器	客户机
192.168.1.0	doc	S1	
192.168.2.0	sales.doc	S2	C1、C2、C3
192.168.3.0	manf.doc	S3	C4、C5、C6

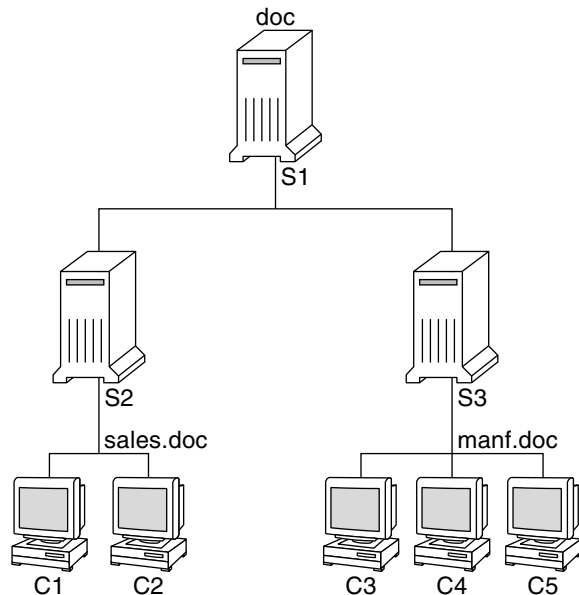
现在，假设创建了第三个部门（测试部门），该部门从其他两个部门借入一些资源，但并未创建第三个子网。物理网络将不再与公司结构类似。



测试部门的流量原本将不具有自己的子网，而是将在 192.168.2.0 与 192.168.3.0 之间拆分。但是，通过网络信息服务，测试部门的流量可以具有自己的专用网络。



因此，当组织更改时，其网络信息服务可以按此处所示更改其映射。



现在，客户机 C1 和 C2 将从服务器 S2 中获取信息。C3、C4 和 C5 将从服务器 S3 中获取信息。

通过更改网络信息结构，无需重新组织网络结构，即可适应组织结构的后续更改。

Solaris 命名服务

Solaris 平台可提供以下命名服务。

- DNS，域名系统（请参见第 27 页中的“DNS 命名服务的说明”）
- /etc 文件，原始的 UNIX 命名系统（请参见第 27 页中的“/etc 文件命名服务的说明”）
- NIS，网络信息服务（请参见第 27 页中的“NIS 命名服务的说明”）
- NIS+，网络信息服务扩充版本（请参见《System Administration Guide: Naming and Directory Services (NIS+)》）
- LDAP，轻量目录访问协议（请参见第 4 部分 LDAP Naming Services Setup and Administration）

大多数现代网络都组合使用上述两种或更多种服务。使用多种服务时，这些服务将由第 2 章，名称服务转换器（概述）中讨论的 `nsswitch.conf` 文件来协调。

DNS 命名服务的说明

DNS 是 Internet 为 TCP/IP 网络提供的命名服务。开发 DNS 后，网络中的计算机可由通用名称而非 Internet 地址来标识。DNS 可在本地管理域中的主机与跨域边界的主机之间执行命名。

使用 DNS 的联网计算机的集合称为 **DNS 名称空间**。DNS 名称空间可以划分为**域**的分层结构。DNS 域是一组计算机。每个域由两台或多台**名称服务器**支持，其中包括一台主服务器以及一台或多台辅助服务器。每台服务器都通过运行 `in.named` 守护进程来运行 DNS。在客户端，DNS 通过“解析程序”来实现。解析程序的功能是解析用户的查询。解析程序将查询名称服务器，然后名称服务器会返回请求的信息或对其他服务器的引用。

/etc 文件命名服务的说明

基于主机的原始 UNIX 命名系统是为独立的 UNIX 计算机开发的，后来修改为可用于网络。许多旧的 UNIX 操作系统和计算机仍在使用此系统，但是此系统并不适用于大型的复杂网络。

NIS 命名服务的说明

网络信息服务 (Network Information Service, NIS) 是独立于 DNS 开发的。DNS 通过使用计算机名代替数字 IP 地址来简化通信。NIS 的主要作用是通过对各种网络信息进行集中控制来更好地管理网络。NIS 存储有关网络、计算机名称和地址、用户、以及网络服务的信息。这种网络信息的集合被称为 **NIS 名称空间**。

NIS 名称空间信息存储在 NIS 映射中。NIS 映射旨在替换 UNIX `/etc` 文件以及其他配置文件。NIS 除了存储名称和地址外，还存储大量的其他信息。因此，NIS 名称空间存在大量映射。有关更多信息，请参见第 95 页中的“使用 NIS 映射”。

NIS 使用与 DNS 类似的客户机/服务器布局。复制的 NIS 服务器可向 NIS 客户机提供服务。主要的服务器称为主服务器，为确保其可靠，主服务器还具有备份，即从属服务器。主服务器和从属服务器都使用 NIS 检索软件，并且都存储 NIS 映射。有关 NIS 体系结构和 NIS 管理的更多信息，请参见第 5 章，**设置和配置 NIS 服务**和第 6 章，**管理 NIS（任务）**。

NIS+ 命名服务的说明

网络信息服务扩充版本 (Network Information Service Plus, NIS+) 与 NIS 相似，但具有更多功能。但是，NIS+ 不是 NIS 的扩展。

NIS+ 命名服务旨在符合组织的结构。与 NIS 不同，NIS+ 名称空间是动态的，因为可以进行更新并可随时由任何授权用户使更新生效。

通过 NIS+ 可将有关计算机地址的信息、安全信息、邮件信息、以太网接口和网络服务存储在一个集中位置。这种网络信息的配置称为 NIS+ **名称空间**。

NIS+ 名称空间是分层的。NIS+ 名称空间在结构上与 UNIX 目录文件系统相似。通过这种分层结构，NIS+ 名称空间可以配置为符合组织的逻辑分层结构。名称空间的信息布局与其**物理**布局无关。因此，一个 NIS+ 名称空间可以划分为多个可独立管理的域。如果客户机具有适当的权限，可以访问除了自身的域之外的域中的信息。

NIS+ 使用客户机/服务器模型来存储和访问 NIS+ 名称空间中包含的信息。每个域由一组服务器提供支持。主要的服务器称为**主服务器**。备份服务器称为**辅助服务器**。网络信息存储在内部 NIS+ 数据库的 16 个标准 NIS+ 表中。主服务器和辅助服务器都运行 NIS+ 服务器软件，并且都保留 NIS+ 表的副本。对主服务器中 NIS+ 数据所做的更改将自动以增量方式传播到辅助服务器。

NIS+ 包括一个复杂的安全系统，用来保护名称空间的结构及其信息。NIS+ 使用身份验证和授权来验证是否应满足客户机对信息的请求。**身份验证**确定信息请求者是否是网络中的有效用户。**授权**确定是否允许特定用户拥有或修改请求的信息。有关 NIS+ 安全的更详细说明，请参见《[System Administration Guide: Naming and Directory Services \(NIS+\)](#)》。

有关进行从 NIS+ 到 LDAP 的转换的信息，请参见第 16 章，[从 NIS+ 转换为 LDAP](#)。

LDAP 命名服务的说明

Solaris 操作系统与 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）以及其他 LDAP 目录服务器一起支持 LDAP（Lightweight Directory Access Protocol，轻量目录访问协议）。

有关 LDAP 命名服务的信息，请参见第 8 章，[LDAP 命名服务介绍（概述/参考）](#)。

有关从 NIS 转换到 LDAP 或从 NIS+ 转换到 LDAP 的信息，请参见第 15 章，[从 NIS 转换为 LDAP（概述/任务）](#)或第 16 章，[从 NIS+ 转换为 LDAP](#)。

有关单点登录以及设置和维护 Kerberos 验证服务的信息，请参见《[系统管理指南：安全性服务](#)》中的第 VI 部分，“[Kerberos 服务](#)”。

命名服务：简要比较

	DNS	NIS	NIS+	LDAP
名称空间	分层	不分层	分层	分层
数据存储	文件/资源记录	包含 2 列的映射	包含多列的表	目录 [视情况而定]

	DNS	NIS	NIS+	LDAP
服务器名	主/从	主/从	根主/非根主；主/辅助；高速缓存/存根	主/副本
安全性	无	无（根或不包含任何内容）	安全 RPC (AUTH_DH) 验证	SSL（视情况而定）
传输	TCP/IP	RPC	RPC	TCP/IP
范围	全局	LAN	LAN	全局

名称服务转换器（概述）

本章介绍名称服务转换器。名称服务转换器可用于协调各个命名服务的使用。

关于名称服务转换器

名称服务转换器是一个名为 `nsswitch.conf` 的文件，用于控制客户机或应用程序获取网络信息的方式。名称服务转换器由调用任意 `getXbyY()` 接口（如以下接口）的客户机应用程序使用。

- `gethostbyname()`
- `getpwuid()`
- `getpwnam()`
- `getaddrinfo()`

每台计算机的 `/etc` 目录中都有一个转换器文件。该文件中的每一行都标识一条特定类型的网络信息（如主机、口令和组），后跟该信息的一个或多个位置。

客户机可以从转换器的一个或多个源获取命名信息。例如，NIS 客户机可以从 NIS 映射获取其主机信息，从本地 `/etc` 文件获取其口令信息。另外，客户机可以指定转换器在哪些条件下必须使用每个源。请参见表 2-1。

在安装过程中，Solaris 系统会自动将 `nsswitch.conf` 文件装入每台计算机的 `/etc` 目录中。对于 LDAP、NIS、NIS+ 或文件，转换器文件的四个备用（模板）版本也将装入 `/etc` 中。请参见第 36 页中的“`nsswitch.conf` 模板文件”。

这四个文件是备用的缺省转换器文件。每个文件旨在处理不同的主命名服务：`/etc` 文件、NIS、NIS+ 或 LDAP。将 Solaris 软件首次安装到计算机上时，安装程序会选择计算机的缺省命名服务：NIS+、NIS、本地文件或 LDAP。在安装过程中，会将对应的模板文件复制到 `nsswitch.conf` 中。例如，对于使用 LDAP 的客户机，会在安装过程中将 `nsswitch.ldap` 复制到 `nsswitch.conf` 中。除非您拥有特殊的名称空间，否则复制到 `nsswitch.conf` 中的缺省模板文件应该能够满足正常的操作。

系统没有为 DNS 提供缺省文件，但是可以编辑上述任何文件以使用 DNS。有关更多信息，请参见第 41 页中的“DNS 和 Internet 访问”。

如果要稍后更改计算机的主命名服务，应将适当的备用转换器文件复制到 `nsswitch.conf`。请参见第 36 页中的“`nsswitch.conf` 模板文件”。您也可以通过编辑 `/etc/nsswitch.conf` 文件的相应行来更改客户机使用的特定类型网络信息的源。下面的内容介绍语法，其他说明在第 40 页中的“如何修改名称服务转换器”中予以介绍。

nsswitch.conf 文件的格式

`nsswitch.conf` 文件实质上就是一个列表，其中包含 16 种信息和 `getXXbyYY()` 例程在其中搜索这些信息的源。下面是这 16 种信息（并非必须按此顺序）。

- `aliases`
- `bootparams`
- `ethers`
- `group`
- `hosts`
- `ipnodes`
- `netgroup`
- `netmasks`
- `networks`
- `passwd`（包括 shadow 信息）
- `protocols`
- `publickey`
- `rpc`
- `services`
- `automount`
- `sendmailvars`

下表说明了可以列在上述信息类型的转换器文件中的源的种类。

表 2-1 转换器文件中的信息源

信息源	说明
<code>files</code>	存储在客户机的 <code>/etc</code> 目录中的文件。例如， <code>/etc/passwd</code>
<code>nisplus</code>	一个 NIS+ 表。例如， <code>hosts</code> 表。
<code>nis</code>	一个 NIS 映射。例如， <code>hosts</code> 映射。
<code>compat</code>	<code>compat</code> 可用于口令和组信息，以支持 <code>/etc/passwd</code> 、 <code>/etc/shadow</code> 和 <code>/etc/group</code> 文件中的旧式 + 或 - 语法。
<code>dns</code>	可用于指定从 DNS 获取主机信息。
<code>ldap</code>	可用于指定从 LDAP 目录获取项。

搜索条件

单个源。如果某个信息类型只有一个源（如 `nisplus`），使用转换器的例程将只在该源中搜索信息。如果该例程找到了此信息，将返回 `success` 状态消息。如果该例程未找到此信息，则会停止搜索并返回不同的状态消息。不同的例程处理状态消息的方式也不同。

多个源。如果表中包含给定信息类型的多个源，则转换器会指示此例程在列出的第一个源中搜索。如果该例程找到了此信息，将返回 `success` 状态消息。如果该例程未在第一个源中找到此信息，将尝试在下一个源中进行搜索。该例程将依次搜索所有的源，直到找到此信息或者该例程由于指定了 `return` 而停止。如果在搜索了列出的所有源之后仍未找到此信息，该例程将停止搜索并返回 `non-success` 状态消息。

转换器状态消息

如果某例程找到了此信息，则该例程返回 `success` 状态消息。如果该例程未找到此信息，则会返回下面的三个错误状态消息之一。下表列出了可能的状态消息。

表 2-2 转换器搜索状态消息

状态消息	消息的含义
SUCCESS	在指定的源中找到了所请求的项。
UNAVAIL	该源不响应或者不可用。换言之，NIS+ 表、NIS 映射和 <code>/etc</code> 文件均无法找到或访问。
NOTFOUND	该源给出 "No such entry"（该项不存在）的响应。换言之，已经访问该表、映射或文件，但是未找到所需的信息。
TRYAGAIN	该源正忙。该源下次可能会响应。换言之，找到了该表、映射或文件，但是此时无法响应查询。

转换器操作选项

可以指示转换器用下表中显示的两个**操作**之一来响应状态消息。

表 2-3 对转换器状态消息的响应

操作	含义
<code>return</code>	停止查找信息。
<code>continue</code>	尝试在下一个源中查找。

缺省搜索条件

`nsswitch.conf` 文件状态消息和操作选项一起决定例程在每个步骤执行的操作。状态和操作的组合构成了**搜索条件**。

对于每个源来说，转换器的缺省搜索条件完全相同。有关上面列出的状态消息的说明，请参见以下内容。

- **SUCCESS=return**。停止查找信息。使用已经找到的信息以继续。
- **UNAVAIL=continue**。转至下一个 `nsswitch.conf` 文件源并继续搜索。如果该源是最后一个源或唯一的源，将返回 **NOTFOUND** 状态。
- **NOTFOUND=continue**。转至下一个 `nsswitch.conf` 文件源并继续搜索。如果该源是最后一个源或唯一的源，将返回 **NOTFOUND** 状态。
- **TRYAGAIN=continue**。转至下一个 `nsswitch.conf` 文件源并继续搜索。如果该源是最后一个源或唯一的源，将返回 **NOTFOUND** 状态。

可以通过使用上面显示的 `STATUS=action` 语法明确指定某个其他条件来更改缺省搜索条件。例如，**NOTFOUND** 条件的缺省操作是继续搜索下一个源。例如，要指定 `networks` 在遇到 **NOTFOUND** 条件时停止搜索，请编辑转换器文件的 `networks` 行。该行应如下所示。

```
networks: nis [NOTFOUND=return] files
```

`networks: nis [NOTFOUND=return] files` 行可以为 **NOTFOUND** 状态指定非缺省条件。非缺省条件由方括号分隔。

在本示例中，搜索条件按如下方式工作：

- 如果 `networks` 映射可用且包含需要的信息，搜索例程将返回 **SUCCESS** 状态消息。
- 如果 `networks` 映射不可用，搜索例程将返回 **UNAVAIL** 状态消息。缺省情况下，搜索例程会继续搜索相应的 `/etc` 文件。
- 如果 `networks` 映射可用且已找到，但是不包含需要的信息，搜索例程将返回 **NOTFOUND** 消息。但是，搜索例程将停止搜索，而不会继续搜索相应的 `/etc` 文件（这是缺省行为）。
- 如果 `networks` 映射正忙，搜索例程将返回 **TRYAGAIN** 状态消息，而且在缺省情况下会继续搜索相应的 `/etc` 文件。

注 – 在 `nsswitch.conf` 文件中的查找是按照项的列出顺序进行的。但是，口令更新将以相反的顺序进行，除非使用 `passwd - r repository` 命令另行指定了更新顺序。有关更多信息，请参见第 42 页中的“转换器文件和口令信息”。

语法有误时该怎么办？

客户机库例程中包含经过编译的缺省项，如果 `nsswitch.conf` 文件中缺少某项或者该项语法有误，则会使用这些缺省项。这些项与转换器文件的缺省值相同。

名称服务转换器假设表名和源名的拼写正确无误。如果表名或源名的拼写有误，转换器将使用缺省值。

Auto_home 和 Auto_master

auto_home 和 auto_master 表和映射的转换器搜索条件组合成一个名为 automount 的类别。

Timezone 表和转换器文件

timezone 表不使用转换器，因此该表不包含在转换器文件的列表中。

nsswitch.conf 文件中的注释

nsswitch.conf 文件中任何以注释字符（#）开头的行都被解释为注释行。搜索该文件的例程会忽略注释行。

注释标记前面的字符会由搜索 nsswitch.conf 文件的例程进行解释。注释标记右侧的字符会被解释为注释并被忽略。

表 2-4 转换器文件的注释示例

行的类型	示例
注释行。	# hosts: nisplus [NOTFOUND=return] files
解释的行。	hosts: nisplus [NOTFOUND=return] file
部分解释的行。files 元素不会被解释。	hosts: nisplus [NOTFOUND=return] # files

密钥服务器和转换器文件中的 publickey 项



注意 - 对 nsswitch.conf 进行更改之后必须重新启动密钥服务器。

只有启动了密钥服务器后，密钥服务器才会读取名称服务转换器配置文件中的 publickey 项。如果更改转换器配置文件，则只有重新启动密钥服务器后，才会在密钥服务器上注册所做的更改。

nsswitch.conf 模板文件

Solaris 系统提供了四个转换器模板文件以适用于不同的命名服务。每个文件都提供一组不同的缺省信息源。

下面列出了这四个模板文件。

- **LDAP 模板文件。** `nsswitch.ldap` 配置文件指定 LDAP 目录作为计算机的主要信息源。

注 – 为了使用 LDAP 命名服务，除了修改 `nsswitch.conf` 以外，还必须正确地配置所有的 LDAP 客户机。有关更多信息，请参见第 12 章，[设置 LDAP 客户机（任务）](#)。

- **NIS+ 模板文件。** `nsswitch.nisplus` 配置文件指定 NIS+ 作为所有信息（`passwd`、`group`、`automount` 和 `aliases` 除外）的主要源。对于这四个例外文件，主要源是本地 `/etc` 文件，次要源是 NIS+ 表。`[NOTFOUND=return]` 搜索条件指示转换器在获得 "No such entry"（该项不存在）消息时停止在 NIS+ 表中进行搜索。只有当 NIS+ 服务器不可用时，转换器才在本地文件中进行搜索。

- **NIS 模板文件。** `nsswitch.nis` 配置文件指定 NIS 作为所有信息（`passwd`、`group`、`automount` 和 `aliases` 除外）的主要源。对于这四个例外文件，主要源是本地 `/etc` 文件，次要源是 NIS 映射。`[NOTFOUND=return]` 搜索条件指示转换器在获得 "No such entry"（该项不存在）消息时停止在 NIS 映射中进行搜索。只有当 NIS 服务器不可用时，转换器才在本地文件中进行搜索。

因为 `passwd` 和 `group` 的搜索顺序是 `files nis`，所以您不必将 `+` 项放在 `/etc/passwd` 和 `/etc/group` 文件中。

- **文件模板文件。** `nsswitch.files` 配置文件指定本地 `/etc` 文件作为计算机的唯一信息源。`netgroup` 没有“文件”源，因此客户机在转换器文件中不使用该项。

将最符合要求的模板文件复制到 `nsswitch.conf` 配置文件中，然后根据需要修改该文件。

例如，要使用 LDAP 模板文件，可键入以下命令。

```
mymachine# cp /etc/nsswitch.ldap /etc/nsswitch.conf
```

缺省的转换器模板文件

下面列出了随 Solaris 产品提供的四个转换器文件。

示例 2-1 NIS+ 转换器文件模板：`nsswitch.nisplus`

```
#
#
# /etc/nsswitch.nisplus:
```

示例 2-1 NIS+ 转换器文件模板：nsswitch.nisplus (续)

```
#
#
# An example file that could be copied over to /etc/nsswitch.conf;
# it uses NIS+ (NIS Version 3) in conjunction with files.
#
# "hosts:" and "services:" in this file are used only if the
# /etc/netconfig file has a "-" for nametoaddr_libs of "inet"
# transports.

# the following two lines obviate the "+" entry in /etc/passwd
# and /etc/group.
passwd: files nisplus
group: files nisplus
# consult /etc "files" only if nisplus is down.
hosts: nisplus [NOTFOUND=return] files
# Uncomment the following line, and comment out the above, to use
# both DNS and NIS+. You must also set up the /etc/resolv.conf
# file for DNS name server lookup. See resolv.conf(4).
# hosts: nisplus dns [NOTFOUND=return] files
services: nisplus [NOTFOUND=return] files
networks: nisplus [NOTFOUND=return] files
protocols: nisplus [NOTFOUND=return] files
rpc: nisplus [NOTFOUND=return] files
ethers: nisplus [NOTFOUND=return] files
netmasks: nisplus [NOTFOUND=return] files
bootparams: nisplus [NOTFOUND=return] files
publickey: nisplus
netgroup: nisplus
automount: files nisplus
aliases: files nisplus
sendmailvars: files nisplus
```

注- 对于 publickey 项，nisplus 值必须是值列表中的第一个值。例如，对于多个 NIS+ 域会查询的 nsswitch.conf 文件，publickey: nisplus files 是一个正确的项。

示例 2-2 NIS 转换器文件模板

```
#
# /etc/nsswitch.nis:
#
# An example file that could be copied over to /etc/nsswitch.conf;
# it uses NIS (YP) in conjunction with files.
#
# "hosts:" and "services:" in this file are used only if the
# /etc/netconfig file has a "-" for nametoaddr_libs of "inet"
# transports.
#
# the following two lines obviate the "+" entry in /etc/passwd
# and /etc/group.
passwd: files nis
group: files nis
# consult /etc "files" only if nis is down.
```

示例2-2 NIS转换器文件模板 (续)

```
hosts: nis [NOTFOUND=return] files
networks: nis [NOTFOUND=return] files
protocols: nis [NOTFOUND=return] files
rpc: nis [NOTFOUND=return] files
ethers: nis [NOTFOUND=return] files
netmasks: nis [NOTFOUND=return] files
bootparams: nis [NOTFOUND=return] files
publickey: nis [NOTFOUND=return] files
netgroup: nis
automount: files nis
aliases: files nis
# for efficient getservbyname() avoid nis
services: files nis
sendmailvars: files
```

示例2-3 文件转换器文件模板

```
#
# /etc/nsswitch.files:
#
# An example file that could be copied over to /etc/nsswitch.conf;
# it does not use any naming service.
#
# "hosts:" and "services:" in this file are used only if the
# /etc/netconfig file has a "-" for nametoaddr_libs of "inet"
# transports.
passwd: files
group: files
hosts: files
networks: files
protocols: files
rpc: files
ethers: files
netmasks: files
bootparams: files
publickey: files
# At present there isn't a 'files' back end for netgroup;
# the system will figure it out pretty quickly, and will not use
# netgroups at all.
netgroup: files
automount: files
aliases: files
services: files
sendmailvars: files
```

示例2-4 LDAP转换器文件模板

```
#
# /etc/nsswitch.ldap:
#
# An example file that could be copied over to /etc/nsswitch.conf; it
# uses LDAP in conjunction with files.
#
# "hosts:" and "services:" in this file are used only if the
# /etc/netconfig file has a "-" for nametoaddr_libs of "inet" transports.
```

示例 2-4 LDAP 转换器文件模板 (续)

```
# the following two lines obviate the "+" entry in /etc/passwd
and /etc/group.
passwd:    files ldap
group:     files ldap

hosts:     ldap [NOTFOUND=return] files

networks:  ldap [NOTFOUND=return] files
protocols: ldap [NOTFOUND=return] files
rpc:       ldap [NOTFOUND=return] files
ethers:    ldap [NOTFOUND=return] files
netmasks: ldap [NOTFOUND=return] files
bootparams: ldap [NOTFOUND=return] files
publickey: ldap [NOTFOUND=return] files

netgroup:  ldap

automount: files ldap
aliases:   files ldap

# for efficient getservbyname() avoid ldap
services:  files ldap
sendmailvars: files
```

nsswitch.conf 文件

随 Solaris 软件安装的缺省 `nsswitch.conf` 文件由安装过程中选择的命名服务确定。每一行都标识特定类型的网络信息（如主机、口令和组）以及信息源（如 NIS+ 表、NIS 映射、DNS 主机表或本地 `/etc`）。在选择某个命名服务时，会复制该服务的转换器模板文件以创建新的 `nsswitch.conf` 文件。例如，如果选择 NIS，则会复制 `nsswitch.nis` 文件以创建新的 `nsswitch.conf` 文件。

Solaris 9 发行版软件会自动将 `nsswitch.conf` 文件与下列备用（模板）版本一起装入每台计算机的 `/etc` 目录中。

- `/etc/nsswitch.nisplus`
- `/etc/nsswitch.nis`
- `/etc/nsswitch.files`
- `/etc/nsswitch.ldap`

这些备用模板文件中包含由 NIS+ 服务、NIS 服务、本地文件和 LDAP 使用的缺省转换器配置。系统没有为 DNS 提供缺省文件，但是可以编辑其中的任何文件以使用 DNS。Solaris 软件首次安装到计算机上时，安装程序会选择计算机的缺省命名服务。在安装过程中，会将对应的模板文件复制到 `/etc/nsswitch.conf` 中。例如，对于使用 NIS 的客户机，会在安装过程中将 `nsswitch.nis` 复制到 `nsswitch.conf` 中。

如果网络连接到 Internet，而且用户必须使用 DNS 访问 Internet 主机，则必须启用 DNS 转发。

除非您拥有特殊的名称空间，否则复制到 `nsswitch.conf` 中的缺省模板文件应该能够满足正常的操作。

选择其他配置文件

更改计算机的命名服务时，需要相应地修改计算机的转换器文件。例如，如果将计算机的命名服务从文件更改为 NIS，需要安装一个适用于 NIS 的转换器文件。您可以通过将对应的模板文件复制到 `nsswitch.conf` 中来更改转换器文件。

如果要使用 NIS+ 安装脚本在计算机上安装 NIS+，系统会将 NIS+ 模板脚本复制到 `nsswitch.conf` 中。在这种情况下，除非您希望进行定制，否则不必对转换器文件进行配置。

在更改转换器文件之前，确保文件中列出的源已进行正确设置。换言之，如果打算选择 NIS 版本，客户机必须最终能够访问 NIS 服务。如果打算选择本地文件版本，则必须在客户机上正确设置需要的文件。

▼ 如何修改名称服务转换器

要更改转换器文件，请执行以下步骤。

注 – 为了使用 LDAP 命名服务，除了修改 `nsswitch.conf` 以外，还必须正确地配置所有的 LDAP 客户机。有关更多信息，请参见第 12 章，[设置 LDAP 客户机（任务）](#)。

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 将适用于计算机命名服务的相应备用文件复制到 `nsswitch.conf` 文件中。

NIS+ 版本（由 NIS+ 脚本自动完成）

```
client1# cd /etc
client1# cp nsswitch.nisplus nsswitch.conf
```

NIS 版本

```
client1# cd /etc
client1# cp nsswitch.nis nsswitch.conf
```

本地 /etc 文件版本

```
client1# cd /etc
client1# cp nsswitch.files nsswitch.conf
```


3 重新引导计算机。

nscd 守护进程对转换器信息进行缓存。有关相应的信息，请参见 [nscd\(1M\)](#) 手册页。

某些库例程不会通过定期检查 `nsswitch.conf` 文件来查看该文件是否已被更改。必须重新引导计算机以确保此守护进程和这些例程具有该文件中的最新信息。

DNS 和 Internet 访问

`nsswitch.conf` 文件还控制客户机的 DNS 转发，如以下各小节中所述。DNS 转发允许客户机访问 Internet。有关如何为具有 NIS 兼容性的 NIS+ 设置 DNS 转发的信息，请参见 [《System Administration Guide: Naming and Directory Services \(NIS+\)》](#)。

IPv6 和 Solaris 命名服务

NIS、NIS+ 和 LDAP 支持存储 IPv6 数据，还支持将 IPv6 传输机制用于协议通信。从 BIND 8.3.3 版开始，Solaris OS 上的 DNS 支持在客户端使用 IPv6 传输机制。截至 BIND 版本 8.4.2，Solaris 上的 DNS 可以通过 IPv6 网络提供完整的客户机/服务器解决方案。

`nsswitch.conf` 文件控制 IPv6 地址的搜索条件。IPv6 将 IP 地址的大小从 32 位增加到 128 位，从而可以支持更多层的寻址分层结构。地址大小越大，提供的可寻址节点越多。有关 IPv6 及其配置和实现的更多信息，请参见 [《系统管理指南：IP 服务》](#)。

将新的 `ipnodes` 源用于 IPv6 地址。`/etc/inet/ipnodes` 文件同时存储 IPv4 和 IPv6 地址。`/etc/inet/ipnodes` 文件与 `/etc/hosts` 文件使用相同的格式约定。

能够识别 IPv6 的命名服务将新的 `ipnodes` 源用于搜索转发。例如，如果 LDAP 能够识别 IPv6 地址，请指定以下内容。

```
ipnodes: ldap [NOTFOUND=return] files
```



注意 – 潜在的延迟问题：

- `ipnodes` 的缺省值为 `files`。从 IPv4 转换为 IPv6 的过程中，如果所有命名服务均不能识别 IPv6 地址，将采用缺省值 `files`。否则，在解析地址的过程中可能会出现不必要的延迟（如引导计时延迟）。
- 应用程序会首先搜索所有 `ipnodes` 数据库，然后再搜索 `hosts` 数据库，以查找 IPv4 地址。在指定 `ipnodes` 之前，请考虑在这两个数据库中搜索 IPv4 地址所固有的延迟。

确保与 +/- 语法兼容

如果在 `/etc/passwd`、`/etc/shadow` 和 `/etc/group` 文件中使用了 +/-，则需要修改 `nsswitch.conf` 文件以确保与 +/- 兼容。

- **NIS+**。要向 NIS+ 提供 +/- 语义，请将 `passwd` 和 `groups` 源更改为 `compat`。然后，将 `passwd_compat: nisplus` 项添加到 `nsswitch.conf` 文件中 `passwd` 或 `group` 项的后面，如下所示。

```
passwd: compat
passwd_compat: nisplus
group: compat
group_compat: nisplus
```

上面的项指定客户机例程从 `/etc` 文件和这些文件中的 +/- 项所指示的 NIS+ 表获取其网络信息。

- **NIS**。要提供与 Solaris 4.x 发行版中相同的语法，请将 `passwd` 和 `groups` 源更改为 `compat`。

```
passwd: compat
group: compat
```

指定 `/etc` 文件和由这些文件中的 +/- 项指定的 NIS 映射。

注 - 如果用户所使用的客户机由运行在 NIS 兼容模式下的 NIS+ 服务器提供服务，用户将无法针对 `netgroup` 表运行 `ypcat`。如果运行，即使该表中有项，所得到的结果也与该表为空时一样。

转换器文件和口令信息

可以将口令信息包括在多个系统信息库（如 `files` 和 `nisplus`）中并访问它们。使用 `nsswitch.conf` 文件可以为这些信息设置查找顺序。



注意 - `files` 必须是 `nsswitch.conf` 文件中 `passwd` 信息的第一个源。

在 NIS+ 环境中，`nsswitch.conf` 文件中的 `passwd` 行应当按以下顺序列出系统信息库。

```
passwd: files nisplus
```

在 NIS 环境中，`nsswitch.conf` 文件中的 `passwd` 行应当按以下顺序列出系统信息库：

```
passwd: files nis
```

提示 – 首先列出 `files` 可允许 `root` 在大多数情况下登录，即使系统遇到某些网络问题或命名服务问题也如此。

建议不要为同一个用户维护多个系统信息库。通过为每个用户在单个系统信息库中维护集中的口令管理，可以减少发生混淆和错误的几率。如果选择为每个用户维护多个系统信息库，请使用 `passwd -r` 命令更新口令信息。

`passwd -r repository`

如果没有用 `-r` 选项指定系统信息库，`passwd` 会以相反的顺序更新 `nsswitch.conf` 中列出的系统信息库。

第 2 部分

DNS 设置和管理

本部分介绍 Solaris OS 中 BIND 9 DNS 命名服务的配置和管理。

DNS 设置和管理（参考）

Solaris 操作系统 (Solaris OS) 提供了 BIND 9.x DNS 名称服务器。本章介绍了有关在 Solaris 操作系统中使用 BIND 9 的配置和管理信息。常规的 BIND 和 DNS 信息可从其他许多来源获得，包括第 47 页中的“相关材料”中列出的来源。

本章包含以下主题：

- 第 47 页中的“相关材料”
- 第 48 页中的“从 BIND 8 迁移到 BIND 9”
- 第 49 页中的“DNS 和服务管理工具”
- 第 50 页中的“实现 rndc”
- 第 52 页中的“BIND 9 命令、文件、工具和选项”
- 第 54 页中的“named.conf 选项”

相关材料

有关 DNS 和 BIND 管理的信息，请参见以下文档。

- /usr/share/doc/bind/migration.txt 中的 BIND 9 迁移说明文档
- Internet Systems Consortium (ISC) Web 站点 <http://www.isc.org> 中的 BIND 9 管理员手册
- ISC Web 站点 <http://www.isc.org> 中有关 BIND 功能、已知错误和缺陷的列表以及指向其他材料的链接
- 《DNS and Bind》，由 Paul Albitz 和 Cricket Liu 编著，（第 4 版，O'Reilly 出版社，2001 年）

从 BIND 8 迁移到 BIND 9

BIND 9 可与大多数 BIND 8 功能向上兼容。但是，在升级现有的 BIND 8 安装以使用 BIND 9 时，仍需了解许多注意事项。在安装和使用 BIND 9 之前，请务必阅读整个迁移说明文档。迁移说明位于 `/usr/share/doc/bind/migration.txt` 中。此外，BIND 软件包名称已更改为 SUNWbind 和 SUNWbindr。SUNWbindr 软件包中包含 DNS 服务器 manifest。

以下列表简要概括了 BIND 8 与 BIND 9 之间的区别。迁移说明中提供了详细信息。

- 配置文件兼容性
 - 关于未实现选项的警告消息
 - *transfer-format* 选项已更改
 - 配置文件错误
 - 日志类别已更改
 - 通知消息和刷新查询已更改
 - 多个类已更改
- 区域文件兼容性
 - 区域文件中 TTL 规则更严格
 - SOA（面向服务的体系结构）序列号已更改
 - 引号不配对将引起错误
 - 换行符、语言更改
 - 在域名中使用 `\$` 代替 `$$`
- 新协议功能的互操作性影响
 - BIND 9 中新增 EDNS0
 - 区域传送缺省值已更改
- 不受限的字符集合
 - 对字符集合没有任何限制
 - 安全问题，错误命名
- 服务器管理工具
 - `rndc` 程序取代了 `ndc`
 - `nsupdate`: 多个更新的方式已更改
- 区域之间无信息泄漏
 - 以不同的方式处理粘附 NS 记录
- 未修改 Umask
 - 可能的 umask 权限问题

DNS 和服务管理工具

使用服务管理工具 (Service Management Facility, SMF) 可以管理 DNS/BIND named 服务。有关 SMF 的概述, 请参见《系统管理指南: 基本管理》中的第 18 章“管理服务 (概述)”。有关更多详细信息, 另请参阅 [svcadm\(1M\)](#)、[svcs\(1\)](#) 和 [svccfg\(1M\)](#) 手册页。还可以查看 `/var/svc/manifest/network/dns` 中的 DNS 服务器 manifest `server.xml`。

- 使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。

提示 – 使用 `-t` 选项暂时禁用服务可为服务配置提供一些保护。如果禁用服务时使用了 `-t` 选项, 则服务在重新引导后将恢复原始设置。如果禁用服务时未使用 `-t`, 则服务在重新引导后仍将保持禁用状态。

- 用于 DNS 服务的故障管理资源标识符 (Fault Managed Resource Identifier, FMRI) 是 `svc:/network/dns/server:<instance>` 和 `svc:/network/dns/client:<instance>`。
- 使用 `svcs` 命令可以查询 DNS 服务器和客户机的状态。

- `svcs` 命令和输出示例。

```
# svcs \*dns\*
STATE      STIME      FMRI
online     Nov_16     svc:/network/dns/server:default
online     Nov_16     svc:/network/dns/client:default
```

- `svcs -l` 命令和输出示例。

```
# svcs -l /network/dns/server
fmri      svc:/network/dns/server:default
name      Internet domain name server (DNS)
enabled   true
state     online
next_state none
restarter svc:/system/svc/restarter:default
contract_id 25
dependency require_all/none svc:/system/filesystem/minimal (online)
dependency require_all/none file://localhost/etc/named.conf (online)
dependency require_any/error svc:/network/loopback (online)
dependency optional_all/error svc:/network/physical (online)
```

- 如果需要以不同选项启动 DNS 服务 (例如, 用 `/etc/named.conf` 之外的配置文件), 可以使用 `svccfg` 命令更改 DNS 服务器 manifest 的 *start method* 属性。
- 仅当要运行 BIND 9 名称服务的多个副本时, 才需要多个 SMF 服务实例。在 DNS 服务器 manifest 中可为每个附加实例指定不同的启动方法。

尽管建议使用 `svcadm` 管理服务器, 但也可以使用 `rndc`。SMF 可识别 BIND 9 named 服务的状态更改, 无论使用 `svcadm` 还是 `rndc` 进行管理都是如此。

注 – 如果从命令行手动执行服务，SMF 将不识别 BIND 9 named 服务。

实现 rndc

BIND 8 ndc 和 BIND 9 rndc 名称服务器控制工具不向后兼容。rndc 不能与 BIND 8 名称服务器对话，ndc 不能与 BIND 9 名称服务器对话。功能、选项、缺省操作模式以及配置文件要求都已更改。因此，在 BIND 9 服务器中使用 ndc 会导致功能丧失或操作不安全。有关更多信息，请参见 rndc(1M) 手册页。

rndc.conf 配置文件

BIND 8 中的 ndc 与 BIND 9 中的 rndc 之间最显著的差别在于 rndc 需要自己的配置文件 rndc.conf。此文件可由 rndc-confgen 命令生成。rndc.conf 文件指定将由哪个服务器进行控制及该服务器应使用哪种算法。

示例 3-1 rndc.conf 文件样例

```
options {
    default-server localhost;
    default-key "rndc-key";
};

key "rndc-key" {
    algorithm hmac-md5;
    secret "qPWZ3Ndl81aBRY9AmJhVtU==";
};
```

示例 3-2 用于 rndc 的 named.conf 文件项样例

```
controls {
    inet * allow { any; } keys { "rndc-key"; };
};

key "rndc-key" {
    algorithm hmac-md5;
    secret "qPWZ3Ndl81aBRY9AmJhVtU==";
};
```

控制通道之间的差别

ndc 和 rndc 实用程序都使用控制通道来向名称服务器发送信息以及从该服务器中检索信息。但是，这两个实用程序之间存在差别。

- 在 BIND 8 中，ndc 可以使用 AF_UNIX 域套接字（UNIX 控制通道）或 TCP/IP 套接字（inet 控制通道）。缺省情况下，ndc 不需要 /etc/named.conf 中提供的任何支持，因为 BIND 8 服务器使用 UNIX 域套接字，且路径 (/var/run/ndc.d/ndc) 已编译到 in.named 中。

但对于 BIND 9，rndc 只使用经过验证的 TCP/IP inet 控制通道，因而不与 BIND 8 向后兼容。在 BIND 9 服务器中，不存在对控制通道的 UNIX 域套接字支持。

- 使用 rndc 时，需要指定用于与名称服务器通信的 'key' 子句。BIND 9 服务器和 rndc 客户机必须共享同一密钥（在 /etc/named.conf 和 /etc/rndc.conf 中定义）。在 BIND 9 中使用 BIND 8 控制项将产生错误消息。
- 从 ndc 到 rndc 实现，一些命令选项已更改。其中包括 -c 选项，该选项在 BIND 9 中具有不同的语法。因此，要在 BIND 9 中指定控制通道，请使用 rndc -s <server> -p <port>。

BIND 9 rndc 的命令

以下列表介绍 rndc 命令。

reload	重新装入配置文件和区域
reload zone [class [view]]	重新装入单个区域
refresh zone [class [view]]	安排区域的立即维护
reconfig	仅重新装入配置文件和新区域
stats	将服务器统计信息写入统计文件中
querylog	切换查询日志
dumpdb	将高速缓存转储到转储文件 (named_dump.db)
stop	将暂挂更新保存到主文件并停止服务器
halt	停止服务器，但不保存暂挂更新
trace	将调试级别增加一级
trace level	更改调试级别
notrace	将调试级别设置为 0
flush	刷新服务器的所有高速缓存
flush [view]	为某一视图刷新服务器的高速缓存
status	显示服务器的状态
restart	重新启动服务器（尚未实现）

BIND 9 命令、文件、工具和选项

在 BIND 9 中，有些命令、文件、工具和选项与 BIND 8 中保持相同。但是，有些已被修改，还有一些是新增的。本节介绍 BIND 9 中的许多命令、文件、工具和选项以及每项关联的新增行为或已修改的行为。

BIND 9 工具和配置文件

Solaris 操作系统提供了以下 BIND 9.x 工具。

```
named
nsupdate
rndc
dnssec-keygen
nslookup
dig
dnssec-makekeyset
dnssec-signkey
dnssec-signzone
named-checkconf
named-checkzone
rndc-confgen
host
```

Solaris 10 发行版支持以下 BIND 9.x 配置文件。

```
/etc/rndc.conf
```

BIND 8 和 BIND 9 命令和文件比较

下表对 BIND 8 和 BIND 9 的命令及配置文件进行了比较。

BIND 8 命令	BIND 9.x 替代命令
dnskeygen(1M)	dnssec-keygen(1M)
ndc(1M)	rndc(1M)
named-bootconf(1M)	不需要
nsupdate(1M)	nsupdate(1M)

BIND 8 命令	BIND 9.x 替代命令
nslookup(1M)	nslookup(1M)
named-xfer(1M)	不需要
in.named(1M)	named(1M)
named.conf(4)	named.conf ¹
dig(1M)	dig(1M)

¹ BIND 9.2.4 不包括详细的 named.conf 手册页。第 54 页中的“named.conf 选项”中包含 BIND 9.2.4 支持的 named.conf 选项的摘要。

命令和选项更改的说明

下面列出的所有不兼容项都是等效的 BIND 9 二进制文件所不支持的 BIND 8 功能和接口。此列表不是任何 BIND 9.x 二进制文件的选项、命令行选项或功能的详尽列表。

命令	选项更改
in.named(1M)	不支持 DNS 名称服务器的一些 in.named 命令行选项。 在 BIND 9.x 名称服务器中，不支持 -g <i>group_name</i> 、-q、-r 和 -w <i>directory</i> 选项，并且 -c <i>config_file</i> 替代了 BIND 8.x -b <i>config_file</i> 。有关详细信息，请参见 named 手册页。
dnssec-keygen(1M)	BIND 8.x 中用于生成密钥的 dnskeygen 与 BIND 9.x 中的 dnssec-keygen 没有通用选项。有关详细信息，请参见 dnssec-keygen 手册页。
rndc(1M)	BIND 8.x 中的 ndc 与 BIND 9.x 中的 rndc 存在显著差别。它们之间没有通用选项，并且与 ndc 不同，rndc 需要 /etc/rndc.conf 中的配置文件才能运行。有关详细信息，请参见 rndc、rndc.conf 和 rndc-confgen 手册页。
nsupdate(1M)	在 BIND 9.x 中，nsupdate 中 -k 选项的语法已更改。现在语法已不再是 -k <i>keydir:keyname</i> ，而是 k <i>keyfile</i> 。其他仅有的差别是，以前使用空白行作为向服务器发送输入的信号，而现在使用显式的 send 子命令来执行相同操作。有关详细信息，请参见 nsupdate 手册页。
nslookup(1M)	9.x 版本的 BIND 不支持以下选项：help、host server、set ignoretc、set noignoretc、set srch[list]=N1[/N2/.../N6]、set ro[ot]=host、root、finger [USER]、ls [opt] DOMAIN [> FILE]
named.conf(4)	BIND 9.2.4 中不包括详细的 named.conf 手册页。有些选项不受支持、未实现或其缺省值已更改。有关选项更改列表和 BIND 9.2.4 中支持的所有 named.conf 选项的摘要，请参见第 54 页中的“named.conf 选项”。

named.conf 选项

以下列表比较了 BIND 8 和 BIND 9 的 `named.conf` 选项，并对更改进行了简短说明。“更改”列中的 OK 表示选项在 BIND 9 版本的 `named` 中工作方式不变。

选项{	更改
[version version_string;]	OK
[directory path_name;]	OK
[named-xfer path_name;]	过时 ¹
[dump-file path_name;]	OK
[memstatistics-file path_name;]	未实现
[pid-file path_name;]	OK
[statistics-file path_name;]	OK
[auth-nxdomain yes_or_no;]	OK ²
[dialup yes_or_no;]	OK
[fake-iquery yes_or_no;]	过时
[fetch-glue yes_or_no;]	过时
[has-old-clients yes_or_no;]	过时
[host-statistics yes_or_no;]	未实现
[host-statistics-max number;]	未实现
[multiple-cnames yes_or_no;]	过时
[notify yes_or_no explicit;]	OK
[recursion yes_or_no;]	OK
[rfc2308-type1 yes_or_no;]	未实现
[use-id-pool yes_or_no;]	过时
[treat-cr-as-space yes_or_no;]	过时
[also-notify yes_or_no;]	语法已更改 ³
[forward (only first);]	OK ⁴

¹ 由于体系结构差别而过时。

² BIND 8 中缺省设置为 *yes*，而 BIND 9 中缺省设置为 *no*。

³ 选择 *yes* 时，需要一个 IP 地址。

⁴ 如果不指定转发器，此选项将不起作用；在此情况下，会产生 *no matching 'forwarders' statement*（没有匹配的“forwarders”语句）错误。

选项 {	更改
[forwarders { [in_addr; \	OK ⁵
[in_addr; ...] }];]	
[check-names (master slave \	未实现
response) (warn fail ignore);]	
[allow-query { address_match_list };]	OK
[allow-recursion { address_match_list };]	OK
[allow-transfer { address_match_list };]	OK
[blackhole { address_match_list };]	OK
[listen-on [port ip_port] \	OK
{ address_match_list };]	
[query-source [address (ip_addr *)] \	OK
[port (ip_port *)]];]	OK
[lame-ttl number;]	
[max-transfer-time-in number;]	OK
[max-ncache-ttl number;]	OK
[min-roots number;]	未实现
[transfer-format (one-answer \	OK ⁶
many-answers);]	
[transfers-in number;]	OK
[transfers-out number;]	OK
[transfers-per-ns number;]	OK
[transfer-source ip_addr;]	OK
[maintain-ixfr-base yes_or_no;]	过时
[max-ixfr-log-size number;]	过时 ⁷
[coresize size_spec;]	OK
[datasize size_spec;]	OK
[files size_spec;]	OK

⁵ 请参见 [forward] 子句。

⁶ BIND 8 中缺省设置为 *one-answer*，而 BIND 9 中缺省设置为 *many-answers*。

⁷ 不需要此选项，因为 BIND 9 会自动剪裁其日志文件的大小。

选项 {	更改
[stacksize size_spec ;]	OK
[cleaning-interval number ;]	OK
[heartbeat-interval number ;]	OK
[interface-interval number ;]	OK
[statistics-interval number ;]	未实现
[topology { address_match_list ; }]	未实现
[sortlist { address_match_list ; }]	OK
[rrset-order { order_spec ; \	未实现
[order_spec ; ...] } ;]	
};	

BIND 9 中的语句

本节介绍 BIND 8 语句与 BIND 9 语句之间的所有差别。

Controls 语句

unix 是 *ndc* 的缺省设置，并且编译了所有参数。*inet* 是 *rndc* 唯一的选项，并且未编译任何内容。

Syntax	
controls {	
[inet ip_addr	
port ip_port	
allow { address_match_list ; } ;]	OK
[unix path_name	
perm number	
owner number	
group number ;]	Not Implemented
};	

日志语法发生了显著变化。有关 *named.conf* 选项的列表，请参见第 54 页中的“*named.conf* 选项”。

Zone 语句

BIND 8 *named.conf* 手册页中用于区域语句的语法大部分在 BIND 9 中是受支持的，但以下语法除外：

[pubkey number number number string ;]	Obsolete
[check-names (warn fail ignore) ;]	Not Implemented

ACL 语句

该语句的工作方式在 BIND 9 中未变化。

```
Syntax
acl name {
    address_match_list
};
```

Key 语句

该语句的工作方式在 BIND 9 中未变化。

```
Syntax
key key_id {
    algorithm algorithm_id;
    secret secret_string;
};
```

Trusted-Keys 语句

工作方式未变化，但使用此语句的代码在 BIND 9.2.4 中已禁用。

```
Syntax
trusted-keys {
    [ domain_name flags protocol algorithm key; ]
};
```

Server 语句

support-ixfr 已过时，但以下所有选项的工作方式在 BIND 9 中未变化。请注意，*transfer-format* 的缺省设置已变化。

```
Syntax
server ip_addr {
    [ bogus_yes_or_no; ]
    [ transfers number; ]
    [ transfer-format ( one-answer | many-answers ); ]
    [ keys { key_id [ key_id ... ] }; ]
    [ edns yes_or_no; ]
};
```

Include 语句

该语句的工作方式在 BIND 9 中未变化。

```
Syntax
include path_name;
```

named.conf 选项摘要

BIND 9.2.4 不包括详细的 named.conf 手册页。下面是 BIND 9.2.4 支持的 named.conf 选项的摘要。

```
options {
    blackhole { <address_match_element>; ... };
    coresize <size>;
    datasize <size>;
    deallocate-on-exit <boolean>; // obsolete
    directory <quoted_string>;
    dump-file <quoted_string>;
    fake-iquery <boolean>; // obsolete
    files <size>;
    has-old-clients <boolean>; // obsolete
    heartbeat-interval <integer>;
    host-statistics <boolean>; // not implemented
    host-statistics-max <integer>; // not implemented
    interface-interval <integer>;
    listen-on [ port <integer> ] { <address_match_element>; ... };
    listen-on-v6 [ port <integer> ] { <address_match_element>; ... };
    match-mapped-addresses <boolean>;
    memstatistics-file <quoted_string>; // not implemented
    multiple-cnames <boolean>; // obsolete
    named-xfer <quoted_string>; // obsolete
    pid-file <quoted_string>;
    port <integer>;
    random-device <quoted_string>;
    recursive-clients <integer>;
    rrset-order { [ class <string> ] [ type <string> ] [ name
        <quoted_string> ] <string> <string>; ... }; // not implemented
    serial-queries <integer>; // obsolete
    serial-query-rate <integer>;
    stacksize <size>;
    statistics-file <quoted_string>;
    statistics-interval <integer>; // not yet implemented
    tcp-clients <integer>;
    tkey-dhkey <quoted_string> <integer>;
    tkey-gssapi-credential <quoted_string>;
    tkey-domain <quoted_string>;
    transfers-per-ns <integer>;
    transfers-in <integer>;
    transfers-out <integer>;
    treat-cr-as-space <boolean>; // obsolete
    use-id-pool <boolean>; // obsolete
    use-ixfr <boolean>;
    version <quoted_string>;
    allow-recursion { <address_match_element>; ... };
    allow-v6-synthesis { <address_match_element>; ... };
    sortlist { <address_match_element>; ... };
    topology { <address_match_element>; ... }; // not implemented
    auth-nxdomain <boolean>; // default changed
    minimal-responses <boolean>;
    recursion <boolean>;
    provide-ixfr <boolean>;
    request-ixfr <boolean>;
    fetch-glue <boolean>; // obsolete
```

```

rfc2308-type1 <boolean>; // not yet implemented
additional-from-auth <boolean>;
additional-from-cache <boolean>;
query-source <querysource4>;
query-source-v6 <querysource6>;
cleaning-interval <integer>;
min-roots <integer>; // not implemented
lame-ttl <integer>;
max-ncache-ttl <integer>;
max-cache-ttl <integer>;
transfer-format ( many-answers | one-answer );
max-cache-size <size_no_default>;
check-names <string> <string>; // not implemented
cache-file <quoted_string>;
allow-query { <address_match_element>; ... };
allow-transfer { <address_match_element>; ... };
allow-update-forwarding { <address_match_element>; ... };
allow-notify { <address_match_element>; ... };
notify <notifytype>;
notify-source ( <ipv4_address> | * ) [ port ( <integer> | * ) ];
notify-source-v6 ( <ipv6_address> | * ) [ port ( <integer> | * ) ];
also-notify [ port <integer> ] { ( <ipv4_address> | <ipv6_address>
    ) [ port <integer> ]; ... };
dialup <dialuptype>;
forward ( first | only );
forwarders [ port <integer> ] { ( <ipv4_address> | <ipv6_address> )
    [ port <integer> ]; ... };
maintain-ixfr-base <boolean>; // obsolete
max-ixfr-log-size <size>; // obsolete
transfer-source ( <ipv4_address> | * ) [ port ( <integer> | * ) ];
transfer-source-v6 ( <ipv6_address> | * ) [ port ( <integer> | * ) ];
max-transfer-time-in <integer>;
max-transfer-time-out <integer>;
max-transfer-idle-in <integer>;
max-transfer-idle-out <integer>;
max-retry-time <integer>;
min-retry-time <integer>;
max-refresh-time <integer>;
min-refresh-time <integer>;
sig-validity-interval <integer>;
zone-statistics <boolean>;
};

controls {
    inet ( <ipv4_address> | <ipv6_address> | * ) [ port ( <integer> | *
        ) ] allow { <address_match_element>; ... } [ keys { <string>; ... } ];
    unix <unsupported>; // not implemented
};

acl <string> { <address_match_element>; ... };

logging {
    channel <string> {
        file <logfile>;
        syslog <optional_facility>;
        null;
        stderr;
        severity <logseverity>;
        print-time <boolean>;
    };
};

```

```

        print-severity <boolean>;
        print-category <boolean>;
    };
    category <string> { <string>; ... };
};

view <string> <optional_class> {
    match-clients { <address_match_element>; ... };
    match-destinations { <address_match_element>; ... };
    match-recursive-only <boolean>;
    key <string> {
        algorithm <string>;
        secret <string>;
    };
    zone <string> <optional_class> {
        type ( master | slave | stub | hint | forward );
        allow-update { <address_match_element>; ... };
        file <quoted_string>;
        ixfr-base <quoted_string>; // obsolete
        ixfr-tmp-file <quoted_string>; // obsolete
        masters [ port <integer> ] { ( <ipv4_address> |
            <ipv6_address> ) [ port <integer> ] [ key <string> ]; ... };
        pubkey <integer> <integer> <integer> <quoted_string>; //
            obsolete
        update-policy { ( grant | deny ) <string> ( name |
            subdomain | wildcard | self ) <string> <rrtypelist>; ... };
        database <string>;
        check-names <string>; // not implemented
        allow-query { <address_match_element>; ... };
        allow-transfer { <address_match_element>; ... };
        allow-update-forwarding { <address_match_element>; ... };
        allow-notify { <address_match_element>; ... };
        notify <notifytype>;
        notify-source ( <ipv4_address> | * ) [ port ( <integer> | *
            ) ];
        notify-source-v6 ( <ipv6_address> | * ) [ port ( <integer>
            | * ) ];
        also-notify [ port <integer> ] { ( <ipv4_address> |
            <ipv6_address> ) [ port <integer> ]; ... };
        dialup <dialuptype>;
        forward ( first | only );
        forwarders [ port <integer> ] { ( <ipv4_address> |
            <ipv6_address> ) [ port <integer> ]; ... };
        maintain-ixfr-base <boolean>; // obsolete
        max-ixfr-log-size <size>; // obsolete
        transfer-source ( <ipv4_address> | * ) [ port ( <integer> |
            * ) ];
        transfer-source-v6 ( <ipv6_address> | * ) [ port (
            <integer> | * ) ];
        max-transfer-time-in <integer>;
        max-transfer-time-out <integer>;
        max-transfer-idle-in <integer>;
        max-transfer-idle-out <integer>;
        max-retry-time <integer>;
        min-retry-time <integer>;
        max-refresh-time <integer>;
        min-refresh-time <integer>;
        sig-validity-interval <integer>;
        zone-statistics <boolean>;
    };
};

```

```

};
server {
    bogus <boolean>;
    provide-ixfr <boolean>;
    request-ixfr <boolean>;
    support-ixfr <boolean>; // obsolete
    transfers <integer>;
    transfer-format ( many-answers | one-answer );
    keys <server_key>;
    edns <boolean>;
};
trusted-keys { <string> <integer> <integer> <integer>
    <quoted_string>; ... };
allow-recursion { <address_match_element>; ... };
allow-v6-synthesis { <address_match_element>; ... };
sortlist { <address_match_element>; ... };
topology { <address_match_element>; ... }; // not implemented
auth-nxdomain <boolean>; // default changed
minimal-responses <boolean>;
recursion <boolean>;
provide-ixfr <boolean>;
request-ixfr <boolean>;
fetch-glue <boolean>; // obsolete
rfc2308-type1 <boolean>; // not yet implemented
additional-from-auth <boolean>;
additional-from-cache <boolean>;
query-source <querysource4>;
query-source-v6 <querysource6>;
cleaning-interval <integer>;
min-roots <integer>; // not implemented
lame-ttl <integer>;
max-ncache-ttl <integer>;
max-cache-ttl <integer>;
transfer-format ( many-answers | one-answer );
max-cache-size <size_no_default>;
check-names <string> <string>; // not implemented
cache-file <quoted_string>;
allow-query { <address_match_element>; ... };
allow-transfer { <address_match_element>; ... };
allow-update-forwarding { <address_match_element>; ... };
allow-notify { <address_match_element>; ... };
notify <notifytype>;
notify-source ( <ipv4_address> | * ) [ port ( <integer> | * ) ];
notify-source-v6 ( <ipv6_address> | * ) [ port ( <integer> | * ) ];
also-notify [ port <integer> ] { ( <ipv4_address> | <ipv6_address>
    ) [ port <integer> ]; ... };
dialup <dialuptype>;
forward ( first | only );
forwarders [ port <integer> ] { ( <ipv4_address> | <ipv6_address> )
    [ port <integer> ]; ... };
maintain-ixfr-base <boolean>; // obsolete
max-ixfr-log-size <size>; // obsolete
transfer-source ( <ipv4_address> | * ) [ port ( <integer> | * ) ];
transfer-source-v6 ( <ipv6_address> | * ) [ port ( <integer> | * ) ];
max-transfer-time-in <integer>;
max-transfer-time-out <integer>;
max-transfer-idle-in <integer>;
max-transfer-idle-out <integer>;
max-retry-time <integer>;

```

```

min-retry-time <integer>;
max-refresh-time <integer>;
min-refresh-time <integer>;
sig-validity-interval <integer>;
zone-statistics <boolean>;
};

lwres {
    listen-on [ port <integer> ] { ( <ipv4_address> | <ipv6_address> )
        [ port <integer> ]; ... };
    view <string> <optional_class>;
    search { <string>; ... };
    ndots <integer>;
};

key <string> {
    algorithm <string>;
    secret <string>;
};

zone <string> <optional_class> {
    type ( master | slave | stub | hint | forward );
    allow-update { <address_match_element>; ... };
    file <quoted_string>;
    ixfr-base <quoted_string>; // obsolete
    ixfr-tmp-file <quoted_string>; // obsolete
    masters [ port <integer> ] { ( <ipv4_address> | <ipv6_address> ) [
        port <integer> ] [ key <string> ]; ... };
    pubkey <integer> <integer> <integer> <quoted_string>; // obsolete
    update-policy { ( grant | deny ) <string> ( name | subdomain |
        wildcard | self ) <string> <rrtypelist>; ... };
    database <string>;
    check-names <string>; // not implemented
    allow-query { <address_match_element>; ... };
    allow-transfer { <address_match_element>; ... };
    allow-update-forwarding { <address_match_element>; ... };
    allow-notify { <address_match_element>; ... };
    notify <notifytype>;
    notify-source ( <ipv4_address> | * ) [ port ( <integer> | * ) ];
    notify-source-v6 ( <ipv6_address> | * ) [ port ( <integer> | * ) ];
    also-notify [ port <integer> ] { ( <ipv4_address> | <ipv6_address>
        ) [ port <integer> ]; ... };
    dialup <dialuptype>;
    forward ( first | only );
    forwarders [ port <integer> ] { ( <ipv4_address> | <ipv6_address> )
        [ port <integer> ]; ... };
    maintain-ixfr-base <boolean>; // obsolete
    max-ixfr-log-size <size>; // obsolete
    transfer-source ( <ipv4_address> | * ) [ port ( <integer> | * ) ];
    transfer-source-v6 ( <ipv6_address> | * ) [ port ( <integer> | * ) ];
    max-transfer-time-in <integer>;
    max-transfer-time-out <integer>;
    max-transfer-idle-in <integer>;
    max-transfer-idle-out <integer>;
    max-retry-time <integer>;
    min-retry-time <integer>;
    max-refresh-time <integer>;
    min-refresh-time <integer>;
    sig-validity-interval <integer>;
};

```

```
        zone-statistics <boolean>;
};

server {
    bogus <boolean>;
    provide-ixfr <boolean>;
    request-ixfr <boolean>;
    support-ixfr <boolean>; // obsolete
    transfers <integer>;
    transfer-format ( many-answers | one-answer );
    keys <server_key>;
    edns <boolean>;
};

trusted-keys { <string> <integer> <integer> <integer> <quoted_string>; ... };
```


第 3 部分

NIS 设置和管理

本部分概述了 NIS 命名服务，还介绍了有关 Solaris OS 中 NIS 的设置、管理及疑难解答。

网络信息服务 (Network Information Service, NIS) (概述)

本章概述了网络信息服务 (Network Information Service, NIS)。

NIS 是一种分布式命名服务，提供了一种标识和定位网络对象及资源的机制。NIS 以使用传输协议且独立于介质的方式为网络范围内的信息提供统一的存储和检索方法。

本章包含以下主题：

- 第 67 页中的“NIS 介绍”
- 第 68 页中的“NIS 计算机类型”
- 第 69 页中的“NIS 元素”
- 第 75 页中的“NIS 绑定”

NIS 介绍

通过运行 NIS，系统管理员可在各种服务器（**主服务器**和**从属服务器**）中分布管理数据库，这些数据库称为**映射**。管理员可以通过一种自动而且可靠的方式从一个集中位置更新这些数据库，以确保整个网络中的所有客户机都一致共享相同的命名服务信息。

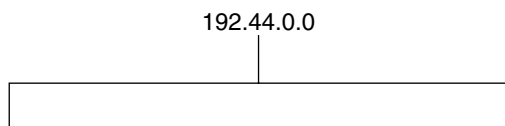
NIS 是独立于 DNS 开发的，并且其侧重点也稍有不同。DNS 侧重于使用计算机名称而不是数字 IP 地址来简化通信，而 NIS 侧重于对各种网络信息进行集中控制来更好地管理网络。NIS 不仅存储有关计算机名和地址的信息，还存储有关用户、网络本身以及网络服务的信息。这种网络**信息**的集合称为 NIS **名称空间**。

注 - 在一些上下文中，**计算机名称**是指**主机名**或**计算机名称**。本讨论中使用**计算机**，但一些屏幕消息或 NIS 映射名中可能使用**主机**或**计算机**。

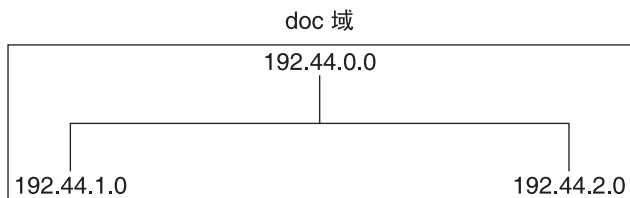
NIS 体系结构

NIS 使用客户机/服务器方案。NIS 服务器向 NIS 客户机提供服务。主要的服务器称为主服务器，并且为确保其可靠性，主服务器还具有备份，即从属服务器。主服务器和从属服务器都使用 NIS 信息检索软件，并且都存储 NIS 映射。

NIS 使用域来编排其名称空间中的计算机、用户和网络。但它并不使用域分层结构：NIS 名称空间无层次。



因此，此物理网络将被编排为一个NIS域。



NIS 域不能仅使用 NIS 直接与 Internet 连接。但是，如果组织要使用 NIS 并且希望连接到 Internet，可以将 NIS 与 DNS 结合使用。您可以使用 NIS 管理所有本地信息，使用 DNS 进行 Internet 主机查找。NIS 可提供转发服务，当 NIS 映射中找不到信息时，该服务会将主机查找转发给 DNS。Solaris 系统还允许您设置 `nsswitch.conf` 文件，指定仅将主机查找发送至 DNS；或是先发送至 DNS，当 DNS 找不到时再发送至 NIS；或是先发送至 NIS，当 NIS 找不到时再发送至 DNS。有关详细信息，请参见第 2 章，名称服务转换器（概述）。

NIS 计算机类型

NIS 计算机有三种类型。

- 主服务器
- 从属服务器
- NIS 服务器的客户机

任何计算机都可以成为NIS客户机，但只有带磁盘的计算机才能成为NIS服务器，包括主服务器或从属服务器。服务器也是客户机，通常是自身的客户机。

NIS 服务器

NIS 服务器与 NFS 文件服务器不必是同一台计算机。

NIS 服务器分为两种：主服务器和从属服务器。指定为主服务器的计算机包含系统管理员根据需要创建和更新的映射集合。每个 NIS 域必须有且仅有一台主服务器，该服务器可以传播 NIS 更新，并最大程度地减少对性能的影响。

您可以将域中附加的其他 NIS 服务器指定为从属服务器。从属服务器具有主 NIS 映射集合的完整副本。只要主服务器映射进行更新，该更新便会传播到从属服务器。从属服务器可以处理主服务器的任何请求溢出，从而最大程度地减少“服务器不可用”错误。

通常，系统管理员会为所有 NIS 映射指定一台主服务器。但是，由于每个单个 NIS 映射中都对主服务器的计算机名进行了编码，因此，可以指定不同的服务器充当不同映射的主服务器和从属服务器。为了尽量避免混淆，请指定一台单个服务器作为您在一个域中创建的所有映射的主服务器。本章的示例假设将一台服务器用作域中所有映射的主服务器。

NIS 客户机

NIS 客户机运行进程，以向服务器中的映射请求数据。由于所有 NIS 服务器都应具有相同信息，因此客户机并不区分主服务器和从属服务器。

注 – Solaris 操作系统不支持 NIS 客户机与本地 LDAP 客户机共存于同一台客户机上的配置。

NIS 元素

NIS 命名服务由以下元素组成：

- 域（请参见第 69 页中的“NIS 域”）
- 守护进程（请参见第 70 页中的“NIS 守护进程”）
- 实用程序（请参见第 70 页中的“NIS 实用程序”）
- 映射（请参见第 71 页中的“NIS 映射”）
- NIS 命令集合（请参见第 74 页中的“与 NIS 相关的命令”）

NIS 域

NIS 域是共享一组通用 NIS 映射的计算机的集合。每个域都有一个域名，共享这组通用映射的每台计算机都属于该域。

任何计算机都可以属于某个给定域，只要同一网络中存在用于该域映射的服务器即可。NIS 客户机在引导过程中获取域名并绑定到 NIS 服务器。

NIS 守护进程

NIS 服务由五个守护进程提供，如表 4-1 所示。NIS 服务由服务管理工具管理。使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

表 4-1 NIS 守护进程

守护进程	功能
<code>ypserv</code>	服务器进程
<code>ypbind</code>	绑定进程
<code>ypxfrd</code>	高速映射传送
<code>rpc.yppasswdd</code>	NIS 口令更新守护进程 **请参见下面的“注”。**
<code>rpc.yppupdated</code>	修改其他映射，如 <code>publickey</code>

注 – `rpc.yppasswdd` 将所有以 `r` 开头的 `shell` 视为受限制的 `shell`。例如，如果您位于 `/bin/rksh` 中，将不允许从该 `shell` 更改到其他 `shell` 中。如果您有以 `r` 开头的 `shell`，但不想受到此类限制，请参阅第 7 章，NIS 疑难解答了解解决方法。

NIS 实用程序

NIS 服务由九个实用程序支持，如表 4-2 所示。

表 4-2 NIS 实用程序

实用程序	功能
<code>makedbm</code>	为 NIS 映射创建 <code>dbm</code> 文件
<code>ypcat</code>	列出映射中的数据
<code>ypinit</code>	生成并安装 NIS 数据库并初始化 NIS 客户机的 <code>ybservers</code> 列表。
<code>ypmatch</code>	查找映射中的特定项
<code>yppoll</code>	从服务器中获取映射顺序编号
<code>yppush</code>	从 NIS 主服务器向 NIS 从属服务器传播数据
<code>ypset</code>	设置到特定服务器的绑定

表 4-2 NIS 实用程序 (续)

实用程序	功能
ypwhich	列出 NIS 服务器的名称以及昵称转换表
ypxfr	从主 NIS 服务器向从属 NIS 服务器传送数据

NIS 映射

NIS 映射中的信息以 ndbm 格式存储。ypfiles(4) 和 ndbm(3C) 对映射文件的这一格式进行了说明。

NIS 映射旨在替换 UNIX /etc 文件以及其他配置文件，因此其中除了存储名称和地址外还存储其他大量信息。在运行 NIS 的网络中，每个 NIS 域的 NIS 主服务器会保留一组 NIS 映射，以供域中的其他计算机查询。NIS 从属服务器也会保留主服务器映射的副本。NIS 客户机可从主服务器或从属服务器获取名称空间信息。

NIS 映射实质上是包含两个列的表。一列为**关键字**，另一列为与该关键字相关的信息。NIS 通过搜索关键字为客户机查找信息。由于每个映射使用不同的关键字，因此有些信息存储在多个映射中。例如，计算机的名称和地址存储在两个映射中：hosts.byname 和 hosts.byaddr。当服务器已知计算机的名称而需要查找其地址时，它将在 hosts.byname 映射中进行查找。当服务器已知计算机的地址而需要查找其名称时，它将在 hosts.byaddr 映射中进行查找。

NIS Makefile 存储在安装时被指定为 NIS 服务器的计算机上的 /var/yp 目录中。在该目录中运行 make 会使 makedbm 根据输入文件创建或修改缺省的 NIS 映射。

注 - 请始终在主服务器上创建映射，因为在从属服务器上创建的映射将不会自动推送到主服务器。

缺省 NIS 映射

Solaris 系统中提供了一组缺省 NIS 映射。您可能要使用所有这些映射，也可能只使用其中的部分映射。NIS 还可以使用您在安装其他软件产品时创建或添加的任何映射。

NIS 域的缺省映射位于每台服务器的 /var/yp/domainname 目录中。例如，属于域 test.com 的映射位于每台服务器的 /var/yp/test.com 目录中。

表 4-3 介绍了缺省 NIS 映射、其中包含的信息以及 NIS 运行时软件是否查看对应的管理文件。

表 4-3 NIS 映射说明

映射名	对应的 NIS 管理文件	说明
audit_user	audit_user	包含用户审计预选数据。

表 4-3 NIS 映射说明 (续)

映射名	对应的 NIS 管理文件	说明
auth_attr	auth_attr	包含授权名称和说明。
bootparams	bootparams	包含客户机在引导期间所需文件的路径名：root、swap，也可能是其他名称。
ethers.byaddr	ethers	包含计算机名和以太网地址。以太网地址是映射中的关键字。
ethers.byname	ethers	与 ethers.byaddr 相同，但关键字是计算机名而非以太网地址。
exec_attr	exec_attr	包含配置文件执行属性。
group.bygid	group	包含以组 ID 作为关键字的组安全信息。
group.byname	group	包含以组名作为关键字的组安全信息。
hosts.byaddr	hosts	包含计算机名和 IP 地址，以 IP 地址作为关键字。
hosts.byname	hosts	包含计算机名和 IP 地址，以计算机（主机）名作为关键字。
mail.aliases	aliases	包含别名和邮件地址，以别名作为关键字。
mail.byaddr	aliases	包含邮件地址和别名，以邮件地址作为关键字。
netgroup.byhost	netgroup	包含组名、用户名和计算机名。
netgroup.byuser	netgroup	与 netgroup.byhost 相同，但关键字为用户名。
netgroup	netgroup	与 netgroup.byhost 相同，但关键字为组名。
netid.byname	passwd, hosts group	用于 UNIX 形式的验证。包含计算机名和邮件地址（包括域名）。如果存在可用的 netid 文件，则除了查询通过其他文件提供的数据外，还会查询该文件。
netmasks.byaddr	netmasks	包含要与 IP 一起提交的网络掩码，以地址作为关键字。
networks.byaddr	networks	包含系统已知的网络的名称及其 IP 地址，以地址作为关键字。
networks.byname	networks	与 networks.byaddr 相同，但关键字为网络的名称。
passwd.adjunct.byname	passwd 和 shadow	包含 C2 客户机的审计信息和隐藏的口令信息。

表 4-3 NIS 映射说明 (续)

映射名	对应的 NIS 管理文件	说明
passwd.byname	passwd 和 shadow	包含以用户名作为关键字的口令信息。
passwd.byuid	passwd 和 shadow	与 passwd.byname 相同，但关键字为用户 ID。
prof_attr	prof_attr	包含执行配置文件的属性。
protocols.byname	protocols	包含网络可识别的网络协议。
protocols.bynumber	protocols	与 protocols.byname 相同，但关键字为协议编号。
rpc.bynumber	rpc	包含系统可识别的 RPC 的程序编号和名称。关键字为 RPC 程序编号。
services.byname	services	列出网络可识别的 Internet 服务。关键字为端口或协议。
services.byservice	services	列出网络可识别的 Internet 服务。关键字为服务名。
user_attr	user_attr	包含用户和角色的扩展属性。
ypservers	N/A	列出网络可识别的 NIS 服务器。

NIS 中添加了新 ipnodes 映射 (ipnodes.byaddr 和 ipnodes.byname)。这些映射可同时存储 IPv4 和 IPv6 地址。

注 – 从 Solaris 10 8/07 发行版开始，Solaris OS 不再具有两个单独的 hosts 文件。/etc/inet/hosts 文件是同时包含 IPv4 项和 IPv6 项的单个 hosts 文件。无需在始终要求同步的两个 hosts 文件中维护 IPv4 项。为了向后兼容，/etc/inet/ipnodes 文件被替换为指向 /etc/inet/hosts 文件的同名符号链接。

有关更多信息，请参见 [hosts\(4\)](#) 手册页。

NIS 客户机和服务器可以使用 IPv4 或 IPv6 RPC 传输方式进行通信。

ageing.byname 映射包含一些信息，这些信息供 yppasswdd 在实现 NIS 到 LDAP 的转换时从 DIT 中读取和向其中写入口令生命期信息。如果不使用口令生命期，可以将其从映射文件中注释掉。有关 NIS 到 LDAP 转换的更多信息，请参见第 15 章，[从 NIS 转换为 LDAP \(概述/任务\)](#)。

使用 NIS 映射

与使用 /etc 文件系统进行更新相比，NIS 可使更新网络数据库变得更加简单。您无需在每次修改网络环境时更改每台计算机中的管理 /etc 文件。

例如，向运行 NIS 的网络中添加新计算机时，只需要更新主服务器中的输入文件并运行 `make` 即可。这将自动更新 `hosts.byname` 和 `hosts.byaddr` 映射。然后，这些映射将传送给所有从属服务器，并可供域中所有客户机及其程序使用。当客户机或应用程序请求计算机名或地址时，NIS 服务器将参阅相应的 `hosts.byname` 或 `hosts.byaddr` 映射，并向该客户机发送请求的信息。

您可以使用 `ypcat` 命令显示映射中的值。`ypcat` 基本格式为：

```
% ypcat mapname
```

其中，*mapname* 是要查看的映射的名称或其**昵称**。如果映射仅由关键字组成（如 `ypservers`），请使用 `ypcat -k`。否则，`ypcat` 将列显空白行。[ypcat\(1\)](#) 手册页介绍了 `ypcat` 的更多选项。

您可以使用 `ypwhich` 命令来确定哪台服务器是特定映射的主服务器。键入以下命令。

```
% ypwhich -m mapname
```

其中，*mapname* 是要查找其主服务器的映射的名称或昵称。`ypwhich` 给出的响应是显示主服务器的名称。有关完整信息，请参阅 [ypwhich\(1\)](#) 手册页。

NIS 映射昵称

昵称是完整映射名的别名。要获得可用映射昵称（如 `passwd.byname` 的 `passwd`）的列表，请键入 `yycat -x` 或 `ypwhich -x`。

昵称存储在 `/var/yp/nicknames` 文件中，该文件中包含映射昵称，后跟映射的完全指定名称，两者之间以空格分隔。您可以对此列表进行添加或修改。目前，昵称限制在 500 个以内。

与 NIS 相关的命令

NIS 服务包括专用守护进程、系统程序和命令，下表对其进行了汇总。

表 4-4 NIS 命令汇总

命令	说明
ypserv	通过 NIS 映射为 NIS 客户机的信息请求提供服务。 <code>ypserv</code> 是在具有一整套映射的 NIS 服务器上运行的守护进程。网络中必须至少存在一个 <code>ypserv</code> 守护进程，NIS 服务才能正常运行。
ypbind	向客户机提供 NIS 服务器绑定信息。该守护进程通过在请求客户机的域内查找提供映射的 <code>ypserv</code> 进程来提供绑定。 <code>ypbind</code> 必须在所有服务器和客户机上运行。

表 4-4 NIS 命令汇总 (续)

命令	说明
ypinit	自动根据输入文件为 NIS 服务器创建映射。也用于在客户机上构造初始的 <code>/var/yp/binding/domain/ypservers</code> 文件。初次设置主 NIS 服务器和从属 NIS 服务器时请使用 <code>ypinit</code> 。
make	通过读取 <code>Makefile</code> 来更新 NIS 映射（当在 <code>/var/yp</code> 目录中运行时）。您可以使用 <code>make</code> 根据输入文件来更新所有映射或更新个别映射。 ypmake(1M) 手册页中介绍了用于 NIS 的 <code>make</code> 功能。
makedbm	<code>makedbm</code> 接收输入文件并将其转换为 <code>dbm.dir</code> 和 <code>dbm.pag</code> 文件，即 NIS 可以用作映射的有效 <code>dbm</code> 文件。使用 <code>makedbm -u</code> 还可以分解映射，从而可以了解构成它的关键字-值对。
ypxfr	使用 NIS 本身作为传输介质，将 NIS 映射从远程服务器引入到本地 <code>/var/yp/domain</code> 目录。您可以交互方式运行 <code>ypxfr</code> ，或从 <code>crontab</code> 文件中定期运行该命令。 <code>ypserv</code> 也会调用该命令以启动传送。
ypxfrd	为 <code>ypxfr</code> 请求（通常为从属服务器）提供映射传送服务。该命令仅在主服务器上运行。
yppush	将 NIS 映射的新版本从 NIS 主服务器复制到其从属服务器。该命令在主 NIS 服务器上运行。
ypset	通知 <code>ypbind</code> 进程绑定到指定的 NIS 服务器。该命令不能随意使用。出于安全原因，建议不要使用该命令。有关 <code>ypbind</code> 进程的 <code>ypset</code> 和 <code>ypsetme</code> 选项的信息，请参见 ypset(1M) 和 ypbind(1M) 手册页。
yppoll	指明在指定的服务器上运行的 NIS 映射的版本。还会列出映射的主服务器。
ypcat	显示 NIS 映射的内容。
ypmatch	列显 NIS 映射中的一个或多个指定关键字的值。您不能指定要查看的 NIS 服务器映射的版本。
ypwhich	显示客户机当前为取得 NIS 服务而正在使用的 NIS 服务器，如果调用该命令时使用了 <code>-m mapname</code> 选项，则显示作为各映射的主服务器的 NIS 服务器。如果只使用 <code>-m</code> ，则显示所有可用映射的名称及其各自的主服务器。

NIS 绑定

NIS 客户机通过绑定进程从 NIS 服务器获取信息，该进程可以采用以下两种模式之一运行：服务器列表或广播。

- 服务器列表。使用服务器列表模式时，`ypbind` 进程将在 `/var/yp/binding/domain/ypservers` 列表中查询域中所有 NIS 服务器的名称。`ypbind` 进程只绑定到此文件中的服务器。该文件通过运行 `ypinit -c` 来创建。
- 广播。`ypbind` 进程还可以使用 RPC 广播来启动绑定。因为广播只是不会进一步路由的本地子网事件，所以必须至少有一台服务器（主服务器或从属服务器）与客户机位于同一个子网中。服务器本身可能会存在于不同的子网中，因为映射传播可跨越

子网边界。在子网环境中，一种常用方法是使子网路由器成为 NIS 服务器。这样，域服务器便可为任何一个子网接口上的客户机提供服务。

服务器列表模式

在服务器列表模式下，绑定进程的工作过程如下：

1. 在 NIS 客户机上运行的、需要 NIS 映射所提供信息的任何程序向 `ypbind` 请求服务器的名称。
2. `ypbind` 在 `/var/yp/binding/domainname/ypservers` 文件中查找域中 NIS 服务器的列表。
3. `ypbind` 启动到列表中第一台服务器的绑定。如果该服务器不响应，`ypbind` 会尝试第二台，直至找到服务器或找遍整个列表。
4. `ypbind` 将要联系的服务器告知客户机进程。然后，客户机会将请求直接发送到该服务器。
5. NIS 服务器上的 `ypserv` 守护进程通过查询相应映射来处理请求。
6. `ypserv` 将请求的信息发送回客户机。

广播模式

在广播模式下，绑定进程的工作过程如下：

1. `ypbind` 启动时必须设置了广播选项 (`broadcast`)。
2. `ypbind` 发出 RPC 广播来搜索 NIS 服务器。

注 - 为了支持此类客户机，需要让每个请求 NIS 服务的子网具有 NIS 服务器。

3. `ypbind` 启动到最先对广播做出响应的服务器的绑定。
4. `ypbind` 将要联系的服务器告知客户机进程。然后，客户机会将请求直接发送到该服务器。
5. NIS 服务器上的 `ypserv` 守护进程通过查询相应映射来处理请求。
6. `ypserv` 将请求的信息发送回客户机。

通常，客户机一旦绑定到服务器，会保持与该服务器的绑定状态，直到某些原因引起状态更改为止。例如，如果服务器中断服务，它所服务的客户机将绑定到新服务器。

要确定当前正在为特定客户机提供服务的 NIS 服务器，请使用以下命令。

%ypwhich *machinename*

其中，*machinename* 是客户机的名称。如果不提供计算机名，`ypwhich` 将缺省认为是本地计算机（即运行命令时所在的计算机）。

设置和配置 NIS 服务

本章介绍网络信息服务 (Network Information Service, NIS) 的初始设置和配置。

注- 在一些上下文中，**计算机**名称是指**主机**名或**计算机**名称。本讨论中使用“计算机”，但一些屏幕消息或 NIS 映射名中可能使用**主机**或**计算机**。

本章包含以下主题：

- 第 79 页中的“配置 NIS 任务列表”
- 第 80 页中的“配置 NIS 之前的准备工作”
- 第 81 页中的“规划 NIS 域”
- 第 82 页中的“准备主服务器”
- 第 87 页中的“在主服务器上启动或停止 NIS 服务”
- 第 88 页中的“设置 NIS 从属服务器”
- 第 89 页中的“设置 NIS 客户机”

配置 NIS 任务列表

任务	有关说明，请转至
为转换准备源文件。	第 83 页中的“为向 NIS 映射的转换准备源文件”
使用 ypinit 设置主服务器	第 85 页中的“使用 ypinit 设置主服务器”
在主服务器上启动 NIS。	第 87 页中的“在主服务器上启动或停止 NIS 服务”
设置从属服务器。	第 88 页中的“设置从属服务器”
设置 NIS 客户机。	第 89 页中的“设置 NIS 客户机”

配置 NIS 之前的准备工作

在配置 NIS 名称空间之前，必须执行以下操作。

- 在所有将要使用 NIS 的计算机上安装正确配置的 `nsswitch.conf` 文件。有关详细信息，请参见第 2 章，名称服务转换器（概述）。
- 规划 NIS 域。

NIS 和服务管理工具

NIS 服务由服务管理工具管理。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

- 使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。不过，也可以从命令行使用 `ypstart` 和 `ypstop` 来启动和停止 NIS。有关更多信息，请参见 `ypstart(1M)` 和 `ypstop(1M)` 手册页。

提示 – 使用 `-t` 选项暂时禁用服务可为服务配置提供一些保护。如果禁用服务时使用了 `-t` 选项，则服务在重新引导后将恢复原始设置。如果禁用服务时未使用 `-t`，则服务在重新引导后仍将保持禁用状态。

- NIS 服务器的 NIS 故障管理资源标识符 (Fault Managed Resource Identifier, FMRI) 是 `svc:/network/nis/server:<instance>`，NIS 客户机的 NIS 故障管理资源标识符是 `svc:/network/nis/client:<instance>`。
- 使用 `svcs` 命令可查询 NIS 的状态。

- `svcs` 命令和输出示例。

```
# svcs network/nis/server
STATE      STIME      FMRI
online     Jan_10     svc:/network/nis/server:default

# svcs \*nis\*
STATE      STIME      FMRI
disabled   12:39:18   svc:/network/rpc/nisplus:default
disabled   12:39:18   svc:/network/nis/server:default
disabled   12:39:20   svc:/network/nis/passwd:default
disabled   12:39:20   svc:/network/nis/update:default
disabled   12:39:20   svc:/network/nis/xfr:default
online     12:42:16   svc:/network/nis/client:default
```

- `svcs -l` 命令和输出示例。

```
# svcs -l /network/nis/client
fmri      svc:/network/nis/client:default
enabled   true
state     online
```



```
next_state    none
restarter     svc:/system/svc/restarter:default
contract_id   99
dependency    exclude_all/none svc:/network/nis/server (offline)
dependency    require_all/none svc:/system/identity:domain (online)
dependency    require_all/restart svc:/network/rpc/bind (online)
dependency    require_all/none svc:/system/filesystem/minimal (online)
```

- 使用 `svccfg` 实用程序可获取有关服务的更多详细信息。请参见 [svccfg\(1M\)](#) 手册页。
- 使用 `ps` 命令可检查守护进程是否存在。

```
# ps -e | grep rpcbind
daemon 100806      1   0   Sep 01 ?          25:28   /usr/sbin/rpcbind
```

注 – 不要将 `-f` 选项与 `ps` 结合使用，因为此选项会尝试将用户 ID 转换为名称，而这将导致更多命名服务查找可能无法成功。

规划 NIS 域

在将计算机配置为 NIS 服务器或客户机之前，必须规划 NIS 域。

您需要决定 NIS 域中要包括哪些计算机。NIS 域不必与您的网络完全一致。一个网络可以有多个 NIS 域，并且网络中的计算机可以位于 NIS 域之外。

选择一个 NIS 域名，域名的长度可为 256 个字符。比较好的做法是将域名长度限制在 32 个字符之内。域名区分大小写。为方便起见，可以根据 Internet 域名来命名 NIS 域名。例如，如果 Internet 域名为 `doc.com`，则可将 NIS 域命名为 `doc.com`。如果要 `doc.com` 划分为两个 NIS 域，一个用于销售部门，另一个用于制造部门，则可将其中一个域命名为 `sales.doc.com`，将另一个域命名为 `manf.doc.com`。

只有设置了正确的 NIS 域名和计算机名，计算机才能使用 NIS 服务。计算机名由计算机中的 `/etc/nodename` 文件设置，计算机的域名由该计算机的 `/etc/defaultdomain` 文件设置。在引导时将读取这些文件，其内容分别由 `uname -S` 命令和 `domainname` 命令使用。无盘计算机从其引导服务器中读取这些文件。

标识 NIS 服务器和客户机

选择一台计算机作为主服务器。决定哪些计算机（如果有）将作为从属服务器。

决定哪些计算机将作为 NIS 客户机。通常情况下，您域中的所有计算机都将设置为 NIS 客户机，虽然这不是必要的。

准备主服务器

以下各节介绍如何为主服务器准备源文件和 `passwd` 文件。

源文件目录

源文件应位于主服务器上的 `/etc` 目录或其他某个目录中。将源文件存储在 `/etc` 中并不合适，因为这样映射中的内容将与主服务器上的本地文件中的内容相同。这是 `passwd` 和 `shadow` 文件的一个特殊问题，因为所有用户都可以访问主服务器映射，因而超级用户口令将通过 `passwd` 映射传递给所有 NIS 客户机。有关其他信息，请参见第 82 页中的“[Passwd 文件和名称空间安全](#)”。

但是，如果将源文件放在其他某个目录中，您必须修改 `/var/yp` 中的 `Makefile`，将 `DIR=/etc` 行更改为 `DIR=/your-choice`，其中，*your-choice* 是将用来存储源文件的目录的名称。这样便可将服务器上的本地文件视为客户机上的本地文件进行处理。（最好先保存原始 `Makefile` 的副本。）

此外，如果要从缺省目录外的其他目录中获取 `audit_user`、`auth_attr`、`exec_attr` 和 `prof_attr`，必须将 `RBACDIR=/etc/security` 修改为 `RBACDIR=/your-choice`。

Passwd 文件和名称空间安全

`passwd` 映射是一种特殊情况。除了早期的 Solaris 1 的 `passwd` 文件格式外，此 NIS 实现还接受 `/etc/passwd` 和 `/etc/shadow` 文件格式作为生成 NIS 口令映射的输入。

出于安全原因，用于生成 NIS 口令映射的文件不应包含 `root` 项，以防止未经授权的超级用户访问。因此，不应使用主服务器 `/etc` 目录中的文件生成口令映射。对于用于生成口令映射的口令文件，应删除其中的 `root` 项，并将它们放置在可免遭未经授权的访问的目录中。

例如，只要主服务器口令输入文件本身不是指向其他文件的链接，并且其位置在 `Makefile` 中已指定，就应该将这些文件存储在诸如 `/var/yp` 等目录或您选择的任何目录中。正确的目录选项会根据 `Makefile` 中指定的配置进行自动设置。



注意 – 确保 `PWDDIR` 指定的目录中的 `passwd` 文件不包含 `root` 项。

如果源文件所在的目录不是 `/etc`，则必须更改 `Makefile` 中的 `PWDIR` 口令宏，以引用 `passwd` 和 `shadow` 文件所在的目录，具体操作是将行 `PWDIR=/etc` 更改为 `PWDIR=/your-choice`，其中 *your-choice* 是要用来存储 `passwd` 映射源文件的目录的名称。

为向 NIS 映射的转换准备源文件

为向 NIS 映射的转换准备源文件。

▼ 如何为转换准备源文件

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 检查主服务器上的源文件，确保它们可以反映系统的最新情况。

检查以下文件：

- auto.home 或 auto_home
- auto.master 或 auto_master
- audit_user
- auth_attr
- bootparams
- ethers
- exec_attr
- group
- hosts
- ipnodes
- netgroup
- netmasks
- networks
- passwd
- protocols
- rpc
- service
- shadow
- user_attr

3 将上述除 passwd 外的所有源文件复制到选定的 DIR 目录中。

4 将 passwd 文件复制到选定的 PWDIR 目录中。

5 将 audit_user、auth_attr、exec_attr 以及 prof_attr 复制到选定的 RBACDIR 目录中。

6 检查 /etc/mail/aliases 文件。

与其他源文件不同，/etc/mail/aliases 文件不能移至其他目录。此文件必须位于 /etc/mail 目录中。有关更多信息，请参阅 [aliases\(4\)](#)。

注 – 通过将 `/var/yp/Makefile` 中的 `ALIASES = /etc/mail/aliases` 项指向其他位置，可以添加特定于 `nis` 的邮件别名文件。然后当您运行 `make` 时，`ALIASES` 项将创建一个 `mail.aliases` 映射。当 `/etc/nsswitch.conf` 文件正确地将除 `files` 之外的 `nis` 作为目标时，`sendmail` 服务除了使用 `/etc/mail/aliases` 文件，还使用此映射。请参阅第 98 页中的“修改和使用 `Makefile`”。

7 清除源文件中的所有注释以及其他多余的行和信息。

这些操作可以通过 `sed` 或 `awk` 脚本或文本编辑器来完成。`Makefile` 可自动执行一些文件清理工作，但最好在运行之前手动检查并清理这些文件。

8 确保所有源文件中的数据都具有正确的格式。

对于该特定文件，源文件数据的格式必须是正确的。请检查各文件对应的手册页，以确保每个文件都具有正确格式。

准备 Makefile

检查源文件并将其复制到源文件目录之后，您现在需要将这些源文件转换为 NIS 服务使用的 `ndbm` 格式的映射。在主服务器上调用 `ypinit` 时，它会自动执行此操作，如第 85 页中的“使用 `ypinit` 设置主服务器”中所述。

`ypinit` 脚本将调用 `make` 程序，该程序将使用 `/var/yp` 目录中的 `Makefile`。`/var/yp` 目录中提供了缺省的 `Makefile`，该文件包含将源文件转换为期望的 `ndbm` 格式映射所需要的命令。

您可以按原样使用缺省的 `Makefile`，如果需要，也可以对其进行修改。（如果确实要修改缺省的 `Makefile`，请确保先复制并存储原始的缺省 `Makefile`，以便将来需要时使用。）您可能需要对 `Makefile` 进行以下一项或多项修改：

■ 非缺省映射

如果创建了自己的非缺省源文件并想将其转换为 NIS 映射，必须将这些源文件添加到 `Makefile`。

■ DIR 值

如果希望 `Makefile` 使用 `/etc` 目录以外的其他目录中存储的源文件（如第 82 页中的“源文件目录”中所述），必须将 `Makefile` 中的 `DIR` 的值更改为希望使用的目录。更改 `Makefile` 中的该值时，请勿使行缩进。

■ PWDIR 值

如果希望 `Makefile` 使用 `/etc` 目录以外的其他目录中存储的 `passwd`、`shadow` 和/或 `adjunct` 源文件，必须将 `Makefile` 中 `PWDIR` 的值更改为希望使用的目录。更改 `Makefile` 中的该值时，请勿使行缩进。

■ 域名解析程序

如果希望 NIS 服务器对不在当前域中的计算机使用域名解析程序，请注释掉 Makefile 行 `B=`，并取消对行 `B=-b` 的注释（激活）。

Makefile 的功能是为列在 `all` 下的每个数据库创建相应的 NIS 映射。数据在通过 `makedbm` 传递之后，收集在 `mapname.dir` 和 `mapname.pag` 两个文件中。这两个文件都位于主服务器上的 `/var/yp/domainname` 目录中。

Makefile 根据需从 `/PWDIR/passwd`、`/PWDIR/shadow` 和 `/PWDIR/security/passwd.adjunct` 文件生成 `passwd` 映射。

使用 `ypinit` 设置主服务器

`ypinit` 脚本用于设置主服务器和从属服务器以及客户机，以使用 NIS。它还会首先运行 `make`，在主服务器上创建映射。

要使用 `ypinit` 在主服务器上生成一组新的 NIS 映射，请执行以下操作。

▼ 如何使用 `ypinit` 设置主服务器

- 1 在主服务器上，成为超级用户或承担等效角色。
角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。
- 2 将 `nsswitch.files` 文件的内容复制到 `nsswitch.conf` 文件中。

```
# cp /etc/nsswitch.files /etc/nsswitch.conf
```
- 3 编辑 `/etc/hosts` 或 `/etc/inet/ipnodes` 文件，添加每台 NIS 服务器的名称和 IP 地址。
- 4 在主服务器上生成新的映射。

```
# /usr/sbin/ypinit -m
```
- 5 当 `ypinit` 提示输入要成为 NIS 从属服务器的其他计算机的列表时，键入正在使用的服务器的名称以及 NIS 从属服务器的名称。
- 6 当 `ypinit` 询问是希望在出现第一个非致命错误时终止过程，还是即使出现非致命错误也仍然继续时，请键入 `y`。
如果选择 `y`，`ypinit` 将在遇到第一个问题时退出，然后您可以修复问题并重新启动 `ypinit`。建议在初次运行 `ypinit` 时这样做。如果希望继续操作，可尝试手动修复出现的所有问题，然后重新启动 `ypinit`。

注 – 当某些映射文件不存在时，会出现非致命错误。此错误不会影响NIS的功能。如果未自动创建这些映射，可能需要手动添加。有关所有缺省NIS映射的说明，请参阅第 71 页中的“缺省NIS映射”。

- 7 **ypinit** 将询问是否可以销毁 `/var/yp/domainname` 目录中现有的文件。

仅当之前已安装NIS时才会显示此消息。

- 8 **ypinit** 构造服务器列表之后，将调用 **make**。

此程序将使用 `/var/yp` 中的 **Makefile**（缺省或修改过的文件）中包含的说明。**make** 命令将清除指定的文件中剩余的所有注释行。它还会对这些文件运行 **makedbm**，创建适当映射并为每个映射建立主服务器的名称。

如果 **Makefile** 推送的映射所对应的域不是主服务器上的 **domainname** 命令返回的域，则可以使用正确的 **DOM** 变量标识启动 **ypinit shell** 脚本中的 **make** 来确保这些映射被推送到正确的域，如下所示：

```
# make DOM=domainname password
```

此命令会将 **password** 映射推送到目标域，而不是主服务器所属的域。

- 9 要启用NIS作为命名服务，请键入以下命令。

```
# cp /etc/nsswitch.nis /etc/nsswitch.conf
```

此命令会将当前的转换器文件替换为缺省的面向NIS的转换器文件。您可以根据需要编辑此文件。

支持多个NIS域的主服务器

通常，一台NIS主服务器只支持一个NIS域。但是，如果要使用一台主服务器来支持多个域，则在设置服务器为更多域提供服务时，必须对第 85 页中的“使用 **ypinit** 设置主服务器”中所述的步骤稍作修改。

在服务器上运行 **domainname** 命令。该命令返回的域名是服务器的缺省域。使用第 85 页中的“使用 **ypinit** 设置主服务器”中所述的步骤可以顺利为该域设置服务。但是要为其其他任何域配置服务，必须按如下所示修改 **ypinit shell** 脚本。

```
# make DOM=correct-domain passwd
```

correct-domain 是要为其设置服务的其他域的名称，**passwd** 是 **make** 目标。此命令会将 **passwd** 映射推送到目标域，而不是主服务器所属的域。

在主服务器上启动或停止 NIS 服务

创建主服务器映射后，可以在主服务器上启动 NIS 守护进程，并启动服务。启用 NIS 服务时，将在服务器上启动 `ypserv` 和 `ypbind`。当客户机向服务器请求信息时，`ypserv` 守护进程将在 NIS 映射中进行查找，然后再回答来自客户机的信息请求。`ypserv` 和 `ypbind` 守护进程作为一个单元来管理。

在服务器中启动或停止 NIS 服务的方法有三种：

- 在引导过程中自动调用 `/usr/lib/netsvc/yp/ypstart` 脚本
- 从命令行使用服务管理工具的 `svcadm enable <fmri>` 和 `svcadm disable <fmri>` 命令有关 SMF 的更多信息，请参见 [svcadm\(1M\)](#)。
- 从命令行使用 [ypstart\(1M\)](#) 和 [ypstop\(1M\)](#) 命令

自动启动 NIS 服务

在运行 `ypinit` 配置了 NIS 主服务器后，将自动调用 `ypstart`，以便在计算机引导时启动 `ypserv`。请参见第 85 页中的“使用 `ypinit` 设置主服务器”。

从命令行启动和停止 NIS

使用服务管理工具 `svcadm` 命令或 `ypstart/ypstop` 命令可从命令行启动和停止 NIS。使用 `svcadm` 时，仅当运行服务的多个实例时才需要实例名称。有关更多信息，请参见第 80 页中的“NIS 和服务管理工具”，或参见 [svcadm\(1M\)](#)、[ypstart\(1M\)](#) 和 [ypstop\(1M\)](#) 手册页。

要从命令行启动 NIS 服务，请运行 `svcadm enable` 命令或 `ypstart` 命令。

```
# svcadm enable network/nis/server:<instance>
# svcadm enable network/nis/client:<instance>
or
# ypstart
```

注 – 因为在启动后 `ypserv` 准备好响应调用之前会有一小段延迟，所以从程序或脚本内部调用 `svcadm` 时，应该在它之后发出一个三到五秒的休眠。

要停止 NIS 服务，请运行 `svcadm disable` 命令或 `ypstop`。

```
# svcadm disable network/nis/server:<instance>
# svcadm disable network/nis/client:<instance>
or
# ypstop
```

要停止并立即重新启动 NIS 服务，请使用 `svcadm restart` 命令。

```
# svcadm restart network/nis/server:<instance>
# svcadm restart network/nis/client:<instance>
```

设置 NIS 从属服务器

您的网络可以有一台或多台从属服务器。使用从属服务器可在主服务器不可用时确保 NIS 服务的连续性。

准备从属服务器

在实际运行 `ypinit` 创建从属服务器之前，应在每台 NIS 从属服务器上运行 `domainname` 命令，以确保域名与主服务器一致。

注 - 域名区分大小写。

配置 NIS 从属服务器之前，请确保网络工作正常。需要特别指出的是，应检查并确保可以使用 `rpc` 从 NIS 主服务器向 NIS 从属服务器发送文件。

设置从属服务器

以下过程说明了如何设置从属服务器。

▼ 如何设置从属服务器

- 1 成为超级用户或承担等效角色。
角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。
- 2 编辑从属服务器上的 `/etc/hosts` 或 `/etc/inet/ipnodes` 文件，添加所有其他 NIS 服务器的名称和 IP 地址。
- 3 在从属服务器上，将目录更改为 `/var/yp`。

注 - 必须先将新的从属服务器配置为 NIS 客户机，它才能首次从主服务器中获取 NIS 映射。有关详细信息，请参见第 89 页中的“设置 NIS 客户机”。

- 4 将从属服务器初始化为客户机。

```
# /usr/sbin/ypinit -c
```


ypinit 命令会提示输入 NIS 服务器的列表。请先输入您正在使用的本地从属服务器的名称，然后输入主服务器的名称，随后按照从物理上最近到最远（从网络角度看）的顺序输入域中其他 NIS 从属服务器的名称。

5 确定 NIS 客户机是否正在运行，然后根据需要启动客户机服务。

```
# svcs network/nis/client
STATE          STIME          FMRI
online          20:32:56      svc:/network/nis/client:default
```

如果 svc:/network/nis/client 显示为 online（联机）状态，则表明 NIS 正在运行。如果该服务的状态为 "disabled"（已禁用），则表明 NIS 未运行。

a. 如果 NFS 客户机正在运行，请重新启动客户机服务。

```
# svcadm restart network/nis/client
```

b. 如果 NFS 客户机未运行，请启动客户机服务。

```
# svcadm enable network/nis/client
```

6 将此计算机初始化为从属服务器。

```
# /usr/sbin/ypinit -s master
```

其中，*master* 是现有的 NIS 主服务器的计算机名。

对要配置为 NIS 从属服务器的每台计算机重复本节所述的过程。

▼ 如何在从属服务器上启动 NIS

以下过程说明了如何在从属服务器上启动 NIS。

● 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

设置 NIS 客户机

下面介绍两种配置客户机使用 NIS 作为其命名服务的方法。

注 – Solaris 操作系统不支持 NIS 客户机与本地 LDAP 客户机共存于同一台客户机上的配置。

- ypinit。配置客户机使用 NIS 的推荐方法是以 root 身份登录计算机并运行 ypinit -C。

```
# ypinit -c
```

运行此命令时，将要求您列举出客户机要从中获取命名服务信息的 NIS 服务器。您可以列出尽量多的主服务器或从属服务器。列出的服务器可以位于域中的任意位置。最好先列出离计算机最近的服务器（从网络角度来说），然后再列出网络中处于更远距离的服务器。

- **广播模式。**配置客户机使用 NIS 的一种较旧方法是以 root 身份登录计算机，使用 domainname 命令设置域名，然后运行 ypbind。

如果 /var/yp/binding/'domainname'/ypservers 文件不存在，ypstart 将自动以广播模式 (ypbind -broadcast) 调用 NIS 客户机。

```
# domainname doc.com
# mv /var/yp/binding/'domainname'/ypservers /var/yp/binding/'domainname'\
/ypservers.bak
# ypstart
```

运行 ypbind 时，它将在本地子网中搜索 NIS 服务器。如果找到服务器，ypbind 将绑定到该服务器。这种搜索被称为**广播**。如果客户机的本地子网中没有 NIS 服务器，ypbind 将无法绑定，并且客户机无法从 NIS 服务获取名称空间数据。

注 – 出于安全和管理控制的原因，最好在客户机的 ypservers 文件中为客户机指定要绑定到的服务器，而不是让客户机通过广播搜索服务器。广播会降低网络和客户机的速度，并会使您无法通过为不同客户机列出不同服务器来平衡服务器负载。

管理 NIS（任务）

本章介绍了如何管理 NIS。本章包含以下主题：

- 第 91 页中的“口令文件和名称空间安全”
- 第 92 页中的“管理 NIS 用户”
- 第 95 页中的“使用 NIS 映射”
- 第 100 页中的“更新和修改现有映射”
- 第 105 页中的“添加从属服务器”
- 第 106 页中的“使用启用 C2 安全性的 NIS”
- 第 107 页中的“更改计算机的 NIS 域”
- 第 108 页中的“将 NIS 与 DNS 结合使用”
- 第 109 页中的“禁用 NIS 服务”

注 - NIS 服务由服务管理工具管理。使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。有关将 SMF 与 NIS 结合使用的更多信息，请参见第 80 页中的“NIS 和服务管理工具”。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

还可以使用 `ypstart` 和 `ypstop` 命令来启动和停止 NIS 服务。有关更多信息，请参见 `ypstart(1M)` 和 `ypstop(1M)` 手册页。

口令文件和名称空间安全

出于安全原因，请遵循以下原则。

- 最好限制对主服务器上的 NIS 映射的访问。
- 用于生成 NIS 口令映射的文件不应包含 `root` 项，目的是防止未经授权的访问。为此，应移除用于生成口令映射的口令文件中的 `root` 项，并将该文件移至不同于主服务器 `/etc` 目录的其他目录中。应确保此目录不会受到未经授权的访问。

例如，只要主服务器口令输入文件本身不是指向其他文件的链接并且在 Makefile 中已指定，就可将这些文件存储在诸如 `/var/yp` 等目录或您选择的任何目录中。使用服务管理工具或 `ypstart` 脚本启动 NIS 服务时，将根据 Makefile 中指定的配置设置正确的目录选项。

注 – 除了早期的 Solaris 1 版本的 `passwd` 文件格式外，此 NIS 实现还接受 Solaris 2 的 `passwd` 和 `shadow` 文件格式作为生成 NIS 口令映射的输入。

管理 NIS 用户

本节包括有关设置用户口令、向 NIS 域添加新用户以及将用户指定给 `netgroups` 的信息。

▼ 如何向 NIS 域添加新 NIS 用户

- 1 在主 NIS 服务器上，成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

- 2 使用 `useradd` 命令创建新用户的登录 ID。

```
# useradd userID
```

`userID` 是新用户的登录 ID。此命令将在主 NIS 服务器上的 `/etc/passwd` 和 `/etc/shadow` 文件中创建项。

- 3 创建新用户的初始口令。

要创建新用户可用来登录的初始口令，请运行 `passwd` 命令。

```
# passwd userID
```

其中，`userID` 是新用户的登录 ID。系统将提示您输入要指定给此用户的口令。

由于 `useradd` 命令创建的口令项已锁定（这意味着新用户无法登录），因此必须执行此步骤。通过指定初始口令，可以解除对该项的锁定。

- 4 如有必要，可将新项复制到服务器的 `passwd` 映射输入文件中。

主服务器上的映射源文件应位于 `/etc` 之外的某个目录中。将新行从 `/etc/passwd` 和 `/etc/shadow` 文件复制并粘贴到服务器上的 `passwd` 映射输入文件中。有关其他信息，请参见第 91 页中的“口令文件和名称空间安全”。

例如，如果您添加了新用户 `brown`，则要从 `/etc/passwd` 复制到 `passwd` 输入文件中的行将如下所示：

```
brown:x:123:10:User brown:/home/brown:/bin/csh:
```

要从 `/etc/shadow` 中复制的有关 `brown` 的行将如下所示：

```
brown:W12345GkHic:6445:::~:
```

5 确保 Makefile 正确指定了口令输入文件所在的目录。

6 如果合适，请从 `/etc/passwd` 和 `/etc/shadow` 输入文件中删除新用户的项。

出于安全原因，请不要在 NIS 主服务器的 `/etc/passwd` 和 `/etc/shadow` 文件中保留用户项。在将新用户的项复制到其他某个目录中存储的 NIS 映射源文件后，请在主服务器上使用 `userdel` 命令删除新用户。

例如，要从主服务器的 `/etc` 文件中删除新用户 `brown`，可以输入以下命令。

```
# userdel brown
```

有关 `userdel` 的更多信息，请参见 `userdel` 手册页。

7 更新 NIS passwd 映射。

在更新主服务器上的 `passwd` 输入文件后，请在包含源文件的目录中运行 `make`，以更新 `passwd` 映射。

```
# userdel brown
# cd /var/yp
# /usr/ccs/bin/make passwd
```

8 通知新用户已为其登录 ID 指定的初始口令。

登录后，新用户随时可以通过运行 `passwd` 设置不同口令。

设置用户口令

用户可以运行 `passwd` 更改自己的口令。

```
% passwd username
```

您必须先主服务器上启动 `rpc.yppasswdd` 守护进程更新口令文件，然后用户才能更改口令。

`rpc.yppasswdd` 守护进程会在主服务器上自动启动。请注意，如果为 `rpc.yppasswdd` 指定了 `-m` 选项，则在 `/var/yp` 中修改文件后将立即强制执行 `make`。如果要避免在每次更改 `passwd` 文件后都执行此 `make`，请在 `ypstart` 脚本的 `rpc.yppasswd` 命令中删除 `-m` 选项，并通过 `crontab` 文件控制 `passwd` 映射的推送。

注 - `rpc.yppasswd -m` 命令后不应包含任何参数。尽管可以通过编辑 `ypstart` 脚本文件实现不同操作，但除了可以选择删除 `-m` 选项外，建议不要修改此文件。此文件调用的所有命令和守护进程都具有适当的命令行参数集合。如果选择编辑此文件，在编辑 `rpc.yppasswd` 命令时应特别小心。如果您添加了对 `passwd.adjunct` 文件的显式调用，则必须使用路径 `$PWDIR/security/passwd.adjunct`；否则，处理结果将不正确。

NIS 网络组

NIS 网络组是您为实现管理目的而定义的用户或计算机组（集合）。例如，您可以创建网络组来执行以下任务。

- 定义一组可以访问特定计算机的用户
- 定义一组要被授予特定文件系统访问权限的 NFS 客户机
- 定义一组要对特定 NIS 域中的所有计算机具有管理员权限的用户

每个网络组都有一个网络组名。网络组不直接设置权限或访问权限。而是由其他 NIS 映射在通常使用用户名或计算机名的地方使用网络组名。例如，假设您创建了一个由网络管理员构成的网络组，名为 `netadmins`。要向 `netadmins` 组的所有成员授予对给定计算机的访问权限，只需向该计算机的 `/etc/passwd` 文件中添加一个 `netadmin` 项即可。网络组名也可以添加到 `/etc/netgroup` 文件中并传播到 NIS `netgroup` 映射。有关使用网络组的更多详细信息，请参见 [netgroup\(4\)](#)。

在使用 NIS 的网络中，主 NIS 服务器上的 `netgroup` 输入文件用于生成三种映射：`netgroup`、`netgroup.byuser` 和 `netgroup.byhost`。`netgroup` 映射包含 `netgroup` 输入文件中的基本信息。另外两种 NIS 映射中包含的信息的格式可在给定计算机或用户的情况下加速网络组信息的查找。

`netgroup` 输入文件中的项格式如下：*name ID*，其中 *name* 是为网络组给定的名称，而 *ID* 用于标识属于该网络组的计算机或用户。您可以根据需要为网络组指定任意多的 ID（成员），ID 之间以逗号分隔。例如，要创建一个具有三个成员的网络组，`netgroup` 输入文件项将采取以下格式：*name ID, ID, ID*。`netgroup` 输入文件项中的成员 ID 采取以下格式。

```
([-|machine], [-|user], [domain])
```

其中，*machine* 是计算机名，*user* 是用户 ID，*domain* 是计算机或用户的 NIS 域。*domain* 元素是可选的，并且仅应用于标识其他某个 NIS 域中的计算机或用户。每个成员项的 *machine* 和 *user* 元素是必需的，但连字符 (-) 用来表示内容为空。项中的计算机和用户元素之间不存在必然联系。

下面是两个 `netgroup` 输入文件项样例，每个样例都创建了一个名为 `admins` 的网络组，网络组由用户 `hauri` 和 `juanita`（后者在远程域 `sales` 中）以及计算机 `altair` 和 `sirius` 组成。

```
admins (altair, hauri), (sirius,juanita,sales)
```

```
admins (altair,-), (sirius,-), (-,hauri), (-,juanita,sales)
```

各种程序会在登录、远程挂载、远程登录以及远程创建 shell 期间使用 NIS 映射进行权限检查。这些程序包括 `mountd`、`login`、`rlogin` 和 `rsh`。`login` 命令在 `passwd` 数据库中遇到网络组名时，会在网络组映射中查询用户分类。`mountd` 守护进程在 `/etc/dfs/dfstab` 文件中遇到网络组名时，会在网络组映射中查询计算机分类。`rlogin` 和 `rsh`（实际上，任何使用 `ruserok` 接口的程序）在 `/etc/hosts.equiv` 或 `.rhosts` 文件中遇到网络组名时，会在网络组映射中同时查询计算机和用户分类。

如果您向网络中添加新 NIS 用户或计算机，请确保在 `netgroup` 输入文件中将其添加到适当的网络组中。然后使用 `make` 和 `yppush` 命令创建网络组映射，再将其推送到所有 NIS 服务器。有关使用网络组和网络组输入文件语法的详细信息，请参见 [netgroup\(4\)](#)。

使用 NIS 映射

本节包含以下信息：

- [第 95 页中的“获取映射信息”](#)
- [第 96 页中的“更改映射的主服务器”](#)
- [第 97 页中的“修改配置文件”](#)
- [第 98 页中的“修改和使用 Makefile”](#)

获取映射信息

用户可以使用 `ypcat`、`ypwhich` 和 `ypmatch` 命令随时从映射中获取有关映射的信息。在下面的示例中，`mapname` 同时指映射的正式名称和昵称（如果有）。

要列出映射中的所有值，请键入以下命令。

```
% ypcat mapname
```

要同时列出映射中的关键字和值（如果有），请键入以下命令。

```
% ypcat -k mapname
```

要列出所有映射昵称，请键入以下任何命令。

```
% ypcat -x
% ypmatch -x
% ypwhich -x
```

要列出所有可用映射及其主服务器，请键入以下命令。

```
% ypwhich -m
```

要列出特定映射的主服务器，请键入以下命令。

```
% ypwhich -m mapname
```

要以映射中的项匹配关键字，请键入以下命令。

```
% ypmatch key mapname
```

如果要查找的项不是映射中的关键字，请键入以下命令。

```
% ypcat mapname | grep item
```

其中，*item* 是要搜索的信息。要获取有关其他域的信息，请使用这些命令的 `-d domainname` 选项。

如果请求缺省域之外的域信息的计算机没有到被请求域的绑定，`ypbind` 将在 `/var/yp/binding/domainname/ypservers` 文件中查询该域的服务器列表。如果此文件不存在，该命令将发出服务器的 RPC 广播。在这种情况下，被请求域必须有一台服务器与请求计算机位于同一子网。

更改映射的主服务器

要更改选定映射的主服务器，必须先在新 NIS 主服务器上生成该映射。由于旧的主服务器名称以关键字-值对的形式出现在现有映射中（此对由 `makedbm` 自动插入），因此将该映射复制到新的主服务器或使用 `ypxfr` 将副本传送到新的主服务器是不够的。您必须将该关键字与新主服务器名重新关联。如果映射具有 ASCII 源文件，应将此文件复制到新的主服务器。

▼ 如何更改映射的主服务器

- 1 在新的主服务器上，成为超级用户或承担等效角色。
角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。
- 2 更改目录。

```
newmaster# cd /var/yp
```
- 3 **Makefile** 必须具有新映射的项，才能指定要进行的映射。否则，请使用名为 `sites.byname` 的映射立即编辑 **Makefile**。
- 4 要更新映射或重新进行映射，请键入以下命令。

```
newmaster# make sites.byname
```
- 5 如果旧的主服务器仍为 NIS 服务器，请远程登录 (`rlogin`) 到该旧的主服务器并编辑 **Makefile**。确保注释掉 **Makefile** 中创建 `sites.byname` 的部分，使其不再被创建。

- 6 如果 `sites.byname` 只作为 `ndbm` 文件存在，请新的主服务器上重新创建它，方法是对来自任何 NIS 服务器的该文件的副本进行反汇编，然后通过 `makedbm` 运行反汇编的版本。

```
newmaster# cd /var/yp
newmaster# ypcat sites.byname | makedbm -domain-/sites.byname
```

在新的主服务器上创建映射后，必须向其他从属服务器发送新映射的副本。不要使用 `yppush`，因为其他从属服务器会试图从旧的主服务器（而不是新的主服务器）中获取新副本。解决此问题的一个典型方法是从新的主服务器向旧的主服务器传送一份映射的副本。为此，请在旧的主服务器上成为超级用户或承担等效角色，并键入以下命令。

```
oldmaster# /usr/lib/netstvc/yp/ypxfr -h newmaster sites.byname
```

现在，即可安全运行 `yppush`。其余所有的从属服务器仍会认为旧的主服务器是当前的主服务器，并尝试从旧的主服务器中获取最新版本的映射。当客户机这样做时，它们将获得新得映射，该映射会将新的主服务器指定为当前主服务器。

如果此方法失败，可以使用超级用户身份登录每台 NIS 服务器并执行上面所示的 `ypxfr` 命令。

修改配置文件

NIS 可以智能解析设置文件。尽管这样可以简化 NIS 管理，但它使 NIS 的行为对设置和配置文件中的更改更敏感。

进行以下任何一项修改时，请使用本节中的过程。

- 修改 `/var/yp/Makefile` 以添加或删除支持的映射
- 添加或删除 `/etc/resolv.conf`，以允许或拒绝 DNS 转发
- 添加或删除 `$PWDIR/security/passwd.adjunct`，以允许或拒绝 C2 安全性（`$PWDIR` 在 `/var/yp/Makefile` 中定义）

▼ 如何修改配置文件

更改 NIS 映射或映射源文件时，无需停止和启动 NIS。

请牢记以下几点。

- 在 NIS 主服务器中删除映射或源文件不会自动在从属服务器中执行同样的删除操作。您必须手动删除从属服务器中的映射和源文件。
- 新映射不会自动推送到现有的从属服务器。你必须在从属服务器中运行 `ypxfr`。

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 停止 NIS 服务器。

```
# svcadm disable network/nis/server
```

3 对文件进行必要的更改。

4 启动 NIS 服务器。

```
# svcadm enable network/nis/server
```

修改和使用 Makefile

您可以修改 `/var/yp` 中缺省提供的 `Makefile` 来满足自己的需要。您可以添加或删除映射，也可以更改某些目录的名称。

提示 – 请保留原始的 `Makefile` 的未修改副本，以供将来参考。

使用 Makefile

要添加新的 NIS 映射，必须将该映射的 `ndbm` 文件的副本放入域中每台 NIS 服务器上的 `/var/yp/domainname` 目录中。此操作通常由 `Makefile` 执行。在决定好将哪台 NIS 服务器用作映射的主服务器之后，请修改主服务器上的 `Makefile`，以便可以方便地重新生成映射。不同的服务器可以作为不同映射的主服务器，但在大多数情况下，这会导致管理上的混乱。请尽量只将一台服务器设置为所有映射的主服务器。

通常，会使用 `awk`、`sed` 或 `grep` 对适合人阅读的文本文件进行过滤，以使其适合输入到 `makedbm`。有关示例，请参阅缺省的 `Makefile`。有关 `make` 命令的一般信息，请参见 [make\(1S\)](#)。

在决定如何创建 `make` 可识别的相关性时，请使用 `Makefile` 中已经存在的机制。请注意，`make` 对于相关性规则中的行首是否存在制表符非常敏感。缺少制表符会使本来格式正确的项无效。

向 `Makefile` 中添加项涉及以下步骤。

- 向 `all` 规则中添加数据库名称
- 编写 `time` 规则
- 为数据库添加规则

例如，为使 `Makefile` 可以处理自动挂载程序输入文件，必须将 `auto_direct.time` 和 `auto_home.time` 映射添加到 NIS 数据库。

要将这些映射添加到 NIS 数据库，需要修改 `Makefile`。

更改 Makefile 宏/变量

通过更改等号 (=) 右侧的值，可以更改 Makefile 顶部定义的变量的设置。例如，如果不想使用 /etc 中的文件作为映射的输入，而想使用另一个目录（如 /var/etc/domainname）中的文件，应将 DIR 由 DIR=/etc 更改为 DIR=/var/etc/domainname。另外，还应将 PWDIR 由 PWDIR=/etc 更改为 PWDIR=/var/etc/domainname。

这些变量如下所示。

- DIR= 包含除 passwd 和 shadow 之外的所有 NIS 输入文件的目录。缺省值为 /etc。由于使用主服务器上 /etc 目录中的文件作为 NIS 输入文件并不是一种很好的做法，因此应更改此值。
- PWDIR= 包含 passwd 和 shadow NIS 输入文件的目录。由于使用主服务器上 /etc 目录中的文件作为 NIS 输入文件并不是一种很好的做法，因此应更改此值。
- DOM= NIS 域名。DOM 的缺省值使用 domainname 命令来设置。但是，大多数 NIS 命令都使用当前计算机的 /etc/defaultdomain 文件中设置的域。

修改 Makefile 项

以下过程介绍如何在 Makefile 中添加和删除数据库。

▼ 如何修改 Makefile 以使用特定数据库

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 修改以单词 all 开头的行，添加要添加的数据库的名称：

```
all: passwd group hosts ethers networks rpc services protocols \
    netgroup bootparams aliases netid netmasks \
    audit_user auth_attr exec_attr prof_attr \
    auto_direct auto_home auto_direct.time auto_home.time
```

各项的顺序不相关，但连续行开头的空白处必须为制表符，而不是空格。

3 在 Makefile 结尾添加以下行：

```
auto_direct: auto_direct.time
auto_home: auto_home.time
```

4 在该文件中间添加 auto_direct.time 项。

```
auto_direct.time: $(DIR)/auto_direct
@ (while read L; do echo $$L; done < $(DIR)/auto_direct
$(CHKPIPE)) | \ (sed -e "/^#/d" -e "s/#.*$$//" -e "/^ *$$/d"
$(CHKPIPE)) | \ $(MAKEDBM) - $(YPDBDIR)/$(DOM)/auto_direct;
```

```
@touch auto_direct.time;
@echo "updated auto_direct";
@if [ ! $(NOPUSH) ]; then $(YPPUSH) auto_direct; fi
@if [ ! $(NOPUSH) ]; then echo "pushed auto_direct"; fi
```

其中，

- CHKPIPE 确保在将结果传输给后面的命令之前管道符号 (|) 左侧的运算已成功完成。如果管道符号左侧的运算未成功完成，该进程将中止，并显示 "NIS make terminated"（NIS make 中止）消息。
- NOPUSH 阻止 makefile 调用 yppush 将新映射传送到从属服务器。如果不设置 NOPUSH，将自动完成推送。

开头的 while 循环旨在消除输入文件中的所有反斜杠扩展行。sed 脚本用于消除注释和空行。

对于其他所有自动挂载程序映射（如 auto_home）或其他任何非缺省映射，应按相同的过程操作。

5 运行 make。

```
# make mapname
```

其中，*mapname* 是要创建的映射的名称。

▼ 如何修改 Makefile 以删除数据库

如果不希望 Makefile 为特定数据库生成映射，请按如下步骤编辑 Makefile。

- 1 从 all 规则中删除数据库的名称。
- 2 为要删除的数据库删除或注释掉数据库规则。
例如，要删除 hosts 数据库，应删除 hosts.time 项。
- 3 删除 time 规则。
例如，要删除 hosts 数据库，应删除 hosts: hosts.time 项。
- 4 从主服务器和从属服务器中删除映射。

更新和修改现有映射

安装 NIS 之后，您可能会发现，有些映射需要频繁更新，而其他映射则从来不需要更改。例如，在大公司的网络中，passwd.byname 映射可能会频繁更改，而 auto_master 映射则只需少量更改，甚至不进行任何更改。

如第 71 页中的“缺省 NIS 映射”中所述，缺省 NIS 映射的缺省位置是主服务器上的 /var/yp/domainname，其中 domainname 是 NIS 域的名称。当您需要更新某个映射时，可以根据该映射是否为缺省映射来使用两个更新过程之一。

- 缺省映射是 `ypinit` 从网络数据库中创建的缺省集合中的映射。
- 非缺省映射可以是以下三种类型之一。
 - 从供应商处购买的应用程序中随附的映射
 - 专门为您的站点创建的映射
 - 根据非文本文件创建的映射

以下各部分介绍如何使用各种更新工具。在实践中，您可能会决定只在系统已启动并运行后添加非缺省映射或更改 NIS 服务器集合时才使用这些工具。

▼ 如何更新随缺省集合提供的映射

使用以下过程可以更新随缺省集合提供的映射。

- 1 成为主服务器上的超级用户。
请始终只在主服务器上修改 NIS 映射。
- 2 编辑要更改的映射的源文件，无论该文件位于 `/etc` 还是您选择的其他某个目录中。
- 3 键入以下命令。

```
# cd /var/yp
# make mapname
```

然后，`make` 命令将根据您在映射对应的文件中所做的更改来更新映射。此命令还会在其他服务器中传播这些更改。

维护已更新映射

以下各节介绍完成更新随缺省集合提供的映射之后进行的其他过程。

传播 NIS 映射

更改映射后，`Makefile` 将使用 `yppush` 向从属服务器传播新映射（除非在 `Makefile` 中设置了 `NOPUSH`）。它会通知 `ypserv` 守护进程并发送映射传送请求以完成此操作。然后，从属服务器上的 `ypserv` 守护进程会启动 `ypxfr` 进程，该进程将与主服务器上的 `ypxfrd` 守护进程联系。在进行了一些基本检查（例如，映射是否真的发生了更改？）后，便会传送映射。然后，从属服务器上的 `ypxfr` 将向 `yppush` 进程发送响应，指明传送是否成功。

注 - 上述过程不适用于新创建的、但在从属服务器中尚不存在的映射。必须通过在从属服务器上运行 `ypxfr` 将新映射发送到从属服务器。

有时候，映射无法传播，这时必须使用 `ypxfr` 手动发送新映射信息。您可以选择以两种不同方式使用 `ypxfr`：通过根 `crontab` 文件定期使用，或在命令行中交互使用。这些方法将在以下各节中进行讨论。

使用 `cron` 进行映射传送

不同映射的更改速度有所不同。例如，有些映射有时候几个月都不更改一次（如缺省映射中的 `protocols.byname` 以及非缺省映射中的 `auto_master`）；但 `passwd.byname` 可能一天就会进行多次更改。使用 `crontab` 命令调度映射传送，可以为各个映射设置特定的传播时间。

要以适合于映射的速度定期运行 `ypxfr`，每台从属服务器中的根 `crontab` 文件应包含适当的 `ypxfr` 项。`ypxfr` 与主服务器联系，并仅在主服务器中的副本比本地副本更新时才传送映射。

注 – 如果主服务器运行带有缺省 `-m` 选项的 `rpc.yppasswdd`，则每次 `yp` 口令发生更改时，`passwd` 守护进程都会运行 `make`，重新生成 `passwd` 映射。

将 Shell 脚本用于 `cron` 和 `ypxfr`

作为为每个映射创建单独的 `crontab` 项的备选方法，您可能更喜欢让根 `crontab` 命令运行 `shell` 脚本来定期更新所有映射。用于更新映射的 `shell` 脚本样例位于 `/usr/lib/netsvc/yp` 目录中。这些脚本名为 `ypxfr_1perday`、`ypxfr_1perhour` 和 `ypxfr_2perday`。您可以修改或替换这些 `shell` 脚本来满足站点需要。[示例 6-1](#) 显示了缺省的 `ypxfr_1perday` `shell` 脚本。

示例 6-1 `ypxfr_1perday` `Shell` 脚本

```
#!/bin/sh
#
# ypxfr_1perday.sh - Do daily yp map check/updates
PATH=/bin:/usr/bin:/usr/lib/netsvc/yp:$PATH
export PATH
# set -xv
ypxfr group.byname
ypxfr group.bygid
ypxfr protocols.byname
ypxfr protocols.bynumber
ypxfr networks.byname
ypxfr networks.byaddr
ypxfr services.byname
ypxfr ypservers
```

如果根 `crontab` 每天执行一次，此 `shell` 脚本会每天更新一次映射。您也可以使脚本以其他频率更新映射（如每周一次、每月一次、每小时一次等等）。但请注意，频繁传播映射会降低性能。

在为 NIS 域配置的每台从属服务器上，以超级用户的身份运行相同的 `shell` 脚本。逐个更改每台服务器上确切的执行时间，以避免使主服务器陷入停顿状态。

如果要从特定的从属服务器中传送映射，请在 shell 脚本中使用 `ypxfr` 的 `-h machine` 选项。放入脚本中的命令的语法如下所示。

```
# /usr/lib/netsvc/yp/ypxfr -h machine [ -c ] mapname
```

其中，*machine* 是要传送的映射所在服务器的名称，*mapname* 是所请求的映射的名称。如果使用了 `-h` 选项但未指定计算机，`ypxfr` 将尝试从主服务器中获取映射。如果执行 `ypxfr` 时 `ypserv` 未在本机运行，则必须使用 `-c` 标志，以使 `ypxfr` 不向本地 `ypserver` 发送清除当前映射的请求。

您可以使用 `-s domain` 选项从其他域向本地域传送映射。这些映射在各个域中应该相同。例如，两个域可能会共享相同的 `services.byname` 和 `services.byaddr` 映射。或者，也可以使用 `rcp` 或 `rdist` 来获取更多控制，以便跨域传送文件。

直接调用 ypxfr

另一种调用 `ypxfr` 的方法是将其作为命令来运行。通常，只在异常情况下才使用此方法——例如，在设置临时 NIS 服务器来创建测试环境时或在尝试使断开服务的 NIS 服务器快速与其他服务器保持一致时。

记录 ypxfr 活动

`ypxfr` 的传送尝试和结果可以捕获并存储在日志文件中。如果存在名为 `/var/yp/ypxfr.log` 的文件，则会向该文件中附加结果。对于该日志文件的大小没有任何限制。为防止日志文件无限制地增大，请键入以下命令定期清空该文件。

```
# cd /var/yp
# cp ypxfr.log ypxfr.log.old
# cat /dev/null > /var/yp/ypxfr.log
```

可让 `crontab` 一周执行一次上述命令。要禁用日志记录，请删除日志文件。

修改非缺省映射

要更新非缺省映射，必须执行下列操作。

1. 创建或编辑对应的文本文件。
2. 生成（或重新生成）新映射或更新的映射。生成映射的方法有两种。
 - 使用 `Makefile`。使用 `Makefile` 是生成非缺省映射的首选方法。如果映射在 `Makefile` 中具有一个项，请运行 `make name`，其中 *name* 是要生成的映射的名称。如果映射中没有 `Makefile` 项，请尝试按照第 98 页中的“[修改和使用 Makefile](#)”中的说明创建一项。
 - 使用 `/usr/sbin/makedbm` 程序。[makedbm\(1M\)](#) 手册页全面描述了此命令。

使用 **makedbm** 修改非缺省映射

如果您没有输入文件，可通过两种不同的方法使用 **makedbm** 修改映射：

- 将 **makedbm -u** 输出重定向到一个临时文件中，修改该文件，然后使用修改过的文件作为 **makedbm** 的输入。
- 在供给 **makedbm** 的流水线中运行 **makedbm -u** 的输出。如果您可以使用附加的 **awk**、**sed** 或 **cat** 更新反编译的映射，则这种方法是适合的。

从文本文件创建新映射

假设使用编辑器或 **shell** 脚本在主服务器上创建了一个文本文件 `/var/yp/mymap.asc`。您希望从此文件中创建一个 NIS 映射并将其置于 *homedomain* 子目录中。为此，请在主服务器上键入以下命令。

```
# cd /var/yp
# makedbm mymap.asc homedomain/mymap
```

mymap 映射现在存在于主服务器上的 *homedomain* 目录中。要将该新映射分发到从属服务器，请运行 **ypxfr**。

向基于文件的映射中添加项

向 *mymap* 中添加项是非常简单的。首先，必须修改文本文件 `/var/yp/mymap.asc`。如果修改实际的 **dbm** 文件而不修改对应的文本文件，所做的修改会丢失。然后，按上面所示运行 **makedbm**。

通过标准输入创建映射

当不存在原始文本文件时，请向 **makedbm** 键入输入，从键盘创建 NIS 映射，如下所示（以 **Ctrl-D** 组合键结束）。

```
ypmaster# cd /var/yp
ypmaster# makedbm -homedomain- /mymap/key1 value1 key2 value2 key3 value3
```

修改通过标准输入创建的映射

如果以后需要修改映射，可以使用 **makedbm** 反编译映射，并创建一个临时的中间文本文件。要反编译映射并创建一个临时文件，请键入下列命令。

```
% cd /var/yp
% makedbm -u homedomain/mymap > mymap.temp
```


在生成的临时文件 *mymap.temp* 中，每行包含一项。您可以根据需要，使用任何文本编辑器编辑此文件。

要更新映射，请键入以下命令将修改过的临时文件的名称提供给 *makedbm*。

```
% makedbm mymap.temp homedomain/mymap
% rm mymap.temp
```

然后，成为超级用户并键入下列命令，将映射传播到从属服务器。

```
# yppush mymap
```

上一段介绍了如何使用 *makedbm* 创建映射；但几乎每项您实际必须执行的操作都可以通过 *ypinit* 和 *Makefile* 完成，除非您在系统启动并运行后向数据库中添加了非缺省映射或更改了 NIS 服务器集合。

无论您在 */var/yp* 中使用 *Makefile* 还是其他某个过程，目标都是相同的。主服务器上的映射目录中最终必须有格式正确的新 *dbm* 文件对。

添加从属服务器

NIS 运行后，您可能需要创建一些之前提供给 *ypinit* 的初始列表中并未包含 NIS 从属服务器。

要添加 NIS 从属服务器：

▼ 如何添加从属服务器

- 1 在主服务器上，成为超级用户或承担等效角色。
角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。
- 2 转至 NIS 域目录。

```
# cd /var/yp/domainname
```
- 3 反编译 *ypservers* 文件。

```
# makedbm -u ypservers >/tmp/temp_file
```


makedbm 命令会将 *ypservers* 从 *ndbm* 格式转换为临时 ASCII 文件 */tmp/temp_file*。
- 4 使用文本编辑器编辑 */tmp/temp_file* 文件。将新的从属服务器的名称添加到服务器列表中。然后，保存并关闭该文件。
- 5 运行 *makedbm* 命令，并以 *temp_file* 和 *ypservers* 分别作为输入文件和输出文件。

```
# makedbm /tmp/temp_file ypservers
```

然后，`makedbm` 会将 `ypservers` 重新转换回 `ndbm` 格式。

- 6 在从属服务器上键入以下命令，验证 `ypservers` 映射是否正确（由于没有针对 `ypservers` 的 ASCII 文件）。

```
slave3# makedbm -u ypservers
```

`makedbm` 命令会在屏幕上显示 `ypservers` 中的每项。

注 – 如果某计算机名不在 `ypservers` 中，则该计算机不会收到映射文件的更新，原因是 `yppush` 需要从此映射中查询从属服务器的列表。

- 7 在新的 NIS 从属服务器上，成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

- 8 设置新从属服务器的 NIS 域目录。

从主服务器复制 NIS 映射集合，然后启动 NIS 客户机。运行 `ypinit` 命令时，请遵循提示并按优先级顺序列出 NIS 服务器。

```
slave3# cd /var/yp
slave3# ypinit -c
slave3# svcadm enable network/nis/client
```

- 9 将此计算机初始化为从属服务器。

```
slave3# /usr/sbin/ypinit -s ypmaster
```

其中，`ypmaster` 是现有的 NIS 主服务器的计算机名。

- 10 停止作为 NIS 客户机运行的计算机。

```
# svcadm disable network/nis/client
```

- 11 启动 NIS 从属服务。

```
# svcadm enable network/nis/server
```

使用启用 C2 安全性的 NIS

如果存在 `$PWDIR/security/passwd.adjunct` 文件，则会自动启动 C2 安全性。（`$PWDIR` 在 `/var/yp/Makefile` 中定义。）C2 安全模式使用 `passwd.adjunct` 文件来创建 `passwd.adjunct` NIS 映射。在此实现中，NIS 允许您使用 `passwd.adjunct` 文件和 `shadow` 文件来管理安全性。仅当您键入下列命令时，才会处理 `passwd.adjunct` 文件。

```
# make passwd.adjunct
```

当您在 C2 安全模式下手动运行 `make` 时，`make passwd` 命令将仅处理 `passwd` 映射，而不处理 `passwd.adjunct` 映射。

绑定到特定 NIS 服务器

使用下列步骤可以绑定到您指定的 NIS 服务器。有关详细信息，请参见 [ypinit\(1M\)](#)、[ypstart\(1M\)](#) 以及 [svcadm\(1M\)](#) 手册页。

1. 将 NIS 服务器的主机名及其 IP 地址添加到 `/etc/hosts` 文件。
2. 运行 `domainname` 命令以置备 `/etc/defaultdomain` 文件。

```
# /usr/bin/domainname name-of-NIS-domain
```

3. 提示输入 NIS 服务器主机名。

```
# /usr/sbin/ypinit -c
Server name:      Type the NIS server hostname
```

4. 执行下列步骤重启 NIS 服务。
 - 对于在系统重新引导期间持续生效的服务，请运行 `svcadm` 命令：


```
# svcadm enable -r svc:/network/nis/client
```
 - 对于在重新引导之前生效的服务，请运行 `ypstop` 和 `ypstart` 命令：

```
# /usr/lib/netsvc/yp/ypstop
# /usr/lib/netsvc/yp/ypstart
```

更改计算机的 NIS 域

要更改计算机的 NIS 域名，请执行以下操作。

▼ 如何更改计算机的 NIS 域名

- 1 成为超级用户或承担等效角色。
角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。
- 2 编辑计算机的 `/etc/defaultdomain` 文件，使用计算机的新域名替换其当前内容。
例如，如果当前域名为 `sales.doc.com`，可将其更改为 `research.doc.com`。
- 3 运行 `domainname 'cat /etc/defaultdomain'`
- 4 将计算机重新初始化为 NIS 客户机、从属服务器或主服务器。
有关详细信息，请参见第 5 章，[设置和配置 NIS 服务](#)。

将 NIS 与 DNS 结合使用

通常，我们使用 `nsswitch.conf` 文件配置 NIS 客户机，以便只使用 NIS 查找计算机名和地址。如果此类查找失败，NIS 服务器可将这些查找转发给 DNS。

▼ 如何配置通过 NIS 和 DNS 执行计算机名和地址查找

- 1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

- 2 两个映射文件（`hosts.byname` 和 `hosts.byaddr`）中必须包括 `YP_INTERDOMAIN` 关键字。要测试此关键字，请编辑 `Makefile` 并修改以下各行。

```
#B=-b  
B=
```

更改为：

```
B=-b  
#B=
```

现在，`makedbm` 在创建映射时将以 `-b` 标志启动，并会在 `ndbm` 文件中插入 `YP_INTERDOMAIN` 关键字。

- 3 运行 `make` 命令，以重新生成映射。

```
# /usr/ccs/bin/make hosts
```

- 4 检查是否所有 NIS 服务器的 `/etc/resolv.conf` 文件都指向有效的名称服务器。

注 - 如果您有不运行 Solaris 发行版 2 的 NIS 服务器，请确保主机映射中存在 `YP_INTERDOMAIN`。

- 5 要启用 DNS 转发，请重新启动每台服务器。

```
# svcadm restart network/nis/server:<instance>
```

在此 NIS 实现中，`ypserv` 将自动以 `-d` 选项启动，将请求转发给 DNS。

处理混合的 NIS 域

如果主服务器和从属服务器并未同时运行 Solaris 2，请参阅下表了解如何避免可能遇到的问题。表示法 "4.0.3+" 指 Solaris OS 的发行版 4.0.3 以及更高版本。`makedm -b` 是对 `Makefile` 中 "B" 变量的引用。

表 6-1 异构NIS域中的NIS/DNS

从属服务器	主服务器		
	4.0.3+	Solaris NIS	
4.0.3+	主服务器：makedbm -b	主服务器：makedbm -b	主服务器：ypserv -d
	从属服务器：ypxfr	从属服务器：ypxfr -b	从属服务器：ypxfr -b
Solaris NIS	主服务器：makedbm -b	主服务器：makedbm -b	主服务器：ypserv -d
	从属服务器：ypxfr	从属服务器：ypxfr	从属服务器：ypxfr（带 resolv.conf）或 ypxfr -b

禁用NIS服务

如果禁用了 NIS 主服务器上的 `ypserv`，您将无法再更新任何 NIS 映射。

- 要禁用客户机上的NIS，请键入以下命令：

```
# svcadm disable network/nis/client
```
- 要禁用特定从属服务器或主服务器上的NIS，请在服务器上键入以下命令：

```
# svcadm disable network/nis/server
```


NIS 疑难解答

本章介绍如何解决在运行 NIS 的网络中遇到的问题，还介绍了 NIS 客户机和 NIS 服务器中存在的问题。

在尝试调试 NIS 服务器或客户机之前，请先阅读第 4 章，[网络信息服务 \(Network Information Service, NIS\) \(概述\)](#)，其中对 NIS 环境进行了介绍。然后，在本节中查找最能恰当描述您所遇到的问题的副标题。

注 - NIS 服务由服务管理工具管理。使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。有关将 SMF 与 NIS 结合使用的更多信息，请参见第 80 页中的“NIS 和服务管理工具”。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务 (概述)”。有关更多详细信息，另请参阅 [svcadm\(1M\)](#) 和 [svcs\(1\)](#) 手册页。

还可以使用 `ypstart` 和 `ypstop` 命令来启动和停止 NIS 服务。有关更多信息，请参见 [ypstart\(1M\)](#) 和 [ypstop\(1M\)](#) 手册页。

NIS 绑定问题

症状

NIS 绑定问题具有以下常见症状。

- 有消息提示 `ypbind` 找不到服务器或无法与服务器通信
- 有消息提示服务器不响应
- 有消息提示 NIS 不可用
- 客户机上的命令在后台模式下缓慢运行，或运行速度远低于正常情况
- 客户机上的命令挂起。有时候，即使整个系统看似正常并且可以运行新命令，命令可能也已挂起

- 客户机上的命令崩溃，同时显示不明消息或根本不显示消息

影响一台客户机的 NIS 问题

如果只有一两台客户机出现了表明 NIS 绑定问题的症状，则可能是这些客户机存在问题。如果许多 NIS 客户机都无法正确绑定，则可能是一台或多台 NIS 服务器存在问题。请参见第 115 页中的“影响多台客户机的 NIS 问题”。

ypbind 未在客户机上运行

一台客户机中存在问题，但同一子网上的其他客户机运行正常。在存在问题的客户机上，对满足以下条件的目录运行 `ls -l`：目录中的文件由许多用户拥有，并且其中一些用户不在该客户机的 `/etc/passwd` 文件中。例如 `/usr` 目录。如果显示结果将不在本地 `/etc/passwd` 中的文件所有者以数字形式（而非名称）列出，表明 NIS 服务未在该客户机上运行。

这些症状通常表示客户机的 `ypbind` 进程未运行。请验证 NIS 客户机服务是否正在运行。

```
client# svcs network/nis/client
STATE          STIME      FMRI
disabled       Sep_01     svc:/network/nis/client:default
```

如果客户机已禁用，请以超级用户或承担等效角色的身份登录，然后启动 NIS 客户机服务。

```
client# svcadm enable network/nis/client
```

缺少域名或域名不正确

一台客户机中存在问题，其他客户机运行正常，但 `ypbind` 正在出问题的客户机上运行。该客户机的域可能设置不正确。

在该客户机上，运行 `domainname` 命令，查看它设置了哪个域名。

```
client7# domainname neverland.com
```

将输出结果与 NIS 主服务器上 `/var/yp` 中的实际域名进行比较。实际的 NIS 域显示为 `/var/yp` 目录中的子目录。

```
Client7# ls /var/yp...
-rwxr-xr-x 1 root Makefile
drwxr-xr-x 2 root binding
drwx----- 2 root doc.com ...
```

如果在计算机上运行 `domainname` 所返回的域名与 `/var/yp` 中作为目录列出的服务器域名不同，则计算机的 `/etc/defaultdomain` 文件中指定的域名不正确。请以超级用户身

份登录或承担等效角色，然后在计算机的 `/etc/defaultdomain` 文件中更正该客户机的域名。这样可以确保计算机每次引导时，域名都是正确的。然后，立即重新引导计算机。

注 - 域名区分大小写。

客户机未绑定到服务器

如果域名设置正确，并且 `ypbind` 正在运行，但命令仍然挂起，请运行 `ypwhich` 命令来确保客户机已绑定到服务器。如果您刚刚启动了 `ypbind`，可多运行几次 `ypwhich`（通常，第一次运行时会报告域未绑定，第二次便会成功）。

没有可用的服务器

如果域名设置正确，并且 `ypbind` 正在运行，但有消息提示客户机无法与服务器通信，则表示问题可能有多种：

- 客户机是否具有 `/var/yp/binding/ domainname/ypservers` 文件（该文件中包含要绑定到的服务器的列表）？如果没有该文件，请运行 `ypinit -c` 并按优先级顺序指定客户机应绑定到的服务器。
- 如果客户机有 `/var/yp/binding/ domainname/ypservers` 文件，该文件中列出的服务器数量是否足以应对一台或两台服务器不可用时的情况？如果没有足够多的服务器，请运行 `ypinit -c` 向列表中添加更多服务器。
- 客户机的 `ypservers` 文件中列出的服务器在 `/etc/hosts` 文件中是否有对应项？如果没有，请将这些服务器添加到 NIS 映射主机的输入文件，然后运行 `yppinit -c` 或 `ypinit -s` 重建映射，如第 95 页中的“使用 NIS 映射”中所述。
- `/etc/nsswitch.conf` 文件是否设置为除了 NIS 外还查询计算机的本地 `hosts` 文件？有关该转换器的更多信息，请参见第 2 章，名称服务转换器（概述）。
- `/etc/nsswitch.conf` 文件是否设置为先在 `files` 中查询 `services` 和 `rpc`？有关该转换器的更多信息，请参见第 2 章，名称服务转换器（概述）。

ypwhich 显示不一致

在同一客户机上多次使用 `ypwhich` 时，所显示的结果会随 NIS 服务器的更改而有所不同。这很正常。当网络或 NIS 服务器繁忙时，NIS 客户机到 NIS 服务器的绑定会不时发生变化。只要可以进行绑定，网络便会在某一时间稳定，所有客户机在此期间从 NIS 服务器获取可接受的响应时间。只要您的客户机能够获得 NIS 服务，服务来源无关紧要。例如，一台 NIS 服务器计算机可以从网络中的其他 NIS 服务器获取其 NIS 服务。

当无法进行服务器绑定时

在无法进行本地服务器绑定的特殊情况下，使用 `ypset` 命令可以暂时允许绑定到其他网络或子网中的另一台服务器（如果可用）。但是，为了使用 `-ypset` 选项，必须使用 `-ypset` 或 `-ypsetme` 选项启动 `ypbind`。有关更多信息，请参见 [ypbind\(1M\)](#) 手册页。

```
# /usr/lib/netsvcs/yp/ypbind -ypset
```

有关其他方法，请参见第 107 页中的“绑定到特定 NIS 服务器”。

注 – 出于安全考虑，应将 -ypset 和 -ypsetme 选项的使用限制为在受控情况下的调试用途。使用 -ypset 和 -ypsetme 选项可能会导致严重的安全违规，因为在守护进程运行时，任何人都可以更改服务器绑定，这会给其他人带来麻烦，并将允许未经授权的用户访问敏感数据。如果必须使用这些选项启动 ypbind，待问题修复后，应立即终止 ypbind，然后在不使用这些选项的情况下重新启动。

要重新启动 ypbind，请使用服务管理工具 (Service Management Facility, SMF)：

```
# svcadm enable -r svc:/network/nis/client:default
```

ypbind 崩溃

如果 ypbind 几乎在每次启动后立即崩溃，请查找系统中其他部分是否存在问题。键入以下内容检查是否存在 rpcbind 守护进程。

```
% ps -e | grep rpcbind
```

如果 rpcbind 不存在、无法持续运行或行为异常，请查阅 RPC 文档。

您也许能够从正常运行的计算机上与存在问题的客户机中的 rpcbind 通信。从正常运行的计算机中，键入以下内容。

```
% rpcinfo client
```

如果存在问题的计算机中的 rpcbind 正常，rpcinfo 将生成以下输出内容。

program	version	netid	address	service	owner
...					
100007	2	udp	0.0.0.0.2.219	ypbind	superuser
100007	1	udp	0.0.0.0.2.219	ypbind	superuser
100007	1	tcp	0.0.0.0.2.220	ypbind	superuser
100007	2	tcp	0.0.0.0.128.4	ypbind	superuser
100007	2	ticotsord	\000\000\020H	ypbind	superuser
100007	2	ticots	\000\000\020K	ypbind	superuser
...					

您的计算机将具有不同地址。如果未显示这些地址，则 ypbind 无法注册其服务。请重新引导计算机并再次运行 rpcinfo。如果存在 ypbind 进程，并且这些进程在每次重新启动 NIS 服务时都会更改，请重新引导系统，即使 rpcbind 守护进程正在运行，也应如此。

影响多台客户机的 NIS 问题

如果只有一两台客户机出现了表明 NIS 绑定问题的症状，则可能是这些客户机存在问题。请参见第 112 页中的“影响一台客户机的 NIS 问题”。如果许多 NIS 客户机都无法正确绑定，则可能是一台或多台 NIS 服务器存在问题。

rpc.yppasswdd 将以 r 开头的非受限 Shell 视为受限制

1. 创建包含以下特殊字符串的
/etc/default/yppasswdd: "check_restricted_shell_name=1"。
2. 如果将 "check_restricted_shell_name=1" 字符串注释掉，将不会进行 "r" 检查。

网络或服务器过载

如果网络或 NIS 服务器过载，导致 ypserv 无法在超时时间段内将响应返回给客户机的 ypbind 进程，则 NIS 将挂起。

在这些情况下，网络中的每台客户机都会遇到相同或相似的问题。在大多数情况下，这种问题是暂时的。当 NIS 服务器重新引导并重新启动 ypserv 时，或 NIS 服务器或网络自身的负载降低时，通常不会再显示该消息。

服务器出现异常

请确保服务器已启动并且正在运行。如果您的物理位置离服务器较远，请使用 ping 命令。

NIS 守护进程未运行

如果服务器已启动并且正在运行，请尝试找一台能够正常工作的客户机计算机，然后运行 ypwhich 命令。如果 ypwhich 不响应，请将其终止。然后以 root 身份登录 NIS 服务器，并输入以下命令检查 NIS 进程是否正在运行。

```
# ps -e | grep yp
```

注 - 不要将 -f 选项与 ps 结合使用，因为此选项会尝试将用户 ID 转换为名称，而这将导致更多命名服务查找可能无法成功。

如果 NIS 服务器 (ypserv) 和 NIS 客户机 (ypbind) 守护进程都未运行，请键入以下命令重新启动它们。

```
# svcadm restart network/nis/server
or
# /usr/lib/netsvc/yp/ypstop
# /usr/lib/netsvc/yp/ypstart
```

如果 `ypserv` 和 `ypbind` 进程都在 NIS 服务器上运行，请运行 `ypwhich`。如果 `ypwhich` 不响应，`ypserv` 可能已挂起，并应重新启动。在以 `root` 身份登录到服务器后，请键入以下命令重新启动 NIS 服务。

```
# svcadm restart network/nis/server
or
# /usr/lib/netsvc/yp/ypstop
# /usr/lib/netsvc/yp/ypstart
```

服务器具有不同版本的 NIS 映射

因为 NIS 在服务器之间传播映射，所以有时您可能会在网络中的不同 NIS 服务器上找到同一个映射的不同版本。如果差别持续的时间不长，则此版本差异正常并且可以接受。

导致映射差异最常见的原因是某些因素阻止了正常的映射传播。例如，NIS 服务器或 NIS 服务器之间的路由器关闭。当所有 NIS 服务器以及它们之间的路由器都在运行时，`ypxfr` 应该能成功运行。

如果服务器和路由器运行正常，请检查以下各项：

- 记录 `ypxfr` 的输出（请参见第 116 页中的“记录 `ypxfr` 输出”）。
- 检查控制文件（请参见第 117 页中的“检查 `crontab` 文件和 `ypxfr` Shell 脚本”）。
- 检查主服务器上的 `ypservers` 映射。请参见第 117 页中的“检查 `ypservers` 映射”。

记录 ypxfr 输出

如果特定从属服务器在更新映射时出现问题，请登录该服务器并以交互方式运行 `ypxfr`。如果 `ypxfr` 运行失败，则会指出失败原因，然后您可以针对原因解决问题。如果 `ypxfr` 运行成功，但您怀疑运行有时失败，请创建一个日志文件以便记录消息。要创建日志文件，请在从属服务器上输入以下命令。

```
ypslave# cd /var/yp
ypslave# touch ypxfr.log
```

这将创建一个 `ypxfr.log` 文件，该文件会保存 `ypxfr` 的所有输出。

该输出与 `ypxfr` 以交互方式运行时所显示的输出类似，但日志文件中的每行都带有时间标记。（您可能会发现时间标记排序不正常。这是正常情况—时间标记会告诉您 `ypxfr` 开始运行的时间。如果 `ypxfr` 的多个副本同时运行，但所用的时间不同，则它们可能实际上按照不同于调用顺序的顺序将摘要状态行写入日志文件。）任何形式的间歇性故障都会在日志中显示。

注—解决问题后，请删除日志文件以关闭记录功能。如果忘记删除该文件，它将继续无限制地增大。

检查 crontab 文件和 ypxfr Shell 脚本

检查根 crontab 文件，并检查该文件调用的 ypxfr shell 脚本。这些文件中的排字错误可能会引起传播问题。在 `/var/spool/cron/crontabs/root` 文件中引用 shell 脚本失败，或者在任何 shell 脚本中引用映射失败，也可能导致错误。

检查 ypservers 映射

另外，请确保 NIS 从属服务器已列在域的主服务器的 ypservers 映射中。否则，从属服务器虽然仍可作为服务器正常运行，但 yppush 不会将映射的更改传播至从属服务器。

解决方法

如果 NIS 从属服务器的问题不明显，可在调试时解决这一问题，方法是使用 rcp 或 ftp 从运行状况良好的 NIS 服务器中复制不一致映射的最新版本。下面显示了如何传送有问题的映射。

```
ypslave# rcp ypmaster:/var/yp/mydomain/map.* /var/yp/mydomain
```

* 字符在命令行中进行了转义，这样它将在 ypmaster 中展开，而不是在 ypslave 本地展开。

ypserv 崩溃

如果 ypserv 进程几乎总是在启动后的瞬间崩溃，并且即使重复启动也无法持续运行，则基本上可按照第 114 页中的“ypbind 崩溃”中所述的过程进行调试。如下所示，检查是否存在 rpcbind 守护进程。

```
ypserver% ps -e | grep rpcbind
```

如果找不到该守护进程，请重新引导服务器。否则，如果该守护进程正在运行，请键入以下命令查找类似输出。

```
% rpcinfo -p ypserv
```

% program	vers	proto	port	service
100000	4	tcp	111	portmapper
100000	3	tcp	111	portmapper
100068	2	udp	32813	cmsd
...				
100007	1	tcp	34900	ypbind
100004	2	udp	731	ypserv
100004	1	udp	731	ypserv
100004	1	tcp	732	ypserv
100004	2	tcp	32772	ypserv

您的计算机可能具有不同的端口号。表示 ypserv 进程的四个项如下所示。

```
100004      2      udp      731      ypserv
100004      1      udp      731      ypserv
100004      1      tcp      732      ypserv
100004      2      tcp      32772    ypserv
```

如果不存在任何项并且 `ypserv` 无法向 `rpcbind` 注册其服务，请重新引导计算机。如果这些项存在，请在重新启动 `ypserv` 之前从 `rpcbind` 取消注册服务。要从 `rpcbind` 中取消注册服务，请在服务器上键入以下命令。

```
# rpcinfo -d number 1
# rpcinfo -d number 2
```

其中，*number* 是 `rpcinfo` 报告的 ID 号（在以上示例中，ID 号为 `100004`）。

第 4 部分

LDAP 命名服务的设置和管理

本部分概述了 LDAP 命名服务。此外，还介绍了有关 Solaris OS 中 LDAP 命名服务的设置、配置、管理及疑难解答，重点介绍如何使用 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）。

LDAP 命名服务介绍（概述/参考）

有关 LDAP 的各章介绍了如何设置 Solaris LDAP 命名服务客户机以便与 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）协作。但是，尽管建议使用 Sun Java System Directory Server，但这不是必需的。第 14 章，[LDAP 一般参考（参考）](#)中简要说明了目录服务器的一般要求。

注 – 目录服务器不一定是 LDAP 服务器。但是，在这些章的上下文中，术语“目录服务器”与“LDAP 服务器”同义。

目标用户

有关 LDAP 命名服务的各章是为已经熟悉 LDAP 的系统管理员编写的。以下列出了用户必须非常熟悉的部分概念。如果不熟悉这些概念，使用本指南在 Solaris 系统中部署 LDAP 命名服务可能会遇到困难。

- LDAP 信息模型（项、对象类、属性、类型和值）
- LDAP 命名模型（目录信息树 (Directory Information Tree, DIT) 结构）
- LDAP 功能模型（搜索参数：基本对象 (base object, DN)、范围、大小限制、时间限制、过滤器（Sun Java System Directory Server 的浏览索引）和属性列表）
- LDAP 安全模型（验证方法和访问控制模型）
- 对 LDAP 目录服务的整体规划和设计（包括如何规划数据以及如何设计 DIT、拓扑、复制和安全性）

建议的背景读物

要更多地了解上述任一概念或者学习 LDAP 及目录服务部署的一般知识，请参阅以下文献：

- 由 Timothy A. Howes 博士和 Mark C. Smith 编著的《Understanding and Deploying LDAP Directory Services》
除了提供 LDAP 目录服务的详细处理方法以外，本书还包括一些有用的 LDAP 部署案例研究。部署示例包括大型大学、大型跨国企业和具有外部网络的企业。
- 随 Sun Java Enterprise System 文档提供的 Sun Java System Directory Server 部署指南
本指南为目录的规划（包括目录设计、架构设计、目录树、拓扑、复制和安全性）提供基础。最后一章提供了部署方案样例，用于帮助您规划简单的小型部署和复杂的全球部署。
- 随 Sun Java Enterprise System 文档提供的 Sun Java System Directory Server 管理指南

其他先决条件

如果您需要安装 Sun Java System Directory Server，请参阅所使用的 Sun Java System Directory Server 版本的安装指南。

LDAP 命名服务与其他命名服务的比较

下表对 DNS、NIS、NIS+ 和 LDAP 命名服务进行了比较。

	DNS	NIS	NIS+	LDAP
名称空间	分层	不分层	分层	分层
数据存储	文件/资源记录	包含 2 列的映射	包含多列的表	目录（视情况而定） 索引数据库
服务器	主/从	主/从	根主/ 非根主；主/ 辅助；高速缓存/存根	主/副本 多主副本
安全性	无	无（根或不包含任何内容）	安全 RPC (AUTH_DH) 验证	SSL（视情况而定）
传输	TCP/IP	RPC	RPC	TCP/IP
范围	全局	LAN	LAN	全局

LDAP 命名服务的优点

- 使用 LDAP，可以通过替换应用程序特定的数据库来整合信息，这可减少要管理的不同数据库的数目。
- LDAP 允许不同的命名服务共享数据。
- LDAP 可提供一個集中的数据信息库。
- LDAP 允许在主服务器和副本服务器之间更频繁地同步数据。
- LDAP 可兼容多种平台以及由多个供应商提供的产品。

LDAP 命名服务的限制

以下是与 LDAP 命名服务相关联的一些限制：

- 不支持 Solaris 8 之前的客户机。
- LDAP 服务器不能作为其自身的客户机。
- LDAP 命名服务的设置和管理更加复杂，需要仔细规划。
- NIS 客户机和本地 LDAP 客户机不能在同一台客户机上共存。

注 – 目录服务器（LDAP 服务器）**不能**作为其自身的客户机。即，不能将运行目录服务器软件的计算机配置为 LDAP 命名服务客户机。

设置 LDAP 命名服务（任务列表）

任务	参考
确认已安装了修补程序	
规划网络模型	第 149 页中的“规划 LDAP 网络模型”
规划 DIT	第 10 章，LDAP 命名服务的规划要求（任务）
设置副本服务器	第 151 页中的“LDAP 和副本服务器”
规划安全模型	第 152 页中的“规划 LDAP 安全模型”
选择客户机配置文件和缺省属性值	第 154 页中的“规划 LDAP 的客户机配置文件和缺省属性值”
规划数据置备	第 154 页中的“规划 LDAP 数据置备”
配置 Sun Java System Directory Server 之后，再将其用于 LDAP 命名服务	Sun ONE Directory Server 5.2（Solaris 版本）

任务	参考
设置 Sun Java System Directory Server 以用于 LDAP 命名客户机	第 11 章，为使用 LDAP 客户机设置 Sun Java System Directory Server（任务）
管理打印机项	第 167 页中的“管理打印机项”
初始化 LDAP 客户机	第 175 页中的“初始化 LDAP 客户机”
使用配置文件初始化客户机	第 175 页中的“使用配置文件初始化客户机”
手动初始化客户机	第 179 页中的“手动初始化客户机”
取消对客户机的初始化	第 180 页中的“取消客户机初始化”
使用服务搜索描述符修改客户机配置文件	第 160 页中的“使用服务搜索描述符修改客户机对各个服务的访问”
检索命名服务信息	第 183 页中的“检索 LDAP 命名服务信息”
定制客户机环境	第 185 页中的“定制 LDAP 客户机环境”

LDAP 基本组件和概念（概述）

本章包含以下主题：

- 第 125 页中的“LDAP 数据交换格式 (LDAP Data Interchange Format, LDIF)”
- 第 128 页中的“随 LDAP 使用全限定域名”
- 第 129 页中的“缺省目录信息树 (Directory Information Tree, DIT)”
- 第 129 页中的“缺省的 LDAP 架构”
- 第 130 页中的“服务搜索描述符 (Service Search Descriptor, SSD) 和架构映射”
- 第 132 页中的“LDAP 客户机配置文件”
- 第 134 页中的“`ldap_cachemgr` 守护进程”
- 第 135 页中的“LDAP 命名服务安全模型”

LDAP 数据交换格式 (LDAP Data Interchange Format, LDIF)

LDIF 是一种基于文本的格式，用于描述目录服务实体及其属性。使用 LDIF 格式，可以通过 `ldapadd` 和 `ldapmodify` 等命令将信息从一个目录移到另一个目录。下面是每个服务的 LDIF 格式的示例。使用 `ldaplist(1)` 命令及 `-l` 选项将显示以下信息。

```
% ldaplist -l hosts myhost
```

```
hosts
```

```
dn: cn=myhost+ipHostNumber=7.7.7.115,ou=Hosts,dc=mydc,dc=mycom,dc=com
cn: myhost
iphostnumber: 7.7.7.115
objectclass: top
objectclass: device
objectclass: ipHost
description: host 1 - floor 1 - Lab a - building b
```

```
% ldaplist -l passwd user1
```

```
passwd
```

```
dn: uid=user1,ou=People,dc=mydc,dc=mycom,dc=com
```

```
uid: user1
cn: user1
userpassword: {crypt}duTx91g7PoNzE
uidnumber: 199995
gidnumber: 20
gecos: Joe Smith [New York]
homedirectory: /home/user1
loginshell: /bin/csh
objectclass: top
objectclass: shadowAccount
objectclass: account
objectclass: posixAccount
```

% ldaplist -l services name

services

```
dn: cn=name+ipServiceProtocol=udp,ou=Services,dc=mydc,dc=mycom,dc=com
cn: name
cn: nameserver
ipServiceProtocol: udp
ipServicePort: 42
objectclass: top
objectclass: ipService
```

% ldaplist -l group mygroup

group

```
dn: cn=mygroup,ou=Group,dc=mydc,dc=mycom,dc=com
cn: mygroup
gidnumber: 4441
memberuid: user1
memberuid: user2
memberuid: user3
userpassword: {crypt}duTx91g7PoNzE
objectclass: top
objectclass: posixGroup
```

% ldaplist -l netgroup mynetgroup

netgroup

```
cn=mynetgroup,ou=netgroup,dc=central,dc=sun,dc=com objectclass=nisNetgroup
-objectclass: -top
-cn: -mynetgroup
-nisnetgrouptriple: -(user1..mydc.mycom.com,-,) nisnetgrouptriple=(user1.,-,)
-membernisnetgroup: -mylab
```

% ldaplist -l networks 200.20.20.0

networks

```
dn: ipNetworkNumber=200.20.20.0,ou=Networks,dc=mydc,dc=mycom,dc=com
cn: mynet-200-20-20
ipnetworknumber: 200.20.20.0
```

```
objectclass: top
objectclass: ipNetwork
description: my Lab Network
ipnetmasknumber: 255.255.255.0
```

```
% ldaplist -l netmasks 201.20.20.0
```

```
netmasks
```

```
dn: ipNetworkNumber=201.20.20.0,ou=Networks,dc=mydc,dc=mycom,dc=com
cn: net-201
ipnetworknumber: 201.20.20.0
objectclass: top
objectclass: ipNetwork
description: my net 201
ipnetmasknumber: 255.255.255.0
```

```
% ldaplist -l rpc ypserv
```

```
rpc
```

```
dn: cn=ypserv,ou=Rpc,dc=mydc,dc=mycom,dc=com
cn: ypserv
cn: ypprog
oncrpcnumber: 100004
objectclass: top
objectclass: oncRpc
```

```
% ldaplist -l protocols tcp
```

```
protocols
```

```
dn: cn=tcp,ou=Protocols,dc=mydc,dc=mycom,dc=com
cn: tcp
ipprotocolnumber: 6
description: transmission control protocol
objectclass: top
objectclass: ipProtocol
```

```
% ldaplist -l bootparams myhost
```

```
bootparams
```

```
dn: cn=myhost,ou=Ethers,dc=mydc,dc=mycom,dc=com
bootparameter: root=boothost:/export/a/b/c/d/e
objectclass: top
objectclass: device
objectclass: bootableDevice
cn: myhost
```

```
% ldaplist -l ethers myhost
```

```
ethers
```

```
dn: cn=myhost,ou=Ethers,dc=mydc,dc=mycom,dc=com
```

```
macaddress: 8:1:21:71:31:c1
objectclass: top
objectclass: device
objectclass: ieee802Device
cn: myhost
```

```
% ldaplist -l publickey myhost
```

```
publickey
```

```
dn: cn=myhost+ipHostNumber=200.20.20.99,ou=Hosts,dc=mydc,dc=mycom,dc=com
cn: myhost
iphonenumber: 200.20.20.99
description: Joe Smith
nispublickey: 9cc01614d929848849add28d090acdaa1c78270aeec969c9
nissecretkey: 9999999998769c999c39e7a6ed4e7afd687d4b99908b4de99
objectclass: top
objectclass: NisKeyObject
objectclass: device
objectclass: ipHost
```

```
% ldaplist -l aliases myname
```

```
aliases
```

```
dn: mail=myname,ou=aliases,dc=mydc,dc=mycom,dc=com
cn: myname
mail: myname
objectclass: top
objectclass: mailgroup
mgrprfc822mailmember: my.name
```

随 LDAP 使用全限定域名

与 NIS 或 NIS+ 客户机不同，LDAP 客户机总是返回主机名的全限定域名 (Fully Qualified Domain Name, FQDN)。LDAP 的 FQDN 与 DNS 返回的 FQDN 相似。例如，假设您的域名为以下形式：

```
west.example.net
```

查找主机名 *server* 时，`gethostbyname()` 和 `getnameinfo()` 都返回 FQDN 版本：

```
server.west.example.net
```

此外，如果您使用特定于接口的别名（例如 `server-#`），则将返回一个全限定主机名的长列表。如果您要使用主机名来共享文件系统或者进行其他类似的检查，必须对这些检查进行说明。例如，如果将非 FQDN 用于本地主机，FQDN 仅用于由 DNS 解析的远程主机，则必须说明二者之间的区别。如果使用与 DNS 不同的域名设置 LDAP，则同一个主机最终可能得到两个不同的 FQDN，具体情况取决于查找源。

缺省目录信息树 (Directory Information Tree, DIT)

缺省情况下，Solaris LDAP 客户机在访问信息时假设 DIT 具有给定的结构。对于 LDAP 服务器支持的每个域，都存在一个具有假定结构的子树。不过，通过指定服务搜索描述符 (Service Search Descriptor, SSD)，可以覆盖该缺省结构。对于一个给定的域，缺省 DIT 将具有一个基本容器，用于存放许多已知容器，这些已知容器存储有特定信息类型的条目。有关这些子树的名称，请参见下表。（此信息可在 RFC 2307 和其他参考资料中找到。）

表 9-1 DIT 缺省位置

缺省容器	信息类型
ou=Ethers	bootparams(4)、ethers(4)
ou=Group	group(4)
ou=Hosts	hosts(4)、ipnodes(4) 和主机的 publickey
ou=Aliases	aliases(4)
ou=Netgroup	netgroup(4)
ou=Networks	networks(4)、netmasks(4)
ou=People	passwd(1)、shadow(4)、user_attr(4)、audit_user(4) 和用户的 publickey
ou=printers	printers(4)
ou=Protocols	protocols(4)
ou=Rpc	rpc(4)
ou=Services	services(4)
ou=SolarisAuthAttr	auth_attr(4)
ou=SolarisProfAttr	prof_attr(4)、exec_attr(4)
ou=projects	project
automountMap=auto_*	auto_*

缺省的 LDAP 架构

架构是一些定义，用于描述哪些类型的信息可以作为条目存储在 LDAP 目录中。要支持 LDAP 命名客户机，目录服务器的架构可能需要进行扩展。[第 14 章，LDAP 一般参考（参考）](#)中提供了有关 IETF 和 Solaris 特定架构的详细消息。您还可以从 IETF Web 站点 <http://www.ietf.org> 访问各种 RFC。

服务搜索描述符 (Service Search Descriptor, SSD) 和架构映射

注- 如果您使用架构映射，请务必谨慎并采用一致的方式。应确保被映射的属性的语法与其映射到的属性的语法一致。换言之，应确保单值属性映射到单值属性，属性的语法保持一致，并且被映射的对象类应该具有正确的强制性属性（可能是映射的属性）。

如上所述，缺省情况下，LDAP 命名服务要求 DIT 以确定方式进行构造。如果需要，您可以指示 Solaris LDAP 命名服务在 DIT 中的非缺省位置进行搜索。另外，您可以指定使用不同的属性和对象类替代缺省架构所指定的属性和对象类。有关缺省过滤器的列表，请参见第 218 页中的“LDAP 命名服务使用的缺省过滤器”。

SSD 说明

`serviceSearchDescriptor` 属性定义 LDAP 命名服务客户机应如何以及应在何处搜索特定服务的信息。`serviceSearchDescriptor` 包含一个服务名称，其后跟一个或多个用分号分隔的 `base-scope-filter`（基-范围-过滤器）三元参数。使用这些 `base-scope-filter`（基-范围-过滤器）三元参数，可以定义仅搜索特定服务并按顺序进行搜索。如果为某个给定服务指定了多个 `base-scope-filter`（基-范围-过滤器），则该服务在查找特定条目时，将使用指定的范围和过滤器在每个基容器中进行搜索。

注- 使用 SSD 时，不会在缺省位置中搜索服务（数据库），除非该 SSD 中包括缺省位置。如果为某个服务指定了多个 SSD，将会产生不可预测的行为。

在以下示例中，Solaris LDAP 命名服务客户机将依次在 `ou=west,dc=example,dc=com` 和 `ou=east,dc=example,dc=com` 中执行单级搜索，以查找 `passwd` 服务。为了查找用户的 `username` 的 `passwd` 数据，将为每个 BaseDN 使用缺省的 LDAP 过滤器 (`&(objectClass=posixAccount)(uid=username)`)。

```
serviceSearchDescriptor: passwd:ou=west,dc=example,dc=com;ou=east,dc=example,dc=com
```

在以下示例中，Solaris LDAP 命名服务客户机将在 `ou=west,dc=example,dc=com` 中执行子树搜索以查找 `passwd` 服务。为了查找用户 `username` 的 `passwd` 数据，将使用 LDAP 过滤器 (`&(fulltimeEmployee=TRUE)(uid=username)`) 搜索 `ou=west,dc=example,dc=com` 子树。

```
serviceSearchDescriptor: passwd:ou=west,dc=example,dc=com?sub?fulltimeEmployee=TRUE
```

还可以将多个容器与一个特定的服务类型相关联。在以下示例中，服务搜索描述符指定在三个容器中搜索口令条目。

```
ou=myuser,dc=example,dc=com
ou=newuser,dc=example,dc=com
ou=extuser,dc=example,dc=com
```

请注意，在下面的示例中，SSD 中的结尾 ';' 表示 `defaultSearchBase` 将附加到相对基容器之后。

```
defaultSearchBase: dc=example,dc=com
serviceSearchDescriptor: \
passwd:ou=myuser,;ou=newuser,;ou=extuser,dc=example,dc=com
```

属性映射

使用 Solaris LDAP 命名服务时，可以重新映射其任何服务的一个或多个属性名。（Solaris LDAP 客户机使用第 14 章，[LDAP 一般参考（参考）](#) 中列出的已知属性。）如果您映射某个属性，必须确保该属性与初始属性具有相同的含义和语法。请注意，映射 `userPassword` 属性可能会产生问题。

出于多种原因，您可能需要使用架构映射。

- 希望映射现有目录服务器中的属性
- 如果您的用户名只存在大小写差异，则必须将忽略大小写的 `uid` 属性映射到不忽略大小写的属性。

此属性的格式为 `service:attribute-name=mapped-attribute-name`。

如果要为给定服务映射多个属性，则可以定义多个 `attributeMap` 属性。

在以下示例中，只要将 `uid` 和 `homeDirectory` 属性用于 `passwd` 服务时，便会使用 `employeeName` 和 `home` 属性。

```
attributeMap: passwd:uid=employeeName
attributeMap: passwd:homeDirectory=home
```

但也存在可以将 `passwd` 服务的 `gecos` 属性映射到多个属性的特殊情况。下面是一个示例：

```
attributemap: gecos=cn sn title
```

以上示例将 `gecos` 值映射到一个以空格分隔的包含 `cn`、`sn` 和 `title` 属性值的列表。

objectClass 映射

使用 Solaris LDAP 命名服务时，可以重新映射其任何服务的对象类。如果要为给定服务映射多个对象类，则可以定义多个 `objectclassMap` 属性。在以下示例中，只要使用 `posixAccount` 对象类时，便会使用 `myUnixAccount` 对象类。

```
objectclassMap: passwd:posixAccount=myUnixAccount
```

LDAP 客户机配置文件

为了简化 Solaris 客户机设置，并避免为每台客户机重新输入同样的信息，可以在目录服务器上创建一个客户机配置文件。这样，通过一个配置文件便可以为所有配置为使用该配置文件的客户机定义配置。之后配置文件属性的任何更改都会按刷新间隔所定义的频率传播到客户机。

这些客户机配置文件应存储在 LDAP 服务器上的已知位置中。给定域的根 DN 必须具有一个对象类 `nisDomainObject` 和一个包含客户机所在域的 `nisDomain` 属性。所有的配置文件都位于相对于此容器的 `ou=profile` 容器中。这些配置文件应可以匿名读取。

客户机的配置文件属性

下表列出了 Solaris LDAP 客户机的配置文件属性，这些属性可以在运行 `idsconfig` 时自动设置。有关如何手动设置客户机配置文件的信息，请参见第 179 页中的“手动初始化客户机”和 `idsconfig(1M)` 手册页。

表 9-2 客户机的配置文件属性

属性	说明
<code>cn</code>	配置文件的名称。该属性没有缺省值。必须指定该属性值。
<code>preferredServerList</code>	首选服务器的主机地址是以空格分隔的服务器地址的列表。（请勿使用主机名。）将先尝试与该列表中的服务器建立连接，然后再尝试与 <code>defaultServerList</code> 中的服务器建立连接，直到成功建立连接。该属性没有缺省值。必须至少在 <code>preferredServerList</code> 或 <code>defaultServerList</code> 中指定一台服务器。
<code>defaultServerList</code>	缺省服务器的主机地址是以空格分隔的服务器地址的列表。（请勿使用主机名。）在尝试与 <code>preferredServerList</code> 中的服务器建立连接之后，会先尝试与客户机所在子网中的缺省服务器建立连接，然后再尝试与其余的缺省服务器建立连接，直到成功建立连接。必须至少在 <code>preferredServerList</code> 或 <code>defaultServerList</code> 中指定一台服务器。只有在尝试与首选服务器列表中的服务器建立连接之后，才会尝试与该列表中的服务器建立连接。该属性没有缺省值。
<code>defaultSearchBase</code>	用于查找已知容器的相对 DN。该属性没有缺省值。不过，对于给定服务，可以使用 <code>serviceSearchDescriptor</code> 属性来覆盖该属性。
<code>defaultSearchScope</code>	定义客户机要搜索的数据库范围。可以使用 <code>serviceSearchDescriptor</code> 属性覆盖该属性。可能的值为 <code>one</code> 或 <code>sub</code> 。缺省值为 <code>one</code> 级别搜索。
<code>authenticationMethod</code>	标识了客户机使用的验证方法。缺省值为 <code>none</code> （无）。有关更多信息，请参见第 139 页中的“选择验证方法”。

表 9-2 客户机的配置文件属性 (续)

属性	说明
credentialLevel	标识了客户机进行验证时应使用的凭证的类型。选项有 anonymous、proxy 或者 self（也称为“每用户”）。缺省值为 anonymous。
serviceSearchDescriptor	定义客户机应如何以及应在何处搜索命名数据库，例如，客户机应在 DIT 中的一个点还是多个点执行查找。缺省情况下，不定义任何 SSD。
serviceAuthenticationMethod	客户机针对指定服务使用的验证方法。缺省情况下，不定义任何服务验证方法。如果某个服务未定义 serviceAuthenticationMethod，则使用 authenticationMethod 的缺省值。
attributeMap	客户机使用的属性映射。缺省情况下，不定义任何 attributeMap。
objectclassMap	客户机使用的对象类映射。缺省情况下，不定义任何 objectclassMap。
searchTimeLimit	客户机上的搜索操作在超时之前可以执行的最长时间（以秒为单位）。这不会影响在 LDAP 服务器上完成搜索所需的时间。缺省值为 30 秒。
bindTimeLimit	客户机与服务器的绑定在超时之前可以持续的最长时间（以秒为单位）。缺省值为 30 秒。
followReferrals	指定客户机是否应遵循 LDAP 引用。可能的值为 TRUE 或 FALSE。缺省值为 TRUE。
profileTTL	ldap_cachemgr(1M) 从 LDAP 服务器刷新客户机配置文件的间隔时间。缺省值为 43200 秒（即 12 小时）。如果指定的值为 0，则不刷新配置文件。

本地客户机属性

下表列出了可以使用 ldapclient 在本地设置的客户机属性。有关更多信息，请参见 [ldapclient\(1M\)](#) 手册页。

自 Solaris 10 10/09 发行版开始，提供了 enableShadowUpdate 开关。有关更多信息，请参见 [第 138 页](#)中的“enableShadowUpdate 开关”。

表 9-3 本地客户机属性

属性	说明
adminDN	指定管理凭证的管理员条目标识名。如果在客户机系统上 enableShadowUpdate 开关的值为 true，且 credentialLevel 的值不是 self，则必须指定 adminDN。

表 9-3 本地客户机属性 (续)	
属性	说明
adminPassword	指定管理凭证的管理员条目口令。如果在客户机系统上 enableShadowUpdate 开关的值为 true，且 credentialLevel 的值不是 self，则必须指定 adminPassword。
domainName	指定客户机的域名（该域将成为此客户机系统的缺省域）。该属性没有缺省值。必须指定该属性值。
proxyDN	代理的标识名。如果为客户机系统配置的 credentialLevel 为 proxy，则必须指定 proxyDN。
proxyPassword	代理的口令。如果为客户机系统配置的 credentialLevel 为 "proxy"，则必须定义 proxyPassword。
certificatePath	本地文件系统中包含证书数据库的目录。如果使用 TLS 为客户机配置了 authenticationMethod 或 serviceAuthenticationMethod，则将使用此属性。缺省值为 /var/ldap。

注 - 如果 SSD 中的 BaseDN 包含一个结尾逗号，则会将其视为 defaultSearchBase 的相对值。在执行搜索之前，会将 defaultSearchBase 的值附加在 BaseDN 后面。

ldap_cachemgr 守护进程

ldap_cachemgr 是在 LDAP 客户机上运行的守护进程。启动 LDAP 客户机时，系统会调用 ldap_cachemgr 守护进程。该守护进程执行以下主要功能：

- 以超级用户身份运行时获取对配置文件的访问权限
- 刷新服务器上的配置文件中存储的客户机配置信息，并从客户机提取这些数据
- 维护要使用的活动 LDAP 服务器的已排序列表
- 缓存不同客户机提交的一些常见查找请求，以提高查找效率
- 提高主机查找的效率
- 从 Solaris 10 10/09 发行版开始，如果 enableShadowUpdate 开关设置为 true，则可获取对所配置的管理员凭证的访问权限，并执行 shadow 数据的更新

注 - ldap_cachemgr 必须一直运行，LDAP 命名服务才能正常工作。

有关详细信息，请参阅 [ldap_cachemgr\(1M\)](#) 手册页。

LDAP 命名服务安全模型

介绍

Solaris LDAP 命名服务能够以两种不同的方式使用 LDAP 系统信息库。一种是同时用作命名服务和验证服务的源。另一种是严格用作命名数据的源。本节讨论了客户机标识、验证方法、`pam_ldap` 和 `pam_unix` 模块的概念，以及将 LDAP 系统信息库同时用作命名服务和验证服务时如何管理帐户。本节还讨论了 LDAP 命名服务如何与 Kerberos 环境（《系统管理指南：安全性服务》中的第 VI 部分，“Kerberos 服务”中的第 VI 部分“Kerberos Service”）以及 `pam_krb5(5)` 模块配合使用。

注 – 以前，如果启用了 `pam_ldap` 帐户管理，所有用户在每次登录系统时都必须提供登录口令以进行验证。因此，使用 `rsh`、`rlogin` 或 `ssh` 等工具进行的不基于口令的登录将会失败。

但是现在，`pam_ldap(5)` 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时，用户可以使用 `rsh`、`rlogin`、`rcp` 和 `ssh` 登录，而不需要提供口令。

`pam_ldap(5)` 现已修改为执行帐户管理和检索用户帐户状态，而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8，它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态，请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI)：

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
    allow (read, search, compare, proxy)
    (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

注 – 如果您使用 Kerberos 作为验证系统并将其与 LDAP 命名系统相集成，则您将能够在您的企业中通过 Kerberos 支持单点登录 (single sign on, SSO)。您还能够在按每用户或每主机查询 LDAP 命名数据时使用同一标识系统。

要访问 LDAP 系统信息库中的信息，客户机可以先通过目录服务器建立标识。此标识可以是匿名的，也可以是 LDAP 服务器能够识别的对象。基于客户机的标识和服务

的访问控制信息 (access control information, ACI)，LDAP 服务器将允许客户机读写目录信息。有关 ACI 的更多信息，请查阅所用的 Sun Java System Directory Server 版本的管理指南。

如果客户机对于任何给定的请求以非匿名方式进行连接，则必须使用自己和服务器均支持的验证方法向服务器证明自己的身份。在确认了自己的身份之后，客户机即可发出各种 LDAP 请求。

使用 `pam_ldap` 时，命名服务访问目录的方式与验证服务 (`pam_ldap`) 访问目录的方式有所不同。命名服务基于预定义的标识从目录中读取各个条目及其属性。验证服务使用用户名和口令向 LDAP 服务器验证用户身份，以确认用户是否输入了正确的口令。有关验证服务的更多信息，请参见 [pam_ldap\(5\)](#) 手册页。

当使用 Kerberos 执行验证并且在 LDAP 命名服务中也启用了验证（这是“每用户”模式所必需的）时，Kerberos 可以提供双重功能。Kerberos 向服务器验证身份，主体（用户或主机）的 Kerberos 标识用于向目录验证身份。这样，用于向系统验证身份的用户标识同样也将用于向目录验证身份，以执行查找或更新，如果需要，管理员可以在目录中使用访问控制信息 (Access Control Information, ACI) 来限制命名服务返回的结果。

传输层安全性 (Transport Layer Security, TLS)

注 - 要为 Solaris LDAP 命名服务使用 TLS，目录服务器必须分别为 LDAP 和 SSL 使用缺省端口 389 和 636。如果您的目录服务器未使用这些端口，则目前尚无法使用 TLS。

TLS 可以用来保护 LDAP 客户机与目录服务器之间的通信安全，既确保保密性又确保数据完整性。TLS 协议是安全套接字层 (Secure Sockets Layer, SSL) 协议的一个超集。Solaris LDAP 命名服务支持 TLS 连接。请注意，使用 SSL 会增加目录服务器和客户机的负荷。

您需要为 SSL 设置目录服务器。有关为 SSL 设置 Sun Java System Directory Server 的更多信息，请参阅您使用的 Sun Java System Directory Server 版本的管理指南。您还需要为 SSL 设置 LDAP 客户机。

如果使用 TLS，必须安装必要的安全数据库。特别的是，需要证书和密钥数据库文件。例如，如果采用 Netscape Communicator 的旧数据库格式，则需要 `cert7.db` 和 `key3.db` 这两个文件。或者，如果使用 Mozilla 提供的较新数据库格式，则需要 `cert8.db`、`key3.db` 和 `secmod.db` 这三个文件。`cert7.db` 或 `cert8.db` 文件中包含可信证书。`key3.db` 文件中包含客户机的密钥。即使 LDAP 命名服务客户机不使用客户机密钥，此文件也必须存在。`secmod.db` 文件中包含安全模块，如 PKCS#11 模块。如果使用的是旧格式，则不需要此文件。

有关更多信息，请参见第 181 页中的“设置 TLS 安全性”。

指定客户机凭证级别

LDAP 命名服务客户机根据客户机的凭证级别向 LDAP 服务器验证身份。可以为 LDAP 客户机指定的凭证级别有四个，客户机将使用它们向目录服务器验证身份。

- `anonymous`
- `proxy`
- `proxy anonymous`
- `self`（在本文档中称为“每用户”）

Anonymous

如果使用 `anonymous` 访问，则只能访问所有人都能使用的数据。在 `anonymous` 模式中，不能执行 LDAP BIND 操作。此外，还应考虑安全问题。允许对目录的某些部分进行 `anonymous` 访问，意味着任何具有该目录访问权限的人都有读取访问权限。如果使用 `anonymous` 凭证级别，您需要允许对所有 LDAP 命名条目和属性进行读取访问。



注意 – 绝对不要允许对目录进行 `anonymous` 写入，因为任何人都可以对其具有写入访问权限的 DIT 中更改信息，包括其他用户的口令或他们自己的身份。

注 – Sun Java System Directory Server 允许您基于 IP 地址、DNS 名称、验证方法和一天中的时段对访问进行限制。您可能希望通过进一步的约束来限制访问。有关更多信息，请参见您使用的 Sun Java System Directory Server 版本的管理指南中的“管理访问控制”。

Proxy

客户机使用单个代理帐户绑定到目录或向目录验证身份。此代理帐户可以是任何允许绑定到目录的条目。此代理帐户需要有足够的权限才能在 LDAP 服务器上执行命名服务功能。此代理帐户是一个按系统共享的资源。也就是说，每个使用代理访问权限登录到某个系统的用户（包括超级用户）与该系统上的所有其他用户看到的结果相同。您需要在使用 `proxy` 凭证级别的每台客户机上配置 `proxyDN` 和 `proxyPassword`。经过加密的 `proxyPassword` 存储在客户机本地。您可以为不同的客户机组设置不同的代理。例如，您可以为所有销售客户机配置一个代理来访问整个公司都可访问的目录以及销售目录，同时阻止销售客户机访问包含工资信息的人力资源目录。或者，在最极端的情况中，您可以为每个客户机指定不同的代理，或者为所有客户机仅指定一个代理。典型的 LDAP 部署应当介于这两种极端情况之间。请仔细考虑您的选择。代理太少可能会限制您对用户访问资源进行控制的能力。但是，代理太多，又会增加系统设置和维护的难度。您需要根据自己的环境向代理用户授予合适的权限。有关如何确定哪种验证方法最适合您的配置的信息，请参见第 139 页中的“凭证存储”。

如果某个代理用户的口令发生了更改，您需要在使用该代理用户的每台客户机上更新该口令。如果您为 LDAP 帐户使用口令生命周期功能，请确保为代理用户关闭此功能。

注 – 请注意，`proxy` 凭证级别应用于任意给定系统上的所有用户和进程。如果两个用户需要使用不同的命名策略，他们必须使用不同的计算机，或者必须使用“每用户”验证模式。

另外，如果客户机使用 `proxy` 凭证进行验证，则 `proxyDN` 在所有服务器上都必须具有相同的 `proxyPassword`。

Proxy anonymous

`proxy anonymous` 是一个多值条目，它定义了多个凭证级别。指定了 `proxy anonymous` 级别的客户机将首先尝试使用其代理标识进行验证。如果客户机由于某种原因（例如，用户锁定、口令失效）而无法作为代理用户进行验证，客户机将使用匿名访问。这可能会导致服务的级别有所不同，具体取决于目录的配置方式。

每用户

“每用户”(self) 验证方式在向目录服务器验证身份时使用 Kerberos 标识（主体）来针对每个用户或每个系统执行查找。使用“每用户”验证，系统管理员可以使用访问控制指令 (access control instruction, ACI)、访问控制列表 (access control list, ACL)、角色、组或其他目录访问控制机制来向特定用户或系统授予或拒绝对特定命名服务数据的访问权限。

注 – 在配置“每用户”模式时，用于启用该模式的配置值是“self”，它表示“每用户”模式。

要使用“每用户”验证模型，必须部署 Kerberos 单点登录服务。此外，部署中使用的一个或多个目录服务器必须支持 SASL 和 SASL/GSSAPI 验证机制。因为 Kerberos 预期使用文件和 DNS（而非 LDAP）进行主机名查找，因此应当在该环境中部署 DNS。另外，要使用“每用户”验证，还必须启用 `nscd`。在该配置中，`nscd` 守护进程不是可选组件。

enableShadowUpdate 开关

从 Solaris 10 10/09 发行版开始，如果在客户机上 `enableShadowUpdate` 开关设置为 `true`，则使用管理凭证来更新阴影数据。阴影数据存储在目录服务器上的 `shadowAccount` 对象类中。管理凭证由 `adminDN` 和 `adminPassword` 属性的值定义，如第 133 页中的“本地客户机属性”中所述。这些管理凭证不用于任何其他用途。

管理凭证具有与 `Proxy` 凭证类似的属性。不同之处在于，对于管理凭证，用户必须具有区域的所有特权或者有效的 `root` UID，才能读取或更新阴影数据。管理凭证可以指定给任何允许绑定到目录的条目。不过，**不要**使用 LDAP 服务器的同一目录管理器标识 (`cn=Directory Manager`)。

具有管理凭证的该条目必须具有足够的访问权限才能读取和写入目录中的阴影数据。因为该条目是按系统共享的资源，因此必须在每个客户机上配置 `adminDN` 和

`adminPassword` 属性。经过加密的 `adminPassword` 存储在客户机本地。口令使用为客户机配置的相同验证方法。给定系统上的所有用户和进程使用管理凭证读取和更新阴影数据。

凭证存储

如果将客户机配置为使用代理标识，则客户机将其 `proxyDN` 和 `proxyPassword` 存储在 `/var/ldap/ldap_client_cred` 中。为了增强安全性，该文件仅限超级用户访问，并且 `proxyPassword` 的值也经过了加密。尽管以前的 LDAP 实现已将代理凭证存储在客户机的配置文件中，但是 Solaris 9 LDAP 命名服务却不是这样。初始化过程中使用 `ldapclient` 设置的所有代理凭证都存储在本地。这样便提高了代理的 DN 和口令信息的安全性。有关设置客户机配置文件的更多信息，请参见第 12 章，设置 LDAP 客户机（任务）。

类似地，如果您将客户机配置为启用阴影数据更新，且客户机凭证级别不是 `self`，则客户机将其 `adminDN` 和 `adminPassword` 属性保存在本地的 `/var/ldap/ldap_client_cred` 文件中。`adminPassword` 的值也是加密的，且只能由 `ldap_cachemgr` 守护进程使用。

如果您将客户机配置为使用“每用户”验证，则在验证期间将使用每个主体（每个用户或主机）的 Kerberos 标识和 Kerberos 票证信息。在这样的环境中，目录服务器将 Kerberos 主体映射到一个 DN，并使用 Kerberos 凭证向该 DN 验证身份。然后，目录服务器根据需要使用其访问控制指令（access control instruction, ACI）机制来允许或拒绝对命名服务数据的访问。在这种情况下，Kerberos 票证信息用于向目录服务器验证身份，系统不会存储用于验证 DN 或口令。因此，对于这种类型的配置来说，在使用 `ldapclient` 命令初始化客户机时，不需要指定 `adminDN` 和 `adminPassword` 属性。

选择验证方法

为客户机指定 `proxy` 或 `proxy-anonymous` 凭证级别时，还需要选择代理向目录服务器验证身份所使用的方法。缺省情况下，验证方法是 `none`，它表示匿名访问。验证方法还可以有与之关联的传输安全选项。

与凭证级别一样，验证方法可以是多值的。例如，在客户机配置文件中，您可以指定客户机首先尝试使用由 TLS 保护的 `simple` 方法进行绑定。如果失败，客户机将尝试使用 `sasl/digest-MD5` 方法进行绑定。这样，`authenticationMethod` 将为 `tls:simple;sasl/digest-MD5`。

LDAP 命名服务支持某些简单身份验证和安全层（Simple Authentication and Security Layer, SASL）机制。这些机制无需 TLS 便可安全地交换口令。但是，这些机制不提供数据完整性和保密性。有关 SASL 的信息，请参见 RFC 2222。

支持的验证机制如下所示：

- `none`

客户机不向目录验证身份。这等效于 `anonymous` 凭证级别。

- **simple**

如果客户机系统使用 **simple** 验证方法，它将通过以明文形式发送用户口令来绑定到服务器。因此，除非会话受 IPsec 保护，否则口令很容易被窥探。使用 **simple** 验证方法的主要优点在于所有目录服务器都支持该验证方法且该方法容易设置。

- **sasl/digest-MD5**

客户机的口令在验证期间会得到保护，但会话不会被加密。某些目录服务器（包括 Sun Java System Directory Server）还支持 **sasl/digest-MD5** 验证方法。**digest-MD5** 的主要优点在于，在验证过程中，口令不会以明文形式通过线路传输，因此比 **simple** 验证方法更安全。有关 **digest-MD5** 的信息，请参见 RFC 2831。因为提高了安全性，**digest-MD5** 被视为是以 **cram-MD5** 为基础的改进。

使用 **sasl/digest-MD5** 时，验证过程是安全的，但会话无法受到保护。

注 – 如果您使用的是 Sun Java System Directory Server，则口令**必须以明文形式存储**在目录中。

- **sasl/cram-MD5**

使用 **sasl/cram-MD5** 执行验证时，不会对 LDAP 会话进行加密，但是在验证期间会保护客户机的口令。

有关 **cram-MD5** 验证方法的信息，请参见 RFC 2195。只有部分目录服务器支持 **cram-MD5**。例如，Sun Java System Directory Server 就不支持 **cram-MD5**。

- **sasl/GSSAPI**

此验证方法与 **self** 凭证模式一起使用可启用“每用户”查找。为使用客户机凭证而为每位用户指定的 **nscd** 使用 **sasl/GSSAPI** 方法和客户机的 Kerberos 凭证绑定到目录服务器。在目录服务器中，可以对每位用户的访问进行控制。

- **tls:simple**

客户机使用 **simple** 方法进行绑定，并且对会话进行加密。口令也将受到保护。

- **tls:sasl/cram-MD5**

对 LDAP 会话进行加密，客户机使用 **sasl/cram-MD5** 向目录服务器验证身份。

- **tls:sasl/digest-MD5**

对 LDAP 会话进行加密，客户机使用 **sasl/digest-MD5** 向目录服务器验证身份。



注意 – Sun Java System Directory Server 要求口令以明文形式存储才能使用 **digest-MD5**。如果验证方法设置为 **sasl/digest-MD5** 或 **tls:sasl/digest-MD5**，则代理用户的口令需要以明文形式存储。要特别注意的是，如果 **userPassword** 属性以明文形式存储，它应具有正确的 ACI，以便它将是不可读的。

下表概述了各种验证方法及其各自的特征。

表 9-4 验证方法

	绑定	线路上的口令	Sun Java System Directory Server 上 的口令	会话
none	否	N/A	N/A	不加密
simple	是	明文	任意	不加密
sasl/digest-MD5	是	加密	明文	不加密
sasl/cram-MD5	是	加密	N/A	不加密
sasl/GSSAPI	是	Kerberos	Kerberos	加密
tls:simple	是	加密	任意	加密
tls:sasl/cram-MD5	是	加密	N/A	加密
tls:sasl/digest-MD5	是	加密	明文	加密

验证和服务

可以在 `serviceAuthenticationMethod` 属性中为给定的服务指定验证方法。目前下列服务支持此功能：

- `passwd-cmd`
`passwd(1)` 使用此服务更改登录口令和口令属性。
- `keyserv`
`chkey(1)` 和 `newkey(1M)` 实用程序使用此服务创建和更改用户的 Diffie-Hellman 密钥对。
- `pam_ldap`
此服务用于使用 `pam_ldap(5)` 验证用户身份。
`pam_ldap` 支持帐户管理。

注 – 如果该服务未设置 `serviceAuthenticationMethod`，则缺省情况下将使用 `authenticationMethod` 属性的值。

注 – 在“每用户”模式中，第 143 页中的“`pam_krb5 服务模块`”(`pam Kerberos`) 用作验证服务。在此操作模式中，`ServiceAuthenticationMethod` 不是必需的。

注 – 如果 `enableShadowUpdate` 开关设置为 `true`，`ldap_cachemgr` 守护进程将使用 `passwd-cmd` 的 `serviceAuthenticationMethod` 参数中定义的验证方法（如果定义了该方法的话）绑定到 LDAP 服务器。否则，将使用 `authenticationMethod`。守护进程不会使用 `none` 验证方法。

下面的示例显示了某个客户机配置文件的一部分，其中，用户将使用 `sasl/digest-MD5` 来向目录服务器验证身份，但将使用 `SSL` 会话更改其口令。

```
serviceAuthenticationMethod=pam_ldap:sasl/digest-MD5
serviceAuthenticationMethod=passwd-cmd:tls:simple
```

可插拔验证方法

通过使用 `PAM` 框架，您可以在几种验证服务之间进行选择，包括 `pam_unix`、`pam_krb5` 和 `pam_ldap`。

如果使用“每用户”验证方法，则必须启用上面列出的三种方法中最强的验证服务 `pam_krb5`。请参见 [pam_krb5\(5\)](#) 和 《系统管理指南：安全性服务》。

即使未启用“每用户”验证，也可以使用 `pam_krb5` 验证系统。如果使用 `proxy` 或 `anonymous` 凭证级别来访问目录服务器数据，则无法基于每个用户限制对目录数据的访问。

当使用 `anonymous` 或 `proxy` 验证方法时，建议优先使用 `pam_ldap` 而非 `pam_unix`，因为前者的灵活性更强、支持更强大的验证方法并且能够使用帐户管理。

pam_unix 服务模块

如果您未更改过 [pam.conf\(4\)](#) 文件，则缺省情况下会启动 `pam_unix` 功能。

注 – `pam_unix` 模块已被删除，`Solaris` 将不再支持该模块。但是，`Solaris` 提供了一组其他服务模块，这些模块将提供等效或更强的功能。因此，在本指南中，`pam_unix` 是指等效的功能，而并非指 `pam_unix` 模块本身。

下面列出了提供等效 `pam_unix` 功能的模块：

```
pam_authok_check(5)
pam_authok_get(5)
pam_authok_store(5)
pam_dhkeys(5)
pam_passwd_auth(5)
```

```
pam_unix_account(5)
pam_unix_auth(5)
pam_unix_cred(5)
pam_unix_session(5)
```

`pam_unix` 遵循传统的 UNIX 验证模型，如下所述。

1. 客户机从名称服务检索用户的加密口令。
2. 系统提示用户输入其口令。
3. 对用户的口令进行加密。
4. 客户机比较这两个经过加密的口令，确定用户是否应通过验证。

另外，使用 `pam_unix` 时还存在以下两个限制：

- 口令必须以 UNIX `crypt` 格式存储，而不应采用其他任何加密方法（包括明文）存储。
- 名称服务必须能够读取 `userPassword` 属性。

例如，如果将凭证级别设置为 `anonymous`，则任何人都必须能够读取 `userPassword` 属性。同样，如果将凭证级别设置为 `proxy`，则代理用户必须能够读取 `userPassword` 属性。

注 – `pam_unix` 与 `sasl` 验证方法 `digest-MD5` 不兼容，因为 Sun Java System Directory Server 要求口令必须以明文形式存储才能使用 `digest-MD5`。`pam_unix` 则要求口令以 `crypt` 格式存储。

注 – 从 Solaris 10 10/09 发行版开始，当 `enableShadowUpdate` 开关设置为 `true` 时，`pam_unix` 可以支持帐户管理。对远程 LDAP 用户帐户实施控制的方式与对 `passwd` 和 `shadow` 文件中定义的本地用户帐户实施控制的方式是相同的。在 `enableShadowUpdate` 模式中，对于 LDAP 帐户，系统将更新并使用 LDAP 服务器上的阴影数据执行口令生命期和帐户锁定功能。当然，本地帐户的阴影数据仅应用于本地客户机系统，而 LDAP 用户帐户的阴影数据将应用于所有客户机系统上的用户。

只有本地客户机支持口令历史记录检查，LDAP 用户帐户不支持此功能。

pam_krb5 服务模块

请参阅 `pam_krb5(5)` 和《系统管理指南：安全性服务》。

pam_ldap 服务模块

实现 `pam_ldap` 时，用户使用 `pam_ldap` 的 `serviceAuthenticationMethod` 参数（如果存在的话）中定义的验证方法绑定到 LDAP 服务器。如果不存在，将使用 `authenticationMethod`。

如果 `pam_ldap` 能够使用用户的身份和提供的口令绑定到服务器，它将验证用户的身份。

注 – 以前，如果启用了 `pam_ldap` 帐户管理，所有用户在每次登录系统时都必须提供登录口令以进行验证。因此，使用 `rsh`、`rlogin` 或 `ssh` 等工具进行的不基于口令的登录将会失败。

但是现在，`pam_ldap(5)` 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时，用户可以使用 `rsh`、`rlogin`、`rcp` 和 `ssh` 登录，而不需要提供口令。

`pam_ldap(5)` 现已修改为执行帐户管理和检索用户帐户状态，而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8，它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态，请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI)：

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
    allow (read, search, compare, proxy)
    (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

`pam_ldap` 不读取 `userPassword` 属性。因此，除非有其他客户机正在使用 `pam_unix`，否则无需授予对 `userPassword` 属性的读取访问权限。此外，`pam_ldap` 不支持 `none` 验证方法。因此，您必须定义 `serviceAuthenticationMethod` 或 `authenticationMethod` 属性，以便客户机可以使用 `pam_ldap`。有关更多信息，请参见 `pam_ldap(5)` 手册页。



注意 – 如果使用 `simple` 验证方法，`userPassword` 属性在传输过程中可能会被第三方读取。

请参见第 197 页中的“`pam_ldap` 的示例 `pam.conf` 文件”。

下表汇总了 `pam_unix`、`pam_ldap` 与 `pam_krb5` 之间的主要区别。

表 9-5 使用 `pam_unix`、`pam_ldap` 和 `pam_krb5` 时 LDAP 中的验证行为

	pam_unix	pam_ldap	pam_krb5
发送的口令	使用 <code>passwd</code> 服务验证方法	使用 <code>passwd</code> 服务验证方法	使用 Kerberos 单点登录技术，不需要口令

表 9-5 使用 pam_unix、pam_ldap 和 pam_krb5 时 LDAP 中的验证行为 (续)

	pam_unix	pam_ldap	pam_krb5
发送的新口令	加密	不加密（除非使用了 TLS）	使用 Kerberos，不通过线路发送口令
存储的新口令	crypt 格式	Sun Java System Directory Server 中定义的口令存储方案	口令由 Kerberos 管理
是否需要读取口令？	是	否	否
更改口令之后，是否与 sasl/digest-MD5 兼容	否。口令不以 clear 形式存储。用户无法验证身份。	是。只要将缺省的存储方案设置为 clear，用户即可验证身份。	否。使用了 sasl/GSSAPI。线路上没有口令，目录服务器中也不存储口令，除非所使用的 Kerberos kdc 在 LDAP 目录服务器中管理其口令数据库。
是否支持口令策略？	是。enableShadowUpdate 必须设置为 true。	是（如果进行了这样的配置）。	请参见 pam_krb5(5)，Kerberos V5 帐户管理模块。

PAM 和更改口令

使用 passwd 命令可以更改口令。如果 enableShadowUpdate 开关未设置为 true，则用户必须可以对 userPassword 属性进行写入。如果 enableShadowUpdate 开关设置为 true，则管理凭证必须能够更新 userPassword 属性。请记住，对于此操作，passwd-cmd 的 serviceAuthenticationMethod 会覆盖 authenticationMethod。当前口令在传输过程中可能是未加密的，具体取决于所使用的验证方法。

对于 pam_unix，新的 userPassword 属性在写入到 LDAP 之前会使用 UNIX crypt 格式进行加密和标记。因此，无论使用哪种验证方法绑定到服务器，在传输过程中都会对新口令进行加密。有关更多信息，请参见 pam_authtok_store(5) 手册页。

如果 enableShadowUpdate 开关设置为 true，则当用户口令被更改时，pam_unix 还会更新相关的 shadow 信息。pam_unix 更新的本地 shadow 文件中的 shadow 字段与更改本地用户口令时 pam_unix 更新的 shadow 字段相同。

从 Solaris 10 软件发行版开始，pam_ldap 不再支持口令更新。之前建议使用的带有 server_policy 选项的 pam_authtok_store 现已取代 pam_ldap 口令更新功能。使用 pam_authtok_store 时，新口令将以明文形式发送到 LDAP 服务器。因此，为了确保保密性，请使用 TLS。如果不使用 TLS，新的 userPassword 将很容易被窥探。如果您为 Sun Java System Directory Server 设置了未标记的口令，则该软件会使用 passwordStorageScheme 属性对口令进行加密。有关 passwordStorageScheme 的更多信息，请参见所用的 Sun Java System Directory Server 版本的管理指南中有关用户帐户管理的章节。

注 – 设置 `passwordStorageScheme` 属性时，需要考虑以下配置问题。如果 NIS、NIS+ 或另一台使用 `pam_unix` 的客户机使用 LDAP 作为系统信息库，则 `passwordStorageScheme` 必须为 `crypt`。此外，如果将 `pam_ldap` 与 `sasl/digest-MD5` 用于 Sun Java System Directory Server，则 `passwordStorageScheme` 必须设置为 `clear`（明文）。

帐户管理

如果您选择 `pam_krb5` 作为帐户和口令管理系统，Kerberos 环境将负责管理您所有的帐户、口令、帐户锁定和其他帐户管理详细信息。请参阅 [pam_krb5\(5\)](#) 和《系统管理指南：安全性服务》。

如果您不使用 `pam_krb5`，则可以配置 LDAP 命名服务以利用 Sun Java System Directory Server 中提供的口令和帐户锁定策略支持。您可以配置 [pam_ldap\(5\)](#) 以支持用户帐户管理。与正确的 PAM 配置一起使用时，[passwd\(1\)](#) 可强制实施由 Sun Java System Directory Server 口令策略设置的口令语法规则。

通过 [pam_ldap\(5\)](#)，可以支持以下帐户管理功能。这些功能取决于 Sun Java System Directory Server 的口令和帐户锁定策略配置。您可以根据需要启用任意多的功能。

- 口令生命期和到期通知
用户必须按照计划更改其口令。如果在配置的时间内未更改口令，口令将过期。过期的口令会导致用户验证失败。
在过期警告期间内登录时，用户每次都会看到一条警告消息。该消息指出口令将在多少小时或多少天之后到期。
- 口令语法检查
新口令必须符合口令的最低长度要求。此外，口令不能与用户目录条目中的 `uid`、`cn`、`sn` 或 `mail` 属性的值相同。
- 口令历史记录检查
用户不能重复使用口令。如果用户尝试将口令更改为以前使用过的口令，[passwd\(1\)](#) 将失败。LDAP 管理员可以配置保留在服务器历史记录列表中的口令数目。
- 用户帐户锁定
连续验证失败达到指定次数后，会锁定用户帐户。如果管理员取消激活了某个用户的帐户，该用户也会被锁定。在帐户锁定时间结束或管理员重新激活帐户之前，验证将一直失败。

注 – 上述帐户管理功能只能用于 Sun Java System Directory Server。有关在服务器上配置口令和帐户锁定策略的信息，请参见所用 Sun Java System Directory Server 版本的管理指南中的“用户帐户管理”一章。另请参见第 198 页中的“为帐户管理配置的 `pam_ldap` 的示例 `pam_conf` 文件”。请勿为 `proxy` 帐户启用帐户管理。

在 Sun Java System Directory Server 上配置口令和帐户锁定策略之前，请确保所有主机都使用具有 `pam_ldap` 帐户管理功能的“最新”LDAP 客户机。

另外，确保客户机上具有已正确配置的 `pam.conf(4)` 文件。否则，当 `proxy` 或用户口令到期后，LDAP 命名服务将无法工作。

注 – 以前，如果启用了 `pam_ldap` 帐户管理，所有用户在每次登录系统时都必须提供登录口令以进行验证。因此，使用 `rsh`、`rlogin` 或 `ssh` 等工具进行的不基于口令的登录将会失败。

但是现在，`pam_ldap(5)` 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时，用户可以使用 `rsh`、`rlogin`、`rcp` 和 `ssh` 登录，而不需要提供口令。

`pam_ldap(5)` 现已修改为执行帐户管理和检索用户帐户状态，而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8，它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态，请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI)：

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
    allow (read, search, compare, proxy)
    (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

使用 `pam_unix` 管理帐户

自 Solaris 10 10/09 发行版开始，提供了 `enableShadowUpdate` 开关。当 `enableShadowUpdate` 设置为 `true` 时，LDAP 提供与文件命名服务相同的帐户管理功能。

如果在客户机上 `enableShadowUpdate` 开关设置为 `true`，则对于本地帐户可用的帐户管理功能也可用于 LDAP 帐户。这些功能包括口令生命期、帐户到期和通知、锁定登录失败的帐户等等。另外，LDAP 中现在支持 `passwd` 命令的 `-dluNfnwx` 选项。因此，LDAP 命名服务支持文件命名服务中的 `passwd` 命令以及 `pam_unix*` 模块的完整功能。`enableShadowUpdate` 开关为在文件和 LDAP 作用域内定义的用户提供了一种统一管理帐户的方法。

为防止用户修改自己的帐户管理数据并借此规避口令策略，LDAP 服务器被配置为阻止用户对服务器上自己的阴影数据进行写入访问。具有管理凭证的管理员将负责为客户机系统执行阴影数据更新。不过，这种配置与 `pam_ldap` 模块冲突，因为该模块要求口令可由用户修改。因此，使用 `pam_ldap` 管理帐户与使用 `pam_unix` 管理帐户不兼容。



注意 – 不要在同一 LDAP 命名域中同时使用 `pam_ldap` 和 `pam_unix`。所有客户机要么都使用 `pam_ldap`，要么都使用 `pam_unix`。因为存在这种限制，您可能需要专用的 LDAP 服务器。例如，某个 web 或电子邮件应用程序可能希望用户能够在 LDAP 服务器上更改自己的口令。

`enableShadowUpdate` 的实现还要求管理凭证（`adminDN` 以及 `adminPassword`）存储在每个客户机本地。尽管 `adminPassword` 已加密，并且只能由 `ldap_cachemgr` 守护进程从 `/var/ldap/ldap_client_cred` 文件中读取，但还是必须采取特殊的措施来保护管理凭证。要保护凭证，请使其不同于服务器的目录管理器（`cn=directory manager`）。另一种可以使用的保护措施是，对于 `passwd-cmd` 服务，将 `serviceAuthenticationMethod` 的值配置为 `tls:simple` 或更高值，使 `adminPassword` 的值不以明文发送，从而不易被窥探。

与使用 `pam_ldap` 进行帐户管理不同，使用 `pam_unix` 进行帐户管理时不需要更改 `/etc/pam.conf` 文件。使用缺省的 `/etc/pam.conf` 文件足以满足要求。

LDAP 命名服务的规划要求（任务）

本章讨论在开始设置和安装服务器与客户机之前应进行的高级规划。

本章包含以下主题：

- 第 149 页中的“LDAP 规划概述”
- 第 149 页中的“规划 LDAP 网络模型”
- 第 150 页中的“规划目录信息树 (Directory Information Tree, DIT)”
- 第 151 页中的“LDAP 和副本服务器”
- 第 152 页中的“规划 LDAP 安全模型”
- 第 154 页中的“规划 LDAP 的客户机配置文件和缺省属性值”
- 第 154 页中的“规划 LDAP 数据置备”

LDAP 规划概述

LDAP 客户机配置文件是配置信息的集合，LDAP 客户机使用这些配置信息访问有关支持 LDAP 服务器的 LDAP 命名服务信息。本章讨论如何规划 LDAP 命名服务的各个方面，其中包括网络模型、目录信息树、安全模型和各种配置文件属性的缺省值，最后讨论如何准备置备数据。

规划 LDAP 网络模型

出于可用性和性能方面的考虑，公司范围的网络的每个子网都应当有各自的 LDAP 服务器，以便为子网中的所有 LDAP 客户机提供服务。这其中只需要有一台服务器作为主 LDAP 服务器即可，其余服务器都可以是主服务器的副本。

要规划网络配置，请考虑可用服务器的数目，客户机如何与各台服务器通信，以及按什么顺序访问各台服务器。如果每个子网只有一台服务器，则可以使用 `defaultServerList` 属性列出所有的服务器，并由 LDAP 客户机排列和安排访问顺序。如果由于速度或数据管理方面的原因而需要按特定顺序访问服务器，您应当使用

`preferredServerList` 属性定义固定的服务器访问顺序。请注意，为了减少主服务器上的负荷，不应将主服务器放在其中任何一个列表中。

此外，在规划服务器和网络配置时，可能会发现还有以下三个值得考虑的属性。`bindTimeLimit` 属性可用于设置 TCP 连接请求的超时值。`searchTimeLimit` 属性可用于设置 LDAP 搜索操作的超时值。`profileTTL` 属性可用于控制 LDAP 客户机从服务器下载其配置文件的频率。对于较慢或不稳定的网络，`bindTimeLimit` 和 `searchTimeLimit` 属性的值可能需要大于缺省值。在部署的早期测试阶段，您可能需要减小 `profileTTL` 属性的值，以便客户机提取对存储在 LDAP 服务器中的配置文件的频繁更改。

规划目录信息树 (Directory Information Tree, DIT)

LDAP 命名服务具有一个缺省的目录信息树 (Directory Information Tree, DIT) 和一个关联的缺省架构。例如，`ou=people` 容器包含用户帐户、口令和 `shadow` 信息。`ou=hosts` 容器包含有关网络中各个系统的信息。`ou=people` 容器中的每一项都包含 `objectclass posixAccount` 和 `shadowAccount`。

缺省的 DIT 是一个设计完善的目录结构，并基于开放标准。它应该能够满足大多数命名服务的需要，因此建议不对其进行更改而直接使用。如果选择使用缺省的 DIT，唯一需要确定的就是要从目录树中的哪个节点（基本 DN）中搜索给定域的命名服务信息。此节点使用 `defaultSearchBase` 属性来指定。另外，您可能还要设置 `defaultSearchScope` 属性，以通知客户机应当在哪个搜索范围内执行命名服务查找。是仅搜索该 DN 下的一层 (one)，还是搜索该 DN 下的所有子树 (sub)？

但有时候，LDAP 命名服务需要更大的灵活性，以便可以处理现有的 DIT 或命名服务数据分散在目录树中的复杂 DIT。例如，用户帐户项可能存在于树的不同部分。客户机配置文件中的 `serviceSearchDescriptor`、`attributeMap` 和 `objectclassMap` 属性旨在处理这些情况。

使用服务搜索描述符可以覆盖特定服务的缺省搜索基础、搜索范围和搜索过滤器。请参见第 130 页中的“服务搜索描述符 (Service Search Descriptor, SSD) 和架构映射”。

`AttributeMap` 和 `ObjectclassMap` 属性提供了一种进行架构映射的方法。这些属性使 LDAP 命名服务可以处理现有的 DIT。例如，您可以将 `posixAccount` 对象类映射到现有的 `myAccount` 对象类，也可以将 `posixAccount` 对象类中的属性映射到 `myAccount` 对象类中的属性。

多台目录服务器

多台 LDAP 服务器可以为一个 DIT 提供服务。例如，DIT 的某些子树驻留在其他 LDAP 服务器上。这种情况下，一台 LDAP 服务器可能会指示 LDAP 客户机向其他服务器请求虽然已知、但自身数据库中没有的命名数据。如果您规划这种 DIT 配置，应当设置客

户机的配置文件属性 `followReferrals`，指示 LDAP 命名服务遵循服务器引用，以继续执行命名服务查找。但是，应尽可能使给定域的所有命名数据都位于一个目录服务器上。

如果希望客户机在大多数时间访问只读副本，仅在必要时才按照引用访问读/写主服务器，则引用功能可能非常有用。这样，主服务器将不会因原本可以由副本服务器处理的过多请求而过载。

与其他应用程序共享数据

要充分利用 LDAP，每个逻辑项都应该有一个 LDAP 项。例如，对于用户，您不但可以拥有公司的用户数据库信息，还可以拥有 Solaris 帐户信息，还可能拥有特定于应用程序的数据。由于 `posixAccount` 和 `shadowAccount` 是辅助对象类，因此它们可以添加到目录内的任何项中。这将需要您仔细规划、设置和管理。

选择目录后缀

有关如何选择适当的目录后缀的信息，请参见 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）文档。

LDAP 和副本服务器

设置副本服务器时可以采用三种不同的策略。

- 单主复制
- 浮动主复制
- 多主复制

单主

在单主复制中，对于任何给定的分区或非分区网络，仅有一台主服务器保存有目录项的可写副本。所有副本服务器拥有目录项的只读副本。副本服务器和主服务器都可以执行搜索、比较和绑定操作，但只有主服务器可以执行写入操作。

单主复制策略的潜在缺点是主服务器会出现单点故障。如果主服务器关闭，任何副本都无法处理写入操作。

浮动主

浮动主策略与单主策略类似，即在任何给定时间内，对于给定的分区或非分区网络，仅有一台主服务器具有写入功能。但是，实现浮动主策略时，如果主服务器关闭，会有一台副本服务器通过某种算法自动转换为主服务器。

浮动主复制策略的潜在缺点是，如果网络成为分区网络并且分区任一端的副本服务器成为主服务器，则网络重新连接后，协调新主服务器的过程会非常复杂。

多主

如果采用多主复制，多台主服务器各自拥有目录项数据的读写副本。尽管多主策略消除了单点故障问题，但服务器之间仍会发生更新冲突。换句话说，如果在两台主服务器上几乎同时修改某项的属性，则必须备有解决更新冲突的策略，如“最后写入者取得权限”。

有关如何设置副本服务器的信息，请参阅所使用的 Sun Java System Directory Server 版本的管理指南。

规划 LDAP 安全模型

要规划安全模型，首先应当考虑 LDAP 客户机与 LDAP 服务器通信时应使用什么身份。例如，您必须确定是希望使用企业范围内的单点登录解决方案（该方案不通过线路发送口令），还是希望基于每个用户对数据进行线路加密并能够访问目录服务器中的控制数据结果。您还必须确定是否希望使用强验证来保护通过网络传输的用户口令，以及/或者是否需要加密 LDAP 客户机与 LDAP 服务器之间的会话来保护传输的 LDAP 数据。

配置文件中的 `credentialLevel` 和 `authenticationMethod` 属性用于实现此目的。`credentialLevel` 有四种可能的凭证级别：`anonymous`、`proxy`、`proxy anonymous` 和 `self`。有关 LDAP 命名服务安全概念的详细讨论，请参见第 135 页中的[“LDAP 命名服务安全模型”](#)。

注 – 以前，如果启用了 `pam_ldap` 帐户管理，所有用户在每次登录系统时都必须提供登录口令以进行验证。因此，使用 `rsh`、`rlogin` 或 `ssh` 等工具进行的不基于口令的登录将会失败。

但是现在，`pam_ldap(5)` 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时，用户可以使用 `rsh`、`rlogin`、`rcp` 和 `ssh` 登录，而不需要提供口令。

`pam_ldap(5)` 现已修改为执行帐户管理和检索用户帐户状态，而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8，它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态，请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI)：

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
      allow (read, search, compare, proxy)
      (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

注 – 如果您启用 `pam_krb5` 和 Kerberos 作为企业范围内的单点登录解决方案，可以设计一个仅在启动会话时需要提供一次登录口令的系统。有关详细信息，请参见《[系统管理指南：安全性服务](#)》。如果您启用了 Kerberos，通常也需要启用 DNS。有关详细信息，请参见本手册中有关 DNS 的各章。

下面列出了在规划安全模型时需要做出的主要决策。

- 是否要使用 Kerberos 和按用户验证？
- LDAP 客户机将使用哪个凭证级别以及哪些验证方法？
- 是否要使用 TLS？
- 是否需要与 NIS 或 NIS+ 向后兼容？也就是说，客户机是要使用 `pam_unix` 还是 `pam_ldap`？
- 如何设置服务器的 `passwordStorageScheme` 属性？
- 如何设置访问控制信息？
有关 ACI 的更多信息，请查阅所用的 Sun Java System Directory Server 版本的管理指南。
- 客户机是要使用 `pam_unix` 还是 `pam_ldap` 进行帐户管理？

规划 LDAP 的客户机配置文件和缺省属性值

完成以上规划步骤（网络模型、DIT 和安全模型）之后，应当对以下配置文件属性的值有一些了解。

- cn
- defaultServerList
- preferredServerList
- bindTimeLimit
- searchTimeLimit
- profileTTL
- defaultSearchBase
- defaultSearchScope
- serviceSearchDescriptor
- attributeMap
- objectclassMap
- followReferrals
- credentialLevel
- authenticationMethod
- serviceCredentialLevel
- serviceAuthenticationMethod

在以上各属性中，仅有 cn、defaultServerList 和 defaultSearchBase 是必需的。这些属性没有缺省值。其余属性是可选的，其中有些具有缺省值。

有关设置 LDAP 客户机的更多信息，请参见第 12 章，[设置 LDAP 客户机（任务）](#)。

规划 LDAP 数据置备

`<indexterm><primary>数据置备</primary></indexterm>`要使用数据置备 LDAP 服务器，请在为 LDAP 服务器配置了正确的 DIT 和架构后，使用新的 `<command>ldapaddent</command>` 工具。使用新的 `ldapaddent` 工具。此工具将根据 `/etc` 文件在 LDAP 容器中创建与文件对应的各项。使用此工具，可以将数据置备到以下数据类型的容器中：`aliases`、`auto_*`、`bootparams`、`ethers`、`group`、`hosts`（包括 IPv6 地址）、`netgroup`、`netmasks`、`networks`、`passwd`、`shadow`、`protocols`、`publickey`、`rpc` 和 `services`。

缺省情况下，`ldapaddent` 从标准输入中读取数据并将其添加到与命令行中指定的数据库关联的 LDAP 容器中。但是，可以使用 `-f` 选项指定一个输入文件，以便从中读取数据。

由于项存储在基于客户机配置的目录中，因此必须将客户机配置为使用 LDAP 命名服务。

为了获得更好的性能，请按以下顺序装入数据库：

1. 装入 `passwd` 数据库后再装入 `shadow` 数据库
2. 装入 `networks` 数据库后再装入 `netmasks` 数据库
3. 装入 `bootparams` 数据库后再装入 `ethers` 数据库

请注意，在添加自动挂载程序项时，数据库名称的形式为 `auto_*`（例如 `auto_home`）。

如果要将来自不同主机的 `/etc` 文件添加到 LDAP 服务器，可以将这些文件全部合并到同一个 `/etc` 文件中，然后在一台主机上使用 `ldapaddent` 来添加这些文件；或者在每台主机都已配置为 LDAP 客户机的前提下，在不同的主机上逐个执行 `ldapaddent`。

如果您的命名服务数据已经位于 NIS 服务器中，并且您要将数据移到 LDAP 服务器上以用于 LDAP 命名服务，请使用 `ypcat`（或 `niscat`）命令将 NIS 映射转储到文件中。然后，对这些文件运行 `ldapaddent`，将数据添加到 LDAP 服务器中。

注 - `ldapaddent` 只能在 LDAP 客户机上运行。

以下过程假定将要 `yp` 客户机提取表。

▼ 如何使用 `ldapaddent` 向服务器置备 `host` 项

- 1 确保使用 `idsconfig` 对 Sun Java System Directory Server 进行了设置。

- 2 在客户机上，成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

- 3 使计算机成为 LDAP 客户机。

```
# ldapclient init -a profileName=new -a domainName=west.example.com \
192.168.0.1
```

- 4 使用数据置备服务器。

```
# ldapaddent -D "cn=directory manager" -f /etc/hosts hosts
```

系统将提示您输入口令。

在本示例中，`ldapaddent` 将使用在配置文件 `"new"` 中配置的验证方法。选择 `"simple"` 将以明文形式发送口令。有关更多信息，请参阅 [ldapaddent\(1M\)](#) 手册页。

为使用 LDAP 客户机设置 Sun Java System Directory Server (任务)

本章介绍如何配置 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）以支持 Solaris LDAP 命名服务客户机的网络。本章中的信息特定于 Sun Java System Directory Server。有关安装和配置目录服务器的信息，请参见 Sun Java Enterprise System 附带的 Sun Java System Directory Server 文档。

注 – 您必须先执行 Sun Java System Directory Server 随附的安装和配置文档中介绍的所有步骤，才能配置 Sun Java System Directory Server 与 Solaris LDAP 客户机协作。

注 – 目录服务器（LDAP 服务器）**不能**作为其自身的客户机。

本章包含以下主题：

- 第 158 页中的“使用 `idsconfig` 配置 Sun Java System Directory Server”
- 第 160 页中的“使用服务搜索描述符修改客户机对各个服务的访问”
- 第 161 页中的“运行 `idsconfig`”
- 第 166 页中的“使用 `ldapaddent` 置备目录服务器”
- 第 167 页中的“管理打印机项”
- 第 167 页中的“向目录服务器置备其他配置文件”
- 第 168 页中的“配置目录服务器以启用帐户管理”
- 第 172 页中的“迁移 Sun Java System Directory Server”

使用 idsconfig 配置 Sun Java System Directory Server

基于服务器安装创建核对表

在服务器的安装过程中，将会定义一些重要变量，在启动 `idsconfig` 之前，应使用这些变量创建类似如下的核对表。您可以使用[第 193 页中的“空白核对表”](#)中提供的空白核对表。

注 – 下面列出的信息将作为与 LDAP 有关的各章中介绍的所有示例的基础。示例域是一个名为 `Example, Inc.` 的装饰品公司，该公司的商店遍布全美。示例涉及到西海岸分公司，它的域是 `west.example.com`

表 11-1 已定义的服务器变量

变量	针对示例网络的定义
安装了目录服务器实例的端口号	389（缺省）
服务器名称	myserver （来自 FQDN <code>myserver.west.example.com</code> 或 <code>192.168.0.1</code> ）
副本服务器（IP 号:端口号）	192.168.0.2 [对于 <code>myreplica.west.example.com</code>]
目录管理器	<code>cn=Directory Manager</code> （缺省值）
要为其提供服务的域名	west.example.com
在超时之前处理客户端请求的最长时间（以秒为单位）	-1-
为每个搜索请求返回的最多项数	-1-

注 – 如果要使用主机名来定义 `defaultServerList` 或 `preferredServerList`，必须确保不使用 LDAP 进行主机查找。这意味着 `ldap` 不得位于 `/etc/nsswitch.conf` `hosts` 行中。

表 11-2 已定义的客户机配置文件变量

变量	针对示例网络的定义
配置文件名（缺省名称是 <code>default</code> ）	WestUserProfile
服务器列表（缺省值为本地子网）	192.168.0.1
首选服务器列表（按照对服务器进行查找的顺序列出）	none

表 11-2 已定义的客户机配置文件变量 (续)

变量	针对示例网络的定义
搜索范围（沿着目录树向下查找的层数。One（缺省值）或 Sub）	one （缺省值）
用于获取服务器访问权限的凭证。缺省值为 anonymous	proxy
是否遵循引用（主服务器不可用时指向另一台服务器的指针）？缺省值为 no 。	y
等待服务器返回信息的搜索时间限制（缺省值为 30 秒）。	default
与服务器进行联系时的绑定时间限制（缺省值为 10 秒）。	default
验证方法（缺省值为 none ）。	simple

注- 客户机配置文件是按每个域进行定义的。必须至少为给定的域定义一个配置文件。

属性索引

idsconfig 为下面列出的属性编制索引以改善性能。

membernissetgroup	pres,eq,sub
nisnetgrouptriple	pres,eq,sub
ipHostNumber	pres,eq,sub
uidNumber	pres,eq
gidNumber	pres,eq
ipNetworkNumber	pres,eq
automountkey	pres,eq
oncRpcNumber	pres,eq

架构定义

[idsconfig\(1M\)](#) 会自动添加必要的架构定义。除非您在 LDAP 管理方面经验丰富，否则请不要手动修改服务器架构。有关 LDAP 命名服务使用的架构的扩展列表，请参见第 14 章，[LDAP 一般参考（参考）](#)。

使用浏览索引

Sun Java System Directory Server 的浏览索引功能（也称为虚拟列表视图 (Virtual List View, VLV)）为客户机提供了一种查看长列表选定组或选定数量的项的方式，从而使每台客户机在搜索过程中花费更少的时间。浏览索引提供优化的、预定义的搜索参数。Solaris LDAP 命名客户机可以使用这些参数更快地访问不同服务中的特定信息。请记住，如果您未创建浏览索引，客户机可能无法获取给定类型的全部项，因为服务器上可能强制实施了搜索时间或项数方面的限制。

VLV 索引是在目录服务器上配置的，代理用户对这些索引具有读取访问权限。

在 Sun Java System Directory Server 上配置浏览索引之前，请考虑与使用这些索引相关联的性能成本。有关更多信息，请参阅所用的 Sun Java System Directory Server 版本的管理指南。

`idsconfig` 为多个 VLV 索引创建项。使用 `directoryserver` 脚本可以停止服务器并创建实际的 VLV 索引。有关更多信息，请参见 [idsconfig\(1M\)](#) 和 [directoryserver\(1M\)](#) 手册页。请参阅 `idsconfig` 命令的输出来确定 `idsconfig` 创建的 VLV 项以及需要运行的对应 `directoryserver` 命令的语法。有关 `idsconfig` 输出的样例，请参见 [第 162 页中的“idsconfig 设置示例”](#)。

使用服务搜索描述符修改客户机对各个服务的访问

服务搜索描述符 (Service Search Descriptor, SSD) 可以将 LDAP 中给定操作的缺省搜索请求更改为您定义的搜索。例如，如果您一直使用的是具有定制容器定义的 LDAP 或其他操作系统，而现在要转为使用最新的 Solaris 发行版，则 SSD 将特别有用。使用 SSD，可以在不必更改现有 LDAP 数据库和数据的情况下配置 Solaris LDAP 命名服务。

使用 `idsconfig` 设置 SSD

假设 Example, Inc. 中的前任管理员已经配置了 LDAP，并将用户存储在 `ou=Users` 容器中。现在要升级到最新的 Solaris 发行版。按照定义，Solaris LDAP 客户机会认为用户项存储在 `ou=People` 容器中。因此，当开始搜索 `passwd` 服务时，LDAP 客户机将搜索 DIT 的 `ou=people` 层，因而不会找到正确的值。

对于上述问题，一个比较繁琐的解决方案是完全覆盖 Example, Inc. 现有的 DIT，并重写 Example, Inc. 网络上现有的所有应用程序，以便它们与新的 LDAP 命名服务兼容。另外一种更可取的解决方案是，使用 SSD 通知 LDAP 客户机在 `ou=Users` 容器（而不是缺省的 `ou=people` 容器）中查找用户信息。

您将在配置 Sun Java System Directory Server 的过程中使用 `idsconfig` 定义所需的 SSD。提示行如下所示：


```

Do you wish to setup Service Search Descriptors (y/n/h? y
A Add a Service Search Descriptor
D Delete a SSD
M Modify a SSD
P Display all SSD's
H Help
X Clear all SSD's

Q Exit menu
Enter menu choice: [Quit] a
Enter the service id: passwd
Enter the base: service ou=user,dc=west,dc=example,dc=com
Enter the scope: one[default]
A Add a Service Search Descriptor
D Delete a SSD
M Modify a SSD
P Display all SSD's
H Help
X Clear all SSD's

Q Exit menu
Enter menu choice: [Quit] p

Current Service Search Descriptors:
=====
Passwd:ou=Users,ou=west,ou=example,ou=com?

Hit return to continue.

A Add a Service Search Descriptor
D Delete a SSD
M Modify a SSD
P Display all SSD's
H Help
X Clear all SSD's

Q Exit menu
Enter menu choice: [Quit] q

```

运行 idsconfig

注 – 运行 `idsconfig` 无需特殊权限，也不必在 LDAP 命名客户机上运行。请记住在准备运行 `idsconfig` 时，按照第 158 页中的“基于服务器安装创建核对表”中的说明创建一份核对表。您不必从服务器或 LDAP 命名服务客户机运行 `idsconfig`。在网络上的任何 Solaris 计算机上都可以运行 `idsconfig`。



注意 – `idsconfig` 以明文形式发送目录管理器的口令。如果不希望如此，必须在目录服务器（而非客户机）上运行 `idsconfig`。

▼ 如何使用 idsconfig 配置 Sun Java System Directory Server

1 确保目标 Sun Java System Directory Server 已启动并且正在运行。

2 运行 `idsconfig`。

```
# /usr/lib/ldap/idsconfig
```

有关使用服务器和客户机核对表中列出的定义运行 `idsconfig` 的示例，请参阅[示例 11-1](#)，这些核对表位于本章开头处的[第 158 页](#)中的“[基于服务器安装创建核对表](#)”中。

3 根据提示回答问题。

请注意 "no" [n] 是缺省的用户输入。如果需要清楚理解任何给定的问题，请键入 `h`

此时将出现一个简短的帮助段落。

在 `idsconfig` 完成了目录的设置之后，您需要在服务器上运行指定的命令，然后才能完成服务器的设置过程，服务器此时即准备就绪，可以为客户机提供服务。

idsconfig 设置示例

本节提供了一个基本的 `idsconfig` 设置的示例，该示例使用了许多缺省值。修改客户机配置文件最复杂的方法就是创建 SSD。有关详细讨论，请参阅[第 160 页](#)中的“[使用服务搜索描述符修改客户机对各个服务的访问](#)”。

提示后面方括号中的数据指示该提示的缺省值。要接受缺省值，请按下 Return 键。

注- 对于摘要屏幕上任何留空的参数将不进行设置。

在 `idsconfig` 完成了目录的设置之后，您需要在服务器上运行指定的命令，然后才能完成服务器的设置过程，服务器此时即准备就绪，可以为客户机提供服务。

示例 11-1 为 Example, Inc. 网络运行 `idsconfig`

以下示例中，在 LDAP 服务器上创建了服务器实例后，`idsconfig` 实用程序将立即运行。

```
# /usr/lib/ldap/idsconfig
It is strongly recommended that you BACKUP the directory server
before running idsconfig.
```

示例 11-1 为 Example, Inc. 网络运行 idsconfig (续)

```

Hit Ctrl-C at any time before the final confirmation to exit.

Do you wish to continue with server setup (y/n/h)? [n] y
Enter the JES Directory Server's hostname to setup: myserver
Enter the port number for iDS (h=help): [389]
Enter the directory manager DN: [cn=Directory Manager]
Enter passwd for cn=Directory Manager :
Enter the domainname to be served (h=help): [west.example.com]
Enter LDAP Base DN (h=help): [dc=west,dc=example,dc=com]
    Checking LDAP Base DN ...
    Validating LDAP Base DN and Suffix ...
    No valid suffixes were found for Base DN dc=west,dc=example,dc=com
Enter suffix to be created (b=back/h=help): [dc=west,dc=example,dc=com]
Enter ldbm database name (b=back/h=help): [west]
    sasl/GSSAPI is not supported by this LDAP server
Enter the profile name (h=help): [default] WestUserProfile
Default server list (h=help): [192.168.0.1]
Preferred server list (h=help):
Choose desired search scope (one, sub, h=help): [one]
The following are the supported credential levels:
    1 anonymous
    2 proxy
    3 proxy anonymous
    4 self
    5 self proxy
    6 self proxy anonymous
Choose Credential level [h=help]: [1] 2
The following are the supported Authentication Methods:
    1 none
    2 simple
    3 sasl/DIGEST-MD5
    4 tls:simple
    5 tls:sasl/DIGEST-MD5
    6 sasl/GSSAPI
Choose Authentication Method (h=help): [1] 2

Current authenticationMethod: simple
Do you want to add another Authentication Method? n
Do you want the clients to follow referrals (y/n/h)? [n]
Do you want to modify the server timelimit value (y/n/h)? [n] y
Enter the time limit for iDS (current=3600): [-1]
Do you want to modify the server sizelimit value (y/n/h)? [n] y
Enter the size limit for iDS (current=2000): [-1]
Do you want to store passwords in "crypt" format (y/n/h)? [n] y
Do you want to setup a Service Authentication Methods (y/n/h)? [n]
Client search time limit in seconds (h=help): [30]
Profile Time To Live in seconds (h=help): [43200]
Bind time limit in seconds (h=help): [10]
Do you want to enable shadow update (y/n/h)? [n]
Do you wish to setup Service Search Descriptors (y/n/h)? [n]

```

Summary of Configuration

```

1 Domain to serve                : west.example.com

```

示例 11-1 为 Example, Inc. 网络运行 idsconfig (续)

```

2 Base DN to setup           : dc=west,dc=example,dc=com
   Suffix to create          : dc=west,dc=example,dc=com
   Database to create         : west
3 Profile name to create     : WestUserProfile
4 Default Server List        : 192.168.0.1
5 Preferred Server List      :
6 Default Search Scope       : one
7 Credential Level           : proxy
8 Authentication Method       : simple
9 Enable Follow Referrals     : FALSE
10 iDS Time Limit             : -1
11 iDS Size Limit             : -1
12 Enable crypt password storage : TRUE
13 Service Auth Method pam_ldap :
14 Service Auth Method keyserve :
15 Service Auth Method passwd-cmd:
16 Search Time Limit          : 30
17 Profile Time to Live       : 43200
18 Bind Limit                 : 10
19 Enable shadow update       : FALSE
20 Service Search Descriptors Menu

Enter config value to change: (1-20 0=commit changes) [0]
Enter DN for proxy agent: [cn=proxyagent,ou=profile,dc=west,dc=example,dc=com]
Enter passwd for proxyagent:
Re-enter passwd:

WARNING: About to start committing changes. (y=continue, n=EXIT) y

1. Changed timelimit to -1 in cn=config.
2. Changed sizelimit to -1 in cn=config.
3. Changed passwordstoragescheme to "crypt" in cn=config.
4. Schema attributes have been updated.
5. Schema objectclass definitions have been added.
6. Database west successfully created.
7. Suffix dc=west,dc=example,dc=com successfully created.
8. NisDomainObject added to dc=west,dc=example,dc=com.
9. Top level "ou" containers complete.
10. automount maps: auto_home auto_direct auto_master auto_shared processed.
11. ACI for dc=west,dc=example,dc=com modified to disable self modify.
12. Add of VLV Access Control Information (ACI).
13. Proxy Agent cn=proxyagent,ou=profile,dc=west,dc=example,dc=com added.
14. Give cn=proxyagent,ou=profile,dc=west,dc=example,dc=com read permission
    for password.
15. Generated client profile and loaded on server.
16. Processing eq,pres indexes:
    uidNumber (eq,pres) Finished indexing.
    ipNetworkNumber (eq,pres) Finished indexing.
    gidnumber (eq,pres) Finished indexing.
    oncrpcnumber (eq,pres) Finished indexing.
    automountKey (eq,pres) Finished indexing.
17. Processing eq,pres,sub indexes:
    ipHostNumber (eq,pres,sub) Finished indexing.
    membernisnetgroup (eq,pres,sub) Finished indexing.
    nisnetgrouptriple (eq,pres,sub) Finished indexing.

```

示例 11-1 为 Example, Inc. 网络运行 idsconfig (续)

```

18. Processing VLV indexes:
west.example.com.getgrent vlv_index      Entry created
west.example.com.gethostent vlv_index     Entry created
west.example.com.getnetent vlv_index      Entry created
west.example.com.getpwent vlv_index       Entry created
west.example.com.getrpcnt vlv_index       Entry created
west.example.com.getspent vlv_index       Entry created
west.example.com.getauhoent vlv_index     Entry created
west.example.com.getsoluent vlv_index     Entry created
west.example.com.getauduent vlv_index     Entry created
west.example.com.getauthent vlv_index     Entry created
west.example.com.getexecent vlv_index     Entry created
west.example.com.getprofent vlv_index     Entry created
west.example.com.getmailent vlv_index     Entry created
west.example.com.getbootent vlv_index     Entry created
west.example.com.getethent vlv_index      Entry created
west.example.com.getngrpent vlv_index     Entry created
west.example.com.getipnent vlv_index      Entry created
west.example.com.getmaskent vlv_index     Entry created
west.example.com.getprent vlv_index       Entry created
west.example.com.getip4ent vlv_index      Entry created
west.example.com.getip6ent vlv_index      Entry created

```

idsconfig: Setup of iDS server myserver is complete.

Note: idsconfig has created entries for VLV indexes.

For DS5.x, use the directoryserver(lm) script on myserver to stop the server. Then, using directoryserver, follow the directoryserver examples below to create the actual VLV indexes.

For DS6.x, use dsadm command delivered with DS6.x on myserver to stop the server. Then, using dsadm, follow the dsadm examples below to create the actual VLV indexes.

```

directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getgrent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.gethostent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getnetent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getpwent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getrpcnt
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getspent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getauhoent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getsoluent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getauduent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getauthent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getexecent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getprofent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getmailent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getbootent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getethent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getngrpent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getipnent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getmaskent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getprent

```

示例 11-1 为 Example, Inc. 网络运行 idsconfig (续)

```

directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getip4ent
directoryserver -s <server-instance> vlvindex -n west -T west.example.com.getip6ent

<install-path>/bin/dsadm reindex -l -t west.example.com.getgrent <directory-instance-path>
dc=west,dc=example,dc=com
<install-path>/bin/dsadm reindex -l -t west.example.com.gethostent <directory-instance-path>
dc=west,dc=example,dc=com
.
.
.
<install-path>/bin/dsadm reindex -l -t west.example.com.getip6ent <directory-instance-path>
dc=west,dc=example,dc=com

```

使用 ldapaddent 置备目录服务器

注 – 在用数据置备目录服务器之前，必须配置服务器，使其在您使用 `pam_unix` 时以 UNIX Crypt 格式存储口令。如果您使用的是 `pam_ldap`，则可以用任何格式存储口令。有关以 UNIX crypt 格式设置口令的更多信息，请参见 Sun Java System Directory Server 文档。

`ldapaddent` 从标准输入（是一个类似 `passwd` 的 `/etc/filename`）中读取数据，然后将该数据放置在与服务关联的容器中。客户机的配置决定了数据的缺省写入方式。

注 – `ldapaddent(1M)` 只能在 LDAP 客户机上运行。第 12 章，设置 LDAP 客户机（任务）介绍了如何为 LDAP 命名服务配置客户机。

▼ 如何使用 ldapaddent 向 Sun Java System Directory Server 置备用户口令数据

请参见 `ldapaddent(1M)`。有关 LDAP 安全性和对目录服务器写入访问权限的信息，请参见第 9 章，LDAP 基本组件和概念（概述）。

- 使用 `ldapaddent` 命令向服务器中添加 `/etc/passwd` 项。

```
# ldapaddent -D "cn=directory manager" -f /etc/passwd passwd
```

管理打印机项

添加打印机

要向 LDAP 目录中添加打印机项，请使用 `printmgr` 配置工具或 `lpset -n ldap` 命令行实用程序。请参见 [lpset\(1M\)](#)。请注意，添加到该目录中的打印机对象仅定义了打印系统客户机所需的打印机连接参数。本地打印服务器配置数据仍保留在文件中。典型的打印机项如下所示：

```
printer-uri=myprinter,ou=printers,dc=mkg,dc=example,dc=com
objectclass=top
objectclass=printerService
objectclass=printerAbstract
objectclass=sunPrinter
printer-name=myprinter
sun-printer-bsdaddr=printsvr.example.com,myprinter,Solaris
sun-printer-kvp=description=HP LaserJet (PS)
printer-uri=myprinter
```

使用 lpget

[lpget\(1M\)](#) 可用于列出 LDAP 客户机的 LDAP 目录已知的所有打印机项。如果 LDAP 客户机的 LDAP 服务器是副本服务器，则列出的打印机可能不同于主 LDAP 服务器中的打印机，具体情况取决于更新复制协议。有关更多信息，请参见 [lpget\(1M\)](#)。

例如，要列出给定的基本 DN 的所有打印机，请键入以下内容：

```
# lpget -n ldap list
myprinter:
  dn=myprinter,ou=printers,dc=mkt,dc=example,dc=com
  bsdaddr=printsvr.example.com,myprinter,Solaris
  description=HP LaserJet (PS)
```

向目录服务器置备其他配置文件

使用带有 `genprofile` 选项的 `ldapclient`，可以基于所指定的属性创建 LDIF 表示形式的配置文件。所创建的配置文件随后可以装入 LDAP 服务器中用作客户机配置文件。客户机可以使用 `ldapclient init` 来下载客户机配置文件。

有关使用 `ldapclient genprofile` 的信息，请参阅 [ldapclient\(1M\)](#)。

▼ 如何使用 **ldapclient** 向目录服务器置备其他配置文件

- 1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第9章“使用基于角色的访问控制（任务）”。

- 2 使用带 **genprofile** 的 **ldapclient** 命令。

```
# ldapclient genprofile \  
-a profileName=myprofile \  
-a defaultSearchBase=dc=west,dc=example,dc=com \  
-a "defaultServerList=192.168.0.1 192.168.0.2:386" \  
  
> myprofile.ldif
```

- 3 将新配置文件上载到服务器。

```
# ldapadd -h 192.168.0.1 -D "cn=directory manager" -f myprofile.ldif
```

配置目录服务器以启用帐户管理

从 Solaris 10 10/09 发行版开始，可以为使用 **pam_ldap** 和 **pam_unix** 的客户机实现帐户管理。



注意 – 不要在同一 LDAP 命名域中同时使用 **pam_ldap** 和 **pam_unix**。所有客户机要么都使用 **pam_ldap**，要么都使用 **pam_unix**。因为存在这种限制，您可能需要专用的 LDAP 服务器。

对于使用 **pam_ldap** 的客户机

为了让 **pam_ldap** 能够正常工作，必须在服务器上正确配置口令和帐户锁定策略。您可以使用 **Directory Server Console** 或 **ldapmodify** 为 LDAP 目录配置帐户管理策略。有关过程和更多信息，请参见所用的 Sun Java System Directory Server 版本的管理指南中的“用户帐户管理”一章。

注 – 以前，如果启用了 `pam_ldap` 帐户管理，所有用户在每次登录系统时都必须提供登录口令以进行验证。因此，使用 `rsh`、`rlogin` 或 `ssh` 等工具进行的不基于口令的登录将会失败。

但是现在，`pam_ldap(5)` 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时，用户可以使用 `rsh`、`rlogin`、`rcp` 和 `ssh` 登录，而不需要提供口令。

`pam_ldap(5)` 现已修改为执行帐户管理和检索用户帐户状态，而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8，它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态，请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI)：

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
    allow (read, search, compare, proxy)
    (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

绝不当允许 `proxy` 用户的口令过期。如果代理口令过期，使用 `proxy` 凭证级别的客户机将无法从服务器检索命名服务信息。为了确保代理用户的口令不过期，请使用以下脚本修改代理帐户：

```
# ldapmodify -h ldapserver -D administrator DN \
-w administrator password <<EOF
dn: proxy user DN
DNchangetype: modify
replace: passwordexpirationtime
passwordexpirationtime: 20380119031407Z
EOF
```

注 – `pam_ldap` 帐户管理依赖 Sun Java System Directory Server 为用户维护和提供口令生命期和帐户过期信息。目录服务器不会解释 `shadow` 项中的对应数据来验证用户帐户。但是，`pam_unix` 会检查阴影数据以确定帐户是否处于锁定状态或者口令是否已失效。由于 LDAP 命名服务或目录服务器不会使阴影数据保持最新，因此 `pam_unix` 不当基于阴影数据授予访问权限。阴影数据是通过 `proxy` 标识检索的。因此，请不要允许 `proxy` 用户对 `userPassword` 属性具有读取访问权限。拒绝 `proxy` 用户对 `userPassword` 的读取访问权限可防止 `pam_unix` 进行无效的帐户验证。

对于使用 **pam_unix** 的客户机

要使 Solaris LDAP 客户机能够使用 **pam_unix** 进行帐户管理，服务器必须设置为启用阴影数据更新。从 Solaris 10 10/09 发行版开始，该项功能可用。与 **pam_ldap** 帐户管理不同，**pam_unix** 不需要额外的配置步骤。所有配置均可运行 **idsconfig** 实用程序来执行。对于基本的 **idsconfig** 运行，请参见[示例 11-1](#)。

下面显示了两个 **idsconfig** 运行的输出。

第一个 **idsconfig** 运行使用现有的客户机配置文件。

```
# /usr/lib/ldap/idsconfig

It is strongly recommended that you BACKUP the directory server
before running idsconfig.

Hit Ctrl-C at any time before the final confirmation to exit.

Do you wish to continue with server setup (y/n/h)? [n] y
Enter the JES Directory Server's hostname to setup: myserver
Enter the port number for iDS (h=help): [389]
Enter the directory manager DN: [cn=Directory Manager]
Enter passwd for cn=Directory Manager :
Enter the domainname to be served (h=help): [west.example.com]
Enter LDAP Base DN (h=help): [dc=west,dc=example,dc=com]
  Checking LDAP Base DN ...
  Validating LDAP Base DN and Suffix ...
  sasl/GSSAPI is not supported by this LDAP server

Enter the profile name (h=help): [default] WestUserProfile

Profile 'WestUserProfile' already exists, it is possible to enable
shadow update now. idsconfig will exit after shadow update
is enabled. You can also continue to overwrite the profile
or create a new one and be given the chance to enable
shadow update later.

Just enable shadow update (y/n/h)? [n] y
Add the administrator identity (y/n/h)? [y]
Enter DN for the administrator: [cn=admin,ou=profile,dc=west,dc=example,dc=com]
Enter passwd for the administrator:
Re-enter passwd:
  ADDED: Administrator identity cn=admin,ou=profile,dc=west,dc=example,dc=com.
  Proxy ACI LDAP_Naming_Services_proxy_password_read does not
  exist for dc=west,dc=example,dc=com.
  ACI SET: Give cn=admin,ou=profile,dc=west,dc=example,dc=com read/write access
  to shadow data.
  ACI SET: Non-Admin access to shadow data denied.

Shadow update has been enabled.
```

第二个 **idsconfig** 运行创建了新的配置文件供以后使用。这里显示的只是部分输出。

```
# /usr/lib/ldap/idsconfig
```

It is strongly recommended that you BACKUP the directory server before running idsconfig.

Hit Ctrl-C at any time before the final confirmation to exit.

```
Do you wish to continue with server setup (y/n/h)? [n] y
Enter the JES Directory Server's hostname to setup: myserver
Enter the port number for iDS (h=help): [389]
Enter the directory manager DN: [cn=Directory Manager]
Enter passwd for cn=Directory Manager :
Enter the domainname to be served (h=help): [west.example.com]
Enter LDAP Base DN (h=help): [dc=west,dc=example,dc=com]
  Checking LDAP Base DN ...
  Validating LDAP Base DN and Suffix ...
  sasl/GSSAPI is not supported by this LDAP server
```

```
Enter the profile name (h=help): [default] WestUserProfile-new
Default server list (h=help): [192.168.0.1]
```

```
.
.
.
```

```
Do you want to enable shadow update (y/n/h)? [n] y
```

Summary of Configuration

```
1 Domain to serve           : west.example.com
2 Base DN to setup          : dc=west,dc=example,dc=com
   Suffix to create         : dc=west,dc=example,dc=com
3 Profile name to create    : WestUserProfile-new
```

```
.
.
.
```

```
19 Enable shadow update     : TRUE
```

```
.
.
.
```

```
Enter DN for the administrator: [cn=admin,ou=profile,dc=west,dc=example,dc=com]
```

```
Enter passwd for the administrator:
```

```
Re-enter passwd:
```

WARNING: About to start committing changes. (y=continue, n=EXIT) y

```
1. Changed timelimit to -1 in cn=config.
2. Changed sizelimit to -1 in cn=config.
```

```
.
.
.
```

```
11. ACI for dc=test1,dc=mpklab,dc=sfbay,dc=sun,dc=com modified to
    disable self modify.
```

```
.
.
.
```

```
15. Give cn=admin,ou=profile,dc=west,dc=example,dc=com write permission for shadow.
```

```
...
```

迁移 Sun Java System Directory Server

Sun Java System Directory Server 5.1 发行版（以前称为 Sun ONE Directory Server）和 Sun Java System Directory Server 5.2 发行版之间实现了架构更改。`ldapaddent` 命令现在向 `ethers/bootparams` 的项中添加了 `objectclass: device`。因此，如果您选择使用 LDAP 命令将目录数据从 Sun Java System Directory Server 5.1 迁移到 5.2，则必须使用 `ldapaddent -d` 导出数据，再使用 `ldapaddent` 导入数据。否则，如果使用 Sun Java System Directory Server 工具 `db2ldif` 和 `ldif2db` 迁移数据，您必须在迁移数据之前为 Sun Java System Directory Server 5.2 应用所有的修补程序，否则数据导入操作将失败。

有关配置 Sun Java System Directory Server 5.2 的信息，请参见 Sun Java Enterprise System 附带的 Sun Java System Directory Server 文档。

设置 LDAP 客户机（任务）

本章介绍如何设置 Solaris LDAP 命名服务客户机。本章包含以下主题：

- 第 173 页中的“LDAP 客户机设置的先决条件”
- 第 174 页中的“LDAP 和服务管理工具”
- 第 175 页中的“初始化 LDAP 客户机”
- 第 183 页中的“检索 LDAP 命名服务信息”
- 第 185 页中的“定制 LDAP 客户机环境”

LDAP 客户机设置的先决条件

要让 Solaris 客户机使用 LDAP 作为命名服务，必须满足以下条件。

- 客户机的域名必须由 LDAP 服务器提供
- `nsswitch.conf` 文件需要指向 LDAP 以获取所需服务
- 客户机需要配置所有给定参数，以用于定义其行为
- `ldap_cachemgr` 需要在客户机上运行
- 至少有一台要为其配置客户机的服务器已启动并正在运行

`ldapclient` 实用程序可以执行上述除启动服务器之外的所有步骤，因此是设置 LDAP 客户机的关键。本章其余部分将举例说明如何使用 `ldapclient` 实用程序设置 LDAP 客户机，以及如何使用其他各种 LDAP 实用程序获取有关 LDAP 客户机的信息并检查其状态。

LDAP 和服务管理工具

LDAP 客户机服务可以使用服务管理工具进行管理。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

- 使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。

提示 – 使用 `-t` 选项暂时禁用服务可为服务配置提供一些保护。如果禁用服务时使用了 `-t` 选项，则服务在重新引导后将恢复原始设置。如果禁用服务时未使用 `-t`，则服务在重新引导后仍将保持禁用状态。

- 用于 LDAP 客户机服务的故障管理资源标识符 (Fault Managed Resource Identifier, FMRI) 为 `svc:/network/ldap/client:<instance>`。
- 使用 `svcs` 命令可以查询 LDAP 客户机和 `ldap_cachemgr` 的状态。
 - `svcs` 命令和输出示例。

```
# svcs \*ldap\*
STATE          STIME          FMRI
online         15:43:46    svc:/network/ldap/client:default
```

- `svcs -l` 命令和输出示例。要获得如下所示的输出，必须在 FMRI 中使用实例名称。

```
# svcs -l network/ldap/client:default
fmri           svc:/network/ldap/client:default
enabled        true
state          online
next_state     none
restarter      svc:/system/svc/restarter:default
contract_id    1598
dependency     require_all/none file://localhost/var/ldap/ldap_client_file (-)
dependency     require_all/none svc:/network/initial (online)
dependency     require_all/none svc:/system/filesystem/minimal (online)
```

- 使用 `ps` 命令可检查守护进程是否存在。

```
# ps -e | grep slapd
root 23320      1   0   Aug 27 ?          16:30 ./ns-slapd -D \
/usr/iplanet/ds5/slapd-lastrev -i /usr/iplanet/ds5/slapd-lastrev/
root 25367 25353   0 15:35:19 pts/1      0:00 grep slapd
```

注 – 不要将 `-f` 选项与 `ps` 结合使用，因为此选项会尝试将用户 ID 转换为名称，而这将导致更多命名服务查找可能无法成功。

初始化 LDAP 客户机

`ldapclient(1M)` 是用于在 Solaris 系统中设置 LDAP 客户机的实用程序。`ldapclient` 假定服务器已配置了适当的客户机配置文件。您必须先安装服务器并用适当的配置文件对其进行配置，然后才能设置客户机。

注 – Solaris 操作系统不支持 NIS 客户机与本地 LDAP 客户机共存于同一客户机系统中的配置。

使用 `ldapclient` 设置客户机主要有两种方法。

- **配置文件**

您至少需要指定包含配置文件以及要使用的域的服务器地址。如果未指定配置文件，则会使用“缺省”的配置文件。服务器将提供其余的必需信息，但代理和证书数据库信息除外。如果客户机的凭证级别为 `proxy` 或 `proxy anonymous`，则必须提供代理的绑定 DN 和口令。有关更多信息，请参见第 137 页中的“指定客户机凭证级别”。

自 Solaris 10 10/09 发行版开始，提供了 `enableShadowUpdate` 开关。要启用阴影数据更新，您必须提供管理凭证（`adminDN` 和 `adminPassword`）。

- **手动**

在客户机上配置配置文件，这意味着要从命令行定义所有参数。这样，配置文件信息将存储在高速缓存文件中，服务器永远不会刷新这些信息。

注 – 尽管您可以手动配置客户机，但并不建议使用此方法。使用配置文件可以降低管理客户机的复杂性和成本。

使用配置文件初始化客户机

▼ 如何使用配置文件初始化客户机

- 1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

- 2 运行带 `init` 的 `ldapclient` 命令。

```
# ldapclient init \
-a profileName=new \
-a domainName=west.example.com 192.168.0.1
System successfully configured
```

使用每用户凭证

注- 请勿直接编辑任何客户机配置文件。请使用 `ldapclient` 命令创建或修改这些文件的内容。

▼ 如何使用每用户凭证初始化客户机

开始之前 使用每用户凭证设置客户机之前，必须已经配置以下各项：

- 一台或多台 Kerberos KDC 服务器必须已配置且正在运行。
- DNS、到 DNS 服务器的客户机访问以及至少一台 DNS 服务器必须已配置且正在运行。
- 客户机上的 Kerberos 必须已配置且已启用。
- 必须存在 Kerberos 客户机安装配置文件。此类配置文件可能如下所示：

```
# cat /usr/tmp/krb5.profile
REALM SPARKS.COM
KDC kdc.example.com
ADMIN super/admin
FILEPATH /usr/tmp/krb5.conf
NFS 1
DNSLOOKUP none
```

- LDAP 服务器必须已安装并配置为支持 sasl/GSSAPI。
- 必须存在适当的标识映射配置。
- 必须在 KDC 中设置了目录服务器和 KDC 的 Kerberos 主机主体。
- `idsconfig` 必须已在要使用的目录服务器 DIT 上运行。
- 必须已创建适当的每用户 `gssapi` 配置文件（如 `gssapi_EXAMPLE.COM`）。

`idsconfig` 中的每用户配置文件的说明如下面的部分示例所示：

```
# /usr/lib/ldap/idsconfig
Do you wish to continue with server setup (y/n/h)? [n] y
Enter the iPlanet Directory Server's (iDS) hostname to setup: kdc.example.com
Enter the port number for iDS (h=help): [389] <Enter your port>
Enter the directory manager DN: [cn=Directory Manager] <Enter your DN>
Enter passwd for cn=Directory Manager : <Enter your password>
Enter the domainname to be served (h=help): [example.com] <Enter your domain>
Enter LDAP Base DN (h=help): [dc=example,dc=com] <Enter your DN>
GSSAPI is supported. Do you want to set up gssapi:(y/n) [n] y
Enter Kerberos Realm: [EXAMPLE.COM] EXAMPLE.COM You can create a sasl/GSSAPI enabled profile with default values now. Do
want to create a sasl/GSSAPI default profile ? [n] y Enter. the profile name
(h=help): [gssapi_EXAMPLE.COM] <Enter>
GSSAPI setup is done. ...
```

- 密钥分发中心 (Key Distribution Center, KDC) 中必须存在所需的用户主体。
- 在客户机上，必须使用含有诸如以下命令的客户机配置文件初始化 Kerberos：

```
# /usr/sbin/kcclient -p /usr/tmp/krb5.profile
```


- `/etc/nsswitch.ldap` 必须配置为针对 `hosts` 和 `ipnodes` 使用 `dns`。如有需要，可使用编辑器修改此文件，如下所示：

```
host: files dns
ipnodes: files dns
```

- `/etc/resolv.conf` 必须已配置，并且 `dns` 服务必须正在运行。有关详细信息，请参见本文档中有关 DNS 的章节。
- 目录服务器 DIT 必须至少预装入此客户机的用户、客户机主机和必要的 `auto_home` LDAP 项。有关如何使用 `ldapaddent` 添加项的详细信息，请参见本手册中的其他节。

1 使用 `gssapi` 配置文件运行 `ldapclient init` 来初始化客户机：

```
# /usr/sbin/ldapclient init -a profilename=gssapi_SPARKS.COM -a \
domainname=example.com 9.9.9.50
```

2 尝试以用户身份登录：

运行 `kinit -p user`。

在用户登录会话中运行 `ldaplist -l passwd user`，然后您将看见 "userpassword"。

但是 `ldaplist -l passwd bar` 可以在不返回 `userpassword` 的同时获取项。缺省情况下，超级用户仍会看到每个人的 `userpassword`。

更多信息 有关使用每用户凭证的注释

- 如果 `syslog` 包含以下消息：`libsldap: Status: 7 Mesg: openConnection: GSSAPI bind failed - 82 Local error`，则标识 Kerberos 可能未初始化，或者其票证已过期。运行 `klist` 进行浏览。运行 `kinit -p foo` 或 `kinit -R -p foo` 并重试。
- 如果需要，可以将 `pam_krb5.so.1` 添加到 `/etc/pam.conf` 中，这样在您登录时，它会自动执行 `kinit`。
- 例如：

```
login    auth optional pam_krb5.so.1
rlogin   auth optional pam_krb5.so.1
other    auth optional pam_krb5.so.1
```

- 如果用户已执行 `kinit` 且 `syslog` 消息指示 `Invalid credential`，则问题可能是主机项 (`root`) 或用户项不在 LDAP 目录中，或者映射规则不正确。
- 执行 `ldapclient init` 时，如果 LDAP 配置文件包含 `self/sasl/GSSAPI` 配置，则会执行一些检查。如果在进行 `/etc/nsswitch.ldap` 检查时失败，通常的原因是 DNS 未添加到 `host:` 和 `ipnodes:`。
- 如果是因为 DNS 客户机未启用而失败，请运行 `svcs -l dns/client` 查看 `/etc/resolv.conf` 是缺失还是仅仅处于禁用状态。运行 `svcadm enable dns/client` 可以将其启用。
- 如果检查是由于 `sasl/GSSAPI` 绑定而失败，请检查 `syslog` 查找失败原因。

有关详细信息，请参见本指南和《系统管理指南：安全性服务》中的其他参考。

使用代理凭证

▼ 如何使用代理凭证初始化客户机

注 - 请勿直接编辑任何客户机配置文件。请使用 `ldapclient` 创建或修改这些文件的内容。

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 运行 `ldapclient`（定义代理值）。

```
# ldapclient init \  
-a proxyDN=cn=proxyagent,ou=profile,dc=west,dc=example,dc=com \  
-a domainName=west.example.com \  
-a profileName=pit1 \  
-a proxyPassword=test1234 192.168.0.1  
System successfully configured
```

如果要使用的配置文件设置为使用 `proxy`，则 `-a proxyDN` 和 `-a proxyPassword` 是必需的。由于凭据并不是存储在服务器上保存的配置文件中，因此您必须在初始化客户机时提供该信息。与原先在服务器上存储代理凭证的方法相比，这种方法更安全。

代理信息将用于创建 `/var/ldap/ldap_client_cred`。其余信息放置在 `/var/ldap/ldap_client_file` 中。

在 LDAP 中启用 shadow 更新

自 Solaris 10 10/09 发行版开始，提供了 `enableShadowUpdate` 开关。

▼ 如何初始化客户机以启用阴影数据更新

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 要设置 enableShadowUpdate 开关并定义管理凭证，请运行 ldapclient 命令。

- 要更新已在运行的客户机，请运行此命令：

```
# ldapclient mod -a enableShadowUpdate=TRUE \
-a adminDN=cn=admin,ou=profile,dc=west,dc=example,dc=com \
-a adminPassword=admin-password
System successfully configured
```

- 要初始化客户机，请运行此命令：

```
# ldapclient init \
-a adminDN=cn=admin,ou=profile,dc=west,dc=example,dc=com \
-a adminPassword=admin-password \
-a domainName=west.example.com \
-a profileName=WestUserProfile \
-a proxyDN=cn=proxyagent,ou=profile,dc=west,dc=example,dc=com \
-a proxyPassword=i<proxy_password> \
192.168.0.1
System successfully configured
```

3 要验证配置，请显示 /var/ldap/ldap_client_cred 文件的内容。

输出中应包含与下面类似的行：

```
# cat /var/ldap/ldap_client_cred

NS_LDAP_BINDDN= cn=proxyagent,ou=profile,dc=west,dc=example,dc=com
NS_LDAP_BINDPASSWD= {NS1}4a3788f8eb85de11
NS_LDAP_ENABLE_SHADOW_UPDATE= TRUE
NS_LDAP_ADMIN_BINDDN= cn=admin,ou=profile,dc=west,dc=example,dc=com
NS_LDAP_ADMIN_BINDPASSWD= {NS1}4a3788f8c053434f
```

手动初始化客户机

超级用户或承担等效角色的管理员可以手动配置客户机。但是在手动配置期间，会跳过许多检查，因此系统配置相对容易出错。此外，您必须更改**每台计算机**上的设置，而不像使用配置文件时那样，只需在一个集中位置进行更改即可。

▼ 如何手动初始化客户机

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 使用 ldapclient manual 初始化客户机。

```
# ldapclient manual \
-a domainName=dc=west.example.com \
-a credentialLevel=proxy \
-a defaultSearchBase=dc=west,dc=example,dc=com \
-a proxyDN=cn=proxyagent,ou=profile,dc=west,dc=example,dc=com \
-a proxyPassword=testtest 192.168.0.1
```

3 使用 `ldapclient list` 进行验证。

```
NS_LDAP_FILE_VERSION= 2.0
NS_LDAP_BINDDN= cn=proxyagent,ou=profile,dc=west,dc=example,dc=com
NS_LDAP_BINDPASSWD= {NS1}4a3788e8c053424f
NS_LDAP_SERVERS= 192.168.0.1
NS_LDAP_SEARCH_BASEDN= dc=west,dc=example,dc=com
NS_LDAP_CREDENTIAL_LEVEL= proxy
```

修改手动客户机配置

▼ 如何修改手动配置

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 使用 `ldapclient mod` 命令将身份验证方法更改为 `simple`。

```
# ldapclient mod -a authenticationMethod=simple
```

3 使用 `ldapclient list` 验证是否已进行了更改。

```
# ldapclient list
NS_LDAP_FILE_VERSION= 2.0
NS_LDAP_BINDDN= cn=proxyagent,ou=profile,dc=west,dc=example,dc=com
NS_LDAP_BINDPASSWD= {NS1}4a3788e8c053424f
NS_LDAP_SERVERS= 192.168.0.1
NS_LDAP_SEARCH_BASEDN= dc=west,dc=example,dc=com
NS_LDAP_AUTH= simple
NS_LDAP_CREDENTIAL_LEVEL= proxy
```

故障排除 您不能使用 `mod` 子命令更改 LDAP 客户机配置的某些属性。例如，您不能更改 `profileName` 和 `profileTTL` 属性。要更改这些属性，请使用 `ldapclient init` 命令创建一个新的配置文件，如第 175 页中的“使用配置文件初始化客户机”中所述，或者运行 `ldapclient manual` 命令，如第 179 页中的“手动初始化客户机”中所述。

取消客户机初始化

`ldapclient uninit` 可将客户机名称服务恢复到它在最近的 `init`、`modify` 或 `manual` 操作之前的状态。换言之，该命令可对采取的上一个步骤执行“撤消”操作。例如，如果客户机被配置为使用 `profile1`，然后更改为使用 `profile2`，则使用 `ldapclient uninit` 将使客户机恢复使用 `profile1`。

▼ 如何取消客户机初始化

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 使用 `ldapclient uninit`。

```
# ldapclient uninit
System successfully recovered
```

设置 TLS 安全性

注 – 安全数据库文件必须可供任何人读取。请勿在 `key3.db` 中包括任何私钥。

如果使用 TLS，必须安装必要的安全数据库。特别的是，需要证书和密钥数据库文件。例如，如果采用 Netscape Communicator 的旧数据库格式，则需要 `cert7.db` 和 `key3.db` 这两个文件。或者，如果使用 Mozilla 提供的较新的数据库格式，则需要 `cert8.db`、`key3.db` 和 `secmod.db` 这三个文件。`cert7.db` 或 `cert8.db` 文件中包含受信任证书。`key3.db` 文件中包含客户机的密钥。即使 LDAP 命名服务客户机不使用客户机密钥，此文件也必须存在。`secmod.db` 文件中包含安全模块，如 PKCS#11 模块。如果使用的是旧格式，则不需要此文件。

注 – 在运行 `ldapclient` 之前，应设置并安装本节中介绍的必需的安全数据库文件。

请参见您所使用的 Sun Java System Directory Server 版本的管理员指南的“管理 SSL”一章中有关配置 LDAP 客户机以使用 SSL 的相关内容，了解如何创建和管理这些文件的信息。配置后，这些文件必须存储在 LDAP 命名服务客户机希望的位置。属性 `certificatePath` 用于确定此位置。缺省值为 `/var/ldap`。

例如，使用 Netscape Communicator 设置所需的 `cert7.db` 和 `key3.db` 文件之后，将文件复制到缺省位置。

```
# cp $HOME/.netscape/cert7.db /var/ldap
# cp $HOME/.netscape/key3.db /var/ldap
```

然后，向所有人授予读取访问权限。

```
# chmod 444 /var/ldap/cert7.db
# chmod 444 /var/ldap/key3.db
```

注 – Netscape 在 `$HOME/.netscape` 目录中管理 `cert7.db` 和 `key3.db` 文件，而 Mozilla 在 `$HOME/.mozilla` 下的一个子目录中管理其 `cert8.db`、`key3.db` 和 `secmod.db` 文件。如果要将这些安全数据库用于 LDAP 命名服务客户机，必须将其副本存储在本地文件系统中。

配置 PAM

`pam_ldap` 是一个用于 LDAP 身份验证和帐户管理的 PAM 模块。请参见 [pam_ldap\(5\)](#) 手册页和附录 A，[Solaris 10 软件中对 DNS、NIS 和 LDAP 的更新](#)，获取更多有关 `pam_ldap` 当前支持的功能的信息。

如果同时选中每用户模式和 `self` 凭证选项，您还必须启用 PAM Kerberos `pam_krb5(5)` `pam` 模块。有关详细信息，请参见 [pam_krb5\(5\)](#) 和《系统管理指南：安全性服务》文档。

配置 PAM 以使用 UNIX policy

要配置 PAM 以使用 UNIX policy，请遵照第 197 页中的“[pam_ldap 的示例 pam.conf 文件](#)”中的样例。向客户机的 `/etc/pam.conf` 文件中添加包含 `pam_ldap.so.1` 的行。有关详细信息，请参见 [pam.conf\(4\)](#) 手册页。

配置 PAM 以使用 LDAP server_policy

要配置 PAM 以使用 LDAP `server_policy`，请遵照第 198 页中的“[为帐户管理配置的 pam_ldap 的示例 pam.conf 文件](#)”中的样例。向客户机的 `/etc/pam.conf` 文件中添加包含 `pam_ldap.so.1` 的行。另外，如果样例 `pam.conf` 文件中有任何 PAM 模块指定了 `binding` 标志和 `server_policy` 选项，请在客户机的 `/etc/pam.conf` 文件中为对应模块使用相同的标志和选项。而且，还要将 `server_policy` 选项添加到包含服务模块 `pam_authtok_store.so.1` 的行中。

注 - 以前, 如果启用了 `pam_ldap` 帐户管理, 所有用户在每次登录系统时都必须提供登录口令以进行验证。因此, 使用 `rsh`、`rlogin` 或 `ssh` 等工具进行的不基于口令的登录将会失败。

但是现在, `pam_ldap(5)` 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时, 用户可以使用 `rsh`、`rlogin`、`rcp` 和 `ssh` 登录, 而不需要提供口令。

`pam_ldap(5)` 现已修改为执行帐户管理和检索用户帐户状态, 而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8, 它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态, 请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI) :

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
    allow (read, search, compare, proxy)
    (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

■ binding 控制标志

使用 `binding` 控制标志允许本地口令覆盖远程 (LDAP) 口令。例如, 如果在本地文件和 LDAP 名称空间中都找到了某一用户帐户, 则与本地帐户关联的口令将优先于远程口令。因此, 如果本地口令过期, 即使远程 LDAP 口令仍然有效, 验证也将失败。

■ server_policy 选项

`server_policy` 选项指示 `pam_unix_auth`、`pam_unix_account` 和 `pam_passwd_auth` 忽略在 LDAP 名称空间中找到的用户, 并允许 `pam_ldap` 执行身份验证或帐户验证。对于 `pam_authtok_store`, 会向 LDAP 服务器传递一个未经加密的新口令。因此, 该口令将根据服务器中配置的口令加密方案存储在目录中。有关更多信息, 请参见 `pam.conf(4)` 和 `pam_ldap(5)`。

检索 LDAP 命名服务信息

使用 `ldaplist` 实用程序可以检索有关 LDAP 命名服务的信息。此 LDAP 实用程序会以 LDIF 格式列出 LDAP 服务器中的命名信息。该实用程序可用于进行故障排除。有关详细信息, 请参见 `ldaplist(1)`。

列出所有 LDAP 容器

`ldaplist` 显示输出时以空白行分隔记录，这对于显示包含多行的大量记录很有帮助。

注 - `ldaplist` 的输出取决于客户机配置。例如，如果 `ns_ldap_search` 的值是 `sub` 而不是 `one`，`ldaplist` 将列出在当前搜索 `baseDN` 下的所有项。

下面是 `ldaplist` 输出的示例。

```
# ldaplist
dn: ou=people,dc=west,dc=example,dc=com

dn: ou=group,dc=west,dc=example,dc=com

dn: ou=rpc,dc=west,dc=example,dc=com

dn: ou=protocols,dc=west,dc=example,dc=com

dn: ou=networks,dc=west,dc=example,dc=com

dn: ou=netgroup,dc=west,dc=example,dc=com

dn: ou=aliases,dc=west,dc=example,dc=com

dn: ou=hosts,dc=west,dc=example,dc=com

dn: ou=services,dc=west,dc=example,dc=com

dn: ou=ethers,dc=west,dc=example,dc=com

dn: ou=profile,dc=west,dc=example,dc=com

dn: automountmap=auto_home,dc=west,dc=example,dc=com

dn: automountmap=auto_direct,dc=west,dc=example,dc=com

dn: automountmap=auto_master,dc=west,dc=example,dc=com

dn: automountmap=auto_shared,dc=west,dc=example,dc=com
```

列出所有用户项属性

要列出特定信息（如用户的 `passwd` 项），请按如下所示使用 `getent`：

```
# getent passwd user1
user1::30641:10:Joe Q. User:/home/user1:/bin/csh
```

如果要列出所有属性，请将 `ldaplist` 与 `-l` 选项结合使用。

```
# ldaplist -l passwd user1dn: uid=user1,ou=People,dc=west,dc=example,dc=com
uid: user1
cn: user1
```



```
uidNumber: 30641
gidNumber: 10
gecos: Joe Q. User
homeDirectory: /home/user1
loginShell: /bin/csh
objectClass: top
objectClass: shadowAccount
objectClass: account
objectClass: posixAccount
shadowLastChange: 6445
```

定制 LDAP 客户机环境

以下各节介绍如何定制客户机环境。

您可以更改任何服务，但一定要小心，因为如果未在服务器上为指定的服务置备数据，服务会停止工作。而且，在某些情况下，可能不会按缺省情况设置文件。

为 LDAP 修改 `nsswitch.conf` 文件

您可以修改 `/etc/nsswitch.conf` 文件，定制每个服务获取信息的位置。缺省设置存储在 `/etc/nsswitch.ldap` 中，在初始化客户机时 `ldapclient` 将使用此文件创建 `/etc/nsswitch.conf` 文件。

为 LDAP 启用 DNS

如果要通过设置 `/etc/resolv.conf` 文件来启用 DNS，请按如下所示将 DNS 添加到 `hosts` 行。

```
hosts:      ldap dns [NOTFOUND=return] files
```

推荐的配置是：

```
hosts: files dns
```

```
ipnodes: files dns
```

如果使用每用户验证方式，则需要为 `sasl/GSSAPI` 和 `Kerberos` 机制配置和启用 DNS 命名服务。有关详细信息，请参见本管理指南中有关 DNS 的章节。

LDAP 疑难解答（参考）

本章描述了配置问题以及建议的问题解决方案。

注 - LDAP 服务由服务管理工具管理。使用 `svcadm` 命令可以对此服务执行启用、禁用或重新启动等管理操作。有关针对 LDAP 使用服务管理工具的更多信息，请参见第 174 页中的“LDAP 和服务管理工具”。有关服务管理工具的概述，请参阅《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

监视 LDAP 客户机状态

以下各节介绍了各种可帮助确定 LDAP 客户机环境状态的命令。有关可以使用的选项的其他信息，另请参见相应的手册页。

有关服务管理工具的概述，请参阅《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

验证 `ldap_cachemgr` 是否正在运行

`ldap_cachemgr` 守护进程必须一直正常运行。否则，系统将无法正常工作。当启动 LDAP 客户机时，客户机会自动启动 `ldap_cachemgr` 守护进程。因此，如果 `ldap_cachemgr` 未运行，LDAP 客户机将被禁用。下面两种方法可以判断 LDAP 客户机是否联机。

- 使用 `svcs` 命令。

```
# svcs \*ldap\*
STATE      STIME      FMRI
disabled   Aug_24     svc:/network/ldap/client:default
```

或

```
# svcs -l network/ldap/client:default
fmri          svc:/network/ldap/client:default
enabled       true
state         online
next_state    none
restarter     svc:/system/svc/restarter:default
contract_id   1598
dependency    require_all/none file://localhost/var/ldap/ldap_client_file (-)
dependency    require_all/none svc:/network/initial (online)
dependency    require_all/none svc:/system/filesystem/minimal (online)
```

- 向 `ldap_cachemgr` 传递 `-g` 选项。

此选项提供更广泛的状态信息，这些信息对于问题的诊断很有帮助。

```
# /usr/lib/ldap/ldap_cachemgr -g
cachemgr configuration:
server debug level      0
server log file "/var/ldap/cachemgr.log"
number of calls to ldapcachemgr 19

cachemgr cache data statistics:
Configuration refresh information:
  Previous refresh time: 2001/11/16 18:33:28
  Next refresh time:    2001/11/16 18:43:28
Server information:
  Previous refresh time: 2001/11/16 18:33:28
  Next refresh time:    2001/11/16 18:36:08
  server: 192.168.0.0, status: UP
  server: 192.168.0.1, status: ERROR
    error message: Can't connect to the LDAP server
Cache data information:
  Maximum cache entries: 256
  Number of cache entries: 2
```

有关 `ldap_cachemgr` 守护进程的更多信息，请参见 [ldap_cachemgr\(1M\)](#) 手册页。

检查当前的配置文件信息

成为超级用户或承担等效角色，然后运行带 `list` 选项的 `ldapclient`。

```
# ldapclient list
NS_LDAP_FILE_VERSION= 2.0
NS_LDAP_BINDDN= cn=proxyagent,ou=profile,dc=west,dc=example,dc=com
NS_LDAP_BINDPASSWD= {NS1}4a3788e8c053424f
NS_LDAP_SERVERS= 192.168.0.1, 192.168.0.10
NS_LDAP_SEARCH_BASEDN= dc=west,dc=example,dc=com
NS_LDAP_AUTH= simple
NS_LDAP_SEARCH_REF= TRUE
NS_LDAP_SEARCH_SCOPE= one
NS_LDAP_SEARCH_TIME= 30
NS_LDAP_SERVER_PREF= 192.168.0.1
NS_LDAP_PROFILE= pit1
NS_LDAP_CREDENTIAL_LEVEL= proxy
NS_LDAP_SERVICE_SEARCH_DESC= passwd:ou=people,?sub
NS_LDAP_SERVICE_SEARCH_DESC= group:ou=group,dc=west,dc=example,dc=com?one
NS_LDAP_BIND_TIME= 5
```

目前，`/var/ldap` 文件采用 ASCII 格式。因为文件有时可更改为二进制格式，所以串联文件会产生问题。`ldapclient list` 是受支持的方法，可以用于访问此信息。有关更多信息，请参见 [ldapclient\(1M\)](#) 手册页。

验证基本的客户机/服务器通信

检查客户机是否正在与 LDAP 服务器通信的最佳方法是使用 `ldaplist` 命令。使用不带任何参数的 `ldaplist` 会转储服务器上的所有容器。只要这些容器存在且不必置备，此方法就起作用。有关更多信息，请参见 [ldaplist\(1\)](#) 手册页。

如果第一步起作用，您可以尝试使用 `ldaplist passwd username` 或 `ldaplist hosts hostname`，但是如果容器中包含大量数据，您可能需要选取一个置备量较小的服务，或者将它们传输到 `head` 或 `more`。

从非客户机检查服务器数据

以上各节中的大多数命令都假设您已经创建了 LDAP 客户机。如果您尚未创建客户机，但仍然希望检查服务器上的数据，请使用 `ldapsearch` 命令。以下示例列出了所有容器。

```
# ldapsearch -h server1 -b "dc=west,dc=example,dc=com" -s one "objectclass=*
```

在 Solaris 9 和早期版本中，在缺省情况下，`ldapsearch` 命令会以非标准文本表示形式生成输出。在之后的 Solaris 发行版中，`ldapsearch` 的缺省输出采取行业标准化的 LDIF 格式，该格式由 RFC-2849 定义。所有版本的 `ldapsearch` 都可以使用 `-L` 选项输出 LDIF 格式。

LDAP 配置问题及解决方案

以下各节描述了 LDAP 的配置问题以及建议的解决方案。

无法解析主机名

在执行主机查找时，Solaris 平台 LDAP 客户机后端返回全限定主机名，如由 `gethostbyname()` 和 `getaddrinfo()` 返回的主机名。如果存储的名称是限定名称（即至少包含一个点），则客户机将按原样返回该名称。例如，如果存储的名称是 `hostB.eng`，则返回的名称是 `hostB.eng`。

如果 LDAP 目录中存储的名称不是限定名称（即不包含点），则客户机后端会在该名称后面附加域名部分。例如，如果存储的名称是 `hostA`，则返回的名称是 `hostA.domainname`。

无法远程访问 LDAP 域中的系统

如果 DNS 域名不同于 LDAP 域名，除非所存储的主机名是全限定名称，否则 LDAP 命名服务不能用于提供主机名。

登录功能不起作用

在登录过程中，LDAP 客户机使用 PAM 模块进行用户验证。在使用标准的 UNIX PAM 模块时，口令是从服务器读取并在客户机端检查的。这可能会由于以下某种原因而失败：

1. ldap 未由 /etc/nsswitch.conf 文件中的 passwd 服务使用。
2. 代理无法读取服务器列表中用户的 userPassword 属性。您需要至少允许一个代理可以读取口令，因为该代理需要将口令返回给客户机进行比较。pam_ldap 不需要对口令具有读取访问权限。
3. 代理可能没有正确的口令。
4. 该项没有 shadowAccount 对象类。
5. 没有为该用户定义口令。

在使用 ldapaddent 时，必须使用 -p 选项确保已向该用户项中添加了口令。如果您使用不带 -p 选项的 ldapaddent，用户的口令将不存储在目录中，除非使用 ldapaddent 另外添加了 /etc/shadow 文件。

6. 没有可访问的 LDAP 服务器。
检查服务器的状态。

```
# /usr/lib/ldap/ldap_cachemgr -g
```

7. pam.conf 的配置有误。
8. 没有在 LDAP 名称空间中定义该用户。
9. NS_LDAP_CREDENTIAL_LEVEL 对于 pam_unix 设置为 anonymous，而且 userPassword 对于匿名用户不可用。
10. 口令没有以 crypt 格式存储。
11. 如果所配置的 pam_ldap 支持帐户管理，则登录失败可能是由以下某种原因引起的：
 - 用户的口令已过期。
 - 用户的帐户由于登录失败尝试的次数过多而被锁定。
 - 用户的帐户已被管理员停用。
 - 用户尝试通过不使用口令的程序（如 rsh、rlogin、ssh 或 sftp）进行登录。
12. 如果使用了每用户验证方式和 sasl/GSSAPI，则 Kerberos 的某个组件或 pam_krb5 配置的设置有误。有关解决这些问题的详细信息，请参见《[系统管理指南：安全性服务](#)》。

查找速度太慢

LDAP 数据库依赖索引来改进搜索性能。如果索引的配置有误，会大大降低性能。本文中包括一组应当编制索引的常见属性。您也可以添加自己的索引来提高站点的性能。

ldapclient 无法绑定到服务器

使用指定了 `profileName` 属性的 `init` 选项时，`ldapclient` 初始化客户机失败。失败的可能原因包括：

1. 命令行上指定的域名有误。
2. 没有在 DIT 中设置 `nisDomain` 属性，该属性表示指定客户机域的入口点。
3. 未在服务器上正确设置访问控制信息，从而无法在 LDAP 数据库中进行匿名搜索。
4. 向 `ldapclient` 命令传递的服务器地址有误。请使用 `ldapsearch` 验证服务器地址。
5. 向 `ldapclient` 命令传递的配置文件名称有误。请使用 `ldapsearch` 验证 DIT 中的配置文件名称。
6. 对客户机网络接口使用 `snoop`，看传出的是哪种通信，并确定哪台服务器正与之通信。

使用 ldap_cachemgr 进行调试

使用带 `-g` 选项的 `ldap_cachemgr` 可能是比较有用的调试方法，因为通过它可以查看当前的客户机配置和统计信息。例如，

```
# ldap_cachemgr -g
```

将按上面提到的那样，在标准输出中列显当前的配置和统计信息（包括所有 LDAP 服务器的状态）。请注意，不必成为超级用户即可执行此命令。

ldapclient 在设置过程中挂起

如果 `ldapclient` 命令挂起，则在恢复先前的环境之后按 `Ctrl-C` 将退出。如果出现这种情况，请与服务器管理员核对，以确保该服务器正在运行。

还要在配置文件中或从命令行检查服务器列表中的属性，并确保服务器信息正确无误。

LDAP 一般参考（参考）

本章包含以下主题：

1. 第 193 页中的“空白核对表”
2. 第 194 页中的“LDAP 升级信息”
3. 第 196 页中的“LDAP 命令”
4. 第 197 页中的“pam_ldap 的示例 pam.conf 文件”
5. 第 198 页中的“为帐户管理配置的 pam_ldap 的示例 pam_conf 文件”
6. 第 200 页中的“LDAP 的 IETF 架构”
7. 第 206 页中的“目录用户代理配置文件 (DUAProfile) 架构”
8. 第 207 页中的“Solaris 架构”
9. 第 210 页中的“LDAP 的 Internet 打印协议信息”
10. 第 218 页中的“LDAP 的常规目录服务器要求”
11. 第 218 页中的“LDAP 命名服务使用的缺省过滤器”

空白核对表

表 14-1 服务器变量定义

变量	针对 _____ 网络的定义
安装目录服务器实例的端口号 (389)	
服务器名称	
副本服务器（IP 号:端口号）	
目录管理器 [dn: cn=directory manager]	
要为其提供服务的域名	
在超时之前处理客户端请求的最长时间（以秒为单位）	

表 14-1 服务器变量定义 (续)

变量	针对 _____ 网络的定义
为每个搜索请求返回的最多项数	

表 14-2 客户机配置文件变量定义

变量	针对 _____ 网络的定义
配置文件名	
服务器列表（缺省值为本地子网）	
首选服务器列表（按照对服务器进行查找的顺序列出）	
搜索范围（沿着目录树向下查找的层数。“One”或“Sub”）	
用于获取服务器访问权限的凭证。缺省值为 <code>anonymous</code>	
是否遵循引用？（主服务器不可用时指向另一台服务器的指针）缺省值为 <code>no</code> 。	
等待服务器返回信息的搜索时间限制（以秒为单位，缺省值为 <code>30</code> ）。	
与服务器进行联系时的绑定时间限制（以秒为单位，缺省值为 <code>30</code> ）。The default is seconds.	
验证方法（缺省值为 <code>none</code> ）。	

LDAP 升级信息

本节提供从 Solaris 8 发行版升级到 Solaris 9 或更高发行版时需要考虑的内容。

兼容性

配置了 Solaris 9 或更高 Solaris 软件发行版的客户机与为服务于 Solaris 8 客户机（仅支持 1.0 版的配置文件）而设置的目录服务器完全兼容。但是，要使用 Solaris 9 和更高发行版中的新功能和较新的安全模型，必须使用 2.0 版配置文件。

服务器可以同时为旧客户机和新客户机提供服务。只要架构映射未启用，而且未将 2.0 版文件配置为使用具有 `serviceSearchDescriptors` 属性的特殊过滤器，不同的客户机便可从服务器得到相同的结果。显然，如果服务器不使用缺省架构，旧客户机便无法使用该服务器，因为 Solaris 8 客户机不能任意映射非缺省架构。

运行 ldap_cachemgr 守护进程

从 Solaris 9 发行版开始，`ldap_cachemgr` 守护进程**必须**一直运行。该守护进程是客户机正常运行所**必需的**。使用服务管理工具的 `svcadm` 命令启动 LDAP 客户机时，会自动调用 `ldap_cachemgr` 守护进程。有关更多信息，请参见 [ldap_cachemgr\(1M\)](#) 手册页。

新的 automount 架构

从 Solaris 9 发行版开始，在缺省情况下，Solaris 软件为 `automount` 项使用新的架构。新架构替代了 Solaris 8 客户机使用的普通 NIS 映射架构。这意味着，如果用 Solaris 9 或更高版本软件中的工具设置服务器，Solaris 8 客户机将看不到 `automount` 项。某些站点上的服务器需要同时为 Solaris 8 客户机和使用更高版本 Solaris 软件的客户机提供服务，对于这类站点，可以在添加自动挂载程序项之前创建一个配置文件，将新架构映射到旧架构。这样将确保 [ldapaddent\(1M\)](#) 可添加使用旧架构的项。但请注意，这还意味着所有基于 Solaris 9 或更高版本软件的客户机都必须使用映射了 `automount` 架构的配置文件。

只有向配置文件中添加下列映射属性后，该映射才能生效。

```
attributeMap:      automount:automountMapName=nisMapName
attributeMap:      automount:automountKey=cn
attributeMap:      automount:automountInformation=nisMapEntry
objectclassMap:    automount:automountMap=nisMap
objectclassMap:    automount:automount=nisObject
```

pam_ldap 更改

Solaris 10 OS 发行版中引入了几项与 `pam_ldap` 有关的改动，下面的内容将介绍这些改动。另请参见 [pam_ldap\(5\)](#) 手册页获得更多信息。

- 从 Solaris 10 软件发行版开始，以前受支持的 `use_first_pass` 和 `try_first_pass` 选项将被废弃。因为以后不再需要这些选项，所以可以从 `pam.conf` 中安全地删除它们，也可以将其自动忽略。以后的发行版中将不再包括这些选项。
- 必须通过以下方式提供口令提示：先将 `pam_authtok_get` 放入验证和口令模块栈中，再将 `pam_ldap` 入栈，并将 `pam_passwd_auth` 放入 `passwd` 服务的 `auth` 栈中。
- 此发行版使用以前推荐使用的、带 `server_policy` 选项的 `pam_authtok_store` 来代替以前支持的口令更新功能。

升级到此发行版不会自动更新现有的 `pam.conf` 文件来反映上述更改。如果现有的 `pam.conf` 文件中包含 `pam_ldap` 配置，则在升级之后，系统会通过 `CLEANUP` 文件通知您。您需要检查 `pam.conf` 文件并根据需要修改该文件。

因为在同一个栈中还使用了其他相关模块，而且还会存在第三方模块，所以不可能为上面列出的更改提供全新的自动更新（主要是口令提示和口令更新）。

有关更多信息，请参见 [pam_passwd_auth\(5\)](#)、[pam_authok_get\(5\)](#)、[pam_authok_store\(5\)](#) 和 [pam.conf\(4\)](#) 手册页。

LDAP 命令

Solaris 系统中存在两组与 LDAP 相关的命令。一组命令是常规 LDAP 工具，它们不要求用 LDAP 命名服务配置客户机。另一组命令使用客户机上的通用 LDAP 配置，因而只有客户机使用 LDAP 作为其命名服务时方可使用。

常规 LDAP 工具

LDAP 命令行工具支持一组通用选项（包括验证和绑定参数）。下列工具支持以通用的文本格式来表示目录信息，这种格式称为 LDAP 数据交换格式 (LDAP Data Interchange Format, LDIF)。您可以使用这些命令直接处理目录项。

```
ldapsearch(1)
ldapmodify(1)
ldapadd(1)
ldapdelete(1)
```

需要 LDAP 命名服务的 LDAP 工具

表 14-3 LDAP 工具

工具	功能
ldapaddent(1M)	用于根据对应的 /etc 文件在 LDAP 容器中创建项。此工具允许根据文件置备目录。例如，它读取 /etc/passwd 格式的文件，并置备目录中的 passwd 项。
ldaplist(1)	用于列出目录中各个服务的内容。
idsconfig(1M)	用于设置 Sun Java System Directory Server，使其为 LDAP 命名服务客户机提供服务。

pam_ldap 的示例 pam.conf 文件

```
#
# Authentication management
#
# login service (explicit because of pam_dial_auth)
#
login    auth requisite      pam_authtok_get.so.1
login    auth required       pam_dhkeys.so.1
login    auth required       pam_dial_auth.so.1
login    auth required       pam_unix_cred.so.1
login    auth sufficient     pam_unix_auth.so.1
login    auth required       pam_ldap.so.1
#
# rlogin service (explicit because of pam_rhost_auth)
#
rlogin    auth sufficient     pam_rhosts_auth.so.1
rlogin    auth requisite     pam_authtok_get.so.1
rlogin    auth required       pam_dhkeys.so.1
rlogin    auth required       pam_unix_cred.so.1
rlogin    auth sufficient     pam_unix_auth.so.1
rlogin    auth required       pam_ldap.so.1
#
# rsh service (explicit because of pam_rhost_auth,
# and pam_unix_auth for meaningful pam_setcred)
#
rsh       auth sufficient     pam_rhosts_auth.so.1
rsh       auth required       pam_unix_cred.so.1
#
# PPP service (explicit because of pam_dial_auth)
#
ppp       auth requisite     pam_authtok_get.so.1
ppp       auth required       pam_dhkeys.so.1
ppp       auth required       pam_dial_auth.so.1
ppp       auth sufficient     pam_unix_auth.so.1
ppp       auth required       pam_ldap.so.1
#
# Default definitions for Authentication management
# Used when service name is not explicitly mentioned for authentication
#
other     auth requisite     pam_authtok_get.so.1
other     auth required       pam_dhkeys.so.1
other     auth required       pam_unix_cred.so.1
other     auth sufficient     pam_unix_auth.so.1
other     auth required       pam_ldap.so.1
#
# passwd command (explicit because of a different authentication module)
#
passwd    auth sufficient     pam_passwd_auth.so.1
passwd    auth required       pam_ldap.so.1
#
# cron service (explicit because of non-usage of pam_roles.so.1)
#
cron      account required    pam_unix_account.so.1
#
# Default definition for Account management
# Used when service name is not explicitly mentioned for account management
#
```

```
other    account requisite    pam_roles.so.1
other    account required     pam_unix_account.so.1
#
# Default definition for Session management
# Used when service name is not explicitly mentioned for session management
#
other    session required     pam_unix_session.so.1
#
# Default definition for Password management
# Used when service name is not explicitly mentioned for password management
#
other    password required     pam_dhkeys.so.1
other    password requisite     pam_authtok_get.so.1
other    password requisite     pam_authtok_check.so.1
other    password required     pam_authtok_store.so.1
#
# Support for Kerberos V5 authentication and example configurations can
# be found in the pam_krb5(5) man page under the "EXAMPLES" section.
#
```

为帐户管理配置的 pam_ldap 的示例 pam_conf 文件

注- 以前，如果启用了 pam_ldap 帐户管理，所有用户在每次登录系统时都必须提供登录口令以进行验证。因此，使用 rsh、rlogin 或 ssh 等工具进行的不基于口令的登录将会失败。

但是现在，pam_ldap(5) 与 Sun Java System Directory Servers DS5.2p4 及更高发行版配合使用时，用户可以使用 rsh、rlogin、rcp 和 ssh 登录，而不需要提供口令。

pam_ldap(5) 现已修改为执行帐户管理和检索用户帐户状态，而不需要用户在登录时向目录服务器进行身份验证。目录服务器上对这一操作的新的控制为 1.3.6.1.4.1.42.2.27.9.5.8，它在缺省情况下是启用的。

要将此控制从缺省状态修改为其他状态，请在目录服务器上添加访问控制指令 (Access Control Instructions, ACI)：

```
dn: oid=1.3.6.1.4.1.42.2.27.9.5.8,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid:1.3.6.1.4.1.42.2.27.9.5.8
cn:Password Policy Account Usable Request Control
aci: (targetattr != "aci")(version 3.0; acl "Account Usable";
      allow (read, search, compare, proxy)
      (groupdn = "ldap:///cn=Administrators,cn=config");)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
```

```
#
# Authentication management
#
```

```

# login service (explicit because of pam_dial_auth)
#
login    auth requisite      pam_authtok_get.so.1
login    auth required       pam_dhkeys.so.1
login    auth required       pam_unix_cred.so.1
login    auth required       pam_dial_auth.so.1
login    auth binding        pam_unix_auth.so.1 server_policy
login    auth required       pam_ldap.so.1
#
# rlogin service (explicit because of pam_rhost_auth)
#
rlogin   auth sufficient     pam_rhosts_auth.so.1
rlogin   auth requisite     pam_authtok_get.so.1
rlogin   auth required       pam_dhkeys.so.1
rlogin   auth required       pam_unix_cred.so.1
rlogin   auth binding        pam_unix_auth.so.1 server_policy
rlogin   auth required       pam_ldap.so.1
#
# rsh service (explicit because of pam_rhost_auth,
# and pam_unix_auth for meaningful pam_setcred)
#
rsh      auth sufficient     pam_rhosts_auth.so.1
rsh      auth required       pam_unix_cred.so.1
rsh      auth binding        pam_unix_auth.so.1 server_policy
rsh      auth required       pam_ldap.so.1
#
# PPP service (explicit because of pam_dial_auth)
#
ppp      auth requisite     pam_authtok_get.so.1
ppp      auth required       pam_dhkeys.so.1
ppp      auth required       pam_dial_auth.so.1
ppp      auth binding        pam_unix_auth.so.1 server_policy
ppp      auth required       pam_ldap.so.1
#
# Default definitions for Authentication management
# Used when service name is not explicitly mentioned for authentication
#
other    auth requisite     pam_authtok_get.so.1
other    auth required       pam_dhkeys.so.1
other    auth required       pam_unix_cred.so.1
other    auth binding        pam_unix_auth.so.1 server_policy
other    auth required       pam_ldap.so.1
#
# passwd command (explicit because of a different authentication module)
#
passwd   auth binding        pam_passwd_auth.so.1 server_policy
passwd   auth required       pam_ldap.so.1
#
# cron service (explicit because of non-usage of pam_roles.so.1)
#
cron     account required    pam_unix_account.so.1
#
# Default definition for Account management
# Used when service name is not explicitly mentioned for account management
#
other    account requisite   pam_roles.so.1
other    account binding     pam_unix_account.so.1 server_policy
other    account required    pam_ldap.so.1
#

```

```
# Default definition for Session management
# Used when service name is not explicitly mentioned for session management
#
other    session required      pam_unix_session.so.1
#
# Default definition for Password management
# Used when service name is not explicitly mentioned for password management
#
other    password required     pam_dhkeys.so.1
other    password requisite     pam_authtok_get.so.1
other    password requisite     pam_authtok_check.so.1
other    password required     pam_authtok_store.so.1 server_policy
#
# Support for Kerberos V5 authentication and example configurations can
# be found in the pam_krb5(5) man page under the "EXAMPLES" section.
#
```

LDAP 的 IETF 架构

架构是一些定义，用于描述哪些类型的信息可作为条目存储在服务器的目录中。

为了使目录服务器支持 Solaris LDAP 命名客户机，本章中定义的架构配置在服务器中，除非该架构使用客户机的架构映射功能进行了映射。

IETF 定义了三个必需的 LDAP 架构：RFC 2307 网络信息服务架构、LDAP 邮件组 Internet 草案和 LDAP Internet 打印协议 (Internet Print Protocol, IPP) 草案架构。要支持命名信息服务，必须将这些架构的定义添加到目录服务器中。您还可以从 IETF Web 站点 <http://www.ietf.org> 访问各种 RFC。

注 - Internet 草案是草案文档，有效期最长六个月，随时可能因其他文档的发布而更新或废弃。

RFC 2307 网络信息服务架构

必须对 LDAP 服务器进行配置，以支持修订后的 RFC 2307。

nisSchema OID 是 1.3.6.1.1。RFC 2307 的属性如下所示。

```
( nisSchema.1.0 NAME 'uidNumber'
DESC 'An integer uniquely identifying a user in an
      administrative domain'
EQUALITY integerMatch SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.1 NAME 'gidNumber'
DESC 'An integer uniquely identifying a group in an
      administrative domain'
EQUALITY integerMatch SYNTAX 'INTEGER' SINGLE-VALUE )
```



```

( nisSchema.1.2 NAME 'gecos'
DESC 'The GECOS field; the common name'
EQUALITY caseIgnoreIA5Match
SUBSTRINGS caseIgnoreIA5SubstringsMatch
SYNTAX 'IA5String' SINGLE-VALUE )

( nisSchema.1.3 NAME 'homeDirectory'
DESC 'The absolute path to the home directory'
EQUALITY caseExactIA5Match
SYNTAX 'IA5String' SINGLE-VALUE )

( nisSchema.1.4 NAME 'loginShell'
DESC 'The path to the login shell'
EQUALITY caseExactIA5Match
SYNTAX 'IA5String' SINGLE-VALUE )

( nisSchema.1.5 NAME 'shadowLastChange'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.6 NAME 'shadowMin'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.7 NAME 'shadowMax'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.8 NAME 'shadowWarning'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.9 NAME 'shadowInactive'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.10 NAME 'shadowExpire'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.11 NAME 'shadowFlag'
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.12 NAME 'memberUid'
EQUALITY caseExactIA5Match
SUBSTRINGS caseExactIA5SubstringsMatch
SYNTAX 'IA5String' )

( nisSchema.1.13 NAME 'memberNisNetgroup'
EQUALITY caseExactIA5Match
SUBSTRINGS caseExactIA5SubstringsMatch
SYNTAX 'IA5String' )

( nisSchema.1.14 NAME 'nisNetgroupTriple'
DESC 'Netgroup triple'
SYNTAX 'nisNetgroupTripleSyntax' )

( nisSchema.1.15 NAME 'ipServicePort'

```

```
EQUALITY integerMatch
SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.16 NAME 'ipServiceProtocol'
  SUP name )

( nisSchema.1.17 NAME 'ipProtocolNumber'
  EQUALITY integerMatch
  SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.18 NAME 'oncRpcNumber'
  EQUALITY integerMatch
  SYNTAX 'INTEGER' SINGLE-VALUE )

( nisSchema.1.19 NAME 'ipHostNumber'
  DESC 'IP address as a dotted decimal, eg. 192.168.1.1
        omitting leading zeros'
  SUP name )

( nisSchema.1.20 NAME 'ipNetworkNumber'
  DESC 'IP network as a dotted decimal, eg. 192.168,
        omitting leading zeros'
  SUP name SINGLE-VALUE )

( nisSchema.1.21 NAME 'ipNetmaskNumber'
  DESC 'IP netmask as a dotted decimal, eg. 255.255.255.0,
        omitting leading zeros'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String{128}' SINGLE-VALUE )

( nisSchema.1.22 NAME 'macAddress'
  DESC 'MAC address in maximal, colon separated hex
        notation, eg. 00:00:92:90:ee:e2'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String{128}' )

( nisSchema.1.23 NAME 'bootParameter'
  DESC 'rpc.bootparamd parameter'
  SYNTAX 'bootParameterSyntax' )

( nisSchema.1.24 NAME 'bootFile'
  DESC 'Boot image name'
  EQUALITY caseExactIA5Match
  SYNTAX 'IA5String' )

( nisSchema.1.26 NAME 'nisMapName'
  SUP name )

( nisSchema.1.27 NAME 'nisMapEntry'
  EQUALITY caseExactIA5Match
  SUBSTRINGS caseExactIA5SubstringsMatch
  SYNTAX 'IA5String{1024}' SINGLE-VALUE )

( nisSchema.1.28 NAME 'nisPublicKey'
  DESC 'NIS public key'
  SYNTAX 'nisPublicKeySyntax' )

( nisSchema.1.29 NAME 'nisSecretKey'
  DESC 'NIS secret key'
```

```

SYNTAX 'nisSecretKeySyntax' )

( nisSchema.1.30 NAME 'nisDomain'
DESC 'NIS domain'
SYNTAX 'IA5String' )

( nisSchema.1.31 NAME 'automountMapName'
DESC 'automount Map Name'
EQUALITY caseExactIA5Match
SUBSTR caseExactIA5SubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )

( nisSchema.1.32 NAME 'automountKey'
DESC 'Automount Key value'
EQUALITY caseExactIA5Match
SUBSTR caseExactIA5SubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )

( nisSchema.1.33 NAME 'automountInformation'
DESC 'Automount information'
EQUALITY caseExactIA5Match
SUBSTR caseExactIA5SubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )

```

nisSchema OID 是 1.3.6.1.1。RFC 2307 objectClasses 如下所示。

```

( nisSchema.2.0 NAME 'posixAccount' SUP top AUXILIARY
DESC 'Abstraction of an account with POSIX attributes'
MUST ( cn $ uid $ uidNumber $ gidNumber $ homeDirectory )
MAY ( userPassword $ loginShell $ gecos $ description ) )

( nisSchema.2.1 NAME 'shadowAccount' SUP top AUXILIARY
DESC 'Additional attributes for shadow passwords'
MUST uid
MAY ( userPassword $ shadowLastChange $ shadowMin
      shadowMax $ shadowWarning $ shadowInactive $
      shadowExpire $ shadowFlag $ description ) )

( nisSchema.2.2 NAME 'posixGroup' SUP top STRUCTURAL
DESC 'Abstraction of a group of accounts'
MUST ( cn $ gidNumber )
MAY ( userPassword $ memberUid $ description ) )

( nisSchema.2.3 NAME 'ipService' SUP top STRUCTURAL
DESC 'Abstraction an Internet Protocol service.
      Maps an IP port and protocol (such as tcp or udp)
      to one or more names; the distinguished value of
      the cn attribute denotes the service's canonical
      name'
MUST ( cn $ ipServicePort $ ipServiceProtocol )
MAY ( description ) )

( nisSchema.2.4 NAME 'ipProtocol' SUP top STRUCTURAL
DESC 'Abstraction of an IP protocol. Maps a protocol number
      to one or more names. The distinguished value of the cn
      attribute denotes the protocol's canonical name'
MUST ( cn $ ipProtocolNumber )
MAY description )

```

```
( nisSchema.2.5 NAME 'oncRpc' SUP top STRUCTURAL
DESC 'Abstraction of an Open Network Computing (ONC)
[RFC1057] Remote Procedure Call (RPC) binding.
This class maps an ONC RPC number to a name.
The distinguished value of the cn attribute denotes
the RPC service's canonical name'
MUST ( cn $ oncRpcNumber $ description )
MAY description )

( nisSchema.2.6 NAME 'ipHost' SUP top AUXILIARY
DESC 'Abstraction of a host, an IP device. The distinguished
value of the cn attribute denotes the host's canonical
name. Device SHOULD be used as a structural class'
MUST ( cn $ ipHostNumber )
MAY ( l $ description $ manager $ userPassword ) )

( nisSchema.2.7 NAME 'ipNetwork' SUP top STRUCTURAL
DESC 'Abstraction of a network. The distinguished value of
the cn attribute denotes the network's canonical name'
MUST ipNetworkNumber
MAY ( cn $ ipNetmaskNumber $ l $ description $ manager ) )

( nisSchema.2.8 NAME 'nisNetgroup' SUP top STRUCTURAL
DESC 'Abstraction of a netgroup. May refer to other netgroups'
MUST cn
MAY ( nisNetgroupTriple $ memberNisNetgroup $ description ) )

( nisSchema.2.9 NAME 'nisMap' SUP top STRUCTURAL
DESC 'A generic abstraction of a NIS map'
MUST nisMapName
MAY description )

( nisSchema.2.10 NAME 'nisObject' SUP top STRUCTURAL
DESC 'An entry in a NIS map'
MUST ( cn $ nisMapEntry $ nisMapName )
MAY description )

( nisSchema.2.11 NAME 'ieee802Device' SUP top AUXILIARY
DESC 'A device with a MAC address; device SHOULD be
used as a structural class'
MAY macAddress )

( nisSchema.2.12 NAME 'bootableDevice' SUP top AUXILIARY
DESC 'A device with boot parameters; device SHOULD be
used as a structural class'
MAY ( bootFile $ bootParameter ) )

( nisSchema.2.14 NAME 'nisKeyObject' SUP top AUXILIARY
DESC 'An object with a public and secret key'
MUST ( cn $ nisPublicKey $ nisSecretKey )
MAY ( uidNumber $ description ) )

( nisSchema.2.15 NAME 'nisDomainObject' SUP top AUXILIARY
DESC 'Associates a NIS domain with a naming context'
MUST nisDomain )

( nisSchema.2.16 NAME 'automountMap' SUP top STRUCTURAL
MUST ( automountMapName )
```

```

MAY description )

( nisSchema.2.17 NAME 'automount' SUP top STRUCTURAL
  DESC 'Automount information'
  MUST ( automountKey $ automountInformation )
  MAY description )

```

邮件别名架构

邮件别名信息使用由 LDAP 邮件组 Internet 草案（以前称为 draft-steinback-ldap-mailgroups 草案）定义的架构。Solaris LDAP 客户机将继续对邮件别名信息使用此架构，直到有新的架构可用。

原来的 LDAP 邮件组架构中包含大量属性和对象类。Solaris 客户机仅使用两个属性和一个对象类。这些属性和对象类如下所示。

邮件别名属性如下所示。

```

( 0.9.2342.19200300.100.1.3
  NAME 'mail'
  DESC 'RFC822 email address for this person'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String(256)'
  SINGLE-VALUE )

( 2.16.840.1.113730.3.1.30
  NAME 'mgrpRFC822MailMember'
  DESC 'RFC822 mail address of email only member of group'
  EQUALITY CaseIgnoreIA5Match
  SYNTAX 'IA5String(256)' )

```

邮件别名 objectClass 如下所示。

```

( 2.16.840.1.113730.3.2.4
  NAME 'mailGroup'
  SUP top
  STRUCTURAL
  MUST mail
  MAY ( cn $ mailAlternateAddress $ mailHost $ mailRequireAuth $
    mgrpAddHeader $ mgrpAllowedBroadcaster $ mgrpAllowedDomain $
    mgrpApprovePassword $ mgrpBroadcasterModeration $ mgrpDeliverTo $
    mgrpErrorsTo $ mgrpModerator $ mgrpMsgMaxSize $
    mgrpMsgRejectAction $ mgrpMsgRejectText $ mgrpNoMatchAddrs $
    mgrpRemoveHeader $ mgrpRFC822MailMember ) )

```

目录用户代理配置文件 (DUAPProfile) 架构

DUAConfSchemaOID 是 1.3.6.1.4.1.11.1.3.1。

```
DESC 'Default LDAP server host address used by a DUA'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

( DUAConfSchemaOID.1.1 NAME 'defaultSearchBase'
DESC 'Default LDAP base DN used by a DUA'
EQUALITY distinguishedNameMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE )

( DUAConfSchemaOID.1.2 NAME 'preferredServerList'
DESC 'Preferred LDAP server host addresses to be used by a
DUA'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

( DUAConfSchemaOID.1.3 NAME 'searchTimeLimit'
DESC 'Maximum time in seconds a DUA should allow for a
search to complete'
EQUALITY integerMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

( DUAConfSchemaOID.1.4 NAME 'bindTimeLimit'
DESC 'Maximum time in seconds a DUA should allow for the
bind operation to complete'
EQUALITY integerMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE )

( DUAConfSchemaOID.1.5 NAME 'followReferrals'
DESC 'Tells DUA if it should follow referrals
returned by a DSA search result'
EQUALITY caseIgnoreIA5Match
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE )

( DUAConfSchemaOID.1.6 NAME 'authenticationMethod'
DESC 'A kestring which identifies the type of
authentication method used to contact the DSA'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE )

( DUAConfSchemaOID.1.7 NAME 'profileTTL'
DESC 'Time to live, in seconds, before a client DUA
should re-read this configuration profile'
'serviceSearchDescriptor'
DESC 'LDAP search descriptor list used by a DUA'
EQUALITY caseExactMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

```

( DUAConfSchemaOID.1.9 NAME 'attributeMap'
  DESC 'Attribute mappings used by a DUA'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

( DUAConfSchemaOID.1.10 NAME 'credentialLevel'
  DESC 'Identifies type of credentials a DUA should
  use when binding to the LDAP server'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
  SINGLE-VALUE )

( DUAConfSchemaOID.1.11 NAME 'objectclassMap'
  DESC 'Objectclass mappings used by a DUA'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

( DUAConfSchemaOID.1.12 NAME 'defaultSearchScope' SINGLE-VALUE )

( DUAConfSchemaOID.1.13 NAME 'serviceCredentialLevel'
  DESC 'Identifies type of credentials a DUA
  should use when binding to the LDAP server for a
  specific service'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

( DUAConfSchemaOID.1.15 NAME 'serviceAuthenticationMethod'
  DESC 'Authentication Method used by a service of the DUA'
  EQUALITY caseIgnoreMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

( DUAConfSchemaOID.2.4 NAME 'DUAConfigProfile'
  SUP top STRUCTURAL
  DESC 'Abstraction of a base configuration for a DUA'
  MUST ( cn )
  MAY ( defaultServerList $ preferredServerList $
  defaultSearchBase $ defaultSearchScope $
  searchTimeLimit $ bindTimeLimit $
  credentialLevel $ authenticationMethod $
  followReferrals $ serviceSearchDescriptor $
  serviceCredentialLevel $ serviceAuthenticationMethod $
  objectclassMap $ attributeMap $
  profileTTL ) )

```

Solaris 架构

Solaris 平台所需的架构如下所示。

- Solaris 项目架构
- 基于角色的访问控制和执行配置文件架构
- 打印机架构

Solaris 项目架构

/etc/project 是与项目相关联的属性的本地源。有关更多信息，请参见 [user_attr\(4\)](#)。

项目属性如下所示。

```
( 1.3.6.1.4.1.42.2.27.5.1.1 NAME 'SolarisProjectID'
  DESC 'Unique ID for a Solaris Project entry'
  EQUALITY integerMatch
  SYNTAX INTEGER SINGLE )

( 1.3.6.1.4.1.42.2.27.5.1.2 NAME 'SolarisProjectName'
  DESC 'Name of a Solaris Project entry'
  EQUALITY caseExactIA5Match
  SYNTAX IA5String SINGLE )

( 1.3.6.1.4.1.42.2.27.5.1.3 NAME 'SolarisProjectAttr'
  DESC 'Attributes of a Solaris Project entry'
  EQUALITY caseExactIA5Match
  SYNTAX IA5String )

( 1.3.6.1.4.1.42.2.27.5.1.30 NAME 'memberGid'
  DESC 'Posix Group Name'
  EQUALITY caseExactIA5Match
  SYNTAX 'IA5String' )
```

项目 objectClass 如下所示。

```
( 1.3.6.1.4.1.42.2.27.5.2.1 NAME 'SolarisProject'
  SUP top STRUCTURAL
  MUST ( SolarisProjectID $ SolarisProjectName )
  MAY ( memberUid $ memberGid $ description $ SolarisProjectAttr ) )
```

基于角色的访问控制和执行配置文件架构

/etc/user_attr 是与用户和角色相关联的扩展属性的本地源。有关更多信息，请参见 [user_attr\(4\)](#)。

基于角色的访问控制属性如下所示。

```
( 1.3.6.1.4.1.42.2.27.5.1.4 NAME 'SolarisAttrKeyValue'
  DESC 'Semi-colon separated key=value pairs of attributes'
  EQUALITY caseIgnoreIA5Match
  SUBSTRINGS caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.7 NAME 'SolarisAttrShortDesc'
  DESC 'Short description about an entry, used by GUIs'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.8 NAME 'SolarisAttrLongDesc'
  DESC 'Detail description about an entry'
```



```

EQUALITY caseIgnoreIA5Match
SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.9 NAME 'SolarisKernelSecurityPolicy'
  DESC 'Solaris kernel security policy'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.10 NAME 'SolarisProfileType'
  DESC 'Type of object defined in profile'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.11 NAME 'SolarisProfileId'
  DESC 'Identifier of object defined in profile'
  EQUALITY caseExactIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.12 NAME 'SolarisUserQualifier'
  DESC 'Per-user login attributes'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.13 NAME 'SolarisReserved1'
  DESC 'Reserved for future use'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.14 NAME 'SolarisReserved2'
  DESC 'Reserved for future use'
  EQUALITY caseIgnoreIA5Match
  SYNTAX 'IA5String' SINGLE-VALUE )

```

基于角色的访问控制 objectClasses 如下所示。

```

( 1.3.6.1.4.1.42.2.27.5.2.3 NAME 'SolarisUserAttr' SUP top AUXILIARY
  DESC 'User attributes'
  MAY ( SolarisUserQualifier $ SolarisAttrReserved1 $ \
        SolarisAttrReserved2 $ SolarisAttrKeyValue ) )

( 1.3.6.1.4.1.42.2.27.5.2.4 NAME 'SolarisAuthAttr' SUP top STRUCTURAL
  DESC 'Authorizations data'
  MUST cn
  MAY ( SolarisAttrReserved1 $ SolarisAttrReserved2 $ \
        SolarisAttrShortDesc $ SolarisAttrLongDesc $ \
        SolarisAttrKeyValue ) )

( 1.3.6.1.4.1.42.2.27.5.2.5 NAME 'SolarisProfAttr' SUP top STRUCTURAL
  DESC 'Profiles data'
  MUST cn
  MAY ( SolarisAttrReserved1 $ SolarisAttrReserved2 $ \
        SolarisAttrLongDesc $ SolarisAttrKeyValue ) )

( 1.3.6.1.4.1.42.2.27.5.2.6 NAME 'SolarisExecAttr' SUP top AUXILIARY
  DESC 'Profiles execution attributes'
  MAY ( SolarisKernelSecurityPolicy $ SolarisProfileType $ \
        SolarisAttrReserved1 $ SolarisAttrReserved2 $ \
        SolarisProfileId $ SolarisAttrKeyValue ) )

```

LDAP 的 Internet 打印协议信息

以下各节提供有关 Internet 打印协议和 Sun 打印机的属性及 ObjectClasses 的信息。

Internet 打印协议 (Internet Printing Protocol, IPP) 属性

```
( 1.3.18.0.2.4.1140
NAME 'printer-uri'
DESC 'A URI supported by this printer.
This URI SHOULD be used as a relative distinguished name (RDN).
If printer-xri-supported is implemented, then this URI value
MUST be listed in a member value of printer-xri-supported.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )

( 1.3.18.0.2.4.1107
NAME 'printer-xri-supported'
DESC 'The unordered list of XRI (extended resource identifiers) supported
by this printer.
Each member of the list consists of a URI (uniform resource identifier)
followed by optional authentication and security metaparameters.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

( 1.3.18.0.2.4.1135
NAME 'printer-name'
DESC 'The site-specific administrative name of this printer, more end-user
friendly than a URI.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} SINGLE-VALUE )

( 1.3.18.0.2.4.1119
NAME 'printer-natural-language-configured'
DESC 'The configured language in which error and status messages will be
generated (by default) by this printer.
Also, a possible language for printer string attributes set by operator,
system administrator, or manufacturer.
Also, the (declared) language of the "printer-name", "printer-location",
"printer-info", and "printer-make-and-model" attributes of this printer.
For example: "en-us" (US English) or "fr-fr" (French in France) Legal values of
language tags conform to [RFC3066] "Tags for the Identification of Languages".
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} SINGLE-VALUE )
```

```
( 1.3.18.0.2.4.1136
NAME 'printer-location'
DESC 'Identifies the location of the printer. This could include
things like: "in Room 123A", "second floor of building XYZ".'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} SINGLE-VALUE )

( 1.3.18.0.2.4.1139
NAME 'printer-info'
DESC 'Identifies the descriptive information about this printer.
This could include things like: "This printer can be used for
printing color transparencies for HR presentations", or
"Out of courtesy for others, please print only small (1-5 page)
jobs at this printer", or even "This printer is going away on July 1, 1997,
please find a new printer".'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127}
SINGLE-VALUE )

( 1.3.18.0.2.4.1134
NAME 'printer-more-info'
DESC 'A URI used to obtain more information about this specific printer.
For example, this could be an HTTP type URI referencing an HTML page
accessible to a Web Browser.
The information obtained from this URI is intended for end user consumption.'
EQUALITY caseIgnoreMatch ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )

( 1.3.18.0.2.4.1138
NAME 'printer-make-and-model'
DESC 'Identifies the make and model of the device.
The device manufacturer MAY initially populate this attribute.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} SINGLE-VALUE )

( 1.3.18.0.2.4.1133
NAME 'printer-ipp-versions-supported'
DESC 'Identifies the IPP protocol version(s) that this printer supports,
including major and minor versions,
i.e., the version numbers for which this Printer implementation meets
the conformance requirements.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} )

( 1.3.18.0.2.4.1132
NAME 'printer-multiple-document-jobs-supported'
DESC 'Indicates whether or not the printer supports more than one
document per job, i.e., more than one Send-Document or Send-Data
operation with document data.'
EQUALITY booleanMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 SINGLE-VALUE )
```

```
( 1.3.18.0.2.4.1109
NAME 'printer-charset-configured'
DESC 'The configured charset in which error and status messages will be
generated (by default) by this printer.
Also, a possible charset for printer string attributes set by operator,
system administrator, or manufacturer.
For example: "utf-8" (ISO 10646/Unicode) or "iso-8859-1" (Latin1).
Legal values are defined by the IANA Registry of Coded Character Sets and
the "(preferred MIME name)" SHALL be used as the tag.
For coherence with IPP Model, charset tags in this attribute SHALL be
lowercase normalized.
This attribute SHOULD be static (time of registration) and SHOULD NOT be
dynamically refreshed attributetypes: (subsequently).'
```

EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{63} SINGLE-VALUE)

```
( 1.3.18.0.2.4.1131
NAME 'printer-charset-supported'
DESC 'Identifies the set of charsets supported for attribute type values of
type Directory String for this directory entry.
For example: "utf-8" (ISO 10646/Unicode) or "iso-8859-1" (Latin1).
Legal values are defined by the IANA Registry of Coded Character Sets and
the preferred MIME name.'
```

EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{63})

```
( 1.3.18.0.2.4.1137
NAME 'printer-generated-natural-language-supported'
DESC 'Identifies the natural language(s) supported for this directory entry.
For example: "en-us" (US English) or "fr-fr" (French in France).
Legal values conform to [RFC3066], Tags for the Identification of Languages.'
```

EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{63})

```
( 1.3.18.0.2.4.1130
NAME 'printer-document-format-supported'
DESC 'The possible document formats in which data may be interpreted
and printed by this printer.
Legal values are MIME types come from the IANA Registry of Internet Media Types.'
```

EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127})

```
( 1.3.18.0.2.4.1129
NAME 'printer-color-supported'
DESC 'Indicates whether this printer is capable of any type of color printing
at all, including highlight color.'
```

EQUALITY booleanMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 SINGLE-VALUE)

```
( 1.3.18.0.2.4.1128
NAME 'printer-compression-supported'
DESC 'Compression algorithms supported by this printer.
For example: "deflate, gzip". Legal values include; "none", "deflate"
attributetypes: (public domain ZIP), "gzip" (GNU ZIP), "compress" (UNIX).'
```

EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{255})

```
( 1.3.18.0.2.4.1127
NAME 'printer-pages-per-minute'
DESC 'The nominal number of pages per minute which may be output by this
printer (e.g., a simplex or black-and-white printer).
This attribute is informative, NOT a service guarantee.
Typically, it is the value used in marketing literature to describe this printer.'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE )

( 1.3.18.0.2.4.1126 NAME 'printer-pages-per-minute-color'
DESC 'The nominal number of color pages per minute which may be output by this
printer (e.g., a simplex or color printer).
This attribute is informative, NOT a service guarantee.
Typically, it is the value used in marketing literature to describe this printer.'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE )

( 1.3.18.0.2.4.1125 NAME 'printer-finishings-supported'
DESC 'The possible finishing operations supported by this printer.
Legal values include; "none", "staple", "punch", "cover", "bind", "saddle-stitch",
"edge-stitch", "staple-top-left", "staple-bottom-left", "staple-top-right",
"staple-bottom-right", "edge-stitch-left", "edge-stitch-top", "edge-stitch-right",
"edge-stitch-bottom", "staple-dual-left", "staple-dual-top", "staple-dual-right",
"staple-dual-bottom".'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{255} )

( 1.3.18.0.2.4.1124 NAME 'printer-number-up-supported'
DESC 'The possible numbers of print-stream pages to impose upon a single side of
an instance of a selected medium. Legal values include; 1, 2, and 4.
Implementations may support other values.'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 )

( 1.3.18.0.2.4.1123 NAME 'printer-sides-supported'
DESC 'The number of impression sides (one or two) and the two-sided impression
rotations supported by this printer.
Legal values include; "one-sided", "two-sided-long-edge", "two-sided-short-edge".'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} )

( 1.3.18.0.2.4.1122 NAME 'printer-media-supported'
DESC 'The standard names/types/sizes (and optional color suffixes) of the media
supported by this printer.
For example: "iso-a4", "envelope", or "na-letter-white".
Legal values conform to ISO 10175, Document Printing Application (DPA), and any
IANA registered extensions.'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{255} )

( 1.3.18.0.2.4.1117 NAME 'printer-media-local-supported'
DESC 'Site-specific names of media supported by this printer, in the language in
"printer-natural-language-configured".
```

For example: "purchasing-form" (site-specific name) as opposed to
(in "printer-media-supported"): "na-letter" (standard keyword from ISO 10175).'
EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{255})

(1.3.18.0.2.4.1121 NAME 'printer-resolution-supported'
DESC 'List of resolutions supported for printing documents by this printer.
Each resolution value is a string with 3 fields:
1) Cross feed direction resolution (positive integer), 2) Feed direction
resolution (positive integer), 3) Resolution unit.
Legal values are "dpi" (dots per inch) and "dpcm" (dots per centimeter).
Each resolution field is delimited by ">". For example: "300> 300> dpi>".'
EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{255})

(1.3.18.0.2.4.1120 NAME 'printer-print-quality-supported'
DESC 'List of print qualities supported for printing documents on this printer.
For example: "draft, normal". Legal values include; "unknown", "draft", "normal",
"high".'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127})

(1.3.18.0.2.4.1110 NAME 'printer-job-priority-supported'
DESC 'Indicates the number of job priority levels supported.
An IPP conformant printer which supports job priority must always support a
full range of priorities from "1" to "100"
(to ensure consistent behavior), therefore this attribute describes the
"granularity".
Legal values of this attribute are from "1" to "100".'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE)

(1.3.18.0.2.4.1118
NAME 'printer-copies-supported'
DESC 'The maximum number of copies of a document that may be printed as a single job.
A value of "0" indicates no maximum limit.
A value of "-1" indicates unknown.'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE)

(1.3.18.0.2.4.1111
NAME 'printer-job-k-octets-supported'
DESC 'The maximum size in kilobytes (1,024 octets actually) incoming print job that
this printer will accept.
A value of "0" indicates no maximum limit. A value of "-1" indicates unknown.'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE)

(1.3.18.0.2.4.1113
NAME 'printer-service-person'
DESC 'The name of the current human service person responsible for servicing this
printer.
It is suggested that this string include information that would enable other humans

```

to reach the service person, such as a phone number.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127}
SINGLE-VALUE )

( 1.3.18.0.2.4.1114
NAME 'printer-delivery-orientation-supported'
DESC 'The possible delivery orientations of pages as they are printed and ejected
from this printer.
Legal values include; "unknown", "face-up", and "face-down".'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} )

( 1.3.18.0.2.4.1115
NAME 'printer-stacking-order-supported'
DESC 'The possible stacking order of pages as they are printed and ejected from
this printer.
Legal values include; "unknown", "first-to-last", "last-to-first".'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} )

( 1.3.18.0.2.4.1116
NAME 'printer-output-features-supported'
DESC 'The possible output features supported by this printer.
Legal values include; "unknown", "bursting", "decollating", "page-collating",
"offset-stacking".'
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} )

( 1.3.18.0.2.4.1108
NAME 'printer-aliases'
DESC 'Site-specific administrative names of this printer in addition the printer
name specified for printer-name.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SUBSTR caseIgnoreSubstringsMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{127} )

( 1.3.6.1.4.1.42.2.27.5.1.63
NAME 'sun-printer-bsdaddr'
DESC 'Sets the server, print queue destination name and whether the client generates
protocol extensions.
"Solaris" specifies a Solaris print server extension. The value is represented b the
following value: server ",", destination ",", Solaris".'
SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' SINGLE-VALUE )

( 1.3.6.1.4.1.42.2.27.5.1.64
NAME 'sun-printer-kvp'
DESC 'This attribute contains a set of key value pairs which may have meaning to the
print subsystem or may be user defined.
Each value is represented by the following: key "=" value.'
SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' )

```

Internet 打印协议 (Internet Printing Protocol, IPP) ObjectClasses

```
objectclasses: ( 1.3.18.0.2.6.2549
NAME 'slpService'
DESC 'DUMMY definition'
SUP 'top' MUST (objectclass) MAY ( ))
```

```
objectclasses: ( 1.3.18.0.2.6.254
NAME 'slpServicePrinter'
DESC 'Service Location Protocol (SLP) information.'
AUXILIARY SUP 'slpService')
```

```
objectclasses: ( 1.3.18.0.2.6.258
NAME 'printerAbstract'
DESC 'Printer related information.'
ABSTRACT SUP 'top' MAY ( printer-name
$ printer-natural-language-configured
$ printer-location
$ printer-info
$ printer-more-info
$ printer-make-and-model
$ printer-multiple-document-jobs-supported
$ printer-charset-configured
$ printer-charset-supported
$ printer-generated-natural-language-supported
$ printer-document-format-supported
$ printer-color-supported
$ printer-compression-supported
$ printer-pages-per-minute
$ printer-pages-per-minute-color
$ printer-finishings-supported
$ printer-number-up-supported
$ printer-sides-supported
$ printer-media-supported
$ printer-media-local-supported
$ printer-resolution-supported
$ printer-print-quality-supported
$ printer-job-priority-supported
$ printer-copies-supported
$ printer-job-k-octets-supported
$ printer-current-operator
$ printer-service-person
$ printer-delivery-orientation-supported
$ printer-stacking-order-supported $ printer! -output-features-supported ))
```

```
objectclasses: ( 1.3.18.0.2.6.255
NAME 'printerService'
DESC 'Printer information.'
STRUCTURAL SUP 'printerAbstract' MAY ( printer-uri
$ printer-xri-supported ))
```

```
objectclasses: ( 1.3.18.0.2.6.257
NAME 'printerServiceAuxClass'
DESC 'Printer information.'
AUXILIARY SUP 'printerAbstract' MAY ( printer-uri $ printer-xri-supported ))
```



```

objectclasses: ( 1.3.18.0.2.6.256
NAME 'printerIPP'
DESC 'Internet Printing Protocol (IPP) information.'
AUXILIARY SUP 'top' MAY ( printer-ipp-versions-supported $
printer-multiple-document-jobs-supported ))

objectclasses: ( 1.3.18.0.2.6.253
NAME 'printerLPR'
DESC 'LPR information.'
AUXILIARY SUP 'top' MUST ( printer-name ) MAY ( printer-aliases))

objectclasses: ( 1.3.6.1.4.1.42.2.27.5.2.14
NAME 'sunPrinter'
DESC 'Sun printer information'
SUP 'top' AUXILIARY MUST (objectclass $ printer-name) MAY
(sun-printer-bsdaddr $ sun-printer-kvp))

```

Sun 打印机属性

```

ATTRIBUTE ( 1.3.6.1.4.1.42.2.27.5.1.63
NAME sun-printer-bsdaddr
DESC 'Sets the server, print queue destination name and whether the
client generates protocol extensions. "Solaris" specifies a
Solaris print server extension. The value is represented by
the following value: server ",", destination ",", Solaris".'
EQUALITY caseIgnoreIA5Match
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
)

```

```

ATTRIBUTE ( 1.3.6.1.4.1.42.2.27.5.1.64
NAME sun-printer-kvp
DESC 'This attribute contains a set of key value pairs which may have
meaning to the print subsystem or may be user defined. Each
value is represented by the following: key "=" value.'
EQUALITY caseIgnoreIA5Match
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )

```

Sun 打印机 ObjectClasses

```

OBJECTCLASS ( 1.3.6.1.4.1.42.2.27.5.2.14
NAME sunPrinter
DESC 'Sun printer information'
SUP top
AUXILIARY
MUST ( printer-name )
MAY ( sun-printer-bsdaddr $ sun-printer-kvp ))

```

LDAP 的常规目录服务器要求

为了支持基于 Solaris 9 或更高 Solaris 版本的 LDAP 客户机，所使用的服务器无论是什么品牌，都必须支持 LDAP 3.0 版协议以及复合命名和辅助对象类。另外，还必须至少支持下列控制之一。

- 简单换页模式 (RFC 2696)
- 虚拟列表视图控制

服务器必须至少支持下列验证方法之一。

```
anonymous
simple
sasl/cram-MD5
sasl/digest-MD5
sasl/GSSAPI
```

如果使用 `pam_unix`，服务器必须支持以 UNIX crypt 格式存储口令。

如果使用 TLS，服务器必须支持 SSL 或 TLS。

如果使用 `sasl/GSSAPI`，服务器必须支持 SASL、GSSAPI 和 Kerberos 5 验证。对 GSS 线上加密的支持是可选的。

LDAP 命名服务使用的缺省过滤器

如果没有使用 SSD 为给定的服务手动指定参数，将使用缺省过滤器。要列出给定服务的缺省过滤器，请使用带 `-v` 选项的 `ldaplist`。

在以下示例中，`filter=(&(objectclass=iphost)(cn=abcde))` 定义了缺省过滤器。

```
database=hosts
filter=(&(objectclass=iphost)(cn=abcde))
user data=(&(%s) (cn=abcde))
```

`ldaplist` 生成以下缺省过滤器列表，其中 `%s` 表示字符串，`%d` 表示数字。

```
hosts
(&(objectclass=iphost)(cn=%s))
-----
passwd
(&(objectclass=posixaccount)(uid=%s))
-----
services
(&(objectclass=ipservice)(cn=%s))
-----
group
```

```

(&(objectclass=posixgroup)(cn=%s))
-----
netgroup
(&(objectclass=nisnetgroup)(cn=%s))
-----
networks
(&(objectclass=ipnetwork)(ipnetworknumber=%s))
-----
netmasks
(&(objectclass=ipnetwork)(ipnetworknumber=%s))
-----
rpc
(&(objectclass=oncrpc)(cn=%s))
-----
protocols
(&(objectclass=ipprotocol)(cn=%s))
-----
bootparams
(&(objectclass=bootableDevice)(cn=%s))
-----
ethers
(&(objectclass=ieee802Device)(cn=%s))
-----
publickey
(&(objectclass=niskeyobject)(cn=%s))
or
(&(objectclass=niskeyobject)(uidnumber=%d))
-----
aliases
(&(objectclass=mailGroup)(cn=%s))
-----

```

表 14-4 getXbyY 调用中使用的 LDAP 过滤器

过滤器	定义
bootparamByName	(&(objectClass=bootableDevice)(cn=%s))
etherByHost	(&(objectClass=ieee802Device)(cn=%s))
etherByEther	(&(objectClass=ieee802Device)(macAddress=%s))
groupByName	(&(objectClass=posixGroup)(cn=%s))
groupByGID	(&(objectClass=posixGroup)(gidNumber=%ld))
groupByMember	(&(objectClass=posixGroup)(memberUid=%s))
hostsByName	(&(objectClass=ipHost)(cn=%s))
hostsByAddr	(&(objectClass=ipHost)(ipHostNumber=%s))
keyByUID	(&(objectClass=nisKeyObject)(uidNumber=%s))
keyByHost	(&(objectClass=nisKeyObject)(cn=%s))
netByName	(&(objectClass=ipNetwork)(cn=%s))

表 14-4 getXbyY 调用中使用的 LDAP 过滤器 (续)

过滤器	定义
netByAddr	(&(objectClass=ipNetwork)(ipNetworkNumber=%s))
nisgroupMember	(membernisnetgroup=%s)
maskByNet	(&(objectClass=ipNetwork)(ipNetworkNumber=%s))
printerByName	(&(objectClass=sunPrinter)((printer-name=%s)(printer-aliases=%s)))
projectByName	(&(objectClass=SolarisProject)(SolarisProjectName=%s))
projectByID	(&(objectClass=SolarisProject)(SolarisProjectID=%ld))
protoByName	(&(objectClass=ipProtocol)(cn=%s))
protoByNumber	(&(objectClass=ipProtocol)(ipProtocolNumber=%d))
passwordByName	(&(objectClass=posixAccount)(uid=%s))
passwordByNumber	(&(objectClass=posixAccount)(uidNumber=%ld))
rpcByName	(&(objectClass=oncrpc)(cn=%s))
rpcByNumber	(&(objectClass=oncrpc)(oncrpcNumber=%d))
serverByName	(&(objectClass=ipService)(cn=%s))
serverByPort	(&(objectClass=ipService)(ipServicePort=%ld))
serverByNameAndProto	(&(objectClass=ipService)(cn=%s)(ipServiceProtocol=%s))
specialByNameserver	(ipServiceProtocol=%s)
ByPortAndProto	(&(objectClass=shadowAccount)(uid=%s))
netgroupByTriple	(&(objectClass=nisNetGroup)(cn=%s))
netgroupByMember	(&(objectClass=nisNetGroup)(cn=%s))
authName	(&(objectClass=SolarisAuthAttr)(cn=%s))
auditUserByName	(&(objectClass=SolarisAuditUser)(uid=%s))
execByName	(&(objectClass=SolarisExecAttr)(cn=%s) (SolarisKernelSecurityPolicy=%s)(SolarisProfileType=%s))
execByPolicy	(&(objectClass=SolarisExecAttr)(SolarisProfileId=%s) (SolarisKernelSecurityPolicy=%s)(SolarisProfileType=%s))
profileByName	(&(objectClass=SolarisProfAttr)(cn=%s))
userByName	(&(objectClass=SolarisUserAttr)(uid=%s))

下表列出了 `getent` 属性过滤器。

表 14-5 `getent` 属性过滤器

过滤器	定义
<code>aliases</code>	<code>(objectClass=rfc822MailGroup)</code>
<code>auth_attr</code>	<code>(objectClass=SolarisAuthAttr)</code>
<code>audit_user</code>	<code>(objectClass=SolarisAuditUser)</code>
<code>exec_attr</code>	<code>(objectClass=SolarisExecAttr)</code>
<code>group</code>	<code>(objectClass=posixGroup)</code>
<code>hosts</code>	<code>(objectClass=ipHost)</code>
<code>networks</code>	<code>(objectClass=ipNetwork)</code>
<code>prof_attr</code>	<code>(objectClass=SolarisProfAttr)</code>
<code>protocols</code>	<code>(objectClass=ipProtocol)</code>
<code>passwd</code>	<code>(objectClass=posixAccount)</code>
<code>printers</code>	<code>(objectClass=sunPrinter)</code>
<code>rpc</code>	<code>(objectClass=oncRpc)</code>
<code>services</code>	<code>(objectClass=ipService)</code>
<code>shadow</code>	<code>(objectClass=shadowAccount)</code>
<code>project</code>	<code>(objectClass=SolarisProject)</code>
<code>usr_attr</code>	<code>(objectClass=SolarisUserAttr)</code>

从 NIS 转换为 LDAP（概述/任务）

本章介绍如何对那些使用 LDAP 目录中存储的命名信息的 NIS 客户机启用支持。按照本章中的过程操作，可以从使用 NIS 命名服务转换为使用 LDAP 命名服务。

要了解转换到 LDAP 的益处，请参见第 122 页中的“LDAP 命名服务与其他命名服务的比较”。

本章将介绍以下信息：

- 第 223 页中的“NIS 到 LDAP 转换服务概述”
- 第 228 页中的“从 NIS 转换为 LDAP（任务列表）”
- 第 228 页中的“NIS 到 LDAP 转换的先决条件”
- 第 229 页中的“设置 NIS 到 LDAP 转换服务”
- 第 235 页中的“使用 Sun Java System Directory Server 进行 NIS 到 LDAP 转换的最佳做法”
- 第 238 页中的“NIS 到 LDAP 转换的限制”
- 第 238 页中的“NIS 到 LDAP 转换的疑难解答”
- 第 242 页中的“恢复为 NIS”

NIS 到 LDAP 转换服务概述

NIS 到 LDAP 转换服务（**N2L 服务**）使用 NIS 到 LDAP 转换守护进程来替换 NIS 主服务器上现有的 NIS 守护进程。N2L 服务还在该服务器上创建一个 NIS 到 LDAP 转换的映射文件。该映射文件指定 NIS 映射项和 LDAP 中目录信息树 (Directory Information Tree, DIT) 等效项之间的映射。已经进行这种转换的 NIS 主服务器称为 **N2L 服务器**。从属服务器上没有 `NISLDAPmapping` 文件，因此它们继续以通常的方式工作。从属服务器定期从 N2L 服务器更新其数据，就好像 N2L 服务器是常规的 NIS 主服务器一样。

N2L 服务的行为由 `ypserv` 和 `NISLDAPmapping` 配置文件控制。借助脚本 `inittyp2l` 可以对这些配置文件进行初始设置。一旦建立了 N2L 服务器，您便可以通过直接编辑这些配置文件来维护 N2L。

N2L 服务支持以下功能：

- 将 NIS 映射导入到 LDAP 目录信息树 (Directory Information Tree, DIT)
- 客户机以 NIS 的速度和可扩展性访问 DIT 信息

在任何命名系统中，只有一个信息源可以是权威来源。在传统的 NIS 中，NIS 源是权威信息。在使用 N2L 服务时，权威数据来源是 LDAP 目录。如第 9 章，[LDAP 基本组件和概念（概述）](#)中所述，该目录使用目录管理工具进行管理。

NIS 源仅保留用于紧急备份或卸载。在使用 N2L 服务之后，您可以逐步淘汰 NIS 客户机。最终，所有的 NIS 客户机都会被 Solaris LDAP 命名服务客户机所取代。

以下各小节中提供了其他概述信息：

- [第 224 页中的“NIS 到 LDAP 转换的目标用户”](#)
- [第 224 页中的“不应使用 NIS 到 LDAP 转换服务的情况”](#)
- [第 225 页中的“NIS 到 LDAP 转换服务对用户造成的影响”](#)
- [第 225 页中的“NIS 到 LDAP 转换术语”](#)
- [第 226 页中的“NIS 到 LDAP 转换的命令、文件和映射”](#)
- [第 227 页中的“支持的标准映射”](#)

NIS 到 LDAP 转换工具和服务管理工具

NIS 和 LDAP 服务由服务管理工具管理。使用 `svcadm` 命令可以对这些服务执行启用、禁用或重新启动等管理操作。使用 `svcs` 命令可以查询服务的状态。有关使用 SMF 对 LDAP 和 NIS 进行管理的更多信息，请参见第 174 页中的[“LDAP 和服务管理工具”](#)和第 80 页中的[“NIS 和服务管理工具”](#)。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章[“管理服务（概述）”](#)。有关更多详细信息，另请参阅 [svcadm\(1M\)](#) 和 [svcs\(1\)](#) 手册页。

NIS 到 LDAP 转换的目标用户

您需要熟悉 NIS 和 LDAP 概念、术语以及 ID 才能执行本章中的过程。有关 NIS 和 LDAP 命名服务的更多信息，请参见本书中的以下两章：

- [第 4 章，网络信息服务 \(Network Information Service, NIS\)（概述）](#)（提供 NIS 的概述）
- [第 8 章，LDAP 命名服务介绍（概述/参考）](#)（提供 LDAP 的概述）

不应使用 NIS 到 LDAP 转换服务的情况

以下情况不要使用 N2L 转换服务：

- 不计划在 NIS 和 LDAP 命名服务客户机之间共享数据的情况下
在这种情况下，N2L 服务器将充当极其复杂的 NIS 主服务器。

- NIS 映射由修改 NIS 源文件的工具（而非 `yppasswd`）进行管理的情况下
从 DIT 映射重新生成 NIS 源是一项不精确的任务，需要手动检查生成的映射。一旦使用了 N2L 服务，所提供的 NIS 源重新生成功能将仅用于卸载 NIS 或恢复为 NIS。
- 没有 NIS 客户机的情况下
在这种情况下，可以使用 Solaris LDAP 命名服务客户机及其对应工具。

NIS 到 LDAP 转换服务对用户造成的影响

仅安装与 N2L 服务相关的文件不会更改 NIS 服务器的缺省行为。在安装时，管理员会看到服务器上的 NIS 手册页会有一些变化并且其中会增加 N2L 帮助脚本 `inityp2l` 和 `ypmap2src`。但是，只要未在 NIS 服务器上运行 `inityp2l` 或未手动创建 N2L 配置文件，NIS 组件便会继续在传统的 NIS 模式下启动，并像往常那样工作。

运行 `inityp2l` 之后，用户会看到服务器和客户机的行为会发生一些变化。以下列表列出了 NIS 和 LDAP 用户的类型，并说明了部署 N2L 服务之后每种类型的用户应当注意到的情况。

用户类型	N2L 服务的影响
NIS 主服务器管理员	NIS 主服务器转换为 N2L 服务器。 <code>NISLDAPmapping</code> 和 <code>ypserv</code> 配置文件将会安装在 N2L 服务器上。建立 N2L 服务器之后，可以使用 LDAP 命令来管理命名信息。
NIS 从属服务器管理员	N2L 转换之后，NIS 从属服务器继续以通常的方式运行 NIS。当 <code>ypmake</code> 调用 <code>yppush</code> 时，N2L 服务器会将更新的 NIS 映射推送到从属服务器。请参见 ypmake(1M) 手册页。
NIS 客户机	NIS 读取操作与传统的 NIS 没有区别。当 Solaris LDAP 命名服务客户机更改 DIT 中的信息时，这些信息会复制到 NIS 映射中。复制操作是在可配置的超时时间过期之后完成的。这类行为与连接到 NIS 从属服务器的常规 NIS 客户机的行为相似。 如果 N2L 服务器无法绑定到 LDAP 服务器进行读取，它将从自身的缓存副本中返回信息。或者，N2L 服务器还可能会返回内部服务器错误。您可以将 N2L 服务器配置为以上述任一方式响应。有关更多详细信息，请参见 ypserv(1M) 手册页。
所有用户	当 NIS 客户机发出更改口令的请求时，所做的更改将立即显示在 N2L 主服务器上并对本地 LDAP 客户机可见。 如果您试图在 NIS 客户机上更改口令，但 LDAP 服务器不可用，更改将被拒绝，并且 N2L 服务器会返回内部服务器错误。此行为可防止将不正确的信息写入高速缓存中。

NIS 到 LDAP 转换术语

以下是与 N2L 服务的实现相关的术语。

表 15-1 与 N2L 转换相关的术语

术语	说明
N2L configuration file（N2L 配置文件）	<code>/var/yp/NISLDAPmapping</code> 和 <code>/var/yp/ypserv</code> 文件， <code>ypserv</code> 守护进程使用这些文件在 N2L 模式下启动主服务器。有关详细消息，请参见 <code>NISLDAPmapping(4)</code> 和 <code>ypserv(4)</code> 手册页。
map（映射）	在 N2L 服务的上下文中，术语“映射”的用法有两种： ■ 指 NIS 用于存储特定类型信息的数据库文件 ■ 描述从 LDAP DIT 映射 NIS 信息或将 NIS 信息映射到 LDAP DIT 的过程
mapping（映射过程）	NIS 项与 LDAP DIT 项之间的相互转换过程。
mapping file（映射文件）	用来指定如何在 NIS 文件和 LDAP 文件之间映射各项的 <code>NISLDAPmapping</code> 文件。
standard map（标准映射）	无需手动修改映射文件即可由 N2L 服务支持的常用 NIS 映射。第 227 页中的“支持的标准映射”中提供了支持的标准映射的列表。
nonstandard map（非标准映射）	经过定制的标准 NIS 映射，这些非标准映射使用 NIS 和 LDAP DIT 之间的映射，而不是 RFC 2307 或其后续版本中标识的映射。
custom map（定制映射）	任何不是标准映射并在从 NIS 转换至 LDAP 时需要手动修改映射文件的映射。
LDAP client（LDAP 客户机）	任何对 LDAP 服务器执行读写操作的传统 LDAP 客户机。传统的 LDAP 客户机是可以对任何 LDAP 服务器执行读写操作的系统。Solaris LDAP 命名服务客户机可处理部分定制的命名信息。
LDAP naming services client（LDAP 命名服务客户机）	用来处理部分定制命名信息的 Solaris LDAP 客户机。
N2L server（N2L 服务器）	已使用 N2L 服务重新配置为 N2L 服务器的 NIS 主服务器。重新配置过程包括替换 NIS 守护进程和添加新配置文件。

NIS 到 LDAP 转换的命令、文件和映射

与 N2L 转换相关联的共有两个实用程序、两个配置文件和一个映射。

表 15-2 N2L 命令、文件和映射的说明

命令/文件/映射	说明
<code>/usr/lib/netsvc/yp/inityp2l</code>	实用程序，可帮助创建 <code>NISLDAPmapping</code> 和 <code>ypserv</code> 配置文件。此实用程序不是用来管理这些文件的通用工具。高级用户可通过使用文本编辑器检查和定制 <code>inityp2l</code> 输出来维护 N2L 配置文件或创建定制映射。请参见 <code>inityp2l(1M)</code> 手册页。
<code>/usr/lib/netsvc/yp/ypmap2src</code>	实用程序，用于将标准 NIS 映射转换为近似等效的 NIS 源文件。 <code>ypmap2src</code> 主要用于将 N2L 转换服务器转换为传统的 NIS。请参见 <code>ypmap2src(1M)</code> 手册页。

表 15-2 N2L 命令、文件和映射的说明 (续)

命令/文件/映射	说明
/var/yp/NISLDAPmapping	配置文件，用于指定 NIS 映射项和 LDAP 中目录信息树 (Directory Information Tree, DIT) 等效项之间的映射。请参见 NISLDAPmapping(4) 手册页。
/var/yp/ypserv	此文件用于为 NIS 到 LDAP 转换守护进程指定配置信息。请参见 ypserv(4) 手册页。
ageing.byname	映射，yppasswdd 在实现 NIS 到 LDAP 转换时使用此映射在 DIT 中读写口令生命期信息。

支持的标准映射

缺省情况下，N2L 服务支持下表所列的映射与 RFC 2307 或其后续版本中 LDAP 各项之间的映射。这些标准映射不需要手动修改映射文件。系统上任何未在下表中列出的映射都被视为定制映射，且需要手动修改。

N2L 服务还支持对 `auto.*` 映射进行自动映射。但是，由于大多数 `auto.*` 文件名和内容都特定于每种网络配置，因此该列表并未指定这些文件。但作为标准映射支持的 `auto.home` 和 `auto.master` 映射除外。

```
audit_user
auth_attr
auto.home
auto.master
bootparams
ethers.byaddr ethers.byname
exec_attr
group.bygid group.byname group.adjunct.byname
hosts.byaddr hosts.byname
ipnodes.byaddr ipnodes.byname
mail.byaddr mail.aliases
netgroup netgroup.byprojid netgroup.byuser netgroup.byhost
netid.byname
netmasks.byaddr
networks.byaddr networks.byname
passwd.byname passwd.byuid passwd.adjunct.byname
printers.conf.byname
prof_attr
project.byname project.byprojectid
protocols.byname protocols.bynumber
publickey.byname
rpc.bynumber
services.byname services.byservicename
timezone.byname
user_attr
```

在 NIS 到 LDAP 转换过程中，yppasswdd 守护进程使用 N2L 特定的映射 `ageing.byname` 在 DIT 中读写口令生命期信息。如果没有使用口令生命期，则会忽略 `ageing.byname` 映射。

从 NIS 转换为 LDAP（任务列表）

下表列出了使用标准的和定制的 NIS 到 LDAP 转换映射来安装和管理 N2L 服务所需的过程。

任务	说明	参考
完成所有先决条件的准备。	确保已经正确配置了 NIS 服务器和 Sun Java System Directory Server（LDAP 服务器）。	第 228 页中的“NIS 到 LDAP 转换的先决条件”
设置 N2L 服务。	在 NIS 主服务器上运行 <code>inityp2l</code> 以设置以下映射之一：	
	标准映射	第 230 页中的“如何使用标准映射设置 N2L 服务”
	定制映射或非标准映射	第 232 页中的“如何使用定制映射或非标准映射设置 N2L 服务”
定制映射。	查看如何为 N2L 转换创建定制映射的示例。	第 234 页中的“定制映射的示例”
使用 N2L 配置 Sun Java System Directory Server。	将 Sun Java System Directory Server 配置并调整为用于 N2L 转换的 LDAP 服务器。	第 235 页中的“使用 Sun Java System Directory Server 进行 NIS 到 LDAP 转换的最佳做法”
排除系统故障。	确定和解决常见的 N2L 问题。	第 238 页中的“NIS 到 LDAP 转换的疑难解答”
恢复为 NIS。	使用相应的映射恢复为 NIS：	
	基于旧 NIS 源文件的映射	第 242 页中的“如何基于旧的源文件恢复到 NIS 映射”
	基于当前 DIT 的映射	第 243 页中的“如何基于当前的 DIT 内容恢复为 NIS 映射”

NIS 到 LDAP 转换的先决条件

在实现 N2L 服务之前，必须检查或完成以下各项操作。

- 运行 `inityp2l` 脚本以启用 N2L 模式之前，确保将系统设置为可正常工作的传统 NIS 服务器。
- 在系统上配置 LDAP 目录服务器。

NIS 到 LDAP 转换迁移工具支持 Oracle Corporation 提供的 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）和兼容版本的目录服务器。如果使用 Sun Java System Directory Server，请在设置 N2L 服务之前，使用 `idsconfig` 命令来配

置服务器。有关 `idsconfig` 的更多信息，请参见第 11 章，为使用 LDAP 客户机设置 [Sun Java System Directory Server（任务）](#) 和 [idsconfig\(1M\)](#) 手册页。

其他（第三方）LDAP 服务器可能会使用 N2L 服务，但是 Oracle 不支持这些服务器。如果使用的是 Sun Java System Directory Server 或兼容 Oracle 服务器以外的 LDAP 服务器，在设置 N2L 服务之前，必须手动配置服务器，使其支持 RFC 2307 或其后续版本中的架构。

- 确保 `nsswitch.conf` 文件至少针对 `hosts` 和 `ipnodes` 项的查找顺序将 `files` 列在 `nis` 之前。
- 确保 N2L 主服务器上的 `hosts` 或 `ipnodes` 文件中提供了 N2L 主服务器和 LDAP 服务器的地址。服务器地址是必须列在 `hosts` 中、`ipnodes` 中还是同时列在两个文件中，取决于系统配置为以何种方式解析本地主机名。

另一种解决方案是在 `ypserv` 中列出 LDAP 服务器地址，而不列出其主机名。这意味着 LDAP 服务器地址列在另一个位置，因此，在更改 LDAP 服务器或 N2L 主服务器的地址时，需要对文件进行额外的修改。

设置 NIS 到 LDAP 转换服务

您可以按照以下两个过程中的说明，使用标准映射或定制映射设置 N2L 服务。

在 NIS 到 LDAP 转换过程中，您需要运行 `inityp2l` 命令。该命令会运行一个交互式脚本，而您必须为该脚本提供配置信息。以下列表列出了需要提供的信息类型。有关这些属性的说明，请参见 [ypserv\(1M\)](#) 手册页。

- 创建的配置文件名称（缺省为 `/etc/default/ypserv`）
- 用来将配置信息存储到 LDAP 中的 DN（缺省为 `ypserv`）
- 用来将数据映射到 LDAP 或从 LDAP 映射数据的首选服务器的列表
- 用来将数据映射到 LDAP 或从 LDAP 映射数据的验证方法
- 用来将数据映射到 LDAP 或从 LDAP 映射数据的传输层安全性 (Transport Layer Security, TLS) 方法
- 用来在 LDAP 中读写数据的代理用户绑定 DN
- 用来在 LDAP 中读写数据的代理用户口令
- LDAP 绑定操作的超时值（以秒为单位）
- LDAP 搜索操作的超时值（以秒为单位）
- LDAP 修改操作的超时值（以秒为单位）
- LDAP 添加操作的超时值（以秒为单位）
- LDAP 删除操作的超时值（以秒为单位）
- LDAP 服务器上搜索操作的时间限制（以秒为单位）
- LDAP 服务器上搜索操作的大小限制（以字节为单位）

- N2L 是否应当遵循 LDAP 引用
- 导致 LDAP 检索错误的操作、尝试检索的次数以及各尝试操作的超时值（以秒为单位）
- 导致存储错误的操作、尝试的次数以及各尝试操作的超时值（以秒为单位）
- 映射文件的名称
- 是否为 `auto_direct` 映射生成映射信息
脚本将与定制映射相关的信息放入映射文件中的相应位置。
- 命名上下文
- 是否启用口令更改功能
- 是否更改所有映射的缺省 TTL 值

注 - 大多数 LDAP 服务器（包括 Sun Java System Directory Server）都不支持 `sasl/cram-md5` 验证。

▼ 如何使用标准映射设置 N2L 服务

如果要转换第 227 页中的“支持的标准映射”中所列的映射，请使用此过程。如果要使用定制映射或非标准映射，请参见第 232 页中的“如何使用定制映射或非标准映射设置 N2L 服务”。

设置 LDAP 服务器之后，请运行 `inityp2l` 脚本并在出现提示时提供配置信息。`inityp2l` 为标准映射和 `auto.*` 映射设置配置和映射文件。

1 完成第 228 页中的“NIS 到 LDAP 转换的先决条件”中所列的先决步骤。

2 在 NIS 主服务器上，成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

3 将 NIS 主服务器转换为 N2L 服务器。

`inityp2l`

在 NIS 主服务器上运行 `inityp2l` 脚本并按照提示操作。有关需要提供的信息的列表，请参见第 229 页中的“设置 NIS 到 LDAP 转换服务”。

有关更多详细信息，请参见 `inityp2l(1M)` 手册页。

4 确定 LDAP 目录信息树 (Directory Information Tree, DIT) 是否已完全初始化。

如果 DIT 中已包含置备 `NISLDAPmapping` 文件中所列全部映射所需要的信息，则表明它已完全初始化。

- 如果不包含，请继续执行[步骤 5](#)并跳过步骤 6。
- 如果已包含，请跳过步骤 5 并转至[步骤 6](#)。

5 初始化 DIT 以便从 NIS 源文件进行转换。

仅当 DIT 尚未完全初始化时，才可执行这些步骤。

a. 确保旧 NIS 映射是最新的版本。

```
# cd /var/yp
# make
```

有关更多信息，请参见 [ypmake\(1M\)](#) 手册页。

b. 停止 NIS 守护进程。

```
# svcadm disable network/nis/server:default
```

c. 将旧映射复制到 DIT 中，然后为这些映射初始化 N2L 支持。

```
# ypserv -Ir
```

等待 ypserv 退出。

提示 – 原始的 NIS dbm 文件不会被覆写。您可以根据需要恢复这些文件。

d. 启动 NIS 守护进程，确保其使用新映射。

```
# svcadm enable network/nis/server:default
```

这样便完成了使用标准映射设置 N2L 服务的过程。您无需完成步骤 6。

6 初始化 NIS 映射。

仅当 DIT 已完全初始化并且跳过了步骤 5 时，才执行这些步骤。

a. 停止 NIS 守护进程。

```
# svcadm disable network/nis/server:default
```

b. 使用 DIT 中的信息初始化 NIS 映射。

```
# ypserv -r
```

等待 ypserv 退出。

提示 – 原始的 NIS dbm 文件不会被覆写。您可以根据需要恢复这些文件。

c. 启动 NIS 守护进程，确保其使用新映射。

```
# svcadm enable network/nis/server:default
```

▼ 如何使用定制映射或非标准映射设置 N2L 服务

如果符合以下情况，请使用此过程：

- 具有第 227 页中的“支持的标准映射”中未列出的映射。
- 具有要映射到非 RFC 2307 LDAP 映射的标准 NIS 映射。

1 完成第 228 页中的“NIS 到 LDAP 转换的先决条件”中所列的先决步骤。

2 在 NIS 主服务器上，成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

3 将 NIS 主服务器配置为 N2L 服务器。

```
# inityp2l
```

在 NIS 主服务器上运行 `inityp2l` 脚本并按照提示操作。有关需要提供的信息的列表，请参见第 229 页中的“设置 NIS 到 LDAP 转换服务”。

有关更多详细信息，请参见 `inityp2l(1M)` 手册页。

4 修改 `/var/yp/NISLDAPmapping` 文件。

有关如何修改映射文件的示例，请参见第 234 页中的“定制映射的示例”。

5 确定 LDAP 目录信息树 (Directory Information Tree, DIT) 是否已完全初始化。

如果 DIT 中已包含置备 `NISLDAPmapping` 文件中所列全部映射所需要的信息，则表明它已完全初始化。

- 如果未包含，请完成步骤 6、步骤 8 和步骤 9。
- 如果已包含，请跳过步骤 6 并完成步骤 7、步骤 8 和步骤 9。

6 初始化 DIT 以便从 NIS 源文件进行转换。

a. 确保旧 NIS 映射是最新的版本。

```
# cd /var/yp
# make
```

有关更多信息，请参见 `ypmake(1M)` 手册页。

b. 停止 NIS 守护进程。

```
# svcadm disable network/nis/server:default
```

c. 将旧映射复制到 DIT 中，然后为这些映射初始化 N2L 支持。

```
# ypserv -Ir
```

等待 `ypserv` 退出。

提示 – 原始的 NIS dbm 文件不会被覆写。您可以根据需要恢复这些文件。

- d. 启动 NIS 守护进程，确保其使用新映射。

```
# svcadm enable network/nis/server:default
```

- e. 跳过步骤 7 并继续执行[步骤 8](#)。

7 初始化 NIS 映射。

仅当 DIT 已完全初始化时，才可以执行此步骤。

- a. 停止 NIS 守护进程。

```
# svcadm disable network/nis/server:default
```

- b. 使用 DIT 中的信息初始化 NIS 映射。

```
# ypserv -r
```

等待 ypserv 退出。

提示 – 原始的 NIS dbm 文件不会被覆写。您可以根据需要恢复这些文件。

- c. 启动 NIS 守护进程，确保其使用新映射。

```
# svcadm enable network/nis/server:default
```

8 检验 LDAP 项是否正确。

如果这些项不正确，LDAP 命名服务器客户机将无法找到这些项。

```
# ldapsearch -h server -s sub -b "ou=servdates, dc=..." \
"objectclass=servDates"
```

9 验证 LDAP_映射的内容。

以下样例输出说明如何使用 makedm 验证 hosts.byaddr 映射的内容。

```
# makedbm -u LDAP_servdate.bynumber
plato: 1/3/2001
johnson: 2/4/2003,1/3/2001
yeats: 4/4/2002
poe: 3/3/2002,3/4/2000
```

如果内容与预期一致，则表明已成功地从 NIS 转换到 LDAP。

请注意，原始的 NIS dbm 文件不会被覆写，因此您始终可以恢复这些文件。有关更多信息，请参见[第 242 页](#)中的“恢复为 NIS”。

定制映射的示例

以下两个示例说明如何定制映射。请使用首选的文本编辑器，根据需要修改 `/var/yp/NISLDAPmapping` 文件。有关文件属性和语法的更多信息，请参见 [NISLDAPmapping\(4\)](#) 手册页以及第 9 章，[LDAP 基本组件和概念（概述）](#) 中的 LDAP 命名服务信息。

示例 1—移动主机项

本示例说明如何将主机项从缺省位置移到 DIT 中的另一个（非标准）位置。

请将 `NISLDAPmapping` 文件中的 `nisLDAPobjectDN` 属性更改为新的 LDAP 基本标识名 (distinguished name, DN)。在本示例中，LDAP 对象的内部结构未更改，因此 `objectClass` 项也不会更改。

将如下内容：

```
nisLDAPobjectDN hosts: \
    ou=hosts,?one?, \
    objectClass=device, \
    objectClass=ipHost
```

更改为：

```
nisLDAPobjectDN hosts: \
    ou=newHosts,?one?, \
    objectClass=device, \
    objectClass=ipHost
```

此更改会导致按如下方式映射这些项：

```
dn: ou=newHosts, dom=domain1, dc=sun, dc=com
```

而不是按如下方式映射：

```
dn: ou=hosts, dom=domain1, dc=sun, dc=com.
```

示例 2—实现定制映射

本示例说明如何实现定制映射。

虚拟映射 `servdate.bynumber` 中包含有关为系统提供服务的日期的信息。此映射根据计算机的序列号（在本示例中为 123）建立索引。每一项都由计算机所有者的姓名、一个冒号和一个用逗号分隔的服务日期列表组成，如 `John Smith:1/3/2001,4/5/2003`。

旧映射的结构将映射到以下形式的 LDAP 项上：

```
dn: number=123,ou=servdates,dc=... \
    number: 123 \
    userName: John Smith \
```

```

date: 1/3/2001 \
date: 4/5/2003 \
.
.
.
objectClass: servDates

```

通过检查 NISLDAPmapping 文件，可以看到与所需模式最接近的映射是 group。可以根据 group 映射建立定制映射的模型。由于仅有一个映射，因此不需要 nisLDAPdatabaseIdMapping 属性。以下是要添加到 NISLDAPmapping 中的属性：

```

nisLDAPentryTtl servdate.bynumber:1800:5400:3600

nisLDAPnameFields servdate.bynumber: \
    ("%s:%s", uname, dates)

nisLDAPobjectDN servdate.bynumber: \
    ou=servdates, ?one? \
    objectClass=servDates:

nisLDAPattributeFromField servdate.bynumber: \
    dn=("number=%s", rf_key), \
    number=rf_key, \
    userName=uname, \
    (date)=(dates, ",")

nisLDAPfieldFromAttribute servdate.bynumber: \
    rf_key=number, \
    uname=userName, \
    dates("%s", (date), ",")

```

使用 Sun Java System Directory Server 进行 NIS 到 LDAP 转换的最佳做法

N2L 服务支持 Oracle Corporation 提供的 Sun Java System Directory Server（以前称为 Sun ONE Directory Server）和兼容版本的目录服务器。其他（第三方）LDAP 服务器可能会使用 N2L 服务，但是 Oracle 不支持这些服务器。如果使用的是 Sun Java System Directory Server 或兼容 Oracle 服务器以外的 LDAP 服务器，您必须手动配置服务器，使其支持 RFC 2307 或其后续版本中的架构。

如果使用的是 Sun Java System Directory Server，则可以增强目录服务器以提高性能。必须对 Sun Java System Directory Server 具有 LDAP 管理员权限才能增强目录服务器。另外，目录服务器可能需要重新引导，此任务必须与服务器的 LDAP 客户机协调进行。Sun Java System Directory Server（以及 Sun ONE 和 iPlanet Directory Server）文档可从 [Sun Java System Directory Server Enterprise Edition 6.2 \(http://docs.sun.com/coll/1224.3\)](http://docs.sun.com/coll/1224.3) 网站获得。

使用 Sun Java System Directory Server 创建虚拟列表视图索引

对于大型映射，必须使用 LDAP 虚拟列表视图 (virtual list view, VLV) 索引来确保 LDAP 搜索可返回全部结果。要了解关于在 Sun Java System Directory Server 上设置 VLV 索引的信息，请参见 [Sun Java System Directory Server Enterprise Edition 6.2 \(http://docs.sun.com/coll/1224.3\)](http://docs.sun.com/coll/1224.3) 文档。

VLV 搜索结果使用固定的页面大小 50000。如果在 Sun Java System Directory Server 上使用 VLV，则 LDAP 服务器和 N2L 服务器都必须可以传送此大小的页面。如果已知所有的映射都小于此限制，则不必使用 VLV 索引。但是，如果使用的映射大于此大小限制，或者不能确定所有映射的大小，请使用 VLV 索引，以避免返回的结果不完整。

如果您使用 VLV 索引，请按如下方式设置适当的大小限制。

- 在 Sun Java System Directory Server 上：必须将 `nsslapd-sizelimit` 属性设置为大于等于 50000 或 -1。请参见 [idsconfig\(1M\)](#) 手册页。
- 在 N2L 服务器上：必须将 `nisLDAPsearchSizelimit` 属性设置为大于等于 50000 或零。有关更多信息，请参见 [NISLDAPmapping\(4\)](#) 手册页。

创建 VLV 索引之后，请立即将其激活，方法是在 Sun Java System Directory Server 上运行带有 `vlvindex` 选项的 `directoryserver`。有关更多信息，请参见 [directoryserver\(1M\)](#) 手册页。

标准映射的 VLV

如果符合以下条件，可以使用 Sun Java System Directory Server 的 `idsconfig` 命令设置 VLV：

- 正在使用 Sun Java System Directory Server。
- 要将标准映射映射到 RFC 2307 LDAP 项。

VLV 特定于域，因此每次运行 `idsconfig` 时，都会为一个 NIS 域创建相应的 VLV。所以，在 NIS 到 LDAP 的转换过程中，必须对 `NISLDAPmapping` 文件中包含的每个 `nisLDAPdomainContext` 属性都运行一次 `idsconfig`。

定制映射和非标准映射的 VLV

如果符合以下条件，则必须为映射手动创建新的 Sun Java System Directory Server VLV，或者复制并修改现有的 VLV 索引：

- 正在使用 Sun Java System Directory Server。
- 具有大型定制映射，或者具有映射到非标准 DIT 位置的标准映射。

要查看现有的 VLV 索引，请键入以下命令：

```
# ldapsearch -h hostname -s sub -b "cn=ldbm database,cn=plugins,cn=config" \
"objectClass=vlvSearch"
```

避免 Sun Java System Directory Server 服务器超时

N2L 服务器在刷新映射时，可能会对 LDAP 目录进行大量访问。如果 Sun Java System Directory Server 的配置不正确，刷新操作可能会因超时而无法完成。要避免目录服务器超时，请手动或者通过运行 `idsconfig` 命令修改以下 Sun Java System Directory Server 属性。

例如，要增加服务器执行搜索请求所需的最短时间（以秒为单位），请修改以下属性：

```
dn: cn=config
nsslapd-timelimit: -1
```

出于测试的目的，您可以使用属性值 `-1`，该值表示没有限制。确定最佳限制值之后，请更改属性值。**请勿**在生产服务器上保留任何值为 `-1` 的属性设置。在没有限制的情况下，服务器可能容易受到拒绝服务攻击。

有关为使用 LDAP 配置 Sun Java System Directory Server 的详细信息，请参见本书的[第 11 章，为使用 LDAP 客户机设置 Sun Java System Directory Server（任务）](#)。

避免 Sun Java System Directory Server 缓冲区溢出

要避免缓冲区溢出，请手动或者通过运行 `idsconfig` 命令修改 Sun Java System Directory Server 属性。

1. 例如，要增加针对客户机搜索查询返回的最大项数，请修改以下属性：

```
dn: cn=config
nsslapd-sizelimit: -1
```

2. 要增加针对客户机搜索查询检验的最大项数，请修改以下属性：

```
dn: cn=config, cn=ldbm database, cn=plugins, cn=config
nsslapd-lookthroughlimit: -1
```

出于测试的目的，您可以使用属性值 `-1`，该值表示没有限制。确定最佳限制值之后，请更改属性值。**请勿**在生产服务器上保留任何值为 `-1` 的属性设置。在没有限制的情况下，服务器可能容易受到拒绝服务攻击。

如果使用 VLV，则应当按照[第 236 页](#)中的“使用 Sun Java System Directory Server 创建虚拟列表视图索引”中的定义设置 `sizelimit` 属性值。如果未使用 VLV，则应当将大小限制设置得足够大，以便可以容纳最大的容器。

有关为使用 LDAP 配置 Sun Java System Directory Server 的详细信息，请参见[第 11 章，为使用 LDAP 客户机设置 Sun Java System Directory Server（任务）](#)。

NIS 到 LDAP 转换的限制

设置 N2L 服务器之后，将不再使用 NIS 源文件。因此，请勿在 N2L 服务器上运行 `ypmake`。如果无意间（例如对于现有的 `cron` 作业）运行了 `ypmake`，N2L 服务不会受到影响。但是，会记录一个警告，提示应当明确调用 `yppush`。

NIS 到 LDAP 转换的疑难解答

本节包括两个方面的疑难解答：

- [第 238 页中的“常见的 LDAP 错误消息”](#)
- [第 239 页中的“NIS 到 LDAP 转换的问题”](#)

常见的 LDAP 错误消息

有时，N2L 服务器会记录与内部 LDAP 问题相关的错误，并生成与 LDAP 相关的错误消息。尽管这些错误不是致命的，但是它们指明有问题需要检查。例如，N2L 服务器可能会继续工作，但是会提供过时或不完整的结果。

以下列表包括一些在实现 N2L 服务时可能遇到的常见 LDAP 错误消息。也包括错误说明、造成这些错误可能的原因和解决方案。

Administrative limit exceeded（超过管理限制）

错误号：11

原因：执行的 LDAP 搜索大于目录服务器的 `nsslapd-sizelimit` 属性所允许的大小。将仅返回部分信息。

解决方案：增大 `nsslapd-sizelimit` 属性的值，或者对失败的搜索实施 VLV 索引。

Invalid DN Syntax（DN 语法无效）

错误号：34

原因：尝试写入的 LDAP 项的 DN 包含非法字符。N2L 服务器尝试对 DN 中生成的非法字符（如 + 号）转义。

解决方案：检查 LDAP 服务器错误日志，确定写入的非法 DN，然后修改生成了非法 DN 的 `NISLDAPmapping` 文件。

Object class violation（对象类违规）

错误号：65

原因：尝试写入的 LDAP 项无效。通常，出现此错误是由于缺少 **MUST** 属性，以下任一情况都可能会导致此错误。

- `NISLDAPmapping` 文件中存在导致所创建的项缺少属性的错误

- 尝试向不存在的对象添加 AUXILIARY 属性

例如，如果仍未从 passwd.byxxx 映射建立用户名，向该用户添加辅助信息的尝试也会失败。

解决方案：对于 NISLDAPmapping 文件中的错误，检查写入服务器错误日志的内容，以确定问题的性质。

Can't contact LDAP server（无法联系 LDAP 服务器）

错误号：81

原因：ypserv 文件可能错误地配置为指向错误的 LDAP 目录服务器。或者，目录服务器当前可能未运行。

解决方案：

- 重新配置 ypserv 文件，使其指向正确的 LDAP 目录服务器。
- 要确认 LDAP 服务器是否正在运行，请在目录服务器上成为超级用户或承担等效角色，然后键入以下命令：

```
# pgrep -l slapd
```

Timeout（超时）

错误号：85

原因：LDAP 操作超时，通常是在从 DIT 更新映射时发生此问题。该映射当前可能包含过时的信息。

解决方案：增大 ypserv 配置文件中的 nisLDAPxxxTimeout 属性。

NIS 到 LDAP 转换的问题

运行 N2L 服务器时可能会出现以下问题。此处提供了可能的原因和解决方案。

调试 NISLDAPmapping 文件

映射文件 NISLDAPmapping 非常复杂。很多潜在的错误可能会导致映射工作不正常。请使用以下技术解决此类问题。

运行 ypserv -ir（或 -Ir）时显示控制台消息

问题：控制台上会显示简单的消息，并且服务器会退出（详细说明将写入 syslog）。

原因：映射文件的语法可能不正确。

解决方案：检查并更正 NISLDAPmapping 文件中的语法。

NIS 守护进程在启动时退出

问题：运行 ypserv 或其他 NIS 守护进程时，会记录一条与 LDAP 相关的错误消息，并且守护进程会退出。

原因：这可能是下列某一原因造成的：

- 无法联系 LDAP 服务器。
- 在 NIS 映射或 DIT 中找到的项与指定的映射不兼容。
- 尝试对 LDAP 服务器执行读写操作时返回错误。

解决方案：检查 LDAP 服务器上的错误日志。请参见第 238 页中的“常见的 LDAP 错误消息”。

NIS 操作产生意外的结果

问题：NIS 操作没有返回期望结果，但是也没有记录错误。

原因：LDAP 或 NIS 映射中可能存在不正确的项，这会导致映射无法按照预期的方式完成。

解决方案：检查并更正 LDAP DIT 以及 N2L 版本的 NIS 映射中的各项。

1. 检查 LDAP DIT 中的项是否正确，并根据需要更正这些项。

如果使用的是 Sun Java System Directory Server，请运行 `directoryserver startconsole` 启动管理控制台。

2. 检查 `/var/yp` 目录中 N2L 版本的 NIS 映射是否包含预期的项，方法是将新生成的映射与原来的映射进行比较。请根据需要更正这些项。

```
# cd /var/yp/domainname
# makedbm -u test.byname
# makedbm -u LDAP_test.byname
```

检查映射的输出时请注意以下情况：

- 在这两个文件中，各项的顺序可能不同。
在对输出进行比较之前，请使用 `sort` 命令。
- 在这两个文件中，空格的用法可能不同。
比较输出时，请使用 `diff -b` 命令。

NIS 映射的处理顺序

问题：出现对象类违规。

原因：运行 `ypserv -i` 命令时，会读取每个 NIS 映射并将其内容写入 DIT 中。同一个 DIT 对象的属性可以由多个映射创建。通常，通过一个映射来创建该对象的大部分属性，包括该对象的所有 **MUST** 属性。其他映射则负责创建其他 **MAY** 属性。

映射是按照 `nisLDAPObjectDN` 属性在 `NISLDAPmapping` 文件中的出现顺序来处理的。如果包含 **MAY** 属性的映射在包含 **MUST** 属性的映射之前处理，会发生对象类违规。要了解关于该错误的更多信息，请参见第 238 页中的“常见的 LDAP 错误消息”中的错误 65。

解决方案：将 `nisLDAPObjectDN` 属性重新排序，以便按照正确的顺序处理这些映射。

临时解决方法是多次重新运行 `ypserv -i` 命令。每次执行该命令，都会增加更多的 LDAP 项。

注 – 如果映射方式会导致不能从至少一个映射创建某个对象的所有 **MUST** 属性，则不支持以这种方式进行映射。

N2L 服务器超时问题

问题：服务器超时。

原因：N2L 服务器在刷新映射时，可能会对 LDAP 目录进行大量访问。如果 Sun Java System Directory Server 的配置不正确，该操作可能会因超时而无法完成。

解决方案：要避免目录服务器超时，请手动或者通过运行 `idsconfig` 命令修改 Sun Java System Directory Server 属性。有关详细消息，请参见第 238 页中的“常见的 LDAP 错误消息”和第 235 页中的“使用 Sun Java System Directory Server 进行 NIS 到 LDAP 转换的最佳做法”。

N2L 锁定文件问题

问题：`ypserv` 命令可以启动，但是不响应 NIS 请求。

原因：N2L 服务器锁定文件不能正确地同步对 NIS 映射的访问。这种情况绝对不应发生。

解决方案：在 N2L 服务器上键入以下命令。

```
# svcadm disable network/nis/server:default
# rm /var/run/yp_maplock /var/run/yp_mapupdate
# svcadm enable network/nis/server:default
```

N2L 死锁问题

问题：N2L 服务器死锁。

原因：如果 `hosts`、`ipnodes` 或 `ypserv` 文件中未正确列出 N2L 主服务器和 LDAP 服务器的地址，则可能会出现死锁问题。请参见第 228 页中的“NIS 到 LDAP 转换的先决条件”，了解关于 N2L 正确地址配置的详细信息。

有关死锁情况的示例，请考虑以下一系列事件：

1. 一台 NIS 客户机试图查找一个 IP 地址。
2. N2L 服务器发现 `hosts` 项已过时。
3. N2L 服务器尝试从 LDAP 更新 `hosts` 项。
4. N2L 服务器从 `ypserv` 获取其 LDAP 服务器的名称，然后使用 `libldap` 进行搜索。
5. `libldap` 尝试通过调用名称服务转换器，将 LDAP 服务器名称转换为 IP 地址。
6. 名称服务转换器可能会对 N2L 服务器进行 NIS 调用，而服务器死锁。

解决方案：在 N2L 主服务器上的 `hosts` 或 `ipnodes` 文件中列出 N2L 主服务器和 LDAP 服务器的地址。必须将服务器地址列在 `hosts`、`ipnodes` 还是同时列在这两个文件中，取决于这些文件配置为以何种方式解析本地主机名。同时，检查 `nsswitch.conf` 文件中的 `hosts` 和 `ipnodes` 项，在查找顺序上将 `files` 列在 `nis` 之前。

此死锁问题的替代解决方案是在 `ypserv` 文件中列出 LDAP 服务器的地址，而不是其主机名。这意味着 LDAP 服务器地址将列在其他位置。因此，更改 LDAP 服务器或 N2L 服务器的地址会使工作量稍有增加。

恢复为 NIS

已使用 N2L 服务从 NIS 转换到 LDAP 的站点将会逐步使用 Solaris LDAP 命名服务客户机替换所有的 NIS 客户机。对 NIS 客户机的支持最终会成为多余。但是，N2L 服务提供了两种在必要时返回传统 NIS 的方法，如以下两个过程中所述。

提示 – 传统的 NIS 会忽略 N2L 版本的 NIS 映射（如果存在这些映射）。恢复为 NIS 之后，如果在服务器上保留 N2L 版本的这些映射，则 N2L 映射不会产生问题。因此，如果您以后决定重新启用 N2L，则保留 N2L 映射可能会非常有用。但是，这些映射确实会占用磁盘空间。

▼ 如何基于旧的源文件恢复到 NIS 映射

- 1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

- 2 停止 NIS 守护进程。

```
# svcadm disable network/nis/server:default
```

- 3 禁用 N2L。

此命令可备份并移动 N2L 映射文件。

```
# mv /var/yp/NISLDAPmapping backup_filename
```

- 4 设置 NOPUSH 环境变量，以便 `yppmake` 不会推送新映射。

```
# NOPUSH=1
```

- 5 创建一组基于旧源的新 NIS 映射。

```
# cd /var/yp
# make
```

- 6 （可选）删除 N2L 版本的 NIS 映射。

```
# rm /var/yp/domainname/LDAP_*
```

- 7 启动 NIS 守护进程。

```
# svcadm enable network/nis/server:default
```

▼ 如何基于当前的 DIT 内容恢复为 NIS 映射

执行此过程之前请先备份旧的 NIS 源文件。

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 停止 NIS 守护进程。

```
# svcadm disable network/nis/server:default
```

3 从 DIT 更新映射。

```
# ypserv -r
```

等待 ypserv 退出。

4 禁用 N2L。

此命令可备份并移动 N2L 映射文件。

```
# mv /var/yp/NISLDAPmapping backup_filename
```

5 重新生成 NIS 源文件。

```
# ypmapping2src
```

6 手动检查重新生成的 NIS 源文件是否具有正确的内容和结构。

7 将重新生成的 NIS 源文件移到适当的目录中。

8 （可选）删除 N2L 版本的映射文件。

```
# rm /var/yp/domainname/LDAP_*
```

9 启动 NIS 守护进程。

```
# svcadm enable network/nis/server:default
```


从 NIS+ 转换为 LDAP

本章介绍如何从使用 NIS+ 命名服务转换为使用 LDAP 命名服务。

NIS+ 到 LDAP 的转换概述

NIS+ 服务器守护进程 `rpc.nisd` 将 NIS+ 数据以专用文件格式存储到 `/var/nis/data` 目录中。尽管完全有可能使 NIS+ 数据与 LDAP 同步，但这样的同步以前需要外部代理。不过，现在使用 NIS+ 守护进程，可以将 LDAP 服务器用作 NIS+ 数据的数据仓库。由于这样一来可以使 NIS+ 客户机和 LDAP 客户机共享相同的命名服务信息，因此更易于从使用 NIS+ 作为主命名服务转换为使用 LDAP 作为主命名服务。

缺省情况下，`rpc.nisd` 守护进程继续按照以前的方式工作，即仅依赖 `/var/nis/data` NIS+ 数据库。如果需要的话，系统管理员可以选择使用 LDAP 服务器作为 NIS+ 数据库任何子集的权威数据仓库。此时，`/var/nis/data` 文件充当 `rpc.nisd` 守护进程的高速缓存，这样可以减少 LDAP 查找的通信流量，而且如果 LDAP 服务器暂时不可用，`rpc.nisd` 还可以继续工作。除了可以在 NIS+ 和 LDAP 之间保持持续同步外，还可以将 NIS+ 数据上载到 LDAP 或者将 LDAP 数据下载到 NIS+。

将数据映射到 LDAP 或从 LDAP 映射数据由灵活的配置文件语法来控制。模板映射文件 `/var/nis/NIS+LDAPmapping.template` 包含所有的标准 NIS+ 表（`client_info.org_dir` 和 `timezone.org_dir` 除外），对于大多数 NIS+ 安装来说，无需更改该文件或只需进行很小的改动即可。（有关 `client_info.org_dir` 和 `timezone.org_dir` 的信息，请参见第 270 页中的“`client_info` 和 `timezone` 表（从 NIS+ 转换为 LDAP）”。）除了提供 NIS+ 数据在 LDAP 目录信息树 (Directory Information Tree, DIT) 中的位置以外，该映射文件还允许为源自 LDAP 的 NIS+ 数据提供生存时间 (time-to-live, TTL)。虽然通常情况下 NIS+ 列的值和 LDAP 属性值之间是一一对应的关系，但是该映射文件还可以用于维护更复杂的关系。

`/etc/default/rpc.nisd` 文件用于选择 LDAP 服务器和验证，并控制 `rpc.nisd` 的一些常规行为。请参见 `rpc.nisd(4)`。映射的详细信息在 `/var/nis/NIS+LDAPmapping` 文件中指

定。有关更多信息，请参见 [NIS+LDAPmapping\(4\)](#)。映射文件的名称可以通过编辑 `/lib/svc/method/nisplus` 文件来更改。有关更多信息，请参见第 247 页中的“[NIS+ 到 LDAP 转换工具和服务管理工具](#)”。

本章使用了以下术语：

- **Container（容器）**

容器是指 LDAP DIT 中用于存储所有相关项的位置。例如，用户帐户信息通常存储在 `ou=People` 容器中，而主机地址信息可以存储在 `ou=Hosts` 容器中。

- **Netname（网络名）**

网络名是安全 RPC（用户或计算机）中可被验证的实体。

- **Mapping（映射）**

映射是指 NIS+ 对象和 LDAP 项之间的关系。例如，`passwd.org_dir` NIS+ 表中 `name` 列的数据（如帐户的用户名）与 `ou=People` 容器中 `posixAccount` 对象类的 LDAP `uid` 属性相对应。该配置可以在 `name` 列和 `uid` 属性之间建立映射。还可以理解为将 `name` 列映射到 `uid` 属性（反之亦然）。

- **Principal（主体）**

主体是指 NIS+（用户或计算机）中可被验证的实体。通常，网络名和主体名之间存在一一对应关系。

rpc.nisd 配置文件

`rpc.nisd` 操作由两个配置文件来控制。

- `/etc/default/rpc.nisd`

此文件包含有关 LDAP 服务器和验证、NIS+ 基本域、LDAP 缺省搜索库、异常处理和常规 `rpc.nisd` 配置（无论 LDAP 映射是否有效，都将应用此配置）的信息。

- `/var/nis/NIS+LDAPmapping`

此文件包含有关在 NIS+ 数据和 LDAP 之间相互映射的信息。模板文件 (`/var/nis/NIS+LDAPmapping.template`) 包含除 `client_info.org_dir` 和 `timezone.org_dir` 以外的所有标准 NIS+ 对象。请参见第 270 页中的“[client_info](#) 和 [timezone](#) 表（从 NIS+ 转换为 LDAP）”和 [NIS+LDAPmapping\(4\)](#)。

配置是通过为预定义的属性赋值来完成的。除了配置文件，配置属性还可以从 LDAP 读取（请参见第 277 页中的“[将配置信息存储到 LDAP 中](#)”），或者在 `rpc.nisd` 命令行中通过 `-x` 选项指定。如果在多个位置中指定了同一个属性，则优先顺序（从高到低）将如下所示：

1. `rpc.nisd -x` 选项
2. 配置文件
3. LDAP

NIS+ 到 LDAP 转换工具和服务管理工具

与 NIS+ 到 LDAP 的转换相关联的大多数命令行管理任务都由服务管理工具来管理。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 [svcadm\(1M\)](#) 和 [svcs\(1\)](#) 手册页。

- 使用 `svcadm` 命令可以对 NIS+ 到 LDAP 转换服务执行启用、禁用或重新启动等管理操作。

提示 – 使用 `-t` 选项暂时禁用服务可为服务配置提供一些保护。如果禁用服务时使用了 `-t` 选项，则服务在重新引导后将恢复原始设置。如果禁用服务时未使用 `-t`，则服务在重新引导后仍将保持禁用状态。

- NIS+ 故障管理资源标识符 (Fault Managed Resource Identifier, FMRI) 是 `svc:/network/rpc/nisplus:<instance>`。LDAP 客户机服务的 FMRI 是 `svc:/network/ldap/client:<instance>`。
- 可以使用 `svcs` 命令来查询 NIS+ 的状态。
 - `svcs` 命令和输出示例。

```
# svcs \*nisplus\*
STATE      STIME      FMRI
online     Sep_01     svc:/network/rpc/nisplus:default
```

- `svcs -l` 命令和输出示例。要获得如下所示的输出，必须在 FMRI 中使用实例名称。

```
# svcs -l network/rpc/nisplus:default
fmri      svc:/network/rpc/nisplus:default
enabled   false
state     disabled
next_state none
restarter svc:/system/svc/restarter:default
dependency require_all/none svc:/network/rpc/keyserv (online)
```

- 使用 `ps` 命令可检查守护进程是否存在。

```
# ps -e | grep rpc.nisd
root 23320      1   0   Aug 27 ?           16:30 ./ns-slapd -D \
/usr/iplanet/ds5/slapd-lastrev -i /usr/iplanet/ds5/slapd-lastrev/
root 25367 25353   0 15:35:19 pts/1    0:00 grep slapd
```

注 – 不要将 `-f` 选项与 `ps` 结合使用，因为此选项会尝试将用户 ID 转换为名称，而这将导致更多命名服务查找可能无法成功。

何时不使用 SMF 进行 NIS+ 到 LDAP 转换

通常，`/usr/sbin/rpc.nisd` 守护进程使用 `svcadm` 命令进行管理。但是，当使用 `-x nisplusLDAPinitialUpdateOnly=yes` 调用 `rpc.nisd` 时，`rpc.nisd` 会执行指定的初始更新操作，然后再退出。即，`rpc.nisd` 不会守护进程化。服务管理工具不应当与 `-x`

`nisplusLDAPinitialUpdateOnly=yes` 结合使用。SMF 可以在您希望启动、停止或重新启动 `rpc.nisd` 守护进程的其他任何时间使用。

以下示例说明如何将 `rpc.nisd` 与 `-x nisplusLDAPinitialUpdateOnly=yes` 结合使用。

```
# /usr/sbin/rpc.nisd -m mappingfile \
-x nisplusLDAPinitialUpdateAction=from_ldap \
-x nisplusLDAPinitialUpdateOnly=yes
```

修改 `/lib/svc/method/nisplus` 文件

如果要在用服务管理工具调用 `rpc.nisd` 守护进程时包含特定选项，可以使用 `svccprop` 命令或者修改 `/lib/svc/method/nisplus` 文件。有关使用 `svccprop` 命令的更多信息，请参见 [svccprop\(1\)](#) 手册页。以下过程说明如何修改 `/lib/svc/method/nisplus` 文件。

▼ 如何修改 `/lib/svc/method/nisplus` 文件

1 成为超级用户或承担等效角色。

角色包含授权和具有一定特权的命令。有关角色的更多信息，请参见《系统管理指南：安全性服务》中的第 9 章“使用基于角色的访问控制（任务）”。

2 停止 NIS+ 服务。

```
# svcadm disable network/rpc/nisplus:default
```

3 打开 `/lib/svc/method/nisplus` 文件。

使用所选的编辑器。

4 编辑该文件并添加所需的选项。

将如下内容：

```
if [ -d /var/nis/data -o -d /var/nis/$hostname ]; then
    /usr/sbin/rpc.nisd || exit $
```

更改为：

```
if [ -d /var/nis/data -o -d /var/nis/$hostname ]; then
    /usr/sbin/rpc.nisd -Y -B || exit $?
```

在本示例中，向 `rpc.nisd` 中添加了 `-Y` 和 `-B` 选项，因此这两个选项将在系统启动时自动实现。

5 保存并退出 `/lib/svc/method/nisplus` 文件。

6 启动 NIS+ 服务。

```
# svcadm enable network/rpc/nisplus:default
```


创建属性和对象类

根据 NIS+/LDAP 映射的配置方式，您可能希望创建许多新的 LDAP 属性和对象类。以下示例说明如何通过指定 LDIF 数据来完成此操作，这些数据可用作 `ldapadd` 命令的输入。创建一个包含 LDIF 数据的文件，然后调用 `ldapadd(1)`。

```
# ldapadd -D bind-DN -f ldif -file
```

此方法适用于 Sun Java System Directory Server，还可能适用于其他 LDAP 服务器。

注—除了 `defaultSearchBase`、`preferredServerList` 和 `authenticationMethod` 属性以及 SYNTAX 规范，本章中使用的对象标识符 (object identifier, OID) 仅用于说明。由于尚未指定任何正式的 OID，因此您可以随意使用任何适当的 OID。

NIS+ 到 LDAP 转换入门

有关开始为 NIS+ 数据使用 LDAP 系统信息库所需的配置的介绍，请参见 [NIS+LDAPmapping\(4\)](#)。本节中的其余部分更详细地介绍配置文件的组织结构。

/etc/default/rpc.nisd 文件

/etc/default/rpc.nisd 文件中所有的赋值均为 `attributeName=value` 类型。

常规配置

下列属性控制 `rpc.nisd` 的常规配置，无论 LDAP 映射是否有效，这些属性都处于活动状态。通常情况下，应保留其缺省值。有关更多信息，请参见 [rpc.nisd\(4\)](#)。

- `nisplusNumberOfServiceThreads`
- `nisplusThreadCreationErrorAction`
- `nisplusThreadCreationErrorAttempts`
- `nisplusThreadCreationErrorTimeout`
- `nisplusDumpErrorAction`
- `nisplusDumpErrorAttempts`
- `nisplusDumpErrorTimeout`
- `nisplusResyncService`
- `nisplusUpdateBatching`
- `nisplusUpdateBatchingTimeout`

来自 LDAP 的配置数据

下列属性控制如何从 LDAP 读取其他配置属性。这些属性本身不能驻留在 LDAP 中。它们只能从命令行或配置文件中读取。有关更多信息，请参见 [rpc.nisd\(4\)](#)。

- `nisplusLDAPconfigDN`

- nisplusLDAPconfigPreferredServerList
- nisplusLDAPconfigAuthenticationMethod
- nisplusLDAPconfigTLS
- nisplusLDAPconfigTLSCertificateDBPath
- nisplusLDAPconfigProxyUser
- nisplusLDAPconfigProxyPassword

服务器选择

- preferredServerList
指定 LDAP 服务器和端口号。

LDAP server can be found at port 389
LDAP server can be found at port 389
on the local machine
preferredServerList=127.0.0.1
Could also be written
preferredServerList=127.0.0.1:389
LDAP server on the machine at IP
address "1.2.3.4", at port 65042
preferredServerList=1.2.3.4:65042

验证和安全性

- authenticationMethod
- nisplusLDAPproxyUser
- nisplusLDAPproxyPassword

指定验证方法，在适用于所选方法的情况下，还可指定要在 `rpc.nisd` 守护进程和 LDAP 服务器之间使用的代理用户（绑定的标识名 (distinguished name, DN)）和口令（密钥或其他共享秘密）。有关更多信息，请参见第 261 页中的“安全性和验证”。

- nisplusLDAPTLS
- nisplusLDAPTLSCertificateDBPath

可以选择使用 SSL，并指定证书文件的位置。有关更多信息，请参见第 263 页中的“使用 SSL”。

LDAP 和 NIS+ 中的缺省位置

- defaultSearchBase
LDAP DIT 中用于存放 RFC 2307 样式的命名服务数据的位置。当未在单独的容器 DN 中指定完整的搜索库时，会使用此缺省位置。有关更多信息，请参见第 254 页中的“[nisplusLDAPobjectDN 属性](#)”。
- nisplusLDAPbaseDomain
当 NIS+ 对象规范（请参见第 252 页中的“[nisplusLDAPdatabaseIdMapping 属性](#)”）未完全限定时使用的缺省 NIS+ 域名。

LDAP 通信的超时/大小限制和引用操作

- `nisplusLDAPbindTimeout`
- `nisplusLDAPmodifyTimeout`
- `nisplusLDAPaddTimeout`
- `nisplusLDAPdeleteTimeout`

以上参数分别是 `ldap bind`、`modify`、`add` 和 `delete` 操作的超时参数。通常情况下，应保留其缺省值。

- `nisplusLDAPsearchTimeout`
- `nisplusLDAPsearchTimeLimit`

以上第一个参数用于设置 LDAP 搜索操作的超时值，第二个参数用于请求服务器端的搜索时间限制。由于 `nisplusLDAPsearchTimeLimit` 将控制 LDAP 服务器花在搜索请求上的时间，因此请确保 `nisplusLDAPsearchTimeLimit` 不小于

`nisplusLDAPsearchTimeout`。根据 NIS+ 服务器、LDAP 服务器以及二者之间连接的性能，您可能需要在缺省值的基础上增大搜索限制的值。请查看 `rpc.nisd` 的超时系统日志消息，并根据消息提示来增大这些值。

- `nisplusLDAPsearchSizeLimit`

以上参数请求 LDAP 搜索请求所返回的 LDAP 数据量的限制值。缺省值为请求无限制。这是服务器端的限制。LDAP 服务器可能会对最大值施加限制，并且这些限制可能会绑定到所使用的代理用户（绑定 DN）。请确保 LDAP 服务器允许 `rpc.nisd` 传送足以置备最大容器（具体取决于站点设置，通常是指用于 `passwd.org_dir`、`mail_aliases.org_dir` 或 `netgroup.org_dir` 的容器）的数据。有关更多信息，请查阅 LDAP 服务器文档。

- `nisplusLDAPfollowReferral`

以上参数用于定义当 LDAP 操作导致需要引用其他 LDAP 服务器时要执行的操作。缺省值为不遵循引用。如果您希望或者需要遵循引用，请启用“遵循引用”。请记住，尽管引用会带来方便，但是它们也会使 `rpc.nisd` 对于每个请求都与多台 LDAP 服务器通信，从而降低操作速度。`rpc.nisd` 通常应当直接指向能够处理 `rpc.nisd` 可能发出的任何 LDAP 请求的 LDAP 服务器。

错误操作

下列参数定义当 LDAP 操作过程中出现错误时要执行的操作。通常情况下，应保留其缺省值。有关更多信息，请参见 `rpc.nisd(4)`。

- `nisplusLDAPinitialUpdateAction`
- `nisplusLDAPinitialUpdateOnly`
- `nisplusLDAPretrieveErrorAction`
- `nisplusLDAPretrieveErrorAttempts`
- `nisplusLDAPretrieveErrorTimeout`
- `nisplusLDAPstoreErrorAction`
- `nisplusLDAPstoreErrorAttempts`

- `nisplusLDAPstoreErrorTimeout`
- `nisplusLDAPrefreshErrorAction`
- `nisplusLDAPrefreshErrorAttempts`
- `nisplusLDAPrefreshErrorTimeout`

常规 LDAP 操作控制

- `nisplusLDAPmatchFetchAction`

以上参数用于确定是否应当为 NIS+ 匹配操作预先提取 LDAP 数据。在大多数情况下，应保留其缺省值。有关更多信息，请参见[rpc.nisd\(4\)](#)。

/var/nis/NIS+LDAPmapping 文件

缺省的 NIS+LDAPmapping 文件充当 NIS+/LDAP 映射的主转换器。

如果您使用非缺省映射文件，则必须编辑 `/lib/svc/method/nisplus` 脚本，以在 `rpc.nisd` 行中使用 `-m mappingfile` 选项指定映射文件的名称。有关更多信息，请参见第 247 页中的“NIS+ 到 LDAP 转换工具和服务管理工具”。

对于应当映射到 LDAP 或从 LDAP 映射的每个 NIS+ 对象，NIS+LDAPmapping 文件将指定两到五个属性，具体取决于该对象以及缺省值是否满足需要。

nisplusLDAPdatabaseIdMapping 属性

您必须设置一个要在其他映射属性中使用的别名。如果 NIS+ 对象名不是全限定名（不以点结尾），则将附加 `nisplusLDAPbaseDomain` 值。

例如，

```
nisplusLDAPdatabaseIdMapping    rpc:rpc.org_dir
```

将数据库 ID `rpc` 定义为 NIS+ `rpc.org_dir` 表的别名。

请注意，NIS+ 表对象可能会针对表对象（如果该对象应当映射到 LDAP）和表项分别出现一次，只是两种情况下的数据库 ID 不同。例如，

```
nisplusLDAPdatabaseIdMapping    rpc_table:rpc.org_dir
nisplusLDAPdatabaseIdMapping    rpc:rpc.org_dir
```

将数据库 ID `rpc_table` 和 `rpc` 定义为 `rpc.org_dir` 表的别名。后面的定义清楚表明将 `rpc_table` 用于 `rpc.org_dir` 表对象，将 `rpc` 用于该表中的各项。

nisplusLDAPentryTtl 属性

由于 `rpc.nisd` 守护进程的本地数据库（位于内存中和磁盘上）充当 LDAP 数据的高速缓存，因此可使用 `nisplusLDAPentryTtl` 属性来设置该高速缓存中各项的生存时间 (time-to-live, TTL) 值。每个数据库 ID 都有三个 TTL。前两个 TTL 用于控制当 `rpc.nisd` 首次从磁盘装入对应的 NIS+ 对象数据时的初始 TTL，第三个 TTL 将在从 LDAP 中读取或刷新对象时指定给该对象。

例如，通过以下语句，`rpc.org_dir` 表对象可获取一个在 21600 秒到 43200 秒的范围内随机选择的初始 TTL。

```
nisplusLDAPentryTtl    rpc_table:21600:43200:43200
```

如果初始 TTL 过期并从 LDAP 中刷新了表对象，则该 TTL 将设置为 43200 秒。

同样，使用以下语句，可在首次装入 `rpc.org_dir` 表时向该表中的各项指定一个介于 1800 秒和 3600 秒之间的初始 TTL。

```
nisplusLDAPentryTtl    rpc:1800:3600:3600
```

每项都会获取各自在指定范围内随机选择的 TTL。如果某个表项过期并已刷新，则该 TTL 将设置为 3600 秒。

选择 TTL 值时，需要综合考虑性能和一致性。如果由 `rpc.nisd` 高速缓存的 LDAP 数据所使用的 TTL 非常长，则性能与 `rpc.nisd` 根本未从 LDAP 映射数据时一样。但是，如果 LDAP 数据被 `rpc.nisd` 以外的某个实体更改，则所做的更改也可能会在很长时间之后才显示在 NIS+ 中。

相反，如果选择非常短（或者甚至为零）的 TTL，则意味着对 LDAP 数据进行的更改会迅速显示在 NIS+ 中，但这也可能会大大降低性能。通常，对于会同时从 LDAP 中读取数据或向 LDAP 中写入数据的 NIS+ 操作而言，所需的时间将至少为没有 LDAP 通信时执行同一操作所需时间的两到三倍（而且还会带来额外的 LDAP 查找开销）。尽管性能会因为硬件资源而大不相同，但是，扫描大型 LDAP 容器（具有几万或几十万个项）以确定应当刷新的 NIS+ 项可能会需要很长时间。`rpc.nisd` 守护进程在后台执行扫描操作，从而可在运行时继续为可能过时的数据提供服务，但是后台扫描仍然会占用 NIS+ 服务器中的 CPU 和内存。

请仔细考虑保持 NIS+ 数据与 LDAP 密切同步的重要性，并选择每个 NIS+ 对象都可接受的最长 TTL。缺省值（未指定 `nisplusLDAPentryTtl` 时的值）是 1 小时。对于表项以外的对象，模板映射文件 `/var/nis/NIS+LDAPmapping.template` 会将该值更改为 12 小时。但是，系统无法自动识别非项对象，因此，如果要为非项对象添加映射，TTL 的缺省值将为 1 小时。

注 - 不存在的对象没有 TTL。因此，无论哪些 TTL 对于 NIS+ 表中的 LDAP 映射项有效，请求 NIS+ 中不存在的项时都将在 LDAP 中查询该项。

nisplusLDAPobjectDN 属性

对于每个映射的 NIS+ 对象，`nisplusLDAPobjectDN` 都会设置对象数据在 LDAP DIT 中所处的位置。它还允许指定在删除 LDAP 项之后要执行的操作。每个 `nisplusLDAPobjectDN` 值都有三部分。第一部分指定从何处读取 LDAP 数据，第二部分指定将 LDAP 数据写入何处，第三部分指定在删除 LDAP 数据时应当发生的情况。请参阅以下示例：

```
nisplusLDAPobjectDN    rpc_table:\
                        cn=rpc,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer:\
cn=rpc,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer,\
                        objectClass=top
```

以上示例说明应当从 DN `cn=rpc,ou=nisPlus` 读取 `rpc.org_dir` 表对象（由于该值以逗号结尾，附加了 `defaultSearchBase` 属性的值），搜索范围为 `base`，`ObjectClass` 属性的值为 `nisplusObjectContainer` 的项处于选中状态。

表对象被写入同一位置。缺少删除规范，这意味着将执行缺省操作，如下所示。如果该 NIS+ 表对象被删除，整个 LDAP 项也应当被删除。

如果应从 LDAP 读取数据，而不应向其中写入数据，请忽略写入部分（以及用于将写入部分与读取部分分开的冒号）。

```
nisplusLDAPobjectDN    rpc_table:\
                        cn=rpc,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer
```

请注意，`nisplusObjectContainer` 对象类不属于 RFC 2307。要使用它，您必须按照第 264 页中的“映射表项以外的 NIS+ 对象”中的详细说明配置 LDAP 服务器。

对于 `rpc.org_dir` 表项，您可以使用以下示例：

```
nisplusLDAPobjectDN    rpc:ou=Rpc,?one?objectClass=oncRpc:\
                        ou=Rpc,?one?objectClass=onRpc,objectClass=top
```

以上示例显示了从基本 `ou=Rpc` 中读取表项和向其中写入表项。而且，结尾的逗号还会附加 `defaultSearchBase` 值。选择 `objectClass` 属性的值为 `oncRpc` 的项。当在 LDAP 中的 `ou=Rpc` 容器中创建项时，您还必须将 `top` 指定为 `objectClass` 值。

请考虑以下示例中说明的非缺省删除规范：

```
nisplusLDAPobjectDN    user_attr:\
                        ou=People,?one?objectClass=SolarisUserAttr,\
                        solarisAttrKeyValue=*\
ou=People,?one?objectClass=SolarisUserAttr:\
                        dbid=user_attr_del
```

`user_attr.org_dir` 数据驻留在 `ou=People` LDAP 容器中，该容器由这些数据和来自其他源（如 `passwd.org_dir` NIS+ 表）的帐户信息共享。

在该容器中选择具有 `solarisAttrKeyValue` 属性的项，因为只有这些项才包含 `user_attr.org_dir` 数据。`nisplusLDAPObjectDN` 的 `dbid=user_attr_del` 部分说明在删除 `user_attr.org_dir` NIS+ 表中的某个项时，应当按照 `user_attr_del` 数据库 ID 所标识的规则集合内的规则删除对应的 LDAP 项（如果有的话）。有关更多信息，请参见第 255 页中的“`nisplusLDAPcolumnFromAttribute` 属性”。

`nisplusLDAPattributeFromColumn` 属性

`nisplusLDAPattributeFromColumn` 指定用于将 NIS+ 数据映射到 LDAP 的规则。其他方向的映射规则由 `nisplusLDAPcolumnFromAttribute` 控制。

`nisplusLDAPcolumnFromAttribute` 属性

`nisplusLDAPcolumnFromAttribute` 指定用于将 LDAP 数据映射到 NIS+ 的规则。

完整的项映射语法可以在 [NIS+LDAPmapping\(4\)](#) 中找到。但是，通过几个示例可以更清楚地理解映射语法。

NIS+ `rpc.org_dir` 表中包含四个名为 `cname`、`name`、`numbe` 和 `comment` 的列。因此，NIS+ RPC 程序号 (100300) 的标准名称为 `nisd`、别名为 `rpc.nisd` 和 `nisplusd` 的项可以由 `rpc.org_dir` 中的下列 NIS+ 项表示：

```
nisd nisd 100300    NIS+ server
nisd rpc.nisd 100300    NIS+ server
nisd nisplusd 100300    NIS+ server
```

假定 `defaultSearchBase` 值为 `dc=some,dc=domain`，对应的 LDAP 项（列在 [ldapsearch\(1\)](#) 中）将如下所示：

```
dn: cn=nisd,ou=Ppc,dc=some,dc=domain
cn: nisd
cn: rpc.nsid
cn: nisplusd
oncRpcNumber: 100300
description: NIS+ server
objectClass: oncRpc
```

这有助于在 NIS+ 数据和 LDAP 数据之间建立简单的一一映射的关系，而从 NIS+ 到 LDAP 的对应映射属性值如下所示：

```
nisplusLDAPattributeFromColumn \
rpc:      dn=("cn=%s,", name), \
          cn=cname, \
          cn=name, \
          oncRpcNumber=number, \
          description=comment
```

这会使该项的 DN 为 `cn=%s`，并用 `cname` 列的值替换 `%s`。

```
cn=nisd,
```

由于该值以逗号结尾，因此将附加从 `nisplusObjectDN` 读取的基值，结果如下所示：

```
cn=nisd,ou=Rpc,dc=some,dc=domain
```

`oncRpcNumber` 和 `description` 属性值只是对应的 NIS+ 列的简单赋值。`rpc.nisd` 会将多个 NIS+ 项收集到一个 LDAP 项中，并用多个 `cn` 值来表示不同的 `name` 列值。

同样，从 LDAP 到 NIS+ 的映射将如下所示：

```
nisplusLDAPcolumnFromAttribute \
rpc:      cname=cn, \
          (name)=(cn), \
          number=oncRpcNumber, \
          comment=description
```

以上示例会将 `oncRpcNumber` 和 `description` 值指定给对应的 NIS+ 列。多值 `cn`（由 `(cn)` 表示）映射到多个 `name` 列值（由 `(name)` 表示）。由于 `name` 列不能具有多值，因此 `rpc.nisd` 会为每个 `cn` 值创建一个 NIS+ 项。

最后，`nisplusLDAPattributeFromColumn` 值是一个用于删除操作的规则集合的示例。

```
nisplusLDAPattributeFromColumn \
user_attr_del:   dn=("uid=%s,", name), \
                SolarisUserQualifier=, \
                SolarisAttrReserved1=, \
                SolarisAttrReserved2=, \
                SolarisAttrKeyValue=
```

而且，`user_attr.org_dir` 数据与其他帐户信息（来自 `passwd.org_dir` 和其他表）共享 `ou=People` 容器。如果 `user_attr.org_dir` 表中的一项被删除，您可能不希望整个 `ou=People` 项被删除。相反，以上删除项假设当 `user_attr.org_dir` 项被删除时，`SolarisUserQualifier`、`SolarisAttrReserved1`、`SolarisAttrReserved2` 和 `SolarisAttrKeyValue` 属性（如果有的话）会从以下规则所指定的 `ou=People` 项中删除：

```
dn=("uid=%s,", name)
```

LDAP 项的其余部分保持不变。

NIS+ 到 LDAP 迁移方案

以下列出了从 NIS+ 迁移到 LDAP 时可能的迁移方案：

- 通过一步操作将所有的 NIS+ 客户机转换为使用 LDAP。您可以使用 `rpc.nisd` 守护进程上载 LDAP 中尚不存在的任何 NIS+ 数据。请参见第 257 页中的[“如何通过一步操作将所有的 NIS+ 数据转换为 LDAP”](#)。

- 从 NIS+ 逐步迁移到 LDAP。首先将 NIS+ 数据转换为 LDAP（请参见第 257 页中的[“如何通过一步操作将所有的 NIS+ 数据转换为 LDAP”](#)）。您可以让 NIS+ 客户机和 LDAP 客户机共享相同的命名服务数据，并使 `rpc.nisd` 自动保持 NIS+ 数据和 LDAP 数据的同步。NIS+ 最初可能是权威的命名服务，LDAP 服务器则维护 NIS+ 数据的副本以便于 LDAP 客户机使用。在适当的时候，可以使 LDAP 成为权威的命名服务，并逐步取消 NIS+ 服务，直到不再有 NIS+ 客户机。
- 已经使用了 LDAP 作为命名服务，因此您需要合并 NIS+ 数据和 LDAP 数据。有三种可能的方法来执行此合并。
 - 将 NIS+ 数据添加到 LDAP 中。将 NIS+ 中存在而 LDAP 中不存在的项添加到 LDAP 中。NIS+ 和 LDAP 中均存在但所含数据不同的项最终将保留 NIS+ 数据。请参见第 257 页中的[“如何通过一步操作将所有的 NIS+ 数据转换为 LDAP”](#)。
 - 使用 LDAP 数据覆写 NIS+ 数据。如果某些项在 NIS+ 中存在，而在 LDAP 中不存在，它们将从 NIS+ 中删除。对于在 NIS+ 和 LDAP 中均存在的项，将保留 LDAP 数据。请参见第 257 页中的[“如何通过一步操作将所有的 LDAP 数据转换为 NIS+”](#)。
 - 合并 NIS+ 数据和 LDAP 数据并逐一解决冲突。请参见第 258 页中的[“合并 NIS+ 数据和 LDAP 数据”](#)。

▼ 如何通过一步操作将所有的 NIS+ 数据转换为 LDAP

- 使用 `rpc.nisd` 上载 LDAP 中尚不存在的任何 NIS+ 数据。

使用以下命令（假设已在缺省位置（`/var/nis/NIS+LDAPmapping`）建立了所有 NIS+/LDAP 数据映射）：

```
# /usr/sbin/rpc.nisd -D \
-x nisplusLDAPinitialUpdateAction=to_ldap \
-x nisplusLDAPinitialUpdateOnly=yes
```

以上命令会指示 `rpc.nisd` 将数据上载到 LDAP，然后退出。NIS+ 数据将不会受到此操作的影响。

请参见 `rpc.nisd(4)` 的 `nisplusLDAPinitialUpdateAction` 属性。

▼ 如何通过一步操作将所有的 LDAP 数据转换为 NIS+

- 使用 `rpc.nisd` 将所有的 LDAP 数据下载到 NIS+，覆写现有的 NIS+ 数据。

使用以下命令（假设已在缺省位置（`/var/nis/NIS+LDAPmapping`）建立了所有 NIS+/LDAP 数据映射）：

```
# /usr/sbin/rpc.nisd -D \
-x nisplusLDAPinitialUpdateAction=from_ldap \
-x nisplusLDAPinitialUpdateOnly=yes
```

以上命令会指示 `rpc.nisd` 守护进程从 LDAP 下载数据，然后退出。LDAP 数据将不会受到此操作的影响。

请参见 `rpc.nisd(4)` 的 `nisplusLDAPinitialUpdateAction` 属性。

合并 NIS+ 数据和 LDAP 数据

第 256 页中的“NIS+ 到 LDAP 迁移方案”说明了当应该让 NIS+ 数据或 LDAP 数据成为权威数据以解决 NIS+ 数据和 LDAP 数据之间的冲突时，如何对 NIS+ 数据和 LDAP 数据进行同步。合并数据需要更复杂的过程。

本节中的示例过程做出了以下假定：

- 您将 NIS+ 数据的备份放在 `/nisbackup` 目录中。
- `/etc/default/rpc.nisd` 和 `/var/nis/tmpmap` 中已经存在有效的映射配置（对于应当合并的表）。
- 合并之前的 NIS+ 数据的平面文件形式存储在 `/before` 中，合并之后的平面文件形式存储在 `/after` 中。
- `niscat` 用于转储以平面文件形式表示的不受 `nisaddent(1M)` 支持的定制 NIS+ 表。您可以使用自己的命令或脚本，从 NIS+ 装入这样的定制表或者将这些表转储到 NIS+ 中。如果是这样，这些命令/脚本应当优先于 `niscat` 使用，因为后者没有能够将数据方便地重新装入 NIS+ 的相应命令。

如果您不得不使用 `niscat(1)` 转储数据，可以使用 `nistbladm(1)` 将项逐个重新装入到 NIS+ 中。

- 您的命令路径中包括 `/usr/lib/nis`（这是 `nisaddent(1M)` 所在的位置）。

▼ 如何合并 NIS+ 数据和 LDAP 数据



注意 – 如果 LDAP 数据在步骤 4 的下载和步骤 10 的上载操作之间发生变化，则上载过程可能会覆写这些更改。因此，应当尽量避免在该过程期间修改 LDAP 数据。有关更多信息，请查阅 LDAP 服务器文档。

- 1 使用 `nisbackup` 命令备份所有的 NIS+ 数据。

```
# nisbackup -a /nisbackup
```

- 2 标识那些包含必须与 LDAP 合并的数据的 NIS+ 表。将这些表的内容转储到平面文件中。例如，使用 `nisaddent`，按以下方式转储 `group.org_dir` 的内容：

```
# nisaddent -d group | sort > /before/group
```

将 `nisaddent` 输出传输到 `sort` 将便于稍后进行比较。

- 3 将 LDAP 数据下载到 NIS+ 中。

```
# /usr/sbin/rpc.nisd -D -m tmpmap \
-x nisplusLDAPinitialUpdateAction=from_ldap \
-x nisplusLDAPinitialUpdateOnly=yes
```

- 4 启动 NIS+ 服务。

```
# svcadm enable network/rpc/nisplus:default
```

rpc.nisd 守护进程现在将为从 LDAP 下载的数据提供服务。如果要解决的冲突很严重，以至于不应让 NIS+ 客户机面对这种冲突，请确保在活动 NIS+ 客户机很少（最好没有）时执行此步骤和后面的几步。

5 转储受影响的表中的 NIS+ 数据。

以下示例使用 group.org_dir 表。

```
# nisaddent -d group | sort > /after/group
```

6 创建这些表的合并版本。

使用您选择的文件合并过程生成合并的表。如果没有其他工具可用，可以使用 `diff(1)` 收集 /before 和 /after 文件之间存在差异的部分，然后用文本编辑器手动合并。

以下示例假定合并结果位于 /after 中。

7 将合并后的数据装入到 NIS+ 中。以下示例使用 group 表。

```
# nisaddent -m -f /after/group group
```

8 删除在合并之后不应当存在的 LDAP 项。

A. 如果某些 LDAP 项在（现在合并的）NIS+ 数据中不存在，但是在上载之后不应当在 LDAP 中存在，则必须删除这些 LDAP 项。

LDAP 服务器可能会提供一种用于删除多项的方便方法，如用于删除容器中所有项的方法。如果不属于这种情况，则可以使用 `ldapsearch(1)` 来为每个容器生成一系列项。例如，要生成 ou=Rpc 容器中所有项的列表，请按以下方式使用 `ldapsearch(1)`。

```
# ldapsearch -h server-address -D bind-DN -w password \
  -b ou=Rpc,search-base 'objectClass=*' dn | \
  grep -i ou=Rpc | grep -v -i \^ou=Rpc > /tmp/delete-dn
```

有关元参数（例如，`server-address` 和 `bind-DN`）的说明，请参见第 263 页中的“性能和索引”。

B. 现在您可以编辑结果文件 (/tmp/delete-dn)，仅指定应当删除的项。或者，为了删除该容器中的所有项，可以按原样使用该文件，并依赖 NIS+ 上载功能恢复 LDAP 数据。无论使用哪种方法，都应当先备份 LDAP 数据，然后再执行下面的 `ldapdelete` 操作。

C. 使用 `ldapdelete` 删除 LDAP 项，将 `stdout`（通常是每个删除的项对应一个空行）重定向到 /dev/null。

```
# ldapdelete -h server-address -D bind-DN -w password \
  /tmp/delete-dn /dev/null
```

D. 对每个包含至少一个必删项的容器，重复上述过程。

主服务器和副本服务器（从 NIS+ 转换为 LDAP）

只有 NIS+ 主服务器可以将数据写入 LDAP 中。NIS+ 副本服务器可以从 NIS+ 主服务器获取更新（主服务器可能已从 LDAP 获取该更新，也可能尚未获取），也可以直接从 LDAP 服务器读取数据。也可能会存在这两种情况的组合。因此，可通过两种主要方法来安排 NIS+ 复制。

- 保持 NIS+ 副本服务器不变，并使其从 NIS+ 主服务器获取数据更新。

这种安排的优点在于配置简单（只有 NIS+ 主服务器需要与 LDAP 服务器建立连接），并且还可保持以前的复制关系（主服务器先于副本服务器获知新数据）。当 NIS+ 始终是命名服务数据的权威源时，这种安排可能是最方便的解决方案。但是，此方案也会延长 LDAP 与 NIS+ 副本服务器之间的路径。

- 让 NIS+ 副本服务器直接从 LDAP 而非 NIS+ 主服务器获取其数据。

在这种情况下，副本服务器可能已在 NIS+ 主服务器之前或之后更新了数据，具体取决于查找通信流量和派生自 LDAP 的数据的 TTL。这种安排更为复杂，但是，如果 LDAP 是权威的命名服务系统信息库，而且很少或不对 NIS+ 数据进行直接更新，则这种安排非常方便。

复制时间标记

如果 NIS+ 副本服务器要从 LDAP 为特定 NIS+ 目录中至少一个对象获取数据，则 `nisping(1M)` 列显的更新时间标记不必指示 NIS+ 主服务器和副本服务器之间数据的一致程度。例如，假定 NIS+ 目录 `dir1` 中包含 `table1` 和 `table2` 表。如果副本服务器是从 NIS+ 主服务器获取 `table1` 和 `table2` 的数据，则可能会看到类似以下的输出：

```
# nisping dir1
```

```
Master server is "master.some.domain."
Last update occurred at Mon Aug  5 22:11:09 2002
```

```
Replica server is "replica.some.domain."
Last Update seen was Mon Aug  5 22:11:09 2002
```

以上输出表明主服务器和副本服务器具有完全相同的数据。但是，如果副本服务器是从 LDAP 获取 `table1` 和 `table2` 中任意一个表的数据或同时获取二者的数据，则输出中将仅显示副本服务器已从主服务器接收了 `NIS_PING`，并出于内务处理的目的更新了自己的重新同步时间标记。如果符合以下任一条件，从 LDAP 映射的一个或多个表中的数据可能会与 NIS+ 主服务器中的数据不同：

- LDAP 数据与 NIS+ 主服务器中的数据不同。
- 副本服务器的高速缓存（NIS+ 数据库的本地版本）中存在尚未过期但与 LDAP 不同步的数据。

如果您不接受这种类型的数据不一致，请允许所有的 NIS+ 副本服务器仅从 NIS+ 主服务器获取其数据。在将 NIS+ 主服务器配置为从 LDAP 获取数据之后，便不需要对副本服务器进行修改。

目录服务器（从 NIS+ 转换为 LDAP）

rpc.nisd 守护进程的 LDAP 映射部分使用 LDAP 协议版本 3 来与 LDAP 服务器通信。缺省的映射配置 (/var/nis/NIS+LDAPmapping.template) 希望 LDAP 服务器支持扩展版本的 RFC 2307。RFC 可从 <http://www.ietf.org/rfc.html> 获取。虽然可以使用 [NIS+LDAPmapping\(4\)](#) 来修改 NIS+ 数据和 LDAP 数据之间的映射，但基本的假设前提是 LDAP 数据按照 RFC 2307 中规定的原则进行组织。

例如，为了在直接 LDAP 客户机和 NIS+ 客户机之间共享帐户信息，LDAP 服务器必须支持以 UNIX crypt 格式存储帐户（用户）口令。如果无法配置 LDAP 服务器实现此操作，您仍可以将 NIS+ 数据（包括帐户）存储在 LDAP 中。但是，您将无法在 NIS+ 用户和 LDAP bindDN 之间完全共享帐户信息。

配置 Sun Java System Directory Server

有关安装、设置和管理 Sun Java System Directory Server 的详细说明，请参阅 Sun Java System Directory Server 文档集合。

您可以使用 [idsconfig\(1M\)](#) 为 LDAP 客户机配置 Sun Java System Directory Server，以便使用 LDAP 作为命名服务。在将 NIS+ 与 LDAP 数据仓库结合使用时，由 [idsconfig\(1M\)](#) 提供的设置也是适用的。

注 – 如果您使用的是 Sun Java System Directory Server 以外的 LDAP 服务器，则必须手动配置该服务器，使其支持 RFC 2307 架构。

指定服务器地址和端口号

/etc/default/rpc.nisd 文件被设置为在端口 389 上使用本地 LDAP 服务器。如果您的配置中不是这样，请为 preferredServerList 属性设置一个新值。例如，要在 IP 地址 192.0.0.1 和端口 65535 上使用 LDAP 服务器，请指定如下内容：

```
preferredServerList=192.0.0.1:65535
```

安全性和验证

当 NIS+ 服务器是从 LDAP 获取数据时，NIS+ 客户机和 NIS+ 服务器之间的验证不受影响。但是，为了保持存储在 LDAP 中的 NIS+ 数据的完整性，请考虑在 rpc.nisd 守护进程和 LDAP 服务器之间配置验证。根据 LDAP 服务器的功能，可以使用几种不同类型的验证。

以下列出了 rpc.nisd 守护进程支持的 LDAP 验证：

- none

none 验证方法为缺省方法。虽然使用 none 无需进行任何设置，但是它并不提供安全性。它仅适用于根本没有安全要求的环境。

要使用 none 验证，请确保 authenticationMethod 属性具有以下值：

```
authenticationMethod=none
```

至少实际上能够提供一定安全性的验证方法通常要求您将共享秘密（口令或密钥）与 LDAP 中的 DN 相关联。您选择的供 rpc.nisd 守护进程使用的 DN 可以是唯一的，也可以用于其他目的。它应当有适当的功能来支持预期的 LDAP 通信流量。例如，如果 rpc.nisd 守护进程应当能够向 LDAP 中写入数据，则所选 DN 必须有权在用于 NIS+ 数据的容器中添加/更新/删除 LDAP 数据。此外，在缺省情况下，LDAP 服务器可能会对资源使用情况施加限制（如搜索时间限制或搜索结果大小限制）。如果情况如此，您选择的 DN 必须具有足够的能力来支持对 NIS+ 数据容器的枚举。

- simple

simple 验证方法通过交换未经加密的口令字符串来提供验证。由于口令在 LDAP 客户机（rpc.nisd 守护进程）和 LDAP 服务器之间以明文形式发送，因此，只有当 NIS+ 和 LDAP 服务器之间的信息交换受到某种其他方法的保护时，simple 方法才适用。

例如，对 LDAP 通信流量进行传输层加密，或者 NIS+ 和 LDAP 服务器是同一个系统，并且 NIS+/LDAP 通信流量保持在内核中（非授权用户无法看到）。

使用要用于 simple 验证的 DN 和口令来修改 rpc.nisd 守护进程的配置。例如，如果 DN 是 cn=nisplusAdmin,ou=People,dc=some,dc=domain，口令是 aword，请设置以下内容：

```
authenticationMethod=simple
nisplusLDAPproxyUser=cn=nisplusAdmin,ou=People,dc=some,dc=domain
nisplusLDAPproxyPassword=aword
```

一定要防止对口令的存储位置进行未经授权的访问。请记住，如果口令是在 rpc.nisd 命令行上指定的，那么，系统上的任何用户都可以通过诸如 [ps\(1\)](#) 之类的命令查看口令。

- sasl/digest-md5

sasl/digest-md5 验证方法使用 digest/md5 算法来提供验证。

有关如何设置授权标识以用于 digest-md5 验证的信息，以及如何修改 /etc/default/rpc.nisd 文件以指定此标识及其相关口令的信息，请查阅 LDAP 服务器文档。

```
authenticationMethod=sasl/digest-md5
nisplusLDAPproxyUser=cn=nisplusAdmin,ou=People,dc=some,dc=domain
nisplusLDAPproxyPassword=aword
```

一定要防止对用于存储口令的文件进行未经授权的访问。

- sasl/cram-md5

使用 `cram/md5` 算法进行验证。可能只有过时的 SunDS LDAP 服务器才支持此验证方法。

有关如何设置绑定 DN 以用于 `cram-md5` 验证的信息，以及如何修改 `/etc/default/rpc.nisd` 文件以指定此 DN 及其相关口令的信息，请查阅 LDAP 服务器文档。

```
authenticationMethod=sasl/cram-md5
nisplusLDAPproxyUser=cn=nisplusAdmin,ou=People,dc=some,dc=domain
nisplusLDAPproxyPassword=aword
```

一定要防止对用于存储口令的文件进行未经授权的访问。

使用 SSL

`rpc.nisd` 守护进程还支持使用 SSL 对 LDAP 通信流量进行传输层加密。有关如何生成用于 LDAP 服务器验证的 SSL 证书的信息，请查阅 LDAP 服务器文档。请将该证书存储到 NIS+ 服务器上的某个文件（例如，`/var/nis/cert7.db`）中并按以下方式修改 `/etc/default/rpc.nisd`：

```
nisplusLDAPTLS=ssl
nisplusLDAPTSLSCertificateDBPath=/var/nis/cert7.db
```

一定要防止对证书文件进行未经授权的访问。请注意，以上方法提供会话加密，还提供 LDAP 服务器到 `rpc.nisd` 的验证。它并不提供 `rpc.nisd` 到 LDAP 服务器的验证，因为证书中不包含用于标识 LDAP 客户机 (`rpc.nisd`) 的任何内容。但是，可以将 SSL 与其他验证方法（`simple` 和 `sasl/digest-md5`）结合使用来实现相互验证。

性能和索引

当 `rpc.nisd` 守护进程被要求枚举一个从 LDAP 映射的 NIS+ 表（例如，使用 `niscat(1)`）时，如果表中至少有一项具有过期的 TTL，该守护进程将枚举对应的 LDAP 容器。尽管容器的枚举是在后台进行的，使 LDAP 性能显得不太重要，但是建立 LDAP 索引的好处是有助于加速对大型容器的枚举。

要获取枚举特定容器所需的大概时间，可以使用如下所示的命令：

```
% /bin/time ldapsearch -h server-address -D bind-DN -w password \
-b container , search-base 'cn=*' /dev/null
```

其中，

- *server-address*
 `/etc/default/rpc.nisd` 中 `preferredServerList` 值的 IP 地址部分
- *bind-DN*
 `/etc/default/rpc.nisd` 中的 `nisplusLDAPproxyUser` 值

- *password*
/etc/default/rpc.nisd 中的 nisplusLDAPproxyPassword 值
- *container*
RFC 2307 容器名（ou=Services、ou=Rpc 等等）之一
- *search-base*
/etc/default/rpc.nisd 中的 defaultSearchBase 值

由 /bin/time 列显的“实际”值是经过的（挂钟）时间。如果此值超出对应表项的 TTL 很大一部分（25% 或更多）（请参见第 250 页中的“验证和安全性”），为 LDAP 容器创建索引可能会非常有用。

rpc.nisd 支持 simple page 和 VLV 索引方法。要查找 rpc.nisd 支持哪种索引方法以及如何创建这样的索引，请参阅 LDAP 服务器文档。

映射表项以外的 NIS+ 对象

可以将表项以外的 NIS+ 对象存储在 LDAP 中。但是，除非您还拥有可从 LDAP 获取这些 NIS+ 对象的 NIS+ 副本服务器，否则这样做没有多大意义。建议使用以下选项：

- **没有副本服务器，或者副本服务器仅从 NIS+ 主服务器获取其数据。**

编辑映射配置文件（请参见 [NIS+LDAPmapping\(4\)](#)）以删除所有非表项对象的下列属性值。

```
nisplusLDAPdatabaseIdMapping
nisplusLDAPentryTtl
nisplusLDAPobjectDN
```

例如，如果是从 /var/nis/NIS+LDAPmapping.template 文件开始操作的，则需要删除（或通过加注释来禁用）以下部分：

```
# Standard NIS+ directories
nisplusLDAPdatabaseIdMapping    basedir:
.
.
.

nisplusLDAPdatabaseIdMapping    user_attr_table:user_attr.org_dir
nisplusLDAPdatabaseIdMapping    audit_user_table:audit_user.org_dir

# Standard NIS+ directories
nisplusLDAPentryTtl             basedir:21600:43200:43200
.
.
.

nisplusLDAPentryTtl             user_attr_table:21600:43200:43200
nisplusLDAPentryTtl             audit_user_table:21600:43200:43200
```



```
# Standard NIS+ directories
nisplusLDAPobjectDN    basedir:cn=basedir,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer:\
                        cn=basedir,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer,\
                        objectClass=top
.
.
.
nisplusLDAPobjectDN    audit_user_table:cn=audit_user,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer:\
                        cn=audit_user,ou=nisPlus,?base?\
                        objectClass=nisplusObjectContainer,\
                        objectClass=top
```

- **NIS+ 副本服务器从 LDAP 服务器获取其数据。**

创建 `nisplusObject` 属性和 `nisplusObjectContainer` 对象类，如以下示例中所示（LDIF 数据适用于 `ldapadd(1)`。属性和对象类 OID 仅用于说明。）

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.1.0 NAME 'nisplusObject'
                  DESC 'An opaque representation of an NIS+ object'
                  SYNTAX 1.3.6.1.4.1.1466.115.121.1.5 SINGLE-VALUE )

dn: cn=schema
changetype: modify
add: objectclasses
objectclasses: (1.3.6.1.4.1.42.2.27.5.42.42.2.0 NAME 'nisplusObjectContainer'
                SUP top STRUCTURAL DESC 'Abstraction of an NIS+ object'
                MUST ( cn $ nisplusObject ) )
```

还需要为 NIS+ 对象创建容器。以下 LDIF 语法说明如何创建 `ou=nisPlus,dc=some,dc=domain` 容器，此语法可用作 `ldapadd(1)` 的输入。

```
dn: ou=nisPlus,dc=some,dc=domain
ou: nisPlus
objectClass: top
objectClass: organizationalUnit
```

NIS+ 项的所有者、组、访问权限和 TTL

如果 NIS+ 表项是从 LDAP 数据创建的，则缺省行为是使用项对象所在表对象中的所有者、组、访问权限和 TTL 来初始化项对象的对应值。这通常是足够的，但也可能会遇到必须单独设置这些 NIS+ 项属性的情况。例如，当站点不使用 `rpc.nispasswd(1M)` 守护进程时。为了允许个别用户更改其 NIS+ 口令（并重新加密存储在 `cred.org_dir` 表中的 Diffie-Hellman 密钥），该用户应当拥有自己的 `passwd.org_dir` 和 `cred.org_dir` 项，而且必须对这些项拥有与项所有者相同的修改权限。

如果您需要将一个或多个 NIS+ 表中表项的所有者、组、访问权限或 TTL 存储在 LDAP 中，则需要执行以下操作。

▼ 如何将其他项属性存储到 LDAP 中

- 1 请查阅 LDAP 服务器文档，并新建下列属性和对象类。（LDIF 数据适用于 `ldapadd`。属性和对象类 OID 仅用于说明。）

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.4.0 NAME 'nisplusEntryOwner' \
DESC 'Opaque representation of NIS+ entry owner' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.4.1 NAME 'nisplusEntryGroup' \
DESC 'Opaque representation of NIS+ entry group' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.4.2 NAME 'nisplusEntryAccess' \
DESC 'Opaque representation of NIS+ entry access' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.4.3 NAME 'nisplusEntryTtl' \
DESC 'Opaque representation of NIS+ entry TTL' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
```

```
dn: cn=schema
changetype: modify
add: objectclasses
```

```
objectclasses:(1.3.6.1.4.1.42.2.27.5.42.42.5.0 NAME 'nisplusEntryData'\
SUP top STRUCTURAL DESC 'NIS+ entry object non-column data'\
```

```
MUST ( cn ) MAY ( nisplusEntryOwner $ nisplusEntryGroup $ \
nisplusEntryAccess $ nisplusEntryTtl ) )
```

- 2 修改相关表的 `nisplusLDAPObjectDN` 属性值，以使写入部分包括新创建的 `nisplusEntryData` 对象类。

例如，对于 `passwd.org_dir` 表，假定使用的是基于 `/var/nis/NIS+LDAPmapping.template` 的映射文件，按以下方式进行编辑：

```
nisplusLDAPObjectDN    passwd:ou=People,?one?objectClass=shadowAccount,\

                        objectClass=posixAccount:\
                        ou=People,?one?objectClass=shadowAccount,\
                        objectClass=posixAccount,\
                        objectClass=account,objectClass=top
```

按如下方式编辑属性值：

```
nisplusLDAPObjectDN    passwd:ou=People,?one?objectClass=shadowAccount,\

                        objectClass=posixAccount:\
                        ou=People,?one?objectClass=shadowAccount,\
                        objectClass=posixAccount,\
                        objectClass=nisplusEntryData,\
                        objectClass=account,objectClass=top
```

3 编辑 nisplusLDAPAttributeFromColumn 和 nisplusLDAPcolumnFromAttribute 属性值，以指定所需的所有者、组、访问权限或TTL的任何子集。

在步骤2中，您创建了用于存储这些值的LDAP属性。对于NIS+，存在几个名称分别为zo_owner、zo_group、zo_access和zo_ttl的预定义伪列。例如，为了将passwd.org_dir项的所有者、组和访问权限存储在LDAP中，请修改以下内容中的nisplusLDAPAttributeFromColumn值：

```
nisplusLDAPAttributeFromColumn \
    passwd:          dn=("uid=%s", name), \
                    cn=name, \
                    uid=name, \
                    userPassword="{crypt$}%s", passwd), \
                    uidNumber=uid, \
                    gidNumber=gid, \
                    gecos=gecos, \
                    homeDirectory=home, \
                    loginShell=shell, \
                    (shadowLastChange,shadowMin,shadowMax, \
                     shadowWarning, shadowInactive,shadowExpire)=\
                     (shadow, ":")
```

按以下方式编辑读取部分：

```
nisplusLDAPAttributeFromColumn \
    passwd:          dn=("uid=%s", name), \
                    cn=name, \
                    uid=name, \
                    userPassword="{crypt$}%s", passwd), \
                    uidNumber=uid, \
                    gidNumber=gid, \
                    gecos=gecos, \
                    homeDirectory=home, \
                    loginShell=shell, \
                    (shadowLastChange,shadowMin,shadowMax, \
                     shadowWarning, shadowInactive,shadowExpire)=\
                     (shadow, ":"), \
                    nisplusEntryOwner=zo_owner, \
                    nisplusEntryGroup=zo_group, \
                    nisplusEntryAccess=zo_access
```

同样，要从LDAP数据中为passwd.org_dir表设置NIS+项的所有者、组和访问权限，请修改以下内容：

```
nisplusLDAPcolumnFromAttribute \
    passwd:          name=uid, \
                    ("{crypt$}%s", passwd)=userPassword, \
                    uid=uidNumber, \
                    gid=gidNumber, \
                    gecos=gecos, \
                    home=homeDirectory, \
                    shell=loginShell, \
                    shadow=("%s:%s:%s:%s:%s:%s", \
                             shadowLastChange, \
                             shadowMin, \
                             shadowMax, \
                             shadowWarning, \
```

```
shadowInactive, \
shadowExpire)
```

按以下方式编辑读取部分：

```
nisplusLDAPcolumnFromAttribute \
passwd:      name=uid, \
             ("crypt%s", passwd)=authPassword, \
             uid=uidNumber, \
             gid=gidNumber, \
             gcos=gecos, \
             home=homeDirectory, \
             shell=loginShell, \
             shadow=("%s:%s:%s:%s:%s:%s", \
                     shadowLastChange, \
                     shadowMin, \
                     shadowMax, \
                     shadowWarning, \
                     shadowInactive, \
                     shadowExpire), \
             zo_owner=nisplusEntryOwner, \
             zo_group=nisplusEntryGroup, \
             zo_access=nisplusEntryAccess
```

- 4 将所有者、组、访问权限和/或 TTL 项数据上载到 LDAP 中。

有关更多信息，请参见第 257 页中的“[如何通过一步操作将所有的 NIS+ 数据转换为 LDAP](#)”。

- 5 重新启动 NIS+ 服务，以便使对映射进行的更改生效。

```
# svcadm restart network/rpc/nisplus:default
```

主体名和网络名（从 NIS+ 转换为 LDAP）

NIS+ 验证依赖主体名（由域名限定的用户名或主机名）和网络名（主体名的安全 RPC 等效名称）来唯一地标识可以进行验证的实体（主体）。尽管 RFC 2307 提供用于存储 NIS+ 验证所用 Diffie-Hellman 密钥的位置，却没有为主体名或网络名指定任何位置。

`/var/nis/NIS+LDAPmapping.template` 文件可通过从 `cred.org_dir` 表的所有者名称（本身就是主体名称）派生主体和网络名的域部分来解决此问题。因此，如果 NIS+ 域是 `x.y.z.`，`cred.org_dir` 表的所有者是 `aaa.x.y.z.`，那么，从 LDAP 数据创建的 NIS+ 项的所有主体名将采用以下形式：

user or system.x.y.z.

网络名采用以下形式：

unix.uid@x.y.z.

unix.nodename@x.y.z.

虽然这种构造主体和网络名的方法可能足以满足大多数 NIS+ 安装要求，但在某些情况下，该方法也会失败，如下所示：

- `cred.org_dir` 表的所有者名称不属于由 `cred.org_dir` 表中的主体和网络名共享的域。对于子域中的 `cred.org_dir` 表，如果所有者是来自父域的主体，可能会出现这种情况。此问题可通过以下方法之一来解决：

- 更改 `cred.org_dir` 表的所有者，使其与该表中项的域一致。
- 更改 `cred.org_dir` 数据库 ID 的映射规则，以使用其他某个 NIS+ 对象（如果不存在合适的对象，可能会为此目的专门创建一个）的所有者。

例如，如果 `<literal>sub.dom.ain.</literal>` 域中的 `<filename>cred.org_dir</filename>` 表由 `<literal>master.dom.ain.</literal>` 拥有，但是 `<literal>cred.org_dir.sub.dom.ain.</literal>` 中的主体和网络名应当属于 `<literal>sub.dom.ain</literal>`，则可以按以下方式创建一个链接对象：

```
# nislnc cred.org_dir.sub.dom.ain. \
credname.sub.dom.ain.
```

按以下方式将链接对象的所有者设置为 `sub.dom.ain.` 中的相应主体：

```
# nischown trusted.sub.dom.ain. credname.sub.dom.ain.
```

编辑映射文件。将以下内容：

```
(nis+:zo_owner[]cred.org_dir, "%.s"), \
```

更改为：

```
(nis+:zo_owner[]credname.sub.dom.ain., "%.s"), \
```

请注意，名为 `credname` 的链接对象仅用于举例说明，可以使用任何有效的对象类型（项对象除外）和对象名称。要点是设置对象的所有者，使其拥有正确的域名。

- 如果不想将所有权（甚至特殊用途对象的所有权）交给来自主体和网络名所用域中的主体，请按照以下详细说明来创建 `nisplusPrincipalName` 和 `nisplusNetname` 属性。
- `cred.org_dir` 表中包含属于多个域的主体和网络名。

请查阅 LDAP 服务器的文档，并创建 `nisplusPrincipalName` 和 `nisplusNetname` 属性以及 `nisplusAuthName` 对象类。（以下是 `ldapadd` 的 LDIF 数据。属性和对象类 OID 仅用于说明。）

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.7.0 NAME 'nisplusPrincipalName' \
DESC 'NIS+ principal name' \
SINGLE-VALUE \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.9.0 NAME 'nisplusNetname' \
DESC 'Secure RPC netname' \
SINGLE-VALUE \
```

```
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
```

```
dn: cn=schema
changetype: modify
add: objectclasses
objectclasses: ( 1.3.6.1.4.1.42.2.27.5.42.42.10.0 NAME 'nisplusAuthName' \
                SUP top AUXILIARY DESC 'NIS+ authentication identifiers' \
                MAY ( nisplusPrincipalName $ nisplusNetname ) )
```

现在, 需要启用 `cred.org_dir` 映射, 以便使用新创建的 `nisplusNetname` 和 `nisplusPrincipalName` 属性。模板映射文件 `/var/nis/NIS+LDAPmapping.template` 中包含出于此目的注释掉的行。请参见 `credlocal`、`creduser` 和 `crednode` 数据库 ID 的 `nisplusObjectDN` 和 `nisplusLDAPattributeFromColumn/nisplusLDAPcolumnFromAttribute` 属性值。在对映射文件进行编辑并获得此效果之后, 重新启动 NIS+ 服务。

client_info 和 timezone 表 (从 NIS+ 转换为 LDAP)

由于 RFC 2307 不为保存在 NIS+ `client_info.org_dir` 和 `timezone.org_dir` 表中的信息提供架构, 因此在缺省情况下, 不会在模板映射文件 (`/var/nis/NIS+LDAPmapping.template`) 中启用对这些表的映射。如果您希望将 `client_info` 和 `timezone` 信息保存到 LDAP 中, 请查阅 LDAP 服务器文档, 并新建以下各节中讨论的属性和对象类。

client_info 属性和对象类

按如下所示创建属性和对象类, 然后为 `client_info` 数据创建容器。建议的容器名称是 `ou=ClientInfo`。LDIF 数据适用于 `ldapadd(1)`。以下内容中使用的属性和对象类 OID 仅用于举例说明。

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.12.0 \
                  NAME 'nisplusClientInfoAttr' \
                  DESC 'NIS+ client_info table client column' \
                  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.12.1 \
                  NAME 'nisplusClientInfoInfo' \
                  DESC 'NIS+ client_info table info column' \
                  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.12.2 \
                  NAME 'nisplusClientInfoFlags' \
                  DESC 'NIS+ client_info table flags column' \
                  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )

dn: cn=schema
changetype: modify
add: objectclasses
```

```
objectclasses: ( 1.3.6.1.4.1.42.2.27.5.42.42.13.0 \
    NAME 'nisplusClientInfoData' \
    DESC 'NIS+ client_info table data' \
    SUP top STRUCTURAL MUST ( cn ) \
    MAY ( nisplusClientInfoAttr $ nisplusClientInfoInfo $ nisplusClientInfoFlags ) )
```

要创建容器，请将以下 LDIF 数据放在一个文件中，并用实际的搜索库替换 *searchBase*。

```
dn: ou=ClientInfo, searchBase
```

```
objectClass: organizationalUnit
```

```
ou: ClientInfo
```

```
objectClass: top
```

使用以上文件作为 `ldapadd` 命令的输入，以创建 `ou=ClientInfo` 容器。例如，如果 LDAP 管理员 DN 是 `cn=directory manager`，而且包含 LDIF 数据的文件名为 `cifile`，请执行以下操作：

```
# ldapadd -D cn="directory manager" -f cifile
```

根据必需的验证方法，`ldapadd` 命令可能会提示您输入口令。

`/var/nis/NIS+LDAPmapping.template` 文件中包含 `client_info.org_dir` 表的已注释掉的定义。请将这些内容复制到实际的映射文件，删除注释字符“`#`”来启用它们，然后重新启动 `rpc.nisd` 守护进程。

```
# svcadm restart network/rpc/nisplus:default
```

如有必要，请按照第 256 页中的“NIS+ 到 LDAP 迁移方案”中的说明同步 NIS+ 和 LDAP 数据。

timezone 属性和对象类

按如下所示创建属性和对象类，然后为 `timezone` 数据创建容器。建议的容器名称是 `ou=Timezone`。（LDIF 数据适用于 `ldapadd(1)`。属性和对象类 OID 仅用于举例说明。）

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.15.0 NAME 'nisplusTimeZone' \
    DESC 'tzone column from NIS+ timezone table' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )

dn: cn=schema
changetype: modify
add: objectclasses
objectclasses: ( 1.3.6.1.4.1.42.2.27.5.42.42.16.0 NAME 'nisplusTimeZoneData' \
```

```
DESC 'NIS+ timezone table data' \
SUP top STRUCTURAL MUST ( cn ) \
MAY ( nisplusTimeZone $ description ) )
```

要创建 `ou=Timezone` 容器，请将以下 LDIF 数据放在一个文件中，并用实际的搜索库替换 `searchBase`。

```
dn: ou=Timezone,searchBase ou: Timezone objectClass: top
```

```
objectClass: organizationalUnit
```

使用以上文件作为 `ldapadd(1)` 的输入，以创建 `ou=Timezone` 容器。例如，如果 LDAP 管理员 DN 是 `cn=directory manager`，而且包含 LDIF 数据的文件名为 `tzfile`。

```
# ldapadd -D cn="directory manager" -f tzfile
```

根据必需的验证方法，`ldapadd` 命令可能会提示您输入口令。

`/var/nis/NIS+LDAPmapping.template` 文件中包含 `timezone.org_dir` 表的已注释掉的定义。请将这些内容复制到实际的映射文件，删除注释字符 `"#"` 来启用它们，然后重新启动 `rpc.nisd` 守护进程。

```
# svcadm restart network/rpc/nisplus:default
```

如有必要，请按照第 256 页中的“NIS+ 到 LDAP 迁移方案”中的说明同步 NIS+ 和 LDAP 数据。

添加新的对象映射（从 NIS+ 转换为 LDAP）

模板映射文件 `/var/nis/NIS+LDAPmapping.template` 中包含所有标准 NIS+ 对象的映射信息。为了支持对特定于站点或应用程序的对象进行映射，您需要添加新的映射项。对于非项（即，目录、组、链接或表）对象来说，此任务非常简单，但是对于项对象来说，如果对应项数据的 LDAP 组织方式与 NIS+ 所使用的组织方式有很大区别，则此任务会变得较为复杂。以下示例说明非项对象的简单情况。

▼ 如何映射非项对象

1 查找要映射的对象的全限定名称。

如果此名称位于由 `nisplusLDAPbaseDomain` 属性指定的域名之下，您可以省略等于 `nisplusLDAPbaseDomain` 值的部分。

例如，如果 `<literal>nisplusLDAPbaseDomain</literal>` 的值为 `<literal>some.domain.</literal>`，要映射的对象是名为 `<literal>nodeinfo.some.domain.</literal>` 的表，则对象名可以缩短为 `<literal>nodeinfo</literal>`。

2 创建用于标识对象的数据库 ID。

数据库 ID 对于所使用的映射配置必须唯一，而没有进行其他方面的解释。它不显示在 LDAP 数据中。为了避免与项对象映射混淆，请创建一个用于标识表对象属性（而非表项）的数据库 ID 并在末尾处使用解释性字符串（如 `_table`）。

在本示例中，使用的是数据库 ID `nodeinfo_table`，并通过添加以下内容在该数据库 ID 和标准映射文件位置（`/var/nis/NIS+LDAPmapping`）中的对象之间建立连接：

```
nisplusLDAPdatabaseIdMapping    nodeinfo_table:nodeinfo.some.domain.
```

假定 `<literal>nisplusLDAPbaseDomain</literal>` 为 `<literal>some.domain.</literal>`，则以下内容也有效：

```
nisplusLDAPdatabaseIdMapping    nodeinfo_table:nodeinfo
```

3 确定适合该对象的 TTL。

在这段时间内，`rpc.nisd` 守护进程会将该对象的本地副本视为有效。在 TTL 过期之后，再次引用该对象会启动 LDAP 查找功能以刷新该对象。

有两个不同的 TTL 值：第一个值是在 `rpc.nisd` 守护进程（在重新引导或重新启动之后）首次从磁盘中装入该对象时设置的，第二个值与从 LDAP 执行的所有刷新有关。第一个 TTL 是从所配置的范围中随机选择的。例如，如果 `nodeinfo_table` 应当在初始装入之后的一到三个小时内以及此后的十二个小时内有效，则需要指定如下内容：

```
nisplusLDAPentryTtl            nodeinfo_table:3600:10800:43200
```

4 确定对象数据在 LDAP 中的存储位置。

模板映射文件建议将非项对象数据放到 `ou=nisPlus` 容器中。

如果您使用此方案，而且尚未创建相应的属性、对象类和容器，请参见第 264 页中的“映射表项以外的 NIS+ 对象”。

例如，假定您希望将 `nodeinfo` 对象存储到 `ou=nisPlus,dc=some,dc=domain` 容器中，而且 LDAP 项应当具有 `cn nodeinfo`。请创建以下 `nisplusLDAPobjectDN`：

```
nisplusLDAPobjectDN    nodeinfo_table:\
                        cn=nodeinfo,ou=nisPlus,dc=some,dc=domain?base?\
                        objectClass=nisplusObjectContainer:\
                        cn=nodeinfo,ou=nisPlus,dc=some,dc=domain?base?\
                        objectClass=nisplusObjectContainer,\
                        objectClass=top
```

由于 NIS+ 副本服务器不将数据写入 LDAP 中，因此您可以为主服务器和副本服务器使用以上 `nisplusLDAPobjectDN`。

- 5 （如果尚未在 NIS+ 中创建要映射的 NIS+ 对象，请跳过此步骤。）将对象数据存储到 LDAP 中。您可以使用 `rpc.nisd` 守护进程来存储对象数据，但是使用 `nisldapmaptest(1M)` 实用程序会更方便，因为这样可以使 `rpc.nisd` 守护进程保持运行状态。

```
# nisldapmaptest -m /var/nis/NIS+LDAPmapping -o -t nodeinfo -r
```

-o 选项指定表对象本身（而非表项）。

- 6 检验对象数据是否存储在 LDAP 中。（本示例假定 LDAP 服务器在本地计算机的端口 389 上运行。

```
# ldapsearch -b ou=nisPlus,dc=some,dc=domain cn=nodeinfo
```

将出现类似以下的输出：

```
dn: cn=nodeinfo,ou=nisPlus,dc=some,dc=domain
objectclass: nisplusObjectContainer
objectclass: top
cn: nodeinfo
nisplusobject=<base 64 encoded data>
```

添加项对象

[NIS+LDAPmapping\(4\)](#) 详细指定了表项映射的语法和语义，还举例说明了如何使用每个语义元素。但是，最简单且最不易出错的方法通常是标识与所需映射相似的现有映射，然后复制并修改这个现有映射。

例如，假定您有一个名为 `nodeinfo` 的 NIS+ 表，该表用于存储节点的清单和所有者信息。假定该 NIS+ 表是通过以下命令创建的。

```
# nistbladm -c -D access=og=rmcd,nw=r -s : nodeinfo_tbl \
cname=S inventory=S owner= nodeinfo.'domainname'.
```

`cname` 列中应当包含节点的标准名称，换言之，与节点的 `hosts.org_dir` 表中 `cname` 列的值相同。

同时还假定：对应的信息保存在 LDAP 中的 `ou=Hosts` 容器中，`nodeInfo` 对象类（在本示例中为新创建的对象类，而不是在 RFC 中定义的对象类）具有 `cn` 作为 MUST 属性，`nodeInventory` 和 `nodeOwner` 是 MAY 属性。

为了将现有的 `nodeinfo` 数据上载到 LDAP 中，在单独的文件中新建映射属性将会很方便。例如，可以使用 `/var/nis/tmpmapping`。

1. 创建一个数据库 ID 来标识要映射的 NIS+ 表。

```
nisplusLDAPdatabaseIdMapping    nodeinfo:nodeinfo
```

2. 为 `nodeinfo` 表中的项设置 TTL。由于极少对这些信息进行更改，因此请将 TTL 设置为十二个小时。当 `rpc.nisd` 守护进程首次从磁盘中加载 `nodeinfo` 表时，该表中项的 TTL 将随机选择为六到十二个小时之间的值。

```
nisplusLDAPentryTtl             nodeinfo:21600:43200:43200
```

3. 标识与要创建的映射具有相似属性的现有映射。在本示例中，映射属性值非常简单（直接赋值），而将 LDAP 数据存储到现有容器中则较为复杂，因此，在删除 `nodeinfo` 数据的过程中一定要格外小心。如果不希望删除整个 `ou=Hosts` 项，而只希望删除 `nodeInventory` 和 `nodeOwner` 属性，则将需要一个特殊的删除规则集合。

总之，就是要查找一个既共享某个容器、又具有删除规则集合的映射。一个可能的备选项就是 `netmasks` 映射，该映射共享 `ou=Networks` 容器，而且确实具有一个删除规则集合。

4. 模板映射 `netmasks` 具有缺省映射（位于 `/var/nis/NIS+LDAPmapping.template` 中），如下所示：

```
nisplusLDAPObjectDN    netmasks:ou=Networks,?one?objectClass=ipNetwork,\
                        ipNetMaskNumber=*\
                        ou=Networks,?one?objectClass=ipNetwork:\
                        dbid=netmasks_del
```

在转移到新的 `nodeinfo` 映射之后，数据库 ID 应当为 `nodeinfo`，容器应当为 `ou=Hosts`，对象类应当为 `nodeInfo`。因此，`nodeinfo` 映射的第一行将变为：

```
nisplusLDAPObjectDN    nodeinfo:ou=Hosts,?one?objectClass=nodeInfo,\
```

`netmasks` 映射的第二行是搜索过滤器的一部分，它只选择那些包含 `ipNetMaskNumber` 属性的 `ou=Networks` 项。在本示例中，它选择那些具有以下 `nodeInventory` 属性的 `ou=Hosts` 项：

```
nodeInventory=*\
```

第三行和第四行是 `nisplusLDAPObjectDN` 的写入部分，它们指定 `nodeinfo` 数据在 LDAP 中的写入位置以及在删除 `nodeinfo` 数据时所使用的规则集合。在本例中，创建一个由数据库 ID `nodeinfo_del` 标识的删除规则集合。因为您总是写入 `ou=Hosts` 中的现有项，所以只需要为 `nodeinfo` 数据属性指定对象类，如下所示：

```
ou=Hosts,?one?objectClass=nodeInfo:\
dbid=nodeinfo_del
```

将所有这些汇总到一起，`nisplusLDAPObjectDN` 将如下所示：

```
nisplusLDAPObjectDN    nodeinfo:ou=Hosts,?one?objectClass=nodeInfo,\
                        nodeInventory=*\
                        ou=Hosts,?one?objectClass=nodeInfo:\
                        dbid=nodeinfo_del
```

5. 创建一个将 `nodeinfo` 数据从 NIS+ 映射到 LDAP 的规则集合。模板（来自 `netmasks`）如下所示：

```
nisplusLDAPAttributeFromColumn \
netmasks:    dn=("ipNetworkNumber=%s", addr), \
              ipNetworkNumber=addr, \
              ipNetmaskNumber=mask, \
              description=comment
```

在本例中，`ou=Hosts` 容器会使情况更复杂，因为 RFC 2307 中规定 `dn` 中应当包含 IP 地址。但是，IP 地址不存储在 `nodeinfo` 表中，因此您必须以其他方式获取它。幸运的是，模板文件中的 `crednode` 映射说明了如何获取 IP 地址。

```
nisplusLDAPAttributeFromColumn \
crednode:    dn=("cn=%s+ipHostNumber=%s", \
                (cname, "%s.*"), \
                ldap:ipHostNumber:?one?("cn=%s", (cname, "%s.*"))), \
```

因此，您可以复制 crednode 映射的这一部分。但是，在本例中，cname 列的值是实际的主机名（而非主体名），因此，您不必单独提取 cname 的某个部分，而是可以明确替换属性名和列名，此时 nodeinfo 映射将变为：

```
nisplusLDAPAttributeFromColumn \
nodeinfo:    dn="cn=%s+ipHostNumber=%s,", cname, \
             ldap:ipHostNumber:?one?("cn=%s", cname)), \
             nodeInventory=inventory, \
             nodeOwner=owner
```

6. 将数据从 LDAP 映射到 NIS+ 时，模板的 netmasks 项如下所示：

```
nisplusLDAPcolumnFromAttribute \
netmasks:    addr=ipNetworkNumber, \
             mask=ipNetmaskNumber, \
             comment=description
```

在替换属性名和列名之后，结果如下所示：

```
nisplusLDAPcolumnFromAttribute \
nodeinfo:    cname=cn, \
             inventory=nodeInventory, \
             owner=nodeOwner
```

7. netmasks 的删除规则集合如下所示：

```
nisplusLDAPAttributeFromColumn \
netmasks_del: dn=("ipNetworkNumber=%s,", addr), \
              ipNetmaskNumber=
```

以上规则集合指定当 NIS+ 中的 netmasks 项被删除时，对应的 ou=Networks LDAP 项中的 ipNetmaskNumber 属性也会被删除。在本例中，删除的是 nodeInventory 和 nodeOwner 属性。因此，使用步骤 5 中的 dn 规范时，结果如下所示：

```
nisplusLDAPAttributeFromColumn \
nodeinfo_del: dn=("cn=%s+ipHostNumber=%s,", cname, \
                 ldap:ipHostNumber:?one?("cn=%s", cname)), \
                 nodeInventory=, \
                 nodeOwner=
```

映射信息完整无缺。

8. 如果 NIS+ nodeinfo 表中已经有数据，请将这些数据上载到 LDAP 中。将新的 nodeinfo 映射信息放到一个单独的文件 /var/nis/tmpmapping 中。

```
# /usr/sbin/rpc.nisd -D -m /var/nis/tmpmapping \
-x nisplusLDAPinitialUpdateAction=to_ldap \
-x nisplusLDAPinitialUpdateOnly=yes
```

9. 将临时文件 /var/nis/tmpmapping 中的映射信息添加到实际映射文件中。这可以借助于编辑器来完成，也可以按以下方式附加数据（假定实际映射文件为 /var/nis/NIS+LDAPmapping）来完成：

```
# cp -p /var/nis/NIS+LDAPmapping \
/var/nis/NIS+LDAPmapping.backup
# cat /var/nis/tmpmapping >> /var/nis/NIS+LDAPmapping
```



注意 - 请注意双箭头 ">>" 表示重定向。单箭头 ">" 表示将覆写目标文件。

将配置信息存储到 LDAP 中

除了将 NIS+/LDAP 配置信息保存在配置文件中 and 命令行上，还可以将配置属性存储到 LDAP 中。如果配置信息由许多 NIS+ 服务器共享，并且将定期进行更改，则这非常有用。

要在 LDAP 中启用对配置属性的存储，请查阅 LDAP 服务器文档并新建下列属性和对象类。配置信息应当位于由 `nisplusLDAPconfigDN` 值（来自 `rpc.nisd` 命令行或来自 `/lib/svc/method/nisplus`）指定的位置中，而且 `cn` 等于 `nisplusLDAPbaseDomain` 值（因为 `rpc.nisd` 守护进程从 LDAP 读取任何配置信息之前，就已经获知了该值）。

LDIF 数据适用于 `ldapadd(1)`（属性和对象类 OID 仅用于举例说明）。

`defaultSearchBase`、`preferredServerList` 和 `authenticationMethod` 属性是从“DUA 配置”草稿架构（将成为 IETF 标准）派生的。在任何情况下，以下定义都可以满足 `NIS+LDAPmapping(4)` 的要求：

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.11.1.3.1.1.1 NAME 'defaultSearchBase' \
DESC 'Default LDAP base DN used by a DUA' \
EQUALITY distinguishedNameMatch \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.11.1.3.1.1.2 NAME 'preferredServerList' \
DESC 'Preferred LDAP server host addresses to be used by a DUA' \
EQUALITY caseIgnoreMatch \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.11.1.3.1.1.6 NAME 'authenticationMethod' \
DESC 'Identifies the authentication method used to connect to the DSA' \
EQUALITY caseIgnoreMatch \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )
```

NIS+/LDAP 配置属性如下所示：

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.0 \
NAME 'nisplusLDAPTLS' \
DESC 'Transport Layer Security' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.1 \
NAME 'nisplusLDAPTLSCertificateDBPath' \
DESC 'Certificate file' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
```

```

attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.2 \
    NAME 'nisplusLDAPproxyUser' \
    DESC 'Proxy user for data store/retrieval' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.3 \
    NAME 'nisplusLDAPproxyPassword' \
    DESC 'Password/key/shared secret for proxy user' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.4 \
    NAME 'nisplusLDAPinitialUpdateAction' \
    DESC 'Type of initial update' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.5 \
    NAME 'nisplusLDAPinitialUpdateOnly' \
    DESC 'Exit after update ?' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.6 \
    NAME 'nisplusLDAPretrieveErrorAction' \
    DESC 'Action following an LDAP search error' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.7 \
    NAME 'nisplusLDAPretrieveErrorAttempts' \
    DESC 'Number of times to retry an LDAP search' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.8 \
    NAME 'nisplusLDAPretrieveErrorTimeout' \
    DESC 'Timeout between each search attempt' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.9 \
    NAME 'nisplusLDAPstoreErrorAction' \
    DESC 'Action following an LDAP store error' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.10 \
    NAME 'nisplusLDAPstoreErrorAttempts' \
    DESC 'Number of times to retry an LDAP store' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.11 \
    NAME 'nisplusLDAPstoreErrorTimeout' \
    DESC 'Timeout between each store attempt' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.12 \
    NAME 'nisplusLDAPrefreshErrorAction' \
    DESC 'Action when refresh of NIS+ data from LDAP fails' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.13 \
    NAME 'nisplusLDAPrefreshErrorAttempts' \
    DESC 'Number of times to retry an LDAP refresh' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.14 \
    NAME 'nisplusLDAPrefreshErrorTimeout' \
    DESC 'Timeout between each refresh attempt' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.15 \
    NAME 'nisplusNumberOfServiceThreads' \
    DESC 'Max number of RPC service threads' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.16 \
    NAME 'nisplusThreadCreationErrorAction' \
    DESC 'Action when a non-RPC-service thread creation fails' \

```

```

        SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.17 \
    NAME 'nisplusThreadCreationErrorAttempts' \
    DESC 'Number of times to retry thread creation' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.18 \
    NAME 'nisplusThreadCreationErrorTimeout' \
    DESC 'Timeout between each thread creation attempt' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.19 \
    NAME 'nisplusDumpErrorAction' \
    DESC 'Action when an NIS+ dump fails' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.20 \
    NAME 'nisplusDumpErrorAttempts' \
    DESC 'Number of times to retry a failed dump' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.21 \
    NAME 'nisplusDumpErrorTimeout' \
    DESC 'Timeout between each dump attempt' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.22 \
    NAME 'nisplusResyncService' \
    DESC 'Service provided during a resync' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.23 \
    NAME 'nisplusUpdateBatching' \
    DESC 'Method for batching updates on master' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.24 \
    NAME 'nisplusUpdateBatchingTimeout' \
    DESC 'Minimum time to wait before pinging replicas' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.25 \
    NAME 'nisplusLDAPmatchFetchAction' \
    DESC 'Should pre-fetch be done ?' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.26 \
    NAME 'nisplusLDAPbaseDomain' \
    DESC 'Default domain name used in NIS+/LDAP mapping' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 SINGLE-VALUE )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.27 \
    NAME 'nisplusLDAPdatabaseIdMapping' \
    DESC 'Defines a database id for an NIS+ object' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.28 \
    NAME 'nisplusLDAPentryTtl' \
    DESC 'TTL for cached objects derived from LDAP' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.29 \
    NAME 'nisplusLDAPobjectDN' \
    DESC 'Location in LDAP tree where NIS+ data is stored' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.30 \
    NAME 'nisplusLDAPcolumnFromAttribute' \
    DESC 'Rules for mapping LDAP attributes to NIS+ columns' \
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
attributetypes: ( 1.3.6.1.4.1.42.2.27.5.42.42.18.31 \
    NAME 'nisplusLDAPattributeFromColumn' \

```

```
DESC 'Rules for mapping NIS+ columns to LDAP attributes' \
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

dn: cn=schema
changetype: modify
add: objectclasses
objectclasses: ( 1.3.6.1.4.1.42.2.27.5.42.42.19.0 NAME 'nisplusLDAPconfig' \
DESC 'NIS+/LDAP mapping configuration' \
SUP top STRUCTURAL MUST ( cn ) \
MAY ( preferredServerList $ defaultSearchBase $
authenticationMethod $ nisplusLDAPTLS $ nisplusLDAPTLSCertificateDBPate
$ nisplusLDAPproxyUser $ nisplusLDAPproxyPassword $ nisplusLDAPinitialUpdateAction
$ nisplusLDAPinitialUpdateOnly $ nisplusLDAPretrieveErrorAction
$ nisplusLDAPretrieveErrorAttempts $ nisplusLDAPretrieveErrorTimeout
$ nisplusLDAPstoreErrorAction $ nisplusLDAPstoreErrorAttempts
$ nisplusLDAPstoreErrorTimeout $ nisplusLDAPrefreshErrorAction
$ nisplusLDAPrefreshErrorAttempts $ nisplusLDAPrefreshErrorTimeout
$ nisplusNumberOfServiceThreads $nisplusThreadCreationErrorAction
$ nisplusThreadCreationErrorAttempts $ nisplusThreadCreationErrorTimeout
$ nisplusDumpErrorAction $ nisplusDumpErrorAttempts
$ nisplusDumpErrorTimeout $ nisplusResyncService $ nisplusUpdateBatching
$ nisplusUpdateBatchingTimeout $ nisplusLDAPmatchFetchAction
$ nisplusLDAPbaseDomain $ nisplusLDAPdatabaseIdMapping $ nisplusLDAPentryTtl
$ nisplusLDAPobjectDN $ nisplusLDAPcolumnFromAttribute !
$ nisplusLDAPattributeFromColumn ) )
```

创建一个包含以下 LDIF 数据的文件（并用实际的搜索库替换 *searchBase*，用全限定域名替换 *domain*）。

```
dn: cn=domain,searchBase

cn: domain

objectClass: top objectClass: nisplusLDAPconfig
```

使用以上文件作为 `ldapadd(1)` 的输入，以创建 NIS+/LDAP 配置项。该项最初为空。使用 `ldapmodify(1)` 添加配置属性。例如，要将 `nisplusNumberOfServiceThreads` 属性设置为 "32"，请创建以下文件（用作 `ldapmodify(1)` 的输入）：

```
dn: cn=domain, searchBase nisplusNumberOfServiceThreads: 32
```


Solaris 10 软件中对 DNS、NIS 和 LDAP 的更新

系统管理指南：命名和目录服务（DNS、NIS 和 LDAP）的 Solaris 10 版本包括对 DNS BIND 和 `pam_ldap` 的更新。它还包含一些对其他内容的些许改动和补充，以及对几个文档错误的更正。

服务管理工具的更改

DNS、NIS 和 LDAP 服务现在通过服务管理工具进行管理。使用 `svcadm` 命令可以对这些服务执行启用、禁用或重新启动等管理操作。使用 `svcs` 命令可以查询该服务的状态。有关 SMF 的概述，请参见《系统管理指南：基本管理》中的第 18 章“管理服务（概述）”。有关更多详细信息，另请参阅 `svcadm(1M)` 和 `svcs(1)` 手册页。

特定于本书中所介绍的每个服务的信息可在以下各节中找到：

- 第 49 页中的“DNS 和服务管理工具”
- 第 80 页中的“NIS 和服务管理工具”
- 第 174 页中的“LDAP 和服务管理工具”
- 第 224 页中的“NIS 到 LDAP 转换工具和服务管理工具”
- 第 247 页中的“NIS+ 到 LDAP 转换工具和服务管理工具”

有关 NIS+ 和服务管理工具的信息，请参见《System Administration Guide: Naming and Directory Services (NIS+)》。

DNS BIND

BIND 8.4.2 随 Solaris 10 发行版提供。Solaris 软件上该版本的 BIND 为 IPv6 网络提供了完整的 DNS 客户机/服务器解决方案。在本指南中，未对 DNS BIND 过程进行任何更改。

Solaris 10 发行版中也支持 BIND 9，它安装在 `/usr/sfw` 目录中。迁移文档位于 `/usr/sfw/doc/bind` 目录中。第 2 部分中的信息和过程（在迁移文档中指出的除外）适用于 BIND 9。

pam_ldap 更改

Solaris 10 OS 发行版中引入了几项与 `pam_ldap` 有关的改动，下面的内容将介绍这些改动。有关更多信息，请参见 [pam_ldap\(5\)](#) 手册页。

- 从 Solaris 10 软件发行版开始，以前受支持的 `use_first_pass` 和 `try_first_pass` 选项将被废弃。因为以后不再需要这些选项，所以可以从 `pam.conf` 中安全地删除它们，也可以将其自动忽略。以后的发行版中将不再包括这些选项。
- 必须通过以下方式提供口令提示：先将 `pam_authok_get` 放入验证和口令模块栈中，再将 `pam_ldap` 入栈，并将 `pam_passwd_auth` 放入 `passwd` 服务的 `auth` 栈中。
- 此发行版使用以前推荐使用的、带 `server_policy` 选项的 `pam_authok_store` 来代替以前支持的口令更新功能。
- `pam_ldap` 帐户管理功能增强了 LDAP 名称服务的整体安全性。具体来说，帐户管理功能执行以下操作：
 - 允许跟踪口令的失效和过期
 - 防止用户选择易破解的或以前使用过的口令
 - 如果口令即将过期，可以向用户发出警告
 - 如果多次登录失败，则禁止用户登录
 - 防止除授权的系统管理员以外的用户停用已初始化的帐户

不可能为上面列出的更改提供全新的自动更新。因此，升级到 Solaris 10 或更高发行版时将不会自动更新现有的 `pam.conf` 文件来反映对 `pam_ldap` 所作的更改。如果现有的 `pam.conf` 文件中包含 `pam_ldap` 配置，系统在通过 CLEANUP 文件升级之后将通知您。您需要检查 `pam.conf` 文件并根据需要修改该文件。

有关更多信息，请参见

[pam_passwd_auth\(5\)](#)、[pam_authok_get\(5\)](#)、[pam_authok_store\(5\)](#) 和 [pam.conf\(4\)](#) 手册页。

文档错误

本发行版中更正了几处文档错误。

词汇表

application-level naming service (应用程序级命名服务)	应用程序级命名服务包含在可提供文件、邮件和打印等服务的应用程序中。应用程序级命名服务绑定在企业级命名服务之下。企业级命名服务提供上下文，应用程序级命名服务的上下文可以绑定在该上下文中。
attribute (属性)	每个 LDAP 项都由许多命名属性组成，每个属性都具有一个或多个值。
authentication (验证)	服务器用于验证客户机身份的方法。
baseDN (基本 DN)	作为部分 DIT 的根元素的 DN。如果是 NIS 域项的 baseDN，它又称为上下文。
cache manager (高速缓存管理器)	用于管理 NIS+ 客户机本地高速缓存 (NIS_SHARED_DIRCACHE) 的程序，这些高速缓存用于存储特定 NIS+ 服务器的位置信息 (包括传输地址、验证信息和生存时间值)，这些 NIS+ 服务器支持客户机最常用的目录。
child domain (子域)	请参见 <i>domain</i> (域)。
client-server model (客户机/服务器模型)	描述网络服务并为这些服务的用户进程 (程序) 建模的常见方法。例如，域名系统 (Domain Name System, DNS) 名称服务器/名称解析程序模式。另请参见 <i>client</i> (客户机)。
client (客户机)	(1) 客户机是从命名服务器请求命名服务的主体 (计算机或用户)。 (2) 在用于文件系统的客户机/服务器模型中，客户机是远程访问计算服务器资源 (如计算能力和大容量内存) 的计算机。 (3) 在客户机/服务器模型中，客户机是从“服务器进程”访问服务的 应用程序 。在该模型中，客户机和服务器可以在同一台计算机上运行，也可以在不同的计算机上运行。
credential (凭证)	与客户机软件向命名服务器发出的每个请求一起发送的验证信息。这些信息用于验证用户或计算机的身份。
data encrypting key (数据加密密钥)	用于对数据进行加密和解密的密钥，适用于执行加密的程序。与 密钥加密密钥 相对。
data encryption standard, DES (数据加密标准)	一种极其复杂的常用算法，由美国国家标准局开发，用于对数据进行加密和解密。另请参见 SUN-DES-1。
DBM	DBM (数据库管理) 是一种数据库，最初用于存储 NIS 映射。
decimal dotted notation (点分十进制表示法)	32 位整数的语法表示形式，该整数由四个以 10 进制表示的 8 位数字组成，它们之间用句点 (点) 分隔。用于将 Internet 中的 IP 地址表示为类似于 192.67.67.20 的形式。

DES	请参见 <i>data encryption standard, DES</i> (数据加密标准) 。
directory cache (目录高速缓存)	一个本地文件, 用于存储与目录对象相关联的数据。
directory information tree (目录信息树)	DIT 指给定网络的分布式目录结构。缺省情况下, Solaris LDAP 客户机在访问信息时假设 DIT 具有给定的结构。LDAP 服务器支持的每个域都有一个具有假设结构的假设子树。
directory (目录)	(1) LDAP 目录是 LDAP 对象的容器。在 UNIX 中, 目录是文件和子目录的容器。
distinguished name (标识名)	标识名是 X.500 目录信息库 (directory information base, DIB) 中的项, 由沿根目录直至指定项的路径, 从树中每一项选择的属性组成。
DIT	请参见 directory information tree (目录信息树) 。
DN	LDAP 中的标识名。LDAP 目录的树状结构化寻址方案, 它赋予每个 LDAP 项一个唯一的名称。
DNS	请参见 <i>Domain Name System</i> (域名系统) 。
DNS-forwarding (DNS 转发)	NIS 服务器或设置了 NIS 兼容性的 NIS+ 服务器将它无法应答的请求转发到 DNS 服务器。
DNS zone files (DNS 区域文件)	一组文件, DNS 软件将域中所有工作站的名称和 IP 地址存储在 其中。
DNS zone (DNS 区域)	网络域中的管理范围, 通常由一个或多个子域组成。
domain name (域名)	指定给本地网络上一组共享 DNS 管理文件的系统的名称。必须要有域名, 网络信息服务数据库才能正常工作。另请参见 <i>domain</i> (域) 。
Domain naming service, DNS (域名服务)	一种服务, 所提供的命名策略和机制可将域名和计算机名映射为企业外部地址 (如 Internet 上的地址) 。DNS 是由 Internet 使用的网络信息服务。
domain (域)	(1) 在 NIS+ 中, 域是指由 NIS+ 管理的一组分层对象。最高层的域 (根域) 有一个, 子域可以有多个也可以根本没有。域和子域可以按地理位置、组织或功能原则进行组织。 ■ 父域。相关术语, 指层次结构中紧邻当前域上方的域。 ■ 子域。相关术语, 指层次结构中紧邻当前域下方的域。 ■ 根域。当前 NIS+ 层次结构中最高层的域。 (2) 在 Internet 中, 命名层次结构的某个部分通常与局域网 (Local Area Network, LAN)、广域网 (Wide Area Network, WAN) 或者类似网络的一部分相对应。从语法上来说, Internet 域名由一系列用句点 (点) 分隔的名称 (标签) 组成。例如, <code>sales.doc.com</code>。 (3) 在国际标准化组织的开放系统互连 (open systems interconnection, OSI) 中, “域”通常用作复杂分布式系统的管理分区, 正如在 MHS 专用管理域 (private management domain, PRMD) 和目录管理域 (directory management domain, DMD) 中一样。
encryption key (加密密钥)	请参见 <i>data encrypting key</i> (数据加密密钥) 。
encryption (加密)	用于保护数据保密性的方法。

enterprise-level network (企业级网络)	“企业级”网络可以是通过电缆、红外线或无线广播通信的单个局域网 (Local Area Network, LAN), 也可以是由两个或多个 LAN 组成的群集 (这些 LAN 通过电缆连接在一起或者直接通过电话线连接在一起)。在企业级网络中, 每台计算机都能在不引用全局命名服务 (如 DNS 或 X.500/LDAP) 的情况下与任何其他计算机进行通信。
entry (项)	数据库表中的一行数据, 如 DIT 中的一个 LDAP 元素。
field (字段)	一个 NIS 映射项可以由许多组成部分和分隔符组成。
GID	请参见 <i>group ID</i> (组 ID)。
global naming service (全局命名服务)	全局命名服务标识全球的企业级网络, 这些网络通过电话、卫星或其他通信系统连接在一起。这个连接在一起的全球网络集合称为 "Internet"。除了命名网络, 全局命名服务还可标识给定网络内的单台计算机和单个用户。
group ID (组 ID)	一个数字, 用于标识用户的缺省组。
indexed name (索引名)	用于标识表中的项的命名格式。
Internet address (Internet 地址)	指定给使用 TCP/IP 的主机的 32 位地址。请参见 <i>decimal dotted notation</i> (点分十进制表示法)。
IP	Internet 协议。Internet 协议套件的 网络层 协议。
IP address (IP 地址)	用于标识网络中每台主机的唯一数字。
key (encrypting) (加密密钥)	用于对其他密钥进行加密和解密的密钥, 它是密钥管理和分布系统的一部分。与 <i>data encrypting key</i> (数据加密密钥) 相对。
key server (密钥服务器)	用于存储私钥的 Solaris 操作环境进程。
LDAP	轻量目录访问协议是一种标准的、可扩展的目录访问协议, 它由 LDAP 命名服务客户机和服务器用于进行相互通信。
local-area network, LAN (局域网)	位于同一地理位置的多个系统, 为了共享和交换数据及软件而连接在一起。
mail exchange record (邮件交换记录)	一些文件, 其中包含 DNS 域名及其对应邮件主机的列表。
mail host (邮件主机)	一个工作站, 充当站点的电子邮件路由器和接收器。
mapping (映射)	NIS 项与 DIT 项之间的相互转换过程。此过程由 映射 文件控制。
master server (主服务器)	用于为特定域维护网络信息服务数据库主副本的服务器。名称空间更改总是针对由域的主服务器保存的命名服务数据库进行。每个域都只有一台主服务器。
MIS	管理信息系统 (或服务)。

N2L server (N2L 服务器)	NIS 到 LDAP 转换服务器。已使用 N2L 服务重新配置为 N2L 服务器的 NIS 主服务器。重新配置过程包括替换 NIS 守护进程和添加新配置文件。
name resolution (名称解析)	将工作站名称或用户名转换为地址的过程。
name server (名称服务器)	运行一个或多个网络命名服务的服务器。
namespace (名称空间)	(1) 名称空间存储用户、工作站和应用程序进行网络通信所必需的信息。 (2) 命名系统中所有名称的集合。
naming service switch (命名服务转换器)	一个配置文件 (<code>/etc/nsswitch.conf</code>)，用于定义命名客户机从中获取其网络信息的源。
naming service (命名服务)	用于处理计算机、用户、打印机、域、路由器以及其他网络的名称和地址的网络服务。
NDBM	NDBM (新数据库管理) 是 DBM 的改进版本。
network mask (网络掩码)	一个数字，软件用它将本地子网地址与给定 Internet 协议地址的其余部分分开。
network password (网络口令)	请参见 Secure RPC password (安全 RPC 口令)。
NIS	一种分布式网络信息服务，其中包含有关网络上的系统和用户的关键信息。NIS 数据库存储在 主服务器 和全部 副本服务器 或 从属服务器 上。
NIS+	一种分布式网络信息服务，其中包含有关网络上的系统和用户的层次结构信息。NIS+ 数据库存储在 主服务器 和全部 副本服务器 上。
NIS-compatibility mode (NIS 兼容模式)	NIS+ 的一种配置，借助该配置，NIS 客户机可以访问存储在 NIS+ 表中的数据。在该模式下时，NIS+ 服务器可以同时应答来自 NIS 和 NIS+ 客户机的信息请求。
NIS maps (NIS 映射)	NIS 用于存储特定类型信息 (例如，网络上所有用户的口令项或者网络上所有主机的名称) 的文件。作为 NIS 服务一部分的程序会查询这些映射。另请参见 <i>NIS</i> 。
parent domain (父域)	请参见 <i>domain (域)</i> 。
preferred server list (首选服务器列表)	一个 <code>client_info</code> 表或一个 <code>client_info</code> 文件。首选服务器列表为客户机或域指定首选服务器。
private key (私钥)	以数学方法生成的一对数字的专用部分，在与公钥合并时，可生成 DES 密钥。DES 密钥又可用于对信息进行编码和解码。发件人的私钥只能由密钥的所有者使用。每个用户或每台计算机都有其各自的公钥/私钥对。
public key (公钥)	以数学方法生成的一对数字的公共部分，在与私钥合并时，可生成 DES 密钥。DES 密钥又可用于对信息进行编码和解码。公钥对所有的用户和计算机公开。每个用户或每台计算机都有其各自的公钥/私钥对。

RDN	相对标识名。DN 的一部分。
record (记录)	请参见 <i>entry</i> (项) 。
remote procedure call, RPC (远程过程调用)	一种易于使用的常见模式，用于实现客户机/服务器分布式计算模型。使用所提供的参数向远程系统发送请求，以执行指定的过程，结果将返回到调用者。
reverse resolution (反向解析)	使用 DNS 软件将工作站 IP 地址转换为工作站名称的过程。
RFC 2307	RFC 的一部分，指定将信息从标准 NIS 映射映射到 DIT 项。
root domain (根域)	请参见 <i>domain</i> (域) 。
RPC	请参见 remote procedure call, RPC (远程过程调用) 。
SASL	简单验证和安全层。用于在应用层协议中协商验证和安全层语义的框架。
schema (架构)	一个规则集合，定义了可存储在任何给定 LDAP DIT 中的数据类型。
searchTriple	一种说明，描述从 DIT 中的什么位置查找给定属性。searchTriple 由“基本 DN”、“范围”和“过滤器”组成。这是在 RFC 2255 中定义的 LDAP URL 格式的一部分。
Secure RPC password (安全 RPC 口令)	安全 RPC 协议所需的口令。此口令用于对私钥进行加密。此口令应当始终与用户的登录口令相同。
server list (服务器列表)	请参见 preferred server list (首选服务器列表) 。
server (服务器)	(1) 在 NIS+、NIS、DNS 和 LDAP 中，服务器是为网络提供命名服务的主机。 (2) 在用于文件系统的客户机/服务器模型中，服务器是具有大容量内存和计算资源的计算机（有时称为计算服务器）。客户机可以远程访问和使用这些资源。在用于窗口系统的客户机/服务器模型中，服务器是为应用程序提供窗口服务的进程或“客户机进程”。在该模型中，客户机和服务器可以在同一台计算机上运行，也可以在不同的计算机上运行。 (3) 用于实际提供文件的守护进程。
slave server (从属服务器)	用于维护 NIS 数据库副本的服务器系统。它包含磁盘以及操作环境的完整副本。 (2) 在 NIS+ 中，从属服务器称为副本服务器。
source (源)	NIS 源文件
SSL	SSL 是安全套接字层 (secure sockets layer) 协议。它是通用的传输层安全机制，旨在使应用协议（如 LDAP）更加安全。
subnet (子网)	为了简化路由将单个逻辑网络分为较小物理网络的解决方案。
suffix (后缀)	在 LDAP 中，后缀是 DIT 的标识名 (distinguished name, DN) 。

table (表)	在 NIS+ 中，表是一个二维（不相关）数据库对象，其行和列中包含 NIS+ 数据。（在 NIS 中，NIS 映射与具有两列的 NIS+ 表相似。）表是 NIS+ 数据的存储格式。NIS+ 提供 16 个预定义的表或系统表。每个表中都存储不同类型的信息。
TCP	请参见 <i>Transport Control Protocol, TCP (传输控制协议)</i> 。
TCP/IP	传输控制协议/接口程序 (Transport Control Protocol/Interface Program) 的首字母缩略词。最初为 Internet 开发的协议套件。它又称作 <i>Internet</i> 协议套件。缺省情况下，Solaris 网络在 TCP/IP 上运行。
Transport Control Protocol, TCP (传输控制协议)	Internet 协议套件中的主要传输协议，提供可靠的、面向连接的全双工流。使用 IP 传送信息。请参见 TCP/IP。
Transport Layer Security, TLS (传输层安全性)	TLS 保护 LDAP 客户机和目录服务器之间的通信，它既提供保密性又提供数据完整性。TLS 协议是一组绝佳的安全套接字层 (Secure Sockets Layer, SSL) 协议。
wide-area network, WAN (广域网)	一种网络，通过电话、光纤或卫星链路连接位于不同地理位置的多个局域网 (local-area network, LAN) 或系统。
X.500	由开放系统互连 (Open Systems Interconnection, OSI) 标准定义的全局级目录服务。LDAP 的前身。
yp	Yellow Pages。NIS 的旧名，仍用在 NIS 代码中。

索引

数字和符号

+/- 语法

- compat, 42

- nsswitch.conf 文件, 42

- passwd_compat, 42

\$PWDIR/security/passwd.adjunct, 97

A

adjunct 文件, 84

aliases 文件, 83

.asc, 104

auto_direct.time 映射, 98

auto_home.time 映射, 98

auto_home 表, nsswitch.conf 文件, 35

auto_master 表, nsswitch.conf 文件, 35

awk, 104

C

CHKPIPE, 100

crontab, NIS, 问题, 117

crontab 文件, 101

- NIS, 问题, 117

D

dbm, 104, 105

defaultdomain 文件, 81

DES, 283, 284

DIR 目录, 83

DNS, 27, 284

- NIS, 68

- NIS, 和, 67

- NIS 与, 108–109

- nsswitch.conf 文件, 32, 41

DNS 区域, 284

DNS 区域文件, 284

DNS 转发, 284

DOM 变量, 86

domainname, 86, 88

E

enableShadowUpdate 开关, 143

/etc/defaultdomain 文件, 81, 112

/etc/hosts, 22, 88

/etc/inet/ipnodes, 22

/etc/mail/aliases 文件, 83

/etc/mail 目录, 83

/etc/nodename 文件, 81

/etc/nsswitch.conf

- nscd 守护进程, 41

- 修改转换器, 40

/etc/nsswitch.files 文件, 39

/etc/nsswitch.ldap 文件, 39

/etc/nsswitch.nis 文件, 39

/etc/nsswitch.nisplus 文件, 39

/etc 文件, 26, 42, 71

F

FMRI

LDAP, 49, 174

NIS, 80

FQDN, 128

ftp, 117

G

getaddrinfo(), 名称服务转换器, 31

gethostbyname(), 名称服务转换器, 31

getpwnam(), 名称服务转换器, 31

getpwuid(), 名称服务转换器, 31

getXbyY(), 31

GID, 285

H

hosts.byaddr, 71

hosts.byname, 71

hosts.byname 映射, 71

hosts 数据库, 100

hosts 文件, 88

I

in.named, 27

inityp2l 脚本, 225, 226

Internet

NIS, 68

nsswitch.conf 文件, 41

Internet 地址, 285

IP, 285

IP 地址, 285

IPv6, nsswitch.conf 文件, 41-42

L

LAN, 285

LDAP

比较所支持的 PAM 模块, 144, 145

LDAP (续)

从 NIS+ 转换为, 245

从 NIS 转换, 223-243

服务管理工具, 174

恢复为 NIS, 242-243

疑难解答, 187-191

在客户机上启用帐户管理, 178-179

在目录服务器上启用帐户管理, 168

帐户管理, 146-148

ldap_cachemgr daemon, 134

LDAP 架构, 193-221

基于角色的属性, 208

LDAP 疑难解答

ldapclient 无法绑定到服务器, 191

查找速度太慢, 191

登录失败, 190

无法解析主机名, 189

无法远程访问 LDAP 域中的系统, 190

ldapaddent 命令, 166

LDIF, 125

/lib/svc/method/nisplus 文件, 248

ls, 112

M

make

C2 安全性和, 106

Makefile 语法, 98

NIS 映射, 74

更新映射后, 101

make 命令

ypinit 和, 86

说明, 75

makedbm, 100, 104

更改映射服务器, 96, 97

makedbm 命令

make 命令, 71

Makefile 和, 85

ypinit 和, 86

说明, 70, 75

添加从属服务器, 105

Makefile 文件

NIS, 71

passwd 映射和, 85

Makefile 文件 (续)

- 非缺省映射
 - 修改, 103
- 更改映射的主服务器, 96
- 更改源目录, 82, 84
- 设置主服务器, 86
- 映射
 - 支持的列表, 97
- 转换为NIS, 84
- 准备, 84
- 自动挂载程序映射以及, 98
- Makefile 中的NOPUSH, 100
- Makefile 文件, NIS 安全, 92
- mapname.dir 文件, 85
- mapname.pag 文件, 85
- MIS, 285

N

- N2L 服务, 223
 - 不应使用, 224
 - 定制映射示例, 234-235
 - 设置, 229-235
 - 支持的映射, 227
- N2L 服务器, 223, 225-226
- N2L 转换, 请参见NIS 到LDAP 的转换
- ndbm, 71, 84
- ndbm 文件, 更改映射服务器, 97
- netgroup.byhost 文件, 94
- netgroup.byuser 文件, 94
- netgroup 文件, 94
 - 项, 示例, 94
- nicknames 文件, 74
- NIS, 27, 67-68, 286
 - C2 安全性, 106
 - DNS, 和, 68
 - DNS 与, 108-109
 - Internet, 68
 - Makefile, 71
 - Makefile 过滤, 98
 - makefile 准备, 84-85
 - ndbm 格式, 71
 - passwd 映射自动更新, 102
 - root 项, 91

NIS (续)

- rpc.yppasswdd, 93
- useradd, 92
- userdel, 93
- /var/yp/, 71
- ypbind 失败, 114
- ypbind 守护进程, 76
- ypbind“无法”消息, 111
- ypinit, 85
- ypservers 文件, 105
- ypwhich, 77
- ypwhich 显示不一致, 113
- 安全, 91-92
- 绑定, 75-77
- 绑定, 服务器列表, 75
- 绑定, 广播, 75
- “不可用”消息, 111
- “不响应”消息, 111
- 从属服务器, 69
- 从属服务器设置, 88-89
- 服务管理工具, 80-81
- 服务器, 68-69, 69
- 服务器, 异常, 115
- 服务器, 映射不同版本, 116-117
- 服务器不可用, 113
- 服务器列表绑定, 76
- 更新passwd 映射, 93
- 广播绑定, 76-77
- 过载的服务器, 115
- 结构, 68
- 客户机, 68-69, 69
- 客户机设置, 89-90
- 客户机问题, 112-114
- 口令, 用户, 93-94
- 口令数据, 82
- 命令挂起, 111
- 启动, 87-88
- 启动, 命令行, 87
- 设置, 准备, 80, 82
- 实用程序, 70-71
- 守护进程, 70
- 守护进程, 启动, 87
- 守护进程, 未运行, 115-116
- 守护进程列表, 70

NIS (续)

- 手动绑定, 107
 - 体系结构, 68
 - 停止, 109
 - 停止, 命令行, 87
 - 网络组, 94-95, 95
 - 问题, 111-118
 - 无法进行服务器绑定, 113-114
 - 修改配置文件, 97-98
 - 用户, 管理, 92-95
 - 用户口令已锁定, 92
 - 域, 68, 69
 - 域, 多个, 86
 - 域名, 81
 - 源文件, 82, 83-84
 - 重新启动, 命令行, 87
 - 主服务器, 69
 - 自动启动, 87
 - 组件, 69-75
- NIS+ 到 LDAP 转换
- 服务管理工具, 247-248
 - 何时不使用 SMF, 247-248
- NIS 从属服务器
- 初始化, 106
 - 添加, 105-106
- NIS 到 LDAP
- 服务管理工具
 - 另请参见 NIS、LDAP
- NIS 到 LDAP 的转换, 223-243
- 另请参见 N2L
- NIS 到 LDAP 转换
- hosts 文件配置, 228
 - ipnodes 文件配置, 228
 - LDAP 错误代码, 238-239
 - nsswitch.conf 文件配置, 228
 - 调试 NISLDAPmapping 文件, 239-241
 - 服务器超时, 237, 241
 - 缓冲区溢出, 237
 - 恢复为 NIS, 242-243
 - 命令, 226-227
 - 配置文件, 226-227
 - 使用 idsconfig 命令, 228
 - 使用 Sun Java System Directory Server, 235-237
 - 使用虚拟列表视图 (virtual list view, VLV), 236

NIS 到 LDAP 转换 (续)

- 死锁, 242
 - 锁定文件, 241
 - 问题, 239-242
 - 先决条件, 228
 - 限制, 238
 - 疑难解答, 238-242
 - 术语, 225-226
- NIS 兼容模式, 286
- NIS 客户机, 未绑定到服务器, 113
- NIS 映射, 286
- Makefile, DIR 变量, 99
 - Makefile, DOM 变量, 99
 - Makefile, PWDIR 变量, 99
 - Makefile 变量, 更改, 99
 - Makefile 过滤, 98
 - Makefile 宏, 更改, 99
 - Makefile 以及, 98-99
 - Makefile 中的 CHKPIPE, 100
 - Makefile 中的 NOPUSH, 100
 - Makefile 中的 yppush, 100
 - /var/yp/, 71
 - 查找, 74
 - 创建, 74
 - 非缺省, 101
 - 格式是 ndbm, 71
 - 更改服务器, 96-97
 - 更新, 73-74
 - 管理, 95-100
 - 昵称, 74
 - 缺省, 71-73
 - 使用, 73-74
 - 显示内容, 74, 95-96
 - 新映射, 从键盘创建, 104
 - 新映射, 从文件创建, 104
 - 修改配置文件, 97-98
 - 有关的命令, 74-75
- NIS 域, 更改, 107
- NIS 域名
- 不正确, 112-113
 - 缺少, 112-113
- NIS 主机, 更改域, 107
- NISLDAPmapping 文件, 223, 227
- nodename 文件, 81

nscd 守护进程, 41
 nsswitch.conf 文件, 26, 35, 80
 +/- 语法, 42
 Auto_home 表, 35
 Auto_master 表, 35
 compat, 42
 DNS, 32, 41
 Internet 访问, 41
 IPv6, 41-42
 keyserver 项, 35
 NIS, 68
 NOTFOUND=continue, 34
 nscd 守护进程, 41
 nsswitch.files 文件, 36
 nsswitch.files 文件和, 36
 nsswitch.nis 文件, 36
 nsswitch.nisplus 文件, 36
 passwd_compat, 42
 publickey 项, 35
 SUCCESS=return, 34
 timezone 表, 35
 TRYAGAIN=continue, 34
 UNAVAIL=continue, 34
 安装, 40-41
 不正确的语法, 34
 操作, 33
 格式, 32
 更新, 43
 介绍, 31
 口令数据, 42
 模板, 31, 36-40
 缺少项, 34
 缺省的模板文件, 36-39
 缺省文件, 39-40
 搜索条件, 33
 消息, 33
 修改, 34
 修改转换器, 40
 选项, 33
 选择文件, 40-41
 注释, 35
 状态消息, 33, 34
 nsswitch.files 文件, 39
 nsswitch ldap 文件, 39

nsswitch.nis 文件, 39
 nsswitch.nisplus 文件, 39

O

objectClass 映射, 131

P

pam_ldap, LDAP 中的帐户管理, 168-170
 pam_unix
 LDAP 中的帐户管理, 147-148, 170-171
 PAM 模块
 LDAP, 142-146
 验证方法, 142-146
 passwd, 93
 NIS 映射已自动更新, 102
 passwd.adjunct 文件, 85, 94, 97, 106
 passwd 文件, Solaris 1.x 格式, 92
 passwd 映射, 82
 用户, 添加, 92
 ping, 115
 proxy anonymous 凭证, 138
 proxy anonymous 凭证级别, 137
 proxy 凭证, 137
 proxy 凭证级别, 137
 PWDIR, 82
 PWDIR/security/passwd.adjunct 文件, 106
 /PWDIR/shadow 文件, 85
 /PWDR/security/passwd.adjunct, 85

R

rcp, 88, 117
 RFC 2307
 对象类, 203
 属性, 200
 RPC, 287
 rpc.nisd 配置文件, 246
 rpc.nisd 属性, 249
 rpc.yppasswdd, 93, 94
 passwd 更新映射, 102

rpc.yppasswdd 守护进程, 说明, 70
rpc.yupdated 守护进程, 说明, 70

S

sed, 104
self 凭证级别, 137
shadow 文件, Solaris 1.x 格式, 92
shadow 文件, 85
sites.byname 文件, 更改映射服务器, 97
SMF, 87
Solaris 命名服务, 26–28
SSD, 130
SSL 协议, 136
Sun Java System Directory Server
 迁移, 172
 使用 idsconfig 设置, 158
Sun Java 系统服务器设置, 将数据装入目录服务器, 166
svcadm, 与 NIS, 106

T

TCP, 288
TCP/IP, 288
timezone 表, 35

U

useradd, 92
 口令已锁定, 92
userdel, 93
/usr/sbin/makedbm, 非缺省映射, 修改, 104

V

/var/spool/cron/crontabs/root 文件, NIS, 问题, 117
/var/yp, 112
/var/yp/, 71, 104
/var/yp/ 目录, 85

/var/yp/binding/ 文件, 113
/var/yp/Makefile, 86
 映射
 支持的列表, 97
/var/yp/nicknames 文件, 74
/var/yp 目录, 82, 84, 88
 NIS 安全, 92

W

WAN, 288

X

X.500, 288

Y

ybind 守护进程
 服务器列表模式, 76
 广播模式, 76, 90
 过载的服务器, 115
 客户机未绑定, 113
 启动 NIS, 87
 失败, 114
 说明, 70, 74
 添加从属服务器, 106
 “无法”消息, 111
ypcat, 42, 74
ypcat 命令
 说明, 70, 75
ypinit 命令
 make 命令, 86
 Makefile 文件, 84
 初始化从属服务器, 88–89
 从属服务器, 88
 客户机设置, 89
 启动 ypserv, 87
 缺省映射, 101
 说明, 70, 75
 添加从属服务器, 106
 主服务器设置, 85

ypmap2src 脚本, 225, 226
 ypmatch 命令
 说明, 70, 75
 yppoll 命令, 说明, 70
 yppush 命令
 Makefile 以及, 100
 yppush 命令, NIS 问题, 117
 yppush 命令
 更改映射服务器, 97
 说明, 70, 75
 ypserv, 76
 故障, 117–118
 ypserv 命令, 广播模式, 76
 ypserv 守护进程, 87
 过载的服务器, 115
 说明, 70, 74
 ypserv 文件, 227
 ypservers 文件
 创建, 105
 添加从属服务器, 105
 ypservers 映射, NIS 问题, 117
 ypset 命令
 说明, 70, 75
 ypstart 脚本, 94
 ypwhich
 显示不一致, 113
 识别绑定服务器, 77
 ypwhich 命令
 说明, 71, 75
 识别主服务器, 74
 ypxfr 命令
 shell 脚本, 117
 更改映射服务器, 96, 97
 记录输出, 116–117
 将新映射分布到从属服务器中, 104
 说明, 71, 75
 ypxfrd 守护进程, 说明, 70
 ypxrfd 守护进程, 说明, 75

安

安全

NIS, 82
 NIS, 以及, 91–92

安全 (续)

NIS 映射中的根, 91
 安全 RPC 口令, 287
 安全性
 C2 安全性
 NIS 和, 106
 NIS, 82

编

编制 LDAP 客户机属性索引, 159

标

标识名, 284

表

表, 288

不

“不可用”消息 (NIS), 111
 “不响应”消息 (NIS), 111

从

从 LDAP 恢复为 NIS, 242–243
 从 NIS 转换为 LDAP, 223–243
 从属服务器, 287

点

点分十进制表示法, 283

对

对象映射, 添加新的, 272

反

反向解析, 287

访

访问控制信息, 135

服

服务管理工具

请参见SMF

 和 LDAP, 174

 和 NIS, 80–81

 和 NIS 到 LDAP 转换工具

另请参见NIS、LDAP

 以及 NIS+ 到 LDAP 转换, 247–248

 何时不使用 SMF, 247–248

服务器, 287

 NIS, 准备, 82

 NIS 从属设置, 88–89

 ypservers 文件, 105

 不可用 (NIS), 113

服务器列表, 287

服务搜索描述符, 130

 定义, 160

副

副本服务器, 260

父

父域, 286

根

根域, 287

公

公钥, 286

基

基于角色的 LDAP 架构, 对象类, 209

基于文件的命名, 27

记

记录, 287

加

加密密钥, 284, 285

架

架构

 RFC 2307, 200

 目录用户代理, 206

 项目, 208

 邮件别名, 205

架构映射, 130

可

可插拔验证模块, 142–146

客

客户机, 283

 NIS, 69

 NIS 设置, 89–90

客户机/服务器模型, 283

□

□ 令

- LDAP, 和, 145
- NIS, 以及, 93–94
- rpc.yppasswdd (NIS), 93

□ 令 -r 命令, 43

□ 令管理, [请参见帐户管理](#)

□ 令数据

- NIS, 82
- NIS, 以及, 91–92
- NIS 映射中的根, 91
- nsswitch.conf 文件, 42

□ 令条目, enableShadowUpdate 开关, 138–139

浏

浏览索引, 160

每

“每用户”凭证, 138

“每用户”索引级别, 137

密

密钥服务器, 285

- nsswitch.conf 文件, 35

名

名称服务器, 286

名称解析, 286

名称空间, 286

- DNS, 27

命

命名, 21–26

- DNS, 27

- NIS, 27

命名 (续)

- Solaris 命名服务, 26–28

- 基于文件, 27

命名服务, 286

命名服务转换器, 286

目

目录, 284

目录服务器, 261

- 迁移, 172

目录高速缓存, 284

目录信息树, 129, 284

配

配置文件, LDAP 客户机, 132

凭

凭证, 283

凭证存储, LDAP 客户机, 139

凭证级别, LDAP 客户机, 137

企

企业级网络, 285

迁

迁移, 目录服务器, 172

全

全局命名服务, 285

设

设置

- NIS, 启动, 87–88
- NIS makefile, 84–85
- NIS 从属服务器, 88–89
- NIS 客户机, 89–90
- NIS 设置, 准备, 80, 82
- 多个 NIS 域, 86
- 转换器文件, 39

守

守护进程

- NIS, 70
- NIS, 启动, 87
- NIS, 未运行, 115–116
- NIS 列表, 70
- nscd, 41

首

- 首选服务器列表, 286

属

- 属性, Internet 打印协议, 210–215
- 属性映射, 131

数

- 数据加密密钥, 283

私

- 私钥, 286

索

- 索引名, 285

网

- 网络口令, 286
- 网络名, 268
- 网络掩码, 286

系

系统信息库

- 更新, 43
- 使用多个, 43

项

- 项, 285

项目

- 对象类, 208
- 属性, 208

验

验证

- digest-MD5, 140
- simple, 140

验证方法

- LDAP 中的服务, 141–142
- none, 139
- PAM 模块, 142–146
- 在 LDAP 中选择, 139–142

引

- 引用, 159

应

- 应用程序级, 282

映

映射文件, NIS 到 LDAP, 223

用

用户

- NIS, 92–95
- useradd, 92
- userdel (NIS), 93
- 更新 passwd 映射, 93
- 口令 (NIS), 93–94
- 网络组, 94–95, 95

邮

邮件交换记录, 285

邮件主机, 285

邮件组

- 对象类, 205
- 属性, 205

域

域, 284

- NIS, 68, 69, 81

- NIS, 多个, 86

域名, 284

帐

帐户管理

- enableShadowUpdate 开关, 143
- LDAP 支持的功能, 146–148
- PAM 模块和 LDAP, 146–148
- 对于使用 pam_ldap 的 LDAP 客户机, 168–170
- 对于使用 pam_unix 的 LDAP 客户机, 170–171
- 配置目录服务器, 168
- 用于 pam_unix 客户机的 LDAP 服务器, 147–148

主

主服务器, 260, 285

主机 (计算机)

- NIS 服务器, 68–69

- NIS 客户机, 68–69

- NIS 域, 更改, 107

主体名, 268

传

传输层安全性, 136, 288

传输控制协议, 288

子

子网, 287

子域, 283

组

组

- 网络组 (NIS), 94–95, 95

组 ID, 285

