



# SUSE LINUX

管理指南

Edition 4 2005

### Copyright ©

本出版物属 Novell Inc. 的知识产权。

可以全部或部分复制本出版物的内容，只要在所有副本的显著位置能够见到版权标志。

本指南力求涵盖所有细节，但这不能确保本指南准确无误。无论是 SUSE LINUX GmbH、作者还是翻译人员都不对任何可能的错误或因错误造成的任何影响负责。

本指南所引用的许多软件和硬件描述都是注册商标。所有商标都受版权限制并且可能是注册商标。SUSE LINUX GmbH 基本沿用制造商的拼写方式。本指南中的产品和商标名称（不论是否采用特定的表示法）同样受商标法和贸易保护法的保护，也可能因此受到版权限制。

请将建议和意见发送至 <documentation@suse.de>。

*Authors:* Stefan Behlert, Frank Bodammer, Stefan Dirsch, Olaf Donjak, Roman Drahtmüller, Torsten Duwe, Thorsten Dubiel, Thomas Fehr, Stefan Fent, Werner Fink, Kurt Garloff, Carsten Gro, Joachim Gleiner, Andreas Grönbacher, Franz Hassels, Andreas Jaeger, Klaus Kmpf, Andi Kleen, Hubert Mantel, Lars Marowsky-Bree, Chris Mason, Johannes Meixner, Lars Müller, Matthias Nagorni, Anas Nashif, Siegfried Olschner, Peter Pml, Thomas Renninger, Heiko Rommel, Marcus Schfer, Nicolaus Schüller, Klaus Singvogel, Hendrik Vogelsang, Klaus G. Wagner, Rebecca Walter, Christian Zoz

*Translators:* Daniel Pisano, Tino Tanner

*Editors:* Jrg Arndt, Antje Faber, Berthold Gunreben, Roland Haidl, Jana Jaeger, Edith Parzefall, Ines Pozo, Thomas Rlz, Thomas Schraitle, Rebecca Walter

*Layout:* Manuela Piotrowski, Thomas Schraitle

*Setting:* DocBook-XML, L<sup>A</sup>T<sub>E</sub>X

This book has been printed on 100 % chlorine-free bleached paper.

# 欢 迎

祝贺您成功安装了新的 Linux 操作系统，同时感谢您选择了 SUSE LINUX 9.3。通过购买此版本，您可以按照 <http://www.novell.com/products/linuxprofessional/support/conditions.html> 所述的电话和电子邮件获得安装支持。要从此项服务中受益，请借助 CD 盒上打印的代码激活 SUSE LINUX Portal (<http://portal.suse.com>) 中的支持授权。

为确保您的系统始终处于安全和最新的状态，我们建议您使用方便的 YaST 在线更新定期更新系统。SUSE 还提供了免费的电子新闻稿，专门提供与安全有关的信息及 SUSE LINUX 提示和诀窍。只需在 <http://www.novell.com/company/subscribe/> 中输入您的电子邮件地址即可订阅。

SUSE LINUX *管理指南* 提供有关您的 SUSE LINUX 系统如何操作的背景信息。本手册为您介绍 Linux 系统管理基础知识，如文件系统、内核、引导进程及 Apache 万维网服务器的配置信息。SUSE LINUX *管理指南* 包含五大类内容：

**安装** 介绍通过 YaST 执行的系统安装和配置、特殊安装类型、LVM、RAID、更新和系统恢复。

**系统** 介绍 SUSE LINUX 的特殊功能、有关内核、引导概念和初始化进程的详细信息、引导装载程序和 X 窗口系统的配置、打印，以及 Linux 中的移动计算。

**服务** 介绍异构网络中的集成、Apache 万维网服务器的配置、文件同步及安全性。

**管理** 介绍文件系统 ACL 和重要的系统监视工具。

**附录** Linux 相关信息的重要来源

SUSE LINUX 手册的数字版本位于目录 `/usr/share/doc/manual/` 中。

# 管理指南中的更改

对上一版本 (SUSE LINUX 9.2) 的文档进行了如下修改：

- 有关 LVM 和分区的章节已经过修订。请参见 第3.7节 “LVM 配置” 和第2.8.5节 “分区”。
- 第8章 引导装载程序 已经过修订，添加了有关 YaST 模块的说明。它还包  
括新的一节，介绍如何使用通配符（第8.3.1节 “使用通配符选择引导内核”  
）。
- 文件系统一章现在包括有关 Reiser4 文件系统的信息。请参见 第20.2.5节  
“Reiser4”。
- 有关网络的部分已经过完全修订和结构重新调整。请参见 第22章 基本联网  
知识 和 后继章节。
- SuSEfirewall2 已经过修订，添加了有关 YaST 新模块的说明。请参见  
第34.1.4节 “用 YaST 进行配置”。
- 第36章 系统监视实用程序 中提到了几个新程序。
- 词汇表已经过修订和更新，另请参见。

## 版式约定

以下是本手册中使用的版式约定：

- `/etc/passwd`：文件名或目录名
- `<placeholder>`：将 `<placeholder>` 替换为实际值
- `PATH`：环境变量 `PATH`
- `ls`：命令
- `--help`：选项和参数
- `user`：用户
- `(Alt)`：按键



- ‘文件’：菜单项、按钮
- Process killed：系统讯息
- man man(1)：参考手册页
- ▶ x86, AMD64  
本部分只与指定的体系结构有关。箭头标记文本块的开始位置和结束位置。◀

## 鸣谢

在大量志愿工作的基础上，Linux 开发人员在全球范围内通力合作共同促进 Linux 的开发。我们对他们的奉献表示感谢 — 没有他们，就不会成就今天这个版本。此外，我们还要感谢 Frank Zappa 和 Pawar。当然，我们也特别感谢 Linus Torvalds 先生。

祝您使用愉快！

SUSE 团队



# Contents

<b>I</b>	<b>安装</b>	<b>1</b>
<b>1</b>	<b>使用 YaST 进行安装</b>	<b>3</b>
1.1	系统启动以进行安装	4
1.1.1	引导选项	4
1.1.2	引导系统时可能出现的问题	4
1.2	引导屏幕	5
1.3	语言选择	7
1.4	安装方式	7
1.5	安装建议	7
1.5.1	安装方式	8
1.5.2	键盘布局	8
1.5.3	鼠标	8
1.5.4	分区	9
1.5.5	软件	16
1.5.6	引导配置	19
1.5.7	时区	19
1.5.8	语言	19
1.5.9	启动安装	20
1.6	完成安装	21
1.6.1	根口令	21

1.6.2	网络配置 . . . . .	21
1.6.3	防火墙配置 . . . . .	22
1.6.4	测试因特网连接 . . . . .	22
1.6.5	装载软件更新程序 . . . . .	23
1.6.6	用户鉴定 . . . . .	24
1.6.7	将主机配置为 NIS 客户机 . . . . .	25
1.6.8	创建本地用户帐户 . . . . .	26
1.6.9	发行说明 . . . . .	28
1.7	硬件配置 . . . . .	28
1.8	图形登录 . . . . .	28
<b>2</b>	<b>使用 YaST 进行系统配置</b>	<b>31</b>
2.1	启动 YaST . . . . .	32
2.2	YaST 控制中心 . . . . .	32
2.3	软件 . . . . .	33
2.3.1	安装和去除软件 . . . . .	33
2.3.2	更改安装源 . . . . .	39
2.3.3	YaST 联机更新 . . . . .	39
2.3.4	增补程序 CD 更新 . . . . .	41
2.3.5	系统更新 . . . . .	42
2.3.6	媒体检查 . . . . .	44
2.4	硬件 . . . . .	44
2.4.1	CD-ROM 和 DVD 驱动器 . . . . .	44
2.4.2	打印机 . . . . .	44
2.4.3	硬盘控制器 . . . . .	45
2.4.4	硬件信息 . . . . .	45
2.4.5	IDE DMA 方式 . . . . .	45
2.4.6	扫描仪 . . . . .	46
2.4.7	声卡 . . . . .	48
2.4.8	电视和无线电卡 . . . . .	49
2.5	网络设备 . . . . .	49

2.6	网络服务	50
2.6.1	邮件传送代理	50
2.6.2	其他可用服务	50
2.7	安全性和用户	52
2.7.1	用户管理	52
2.7.2	组管理	53
2.7.3	安全性设置	54
2.7.4	防火墙	55
2.8	系统	56
2.8.1	系统区域的备份副本	56
2.8.2	恢复系统	57
2.8.3	创建引导磁盘和应急磁盘	57
2.8.4	LVM	58
2.8.5	分区	58
2.8.6	配置文件管理器 (SCPM)	63
2.8.7	Systems Services (RunLevel)	63
2.8.8	Sysconfig 编辑器	63
2.8.9	时区选择	64
2.8.10	语言选择	64
2.9	其他	64
2.9.1	提交支持请求	64
2.9.2	引导日志	64
2.9.3	系统日志	65
2.9.4	装载供应商的驱动程序 CD	65
2.10	文本方式下的 YaST (ncurses)	65
2.10.1	在模块中导航	66
2.10.2	组合键的限制	67
2.10.3	启动单个模块	68
2.10.4	YOU 模块	68
2.11	从命令行联机更新	69

3	特殊安装过程	71
3.1	linuxrc	72
3.1.1	将参数传递给 linuxrc	72
3.2	通过 VNC 进行安装	73
3.2.1	准备进行 VNC 安装	74
3.2.2	用于 VNC 安装的客户程序	74
3.3	通过 YaST 进行基于文本的安装	74
3.4	启动 SUSE LINUX	76
3.4.1	图形 SUSE 屏幕	76
3.4.2	禁用 SUSE 屏幕	76
3.5	提示和技巧	77
3.5.1	使用 rawwritewin 创建引导磁盘	77
3.5.2	使用 rawrite 创建引导磁盘	78
3.5.3	在 UNIX-Type 系统中创建引导磁盘	78
3.5.4	从软盘引导 (SYSLINUX)	79
3.5.5	不支持的 CD-ROM 驱动器	80
3.5.6	Installation from a Network Source	80
3.6	为 SCSI 设备指派永久设备文件名	81
3.7	LVM 配置	81
3.7.1	逻辑卷管理器	82
3.7.2	用 YaST 配置 LVM	83
3.8	软 RAID 配置	87
3.8.1	软 RAID	87
3.8.2	使用 YaST 配置软 RAID	89
3.8.3	查错	90
3.8.4	参考信息	90

<b>4</b>	<b>更新系统和包管理</b>	<b>93</b>
4.1	更新 SUSE LINUX	94
4.1.1	准备工作	94
4.1.2	可能的问题	94
4.1.3	使用 YaST 进行更新	95
4.1.4	更新个别的包	95
4.2	版本间的软件更改	95
4.2.1	从 8.1 到 8.2	96
4.2.2	从 8.2 到 9.0	97
4.2.3	从 9.0 到 9.1	97
4.2.4	从 9.1 到 9.2	103
4.2.5	从 9.2 到 9.3	108
4.3	RPM — 包管理器	110
4.3.1	校验包真实性	110
4.3.2	管理包：安装、更新和卸载	111
4.3.3	RPM 和增补程序	112
4.3.4	增量 RPM 包	113
4.3.5	RPM 查询	114
4.3.6	安装和编译源包	116
4.3.7	使用 build 编译 RPM 包	118
4.3.8	用于 RPM 档案和 RPM 数据库的工具	119
<b>5</b>	<b>系统修复</b>	<b>121</b>
5.1	自动修复	122
5.2	用户定义的修复	123
5.3	专家工具	124
5.4	SUSE 应急系统	124
5.4.1	启动应急系统	125
5.4.2	使用应急系统	125

<b>II</b>	<b>系統</b>	<b>127</b>
<b>6</b>	<b>64 位系统环境中的 32 位和 64 位应用程序</b>	<b>129</b>
6.1	运行时支持	130
6.2	软件开发	130
6.3	Biarch 平台上的软件编译	131
6.4	内核规范	132
<b>7</b>	<b>引导和配置 Linux 系统</b>	<b>133</b>
7.1	Linux 引导进程	134
7.1.1	initrd	135
7.1.2	linuxrc	135
7.1.3	参考信息	136
7.2	init 程序	136
7.3	运行级别	137
7.4	更改运行级别	138
7.5	Init 脚本	139
7.5.1	添加 init 脚本	141
7.6	Systems Servcices (RunLevel)	142
7.7	SuSEconfig 和 /etc/sysconfig	143
7.8	YaST sysconfig 编辑器	145
<b>8</b>	<b>引导装载程序</b>	<b>147</b>
8.1	引导管理	148
8.2	选择引导装载程序	149
8.3	通过 GRUB 引导	149
8.3.1	GRUB 引导菜单	150
8.3.2	文件 device.map	155
8.3.3	文件 /etc/grub.conf	155
8.3.4	GRUB Shell	156
8.3.5	设置引导口令	156



8.4	使用 YaST 配置引导装载程序	157
8.4.1	主窗口	158
8.4.2	引导装载程序配置选项	159
8.5	卸载 Linux 引导装载程序	160
8.6	创建引导 CD	160
8.7	图形 SUSE 屏幕	161
8.8	查错	162
8.9	参考信息	163
<b>9</b>	<b>Linux 内核</b>	<b>165</b>
9.1	内核更新	166
9.2	内核源	166
9.3	内核配置	166
9.3.1	用命令行进行配置	167
9.3.2	用文本方式进行配置	167
9.3.3	在 X 窗口系统中进行配置	167
9.4	内核模块	168
9.4.1	通过 hwinfo 进行硬件检测	168
9.4.2	处理模块	168
9.4.3	/etc/modprobe.conf	169
9.4.4	Kmod — 内核模块装载程序	169
9.5	编译内核	170
9.6	安装内核	170
9.7	在编译后清理硬盘	171
<b>10</b>	<b>SUSE LINUX 的特殊功能</b>	<b>173</b>
10.1	特殊软件包的相关信息	174
10.1.1	包 bash 和 /etc/profile	174
10.1.2	cron 包	174
10.1.3	日志文件：包 logrotate	175
10.1.4	手册页	176

10.1.5	locate 命令	176
10.1.6	ulimit 命令	176
10.1.7	free 命令	177
10.1.8	文件 /etc/resolv.conf	178
10.1.9	GNU Emacs 的设置	178
10.1.10	vi 简介	179
10.2	虚拟控制台	181
10.3	键盘映射	181
10.4	语言和国家/地区特定的设置	182
10.4.1	一些示例	183
10.4.2	语言支持的设置	184
<b>11</b>	<b>X 窗口系统</b>	<b>185</b>
11.1	使用 SaX2 设置 X11	186
11.1.1	桌面	187
11.1.2	图形卡	188
11.1.3	颜色和分辨率	189
11.1.4	虚拟分辨率	190
11.1.5	3D 加速	190
11.1.6	图像位置和大小	191
11.1.7	多头	191
11.1.8	输入设备	192
11.1.9	AccessX	193
11.1.10	参考信息	193
11.1.11	游戏杆	194
11.1.12	键盘布局选择	194
11.1.13	鼠标	194
11.2	优化 X 配置	194
11.2.1	Screen 部分	196
11.2.2	Device 部分	197
11.2.3	Monitor 部分和 Modes 部分	198

11.3	安装和配置字体	199
11.3.1	Xft	199
11.3.2	X11 核心字体	202
11.3.3	CID-Keyed 字体	203
11.4	OpenGL — 3D 配置	204
11.4.1	硬件支持	204
11.4.2	OpenGL 驱动程序	204
11.4.3	诊断工具 3Ddiag	205
11.4.4	OpenGL 测试实用程序	205
11.4.5	查错	205
11.4.6	安装支持	205
11.4.7	其他联机文档	206
<b>12</b>	<b>打印机操作</b>	<b>207</b>
12.1	准备工作和其他考虑事项	208
12.2	打印系统工作流程	209
12.3	连接打印机的方法和协议	209
12.4	安装软件	210
12.5	配置打印机	210
12.5.1	本地打印机	211
12.5.2	网络打印机	213
12.5.3	配置任务	214
12.6	为应用程序进行配置	215
12.6.1	从命令行打印	216
12.6.2	使用命令行工具从应用程序进行打印	216
12.6.3	使用 CUPS 打印系统	216
12.7	SUSE LINUX 中的特殊功能	216
12.7.1	CUPS 服务器和防火墙	216
12.7.2	通过 CUPS 万维网前端进行管理	217
12.7.3	CUPS 打印服务 (cupsd) 中的更改	218
12.7.4	多种包中的 PPD 文件	219

12.8	查错	221
12.8.1	打印机没有标准打印机语言支持	221
12.8.2	没有合适的 PPD 文件可用于 PostScript 打印机	221
12.8.3	并行端口	221
12.8.4	网络打印机连接	222
12.8.5	打印件有问题但没有错误讯息	224
12.8.6	禁用的队列	224
12.8.7	CUPS 浏览：删除打印作业	225
12.8.8	有问题的打印作业和数据传送错误	225
12.8.9	对 CUPS 打印系统进行调试	226
12.8.10	参考信息	226
<b>13</b>	<b>Linux 中的移动计算</b>	<b>227</b>
13.1	便携式计算机	228
13.1.1	便携式计算机硬件的特殊属性	228
13.1.2	省电	228
13.1.3	在变化的操作环境中集成	229
13.1.4	软件选择	230
13.1.5	数据安全性	232
13.2	移动硬件	233
13.3	手提电话和 PDA	234
13.4	参考信息	234
<b>14</b>	<b>PCMCIA</b>	<b>235</b>
14.1	硬件	236
14.2	软件	236
14.2.1	基础模块	236
14.2.2	卡管理器	236
14.3	配置	237
14.3.1	网卡	238
14.3.2	ISDN	238

14.3.3	调制解调器	238
14.3.4	SCSI 和 IDE	238
14.4	实用程序	239
14.5	查错	239
14.5.1	PCMCIA 基础系统不工作	239
14.5.2	PCMCIA 卡工作不正常	240
14.6	参考信息	241
<b>15</b>	<b>系统配置配置文件管理</b>	<b>243</b>
15.1	术语	244
15.2	使用命令行配置 SCPM	244
15.2.1	启动 SCPM 并定义资源组	245
15.2.2	创建和管理配置文件	245
15.2.3	切换配置配置文件	246
15.2.4	高级配置文件设置	246
15.3	YaST 配置文件管理器	247
15.3.1	配置资源组	248
15.3.2	新建配置文件	248
15.3.3	修改现有的配置文件	249
15.3.4	切换配置文件	249
15.4	查错	251
15.4.1	在切换进程中终止	251
15.4.2	更改资源组配置	251
15.5	引导系统时选择配置文件	251
15.6	参考信息	252

<b>16 电源管理</b>	<b>253</b>
16.1 省电功能	254
16.2 APM	255
16.3 ACPI	256
16.3.1 使用 ACPI	256
16.3.2 控制 CPU 性能	259
16.3.3 ACPI 工具	260
16.3.4 查错	260
16.4 硬盘的休眠	261
16.5 powersave 包	262
16.5.1 配置 powersave 包	263
16.5.2 配置 APM 和 ACPI	265
16.5.3 其他 ACPI 功能	266
16.5.4 查错	267
16.6 YaST 电源管理模块	269
<b>17 无线通讯</b>	<b>275</b>
17.1 无线 LAN	276
17.1.1 硬件	276
17.1.2 功能	277
17.1.3 用 YaST 进行配置	279
17.1.4 实用程序	281
17.1.5 建立 WLAN 的提示和技巧	281
17.1.6 查错	282
17.1.7 参考信息	283
17.2 蓝牙	283
17.2.1 基础	283
17.2.2 配置	284
17.2.3 系统部件和实用程序	287
17.2.4 图形应用程序	288
17.2.5 示例	288

17.2.6	查错	290
17.2.7	参考信息	291
17.3	红外线数据传送	291
17.3.1	软件	291
17.3.2	配置	292
17.3.3	用法	292
17.3.4	查错	292
<b>18</b>	<b>热插拔系统</b>	<b>295</b>
18.1	设备和接口	296
18.2	热插拔事件	297
18.3	热插拔代理	297
18.3.1	激活网络接口	298
18.3.2	激活储存设备	298
18.4	自动模块装载	299
18.5	对 PCI 设备进行热插拔	300
18.6	引导脚本冷插拔	300
18.7	错误分析	300
18.7.1	日志文件	300
18.7.2	引导问题	301
18.7.3	事件记录器	301
<b>19</b>	<b>使用 udev 创建动态设备节点</b>	<b>303</b>
19.1	创建规则	304
19.2	使用 NAME 和 SYMLINK 实现自动操作	305
19.3	关键字中的正则表达式	305
19.4	关键字选择	305
19.5	用于大容量储存设备的永久名称	306

<b>20 Linux 中的文件系统</b>	<b>309</b>
20.1 术语	310
20.2 Linux 中的主要文件系统	310
20.2.1 ReiserFS	310
20.2.2 Ext2	311
20.2.3 Ext3	312
20.2.4 将 Ext2 文件系统转换为 Ext3	313
20.2.5 Reiser4	313
20.2.6 JFS	314
20.2.7 XFS	315
20.3 其他一些支持的文件系统	315
20.4 Linux 中对大型文件的支持	316
20.5 参考信息	318
<b>21 通过 PAM 进行鉴定</b>	<b>319</b>
21.1 PAM 配置文件的结构	320
21.2 sshd 的 PAM 配置	321
21.3 PAM 模块的配置	323
21.3.1 pam_unix2.conf	324
21.3.2 pam_env.conf	324
21.3.3 pam_pwcheck.conf	325
21.3.4 limits.conf	325
21.4 参考信息	325
<b>III 服務</b>	<b>327</b>
<b>22 基本联网知识</b>	<b>329</b>
22.1 IP 地址和路由	332
22.1.1 IP 地址	332
22.1.2 网络掩码和路由	333
22.2 IPv6 — 下一代的因特网	335



22.2.1	优点	335
22.2.2	地址类型和结构	336
22.2.3	IPv4 与 IPv6 并存	339
22.2.4	配置 IPv6	340
22.2.5	参考信息	341
22.3	名称解析	341
22.4	使用 YaST 配置网络连接	342
22.4.1	使用 YaST 配置网卡	342
22.4.2	调制解调器	344
22.4.3	ISDN	346
22.4.4	电缆调制解调器	349
22.4.5	DSL	349
22.5	手工配置网络连接	351
22.5.1	配置文件	353
22.5.2	启动脚本	359
22.6	作为拨号助手的 smpppd	359
22.6.1	配置 smpppd	360
22.6.2	配置供远程使用的 kinternet、cinternet 和 qinternet	360
<b>23</b>	<b>网络中的 SLP 服务</b>	<b>363</b>
23.1	注册您自己的服务	364
23.2	SUSE LINUX 中的 SLP 前端	364
23.3	激活 SLP	365
23.4	参考信息	365
<b>24</b>	<b>域名系统</b>	<b>367</b>
24.1	用 YaST 进行配置	368
24.1.1	向导配置	368
24.1.2	专家配置	369
24.2	启动名称服务器 BIND	372
24.3	配置文件 /etc/named.conf	376

24.3.1	重要的配置选项	377
24.3.2	日志记录	378
24.3.3	区域项	379
24.4	区域文件	380
24.5	区域数据的动态更新	382
24.6	安全事务	383
24.7	DNS 安全性	384
24.8	参考信息	384
<b>25</b>	<b>使用 NIS</b>	<b>385</b>
25.1	配置 NIS 服务器	386
25.2	配置 NIS 客户程序	388
<b>26</b>	<b>通过 NFS 共享文件系统</b>	<b>391</b>
26.1	使用 YaST 导入文件系统	392
26.2	手工导入文件系统	392
26.3	使用 YaST 导出文件系统	393
26.4	手工导出文件系统	394
<b>27</b>	<b>DHCP</b>	<b>397</b>
27.1	使用 YaST 配置 DHCP 服务器	398
27.2	DHCP 软件包	399
27.3	DHCP 服务器 dhcpd	400
27.3.1	具有固定 IP 地址的客户机	403
27.3.2	SUSE LINUX 版本	404
27.4	参考信息	404
<b>28</b>	<b>使用 xntp 同步时间</b>	<b>405</b>
28.1	在网络中配置 xntp	406
28.2	设置本地参考时钟	406
28.3	使用 YaST 配置 NTP 客户机	407
28.3.1	快速的 NTP 客户机配置	407
28.3.2	复杂的 NTP 客户机配置	408

<b>29 LDAP — 目录服务</b>	<b>411</b>
29.1 对比 LDAP 和 NIS	413
29.2 LDAP 目录树的结构	413
29.3 使用 slapd.conf 配置服务器	416
29.3.1 slapd.conf 中的全局指令	416
29.3.2 slapd.conf 中的数据库特定指令	419
29.3.3 启动和停止服务器	420
29.4 LDAP 目录中的数据处理	420
29.4.1 将数据插入 LDAP 目录	420
29.4.2 修改 LDAP 目录中的数据	422
29.4.3 搜索或读取 LDAP 目录中的数据	423
29.4.4 删除 LDAP 目录中的数据	423
29.5 YaST LDAP 客户程序	424
29.5.1 标准过程	424
29.5.2 LDAP 客户程序的配置	425
29.5.3 用户和组 — 使用 YaST 配置	429
29.6 参考信息	429
<b>30 Apache 万维网服务器</b>	<b>433</b>
30.1 基础	434
30.1.1 万维网服务器	434
30.1.2 HTTP	434
30.1.3 URL	434
30.1.4 默认页的自动显示	435
30.2 使用 YaST 设置 HTTP 服务器	435
30.3 Apache 模块	435
30.4 线程	436
30.5 安装	437
30.5.1 在 YaST 中选择包	437
30.5.2 激活 Apache	437
30.5.3 用于活动内容的模块	437

30.5.4	其他建议使用的包	438
30.5.5	用 apxs 安装模块	438
30.6	配置	438
30.6.1	用 SuSEconfig 进行配置	438
30.6.2	手工配置	439
30.7	使用 Apache	443
30.8	活动内容	443
30.8.1	服务器端包含	444
30.8.2	通用网关接口	444
30.8.3	GET 和 POST	445
30.8.4	使用模块生成活动内容	445
30.8.5	mod_perl	445
30.8.6	mod_php4	447
30.8.7	mod_python	447
30.8.8	mod_ruby	447
30.9	虚拟主机	448
30.9.1	基于名称的虚拟主机	448
30.9.2	基于 IP 的虚拟主机	449
30.9.3	Apache 的多个实例	450
30.10	安全性	450
30.10.1	最大程度减少风险	450
30.10.2	访问权限	451
30.10.3	保持更新	451
30.11	查错	451
30.12	参考信息	452
30.12.1	Apache	452
30.12.2	CGI	452
30.12.3	安全性	453
30.12.4	其他资源	453

31 文件同步	455
31.1 可用的数据同步软件	456
31.1.1 Unison	456
31.1.2 CVS	456
31.1.3 subversion	457
31.1.4 mailsync	457
31.1.5 rsync	457
31.2 选择程序时的决定性因素	457
31.2.1 客户机/服务器与对等模式	457
31.2.2 可移植性	458
31.2.3 交互与自动	458
31.2.4 冲突：冲突的发生和解决方法	458
31.2.5 选择和添加文件	458
31.2.6 历史记录	459
31.2.7 数据量和硬盘要求	459
31.2.8 GUI	459
31.2.9 用户友好	459
31.2.10 预防攻击	459
31.2.11 防止数据丢失	460
31.3 Unison 简介	460
31.3.1 要求	461
31.3.2 使用 Unison	461
31.3.3 参考信息	462
31.4 CVS 简介	462
31.4.1 配置 CVS 服务器	462
31.4.2 使用 CVS	463
31.4.3 参考信息	464
31.5 Subversion 简介	464
31.5.1 安装 Subversion 服务器	464
31.5.2 用法和操作	465

31.5.3	参考信息	466
31.6	rsync 简介	467
31.6.1	配置和操作	467
31.6.2	参考信息	468
31.7	mailsync 简介	468
31.7.1	配置和用法	469
31.7.2	可能的问题	471
31.7.3	参考信息	471
<b>32</b>	<b>Samba</b>	<b>473</b>
32.1	配置服务器	475
32.1.1	global 部分	475
32.1.2	共享	476
32.1.3	安全性级别	477
32.2	将 Samba 用作登录服务器	478
32.3	使用 YaST 配置 Samba 服务器	479
32.4	配置客户机	480
32.4.1	使用 YaST 配置 Samba 客户机	480
32.4.2	Windows 9x 和 ME	480
32.5	优化	481
<b>33</b>	<b>代理服务器 Squid</b>	<b>483</b>
33.1	作为代理超速缓存的 Squid	484
33.2	有关代理超速缓存的一些事实	484
33.2.1	Squid 和安全性	484
33.2.2	多个超速缓存	484
33.2.3	超速缓存因特网对象	485
33.3	系统要求	485
33.3.1	硬盘	486
33.3.2	磁盘超速缓存的大小	486
33.3.3	RAM	486

33.3.4	CPU	486
33.4	启动 Squid	487
33.4.1	用于启动和停止 Squid 的命令	487
33.4.2	本地 DNS 服务器	488
33.5	配置文件 /etc/squid/squid.conf	489
33.5.1	常规配置选项 (选择)	489
33.5.2	访问控制选项	491
33.6	配置透明代理	493
33.6.1	内核配置	493
33.6.2	/etc/squid/squid.conf 中的配置选项	493
33.6.3	使用 SuSEfirewall2 配置防火墙	494
33.7	cachemgr.cgi	495
33.7.1	设置	495
33.7.2	/etc/squid/squid.conf 中的超速缓存管理器 ACL	495
33.7.3	查看统计数字	496
33.8	squidGuard	497
33.9	使用 Calamaris 生成超速缓存报告	498
33.10	参考信息	499

## IV 管理 501

34	Linux 中的安全性	503
34.1	伪装和防火墙	504
34.1.1	使用 iptables 过滤包	504
34.1.2	关于伪装的基础知识	506
34.1.3	防火墙基础知识	507
34.1.4	SuSEfirewall2	508
34.1.5	参考信息	512
34.2	SSH: 安全的网络操作	512
34.2.1	OpenSSH 软件包	512

34.2.2	ssh 程序 . . . . .	512
34.2.3	scp — 安全复制 . . . . .	513
34.2.4	sftp — 安全的文件传送 . . . . .	513
34.2.5	SSH 守护程序 (sshd) — 服务器端 . . . . .	513
34.2.6	SSH 鉴定机制 . . . . .	514
34.2.7	X、鉴定和转发机制 . . . . .	515
34.3	对分区和文件进行加密 . . . . .	516
34.3.1	应用方案 . . . . .	516
34.3.2	使用 YaST 设置加密文件系统 . . . . .	516
34.3.3	加密可移动媒体的内容 . . . . .	518
34.4	安全性和机密性 . . . . .	518
34.4.1	本地安全和网络安全 . . . . .	519
34.4.2	一些常用的安全提示和技巧 . . . . .	525
34.4.3	使用中央安全报告地址 . . . . .	527
<b>35</b>	<b>Linux 中的访问控制列表</b>	<b>529</b>
35.1	ACL 的优势 . . . . .	530
35.2	定义 . . . . .	530
35.3	处理 ACL . . . . .	531
35.3.1	ACL 项和文件方式权限位 . . . . .	532
35.3.2	具有访问 ACL 的目录 . . . . .	533
35.3.3	具有默认 ACL 的目录 . . . . .	535
35.3.4	ACL 检查算法 . . . . .	537
35.4	应用程序中的 ACL 支持 . . . . .	538
35.5	参考信息 . . . . .	538



<b>36 系统监视实用程序</b>	<b>539</b>
36.1 打开的文件的列表: lsof	540
36.2 访问文件的用户: fuser	541
36.3 文件属性: stat	541
36.4 USB 设备: lsusb	542
36.5 关于 SCSI 设备的信息: scsiinfo	542
36.6 进程: top	543
36.7 进程列表: ps	544
36.8 进程树: pstree	545
36.9 哪些用户在执行哪些操作: w	546
36.10 内存使用: free	547
36.11 内核信号缓冲区: dmesg	547
36.12 文件系统及其使用: mount、df 和 du	548
36.13 /proc 文件系统	549
36.14 vmstat、iostat 和 mpstat	551
36.15 procinfo	551
36.16 PCI 资源: lspci	552
36.17 程序运行的系统调用: strace	553
36.18 程序运行的库调用: ltrace	554
36.19 指定必需的库: ldd	554
36.20 有关 ELF 二进制文件的其他信息	555
36.21 进程间通讯: ipcs	555
36.22 使用 time 进行时间度量	556

**V 附錄** **557**

**A 信息源和文档** **559**

**B File System Checking** **563**

**C GNU 通用公共许可证** **575**

**词汇表** **583**



**Part I**

**安 装**



# 使用 YaST 进行安装

本章将系统地介绍如何使用系统辅助工具 YaST 来安装 SUSE LINUX 系统。在介绍安装过程的准备工作时，将同时提供背景信息来帮助您在各个配置阶段作出正确的决定。

1.1	系统启动以进行安装	4
1.2	引导屏幕	5
1.3	语言选择	7
1.4	安装方式	7
1.5	安装建议	7
1.6	完成安装	21
1.7	硬件配置	28
1.8	图形登录	28

## 1.1 系统启动以进行安装

将第一张 SUSE LINUX CD 或 DVD 插入驱动器。然后重引导计算机，以便从驱动器中的媒体启动安装程序。

### 1.1.1 引导选项

除 CD 或 DVD 之外，还存在其他引导选项，如果从 CD 或 DVD 引导时出现问题，就可以使用这些引导选项。表 1.1 “引导选项” 中介绍了这些选项。

Table 1.1: 引导选项

引导选项	说明
CD-ROM	这是最简单的引导选项。如果系统具有 Linux 支持的本地 CD-ROM 驱动器，则可以使用此选项。
Floppy	用于生成引导软盘的映像位于 CD 1 上的 <code>/boot/</code> 目录中。在同一目录中还提供了一个 <code>README</code> 文件。
PXE 或 BOOTP	这一选项必须得到系统的 BIOS 或固件的支持，而且网络中必须有一个可用的引导服务器。还可以用另一个 SUSE LINUX 系统来执行此任务。
硬盘	SUSE LINUX 也可以从硬盘来进行引导。为此，请将内核 ( <code>linux</code> ) 和安装系统 ( <code>initrd</code> ) 从 CD 1 上的目录 <code>/boot/loader</code> 中复制到硬盘，并向引导装载程序添加适当的项。

### 1.1.2 引导系统时可能出现的问题

如果您使用的是较早的或不受支持的硬件，则从 CD 或 DVD 引导时可能出现問題。您的 CD-ROM 驱动器可能无法读取 CD 1 上的引导映像。在这种情况下，使用 CD 2 引导系统。CD 2 包含传统的 2.88 MB 引导映像，即使不受支持的驱动器也能够读取该映像，并且该映像允许您通过网络执行安装。

BIOS 中引导顺序的设置可能不正确。您的主板的文档中提供如何更改 BIOS 设置的说明。以下几段提供了基本说明。

BIOS 是实现计算机最基本功能的软件。主板厂商提供专门为他们的硬件设计的 BIOS。通常，只能在特定时间（例如引导计算机时）访问 BIOS 设置。在此初

始化阶段，计算机执行若干诊断硬件测试。其中一项测试就是内存检查，由内存计数器指示。当显示计数器时，请查找一行（通常在计数器下面，有时也在底部），该行提到要访问 BIOS 设置需要按的键。通常，要按的键是 (Del) 键、(F1) 键或 (Esc) 键。按此键，直到出现 BIOS 设置屏幕。

### Important

#### BIOS 中的键盘布局

BIOS 配置通常使用 US 键盘布局。

### Important

若要更改 AWARD BIOS 中的引导顺序，请查找 'BIOS FEATURES SETUP' 项。其他制造商可能对该项使用不同的名称，例如 'ADVANCED CMOS SETUP'。当您找到该项后，将其选中并按 (Enter) 键确认。

在所打开的屏幕中，查找名为 'BOOT SEQUENCE' 的子项。引导顺序通常被设置为 C,A 或 A,C 等。在前一种情况中，计算机首先搜索硬盘 (C)，然后搜索软盘驱动器 (A) 以查找可引导媒体。通过按 (PgUp) 键或 (PgDown) 键更改设置，直到顺序为 A,CDROM,C。

通过按 (Esc) 键离开 BIOS 设置屏幕。若要保存更改，请选择 'SAVE & EXIT SETUP' 或按 (F10) 键。若要确认应保存设置，按 (Y) 键。

如果您具有 SCSI CD-ROM 驱动器，则更改 SCSI BIOS 的设置。例如，如果使用的是 Adaptec 主机适配器，则通过按 (Ctrl)-(A) 打开设置。然后选择 '磁盘实用程序'，显示所连接的硬件部件。记录 CD-ROM 驱动器的 SCSI ID。按 (Esc) 键退出菜单，然后打开 '配置适配器设置'。在 '其他选项' 下，选择 '引导设备选项'，然后按 (Enter) 键。输入 CD-ROM 驱动器的 ID，然后再次按 (Enter) 键。随后再按 (Esc) 键两次返回 SCSI BIOS 的开始屏幕。退出此屏幕并通过 '是' 确认以引导计算机。

## 1.2 引导屏幕

引导屏幕将显示安装过程的选项数目。'从硬盘引导' 用于引导已安装系统。计算机默认选择此项目，因为 CD 常被留在驱动器中。要安装系统，请用箭头键选择一个安装选项。相关的选项有：

**安装 常规安装方式。** 将启用所有最新的硬件功能。

**安装—禁用 ACPI** 如果常规安装失败，则可能是因为系统硬件不支持 ACPI（高级配置和电源接口）。如果是这种情况，请使用此选项进行安装，这样将没有 ACPI 支持。

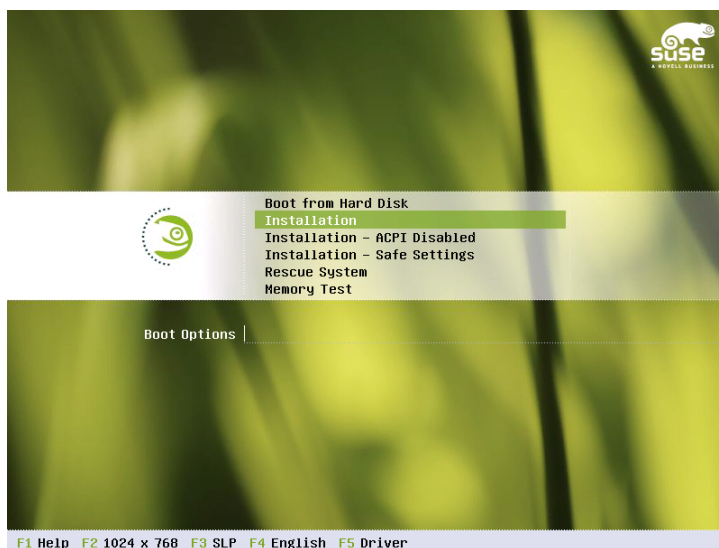


Figure 1.1: 引导屏幕

安装—安全设置 引导使用了 DMA 方式（用于 CD-ROM 驱动器）且禁用了电源管理功能的系统。专家还可以使用命令行来输入或更改内核参数。

使用在屏幕底部一栏中指示的功能键可以更改许多安装设置。

- Ⓛ1 引导屏幕的活动元素的上下文相关帮助。
- Ⓛ2 选择安装的多图形显示方式。如果图形安装出现问题，则可以选择文本方式。
- Ⓛ3 通常情况下都是从插入的安装媒体来执行安装。在这里可以选择其他安装源，如 FTP 或 NFS 服务器。如果在具有 SLP 服务器的网络中执行安装，则可以使用此选项选择服务器上可用的安装源之一。有关 SLP 的信息，请参见第23章 网络中的 SLP 服务。
- Ⓛ4 选择安装的显示语言。
- Ⓛ5 默认情况下，在系统启动期间不会显示 Linux 内核的诊断讯息。您只能看到一个进度条。要显示这些讯息，请选择‘冗长’。有关详细信息（不带附加图形框架），请选择‘本机’。



- Ⓞ6 用于通知系统您有一个可选的、含有 SUSE LINUX 驱动程序更新的磁盘。系统将在安装过程中的适当时间提示您插入更新磁盘。

在启动安装数秒钟之后，SUSE LINUX 将装载一个最小的 Linux 系统来运行安装过程。如果启用了‘本机’或‘冗长’，则将滚动显示讯息和版权声明。在装载进程结束时，YaST 安装程序启动。在数秒钟之后，屏幕将显示图形安装程序。

SUSE LINUX 的实际安装是从此时开始的。所有 YaST 屏幕的布局都基本相同。所有按钮、输入字段和列表都可以通过鼠标或键盘进行访问。如果您的鼠标指针不移动，则表明尚未自动检测到鼠标。在这种情况下，可暂时使用键盘。使用键盘进行导航类似于第2.10.1节“在模块中导航”中的说明。

## 1.3 语言选择

通常可以根据需要配置 YaST 和 SUSE LINUX 使用不同的语言。此处选择的语言也用于键盘布局。另外，YaST 使用此语言设置来猜测系统时钟的时区。这些设置可以在稍候选择要在系统上安装的辅助语言时进行修改。如果鼠标不起作用，请使用箭头键来选择语言，然后按 **(Tab)** 键，直到高亮显示‘接受’选项。然后按 **(Enter)** 键确认语言选择。

## 1.4 安装方式

选择‘全新安装’或‘更新现有系统’。只有在已安装了 SUSE LINUX 系统的情况下才能进行更新。如果已安装了此系统，请选择‘引导已安装系统’来引导它。如果引导已安装系统失败，则可能是某些重要系统配置已被损坏。这时可以选择‘修复已安装系统’使系统重新可引导。如果未安装任何 SUSE LINUX 系统，就只能执行全新安装。请参见图1.3“选择安装方式”。

以下各节将介绍安装新系统的过程。有关系统更新的详细介绍，请参见第2.3.5节“系统更新”。有关系统修复选项的介绍，请参见第5章系统修复。

## 1.5 安装建议

在完成硬件检测后，建议窗口（如图1.4“建议窗口”所示）将显示一些有关已识别出的硬件的信息，并给出许多安装和分区选项。在选择这些项目并在相应的对话框中配置它们后，总是会返回建议窗口，而且每次返回此窗口都会有相应的更新。以下几节将讨论各项设置。

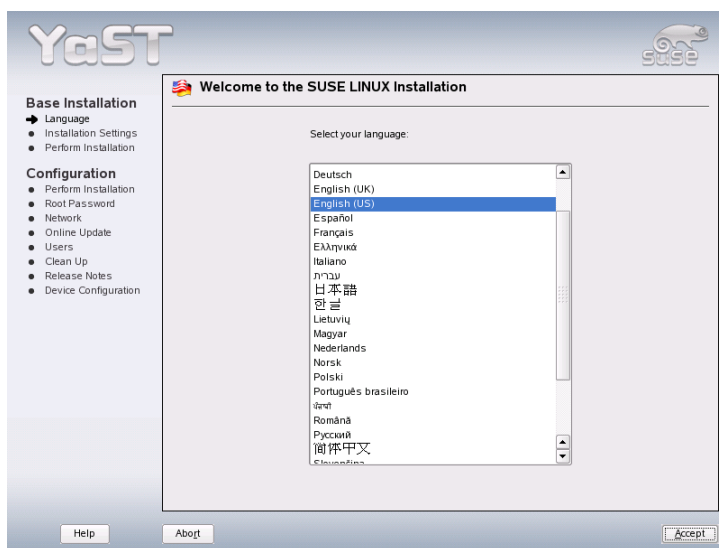


Figure 1.2: 选择语言

### 1.5.1 安装方式

使用此选项可更改先前选择的安装方式。这些选项与第1.4节“安装方式”中介绍的相同。

### 1.5.2 键盘布局

选择键盘布局。默认情况下，键盘布局对应于所选择的语言。在更改布局后，应测试(Y)键、(Z)键和特殊字符以确保选择正确。完成后，选择‘下一步’返回建议窗口。

### 1.5.3 鼠标

如果 YaST 未能自动检测到您的鼠标，请在建议窗口中多次按(Tab)键，直到选中‘鼠标’。然后使用(Space)键来打开用于设置鼠标类型的对话框。图1.5“选择鼠标类型”中显示了此对话框。

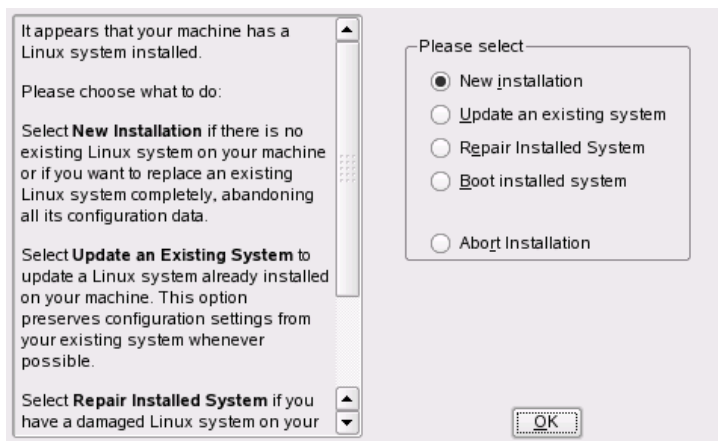


Figure 1.3: 选择安装方式

要选择鼠标类型，请使用 **(?)** 和 **(?)** 键。有关鼠标类型的信息，请参考鼠标的文档。选择鼠标类型后，按 **(Alt)-(T)** 组合键，在未永久性确定选择的情况下测试设备是否工作正常。如果鼠标没有正常工作，请使用键盘选择另一个类型，然后再次测试。使用 **(Tab)** 和 **(Enter)** 键来永久性确定选择。

## 1.5.4 分区

在大多数情况下，YaST 会提出一个无需更改就可以接受的合理的分区方案。也可以使用 YaST 来自定义分区。本节介绍这一过程的必要步骤。

### 分区类型

每个硬盘都有一个分区表，其中有 4 个项。分区表中的一项可以对应于一个主分区或一个扩展分区。但只允许有一个扩展分区项。

主分区由指派给特定操作系统的一系列连续的柱面（物理磁盘区域）组成。仅使用主分区时，限制每个硬盘最多具有 4 个分区，因为超过 4 个分区就不能与分区表相符。这就是使用扩展分区的原因。扩展分区同样是一系列连续的磁盘柱面，但扩展分区本身可以再分为多个逻辑分区。逻辑分区不要求在分区表中有对应的项。换句话说，扩展分区是逻辑分区的容器。

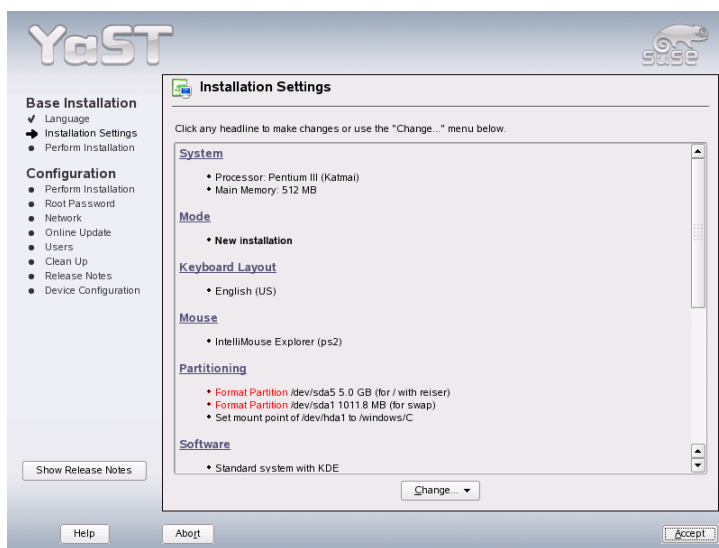


Figure 1.4: 建议窗口

如果需要 4 个以上的分区，请创建一个扩展分区作为第 4 个分区或第 4 个分区之前的分区。这个扩展分区应包括全部剩余的可用柱面范围。然后在扩展分区中创建多个逻辑分区。对于 SCSI、SATA 和 Firewire 磁盘，逻辑分区的最大数目是 15 个，对于 (E)IDE 磁盘是 63 个。对 Linux 使用哪种类型的分区没有什么区别。主分区和逻辑分区都可以。

### Tip

带有 **GPT** 磁盘标签的硬盘

对于使用 GPT 磁盘标签的体系结构，不限制主分区的数目。因此，这种情况没有逻辑分区。

Tip

### 所需磁盘空间

YaST 通常会对充足的磁盘空间提出合理的分区方案。如果希望实施自己的分区方案，应考虑以下与不同系统类型的要求有关的建议。

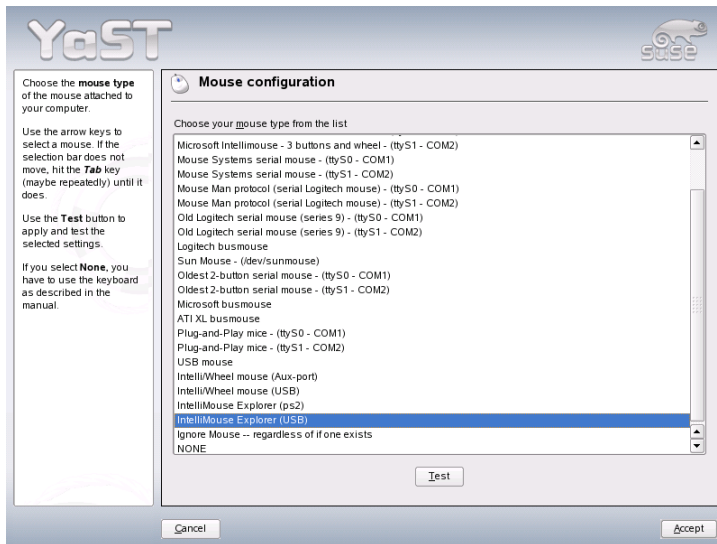


Figure 1.5: 选择鼠标类型

**最小系统：500 MB** 不安装图形界面（X 窗口系统），这意味着只能使用控制台应用程序。而且只选择安装最基本的软件。

**带有图形界面的最小系统：700 MB** 这包括 X 窗口系统和一些应用程序。

**默认系统：2.5 GB** 这包括目前的桌面环境，如 KDE 或 GNOME，并为大型应用程序套件（如 OpenOffice.org、Netscape 或 Mozilla）提供了充足的空间。

要创建的分区取决于可用空间。以下是一些基本的分区原则：

**最大 4 GB：** 一个提供交换空间的分区和一个根分区 (/)。在这种情况下，如果有更多空间可用，根分区就必须为那些经常驻留在它们自己分区上的目录提供空间。

**4 GB 或更多：** 一个交换分区、一个根分区 (1 GB)，以及根据需要与以下每个目录一一对应的分区：/usr（4 GB 或更多）、/opt（4 GB 或更多）和 /var (1 GB)。如果不希望为这些目录提供单独的分区，请向根分区添加建议的磁盘空间。可用空间的其他部分可以用于 /home。

根据硬件的具体情况，也可以创建一个引导分区 (/boot) 来保存引导机制和 Linux 内核，这样做有很大好处。这个分区应位于磁盘的起始位置，而且应至少为 8 MB 或 1 个柱面。作为一条经验性规则，如果在 YaST 的原始建议中包含这样一个分区，则一定要创建该分区。如果对此不太确定，也应创建一个引导分区以确保安全。

还应注意的是，某些（大部分是商业）程序会将数据安装在 /opt 中。因此，要么为 /opt 创建一个单独的分区，要么使根分区足够大。KDE 和 GNOME 也安装在 /opt 中。

### 使用 YaST 进行分区

首次在建议窗口中选择分区项目时，YaST 分区对话框将显示当前建议的分区设置。在继续操作之前，按原样接受这些当前设置或对它们进行更改。也可以丢弃所有这些设置，而从头开始设置。

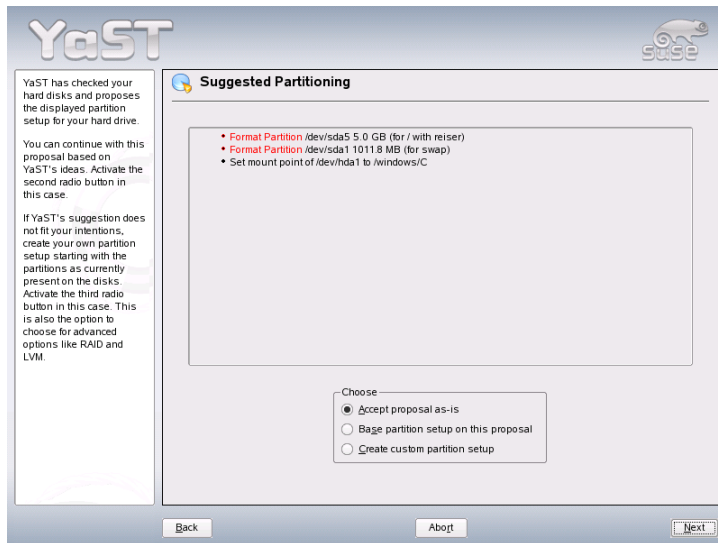


Figure 1.6: 编辑分区设置

如果选择‘按原样接受建议’，则分区设置不会有任何改变。如果选择‘基于此建议调整分区设置’，则将打开‘专家分区工具’。利用它可以精确调整分区设置。第2.8.5节“分区”中介绍了此对话框。这里将 YaST 建议的原始设置作为进行调整的基础。

如果选择‘创建自定义分区设置’，则将打开一个对话框，如图1.7“选择硬盘”所示。使用此列表从系统的现有硬盘中选择一个。SUSE LINUX 将被安装在此对话框中选择的磁盘上。

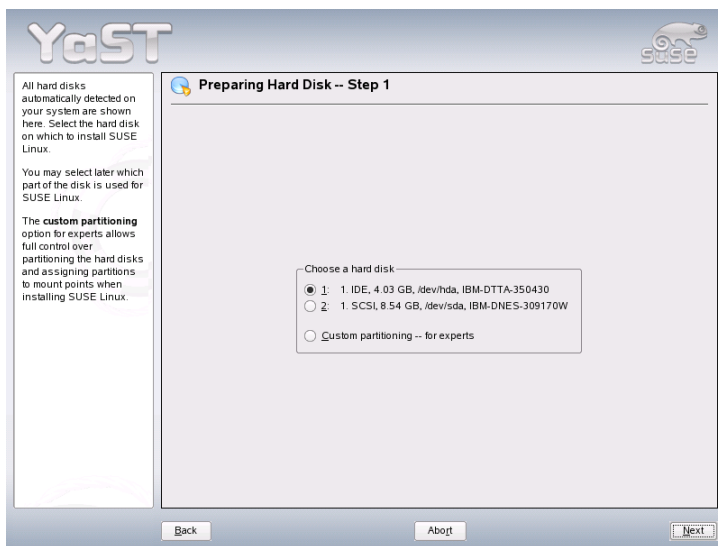


Figure 1.7: 选择硬盘

下一步需要确定的是使用整个磁盘（‘使用整个硬盘’）还是使用任何现有分区（如果有）来进行安装。如果在磁盘上发现 Windows 操作系统，则将询问您是删除此分区还是调整该分区的大小。在执行操作之前，请阅读第1.5.4节“调整 Windows 分区的大小”。如果需要，请转到‘专家分区工具’对话框，在这里创建自定义分区设置（请参见第2.8.5节“分区”）。

### Warning

使用整个硬盘进行安装

如果选择‘使用整个硬盘’，则此磁盘上所有现有的数据在稍后的安装过程中将被完全删除并丢失。

### Warning

YaST 在安装期间会检查磁盘空间是否足够用于安装所选的软件。如果磁盘空间不足，YaST 会自动更改软件选择。这时建议对话框会显示一条相关通知。只要

有足够的磁盘空间可用，YaST 就会接受您的设置并相应地对硬盘进行分区。

### 调整 Windows 分区的大小

如果选择包含 Windows FAT 或 NTFS 分区的硬盘作为安装目标，YaST 将提议删除或收缩此分区。这样，即使当前硬盘上没有足够的空间，您也可以安装 SUSE LINUX。如果所选硬盘只包含一个覆盖整个硬盘的 Windows 分区，则此功能将非常有用。预安装 Windows 的计算机上通常是这种情况。如果 YaST 发现所选硬盘上没有足够的空间，但可以通过删除或收缩 Windows 分区腾出一些空间，则它将显示一个对话框，可以在该对话框中选择以下两个选项之一。

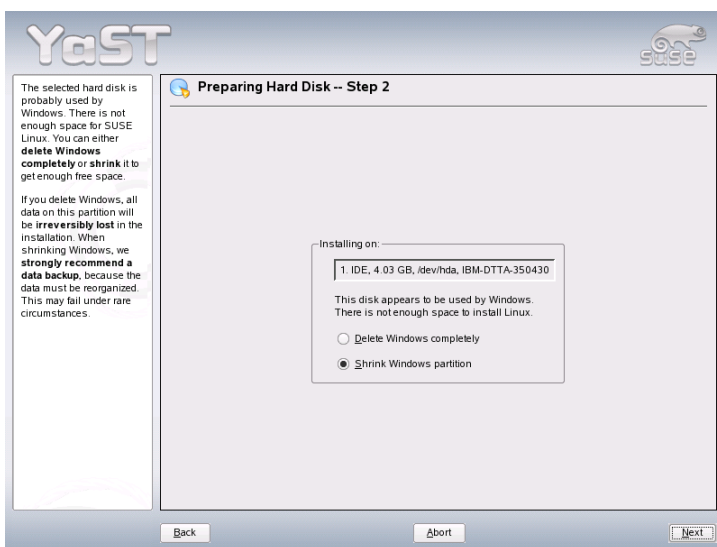


Figure 1.8: 对 Windows 分区进行操作的可能选项

如果选择‘完全删除 Windows’，则会将 Windows 分区标记为要删除，空间将用于安装 SUSE LINUX。

#### Warning

##### 删除 Windows

如果删除 Windows，则在格式化开始后，将丢失所有数据，无法恢复。

Warning



要收缩 Windows 分区，则中断安装并引导 Windows 准备分区。虽然对于 FAT 分区来说此步骤不是必需的，但它可以加快调整进程的速度，同时使之更加安全。对 NTFS 分区来说，这些步骤是必需的。

**FAT 文件系统** 在 Windows 中，首先运行 scandisk 以确保 FAT 分区没有丢失的文件碎片和交联。之后，运行 defrag 将文件移动到分区的开始位置。这将加快在 Linux 中调整大小过程的速度。

如果 Windows 具有优化虚拟内存设置，该设置将连续的交换文件用于相同的初始（最小）和最大大小限制，则考虑使用其他步骤。如果使用这些 Windows 设置，则调整大小可能将交换文件拆分成较小的部分，这些部分将分散在整个 FAT 分区中。另外，调整大小过程中还需要移动整个交换文件，这将使整个过程进度非常缓慢。因此，应该暂时禁用这些 Windows 优化设置，并在调整大小完成后重启用它们。

**NTFS 文件系统** 在 Windows 中，运行 scandisk 和 defrag 将文件移动到硬盘的开始位置。与 FAT 文件系统不同的是，您必须执行这些步骤。否则将不能调整 NTFS 分区的大小。

## Important

### 禁用 Windows 交换文件

如果在 NTFS 文件系统上运行具有永久交换文件的系统，则此文件可能位于硬盘的末尾，即使在执行 defrag 后可能仍保存在原位置。因此，可能无法充分收缩分区。在这种情况下，请临时取消激活该交换文件（Windows 中的虚拟内存）。在调整了分区的大小后，重配置虚拟内存。

## Important

在完成这些准备之后，返回 Linux 分区设置并选择‘收缩 Windows 分区’。在快速检查分区之后，YaST 将打开一个显示有调整 Windows 分区大小建议的对话框。

第一个条形图显示 Windows 当前占用了多少磁盘空间以及仍有多少空间可用。第二个条形图显示根据 YaST 的当前建议，调整大小后空间如何分布。请参见图 1.9 “调整 Windows 分区的大小”。接受建议的设置或使用滑块更改分区大小（在特定限制内）。

如果通过选择‘下一步’退出该对话框，则将储存该设置并将返回到上一个对话框。实际调整大小将在稍后进行，之前将先对硬盘进行格式化。

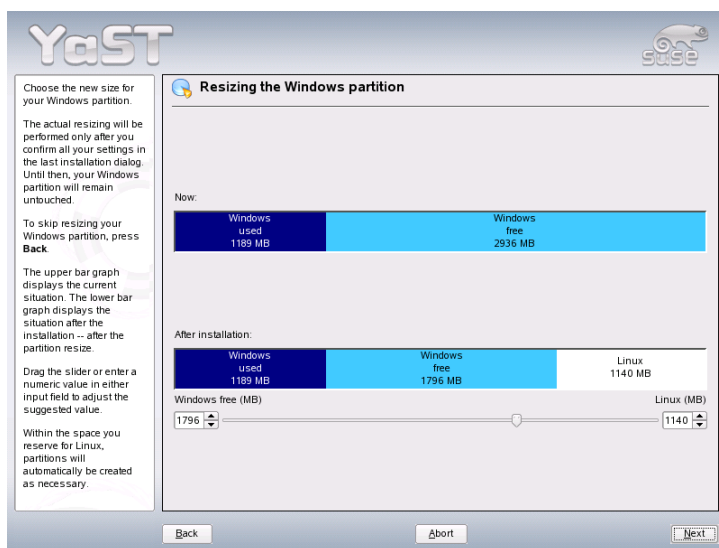


Figure 1.9: 调整 Windows 分区的大小

## Important

### 安装在 NTFS 分区上的 Windows 系统

默认情况下，Windows 版本 NT、2000 和 XP 使用 NTFS 文件系统。与 FAT 文件系统不同，只能从 Linux 读取 NTFS 文件系统。这意味着您可以从 Linux 读取您的 Windows 文件，但不能对它们进行编辑。如果需要对 Windows 数据进行写访问并且不需要 NTFS 文件系统，则请在 FAT32 文件系统中重新安装 Windows。这样，您将具有从 SUSE LINUX 对 Windows 数据的完全访问权限。

Important

## 1.5.5 软件

SUSE LINUX 包含许多用于各种应用目的的软件包。由于逐个选择所需的包会非常繁琐，SUSE LINUX 提供了 3 种具有不同安装范围的系统类型。根据可用的磁盘空间，YaST 会从这些预定义系统中选择一个，然后将它显示在建议窗口中。

**最小系统**（建议仅用于特殊目的） 它基本上只包括核心操作系统和各种服务，没有任何图形用户界面。这时只能使用 ASCII 控制台来操作计算机。此系统类型尤其适用于需要很少直接用户交互的服务器方案。

#### 最小图形系统（无 GNOME 和 KDE）

如果不需要 KDE 和 GNOME 桌面或磁盘空间不足，则可以安装此系统类型。所安装的系统包括 X 窗口系统和一个基本窗口管理器。您可以使用具有图形用户界面的所有程序。不安装任何办公程序。

#### 默认系统（带有 GNOME 和办公套件）

这是最大的预定义系统之一。它包括 GNOME 桌面以及大部分 GNOME 程序和办公程序。

**默认系统（带有 KDE 和办公套件）** 此系统包括 KDE 桌面以及大部分 KDE 程序和办公程序。

在建议窗口中单击‘软件’打开一个对话框，然后在其中选择一个预定义系统。要启动软件安装模块（包管理器）并修改安装范围，请单击‘详细选择’。请参见图 1.10 “使用 YaST 包管理器安装和去除软件”。

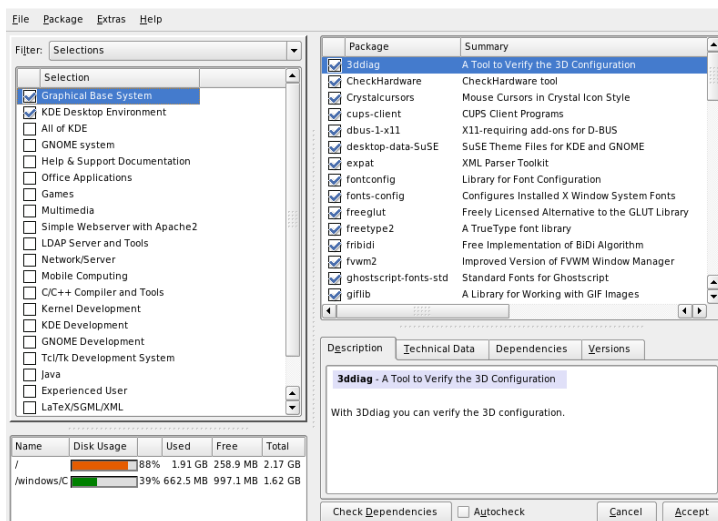


Figure 1.10: 使用 YaST 包管理器安装和去除软件

## 更改安装范围

如果安装默认系统，通常无需添加或去除单独的包。它包括一个软件选择，无需更改即可满足大多数要求。如果有特殊需要，请使用包管理器修改此选择，它可以极大地简化此任务。它提供了多种过滤条件来简化对 SUSE LINUX 中众多包的选择。

过滤器选择框位于菜单栏下的左上方。在启动后，‘选择’过滤器处于活动状态。此过滤器根据应用程序用途对程序包进行排序，如多媒体或办公应用程序。这些组被列在过滤器选择框下。此时预先选择了当前系统类型中包括的包。单击相应的复选框来选择或取消选择要安装的全部选择或组。

窗口右侧显示一个表，其中列出当前选择中包括的各个包。表最左侧的一列中显示每个包的当前状态。以下两个状态标志对安装而言尤为重要：‘安装’（包名称前的复选框被选中）和‘不安装’（复选框未被选中）。要选择或取消选择单个软件包，请单击状态框，直到显示所需的状态。此外，右击包行可以访问一个弹出菜单，其中列出所有可能的状态设置。有关它们的更多信息，请参见第2.3.1节“安装和去除软件”中对此模块的详细介绍。

## 其他过滤器

单击过滤器选择框可以查看其他可用的过滤器。根据‘包组’进行的选择也可以用于安装。此过滤器按照主题将各程序包在左侧的树结构中进行排序。将分支展开得越深入，对包的选择就越具体，在右侧的相关包列表中显示的包就越少。

使用‘搜索’可以搜索某个特定的包。第2.3.1节“安装和去除软件”中对此有详细介绍。

## 包依赖性和冲突

您不能安装软件包的任意组合。要安装的不同软件包必须相互兼容。否则它们可能会互相影响并造成冲突，从而影响整体系统。因此，在此对话框中选择或取消选择软件包后，可能会看到有关未解决的包依赖性 or 冲突的警报。如果首次安装 SUSE LINUX 或不理解这些警报，请参见第2.3.1节“安装和去除软件”，它提供了有关包管理器操作的详细信息并概述了 Linux 中的软件组织。

### Warning

我们根据长期的经验预先选择了要进行安装的软件，它们通常可满足大多数新手和高级家庭用户的需要。通常，无需在这里进行任何更改。但是，如果您决定选择或取消选择任何包，就必须清楚这样做的后果。您特别要注意所有警告，并避免取消选择基础系统的任何包。

Warning

## 退出软件选择

如果对软件选择满意并解决了所有包依赖性冲突，请单击“接受”来应用您的更改并退出此模块。如果是在安装期间，这些更改将被内部记录，并在实际安装启动后应用。

## 1.5.6 引导配置

在安装期间，YaST 会为您的系统建议引导配置。通常，您可以保持这些设置不变。但是，如果您需要自定义设置，则可修改对系统的建议。

您可以配置引导机制，使其依赖于特殊的引导软盘。尽管这意味着引导时引导软盘必须在驱动器内，但它不会影响现有的引导机制。但通常情况下不需要这样做，因为 YaST 也可以配置引导装载程序来引导其他现有的操作系统。此外，还可以更改硬盘上的引导机制的位置。

要更改 YaST 建议的引导配置，请选择“引导”打开一个对话框，可以在其中更改引导机制的许多细节。有关信息，请参见第8.4节“使用 YaST 配置引导装载程序”。只有有经验的计算机用户才能更改引导方法。

## 1.5.7 时区

在此对话框中（如图1.11“选择时区”所示），在“硬件时钟设置”下选择当地时间或 UTC。此选择取决于计算机上设置硬件 (BIOS) 时钟的方式。如果将它设置为与 UTC 相对应的 GMT，则您的系统便可以依赖 SUSE LINUX 在标准时间和夏时制之间自动切换。

## 1.5.8 语言

我们在安装开始时就选择了语言，如第1.3节“语言选择”中所述。但也可以在这里更改此设置并选择要在系统上安装的任何其他语言。在此对话框的上部，选择主要语言。这是将在安装后激活的语言。如果需要，通过选择相应的选中标记使键盘和时区设置适应所选的主要语言。此外，可以使用“细节”来设置用户 root 的语言。这里有 3 个选项：

**ctype only** 为用户 root 采用文件 `/etc/sysconfig/language` 中变量 `LC_CTYPE` 的值。它设置了语言特定的功能调用的本地化。

**yes** root 用户具有与本地用户相同的语言设置。

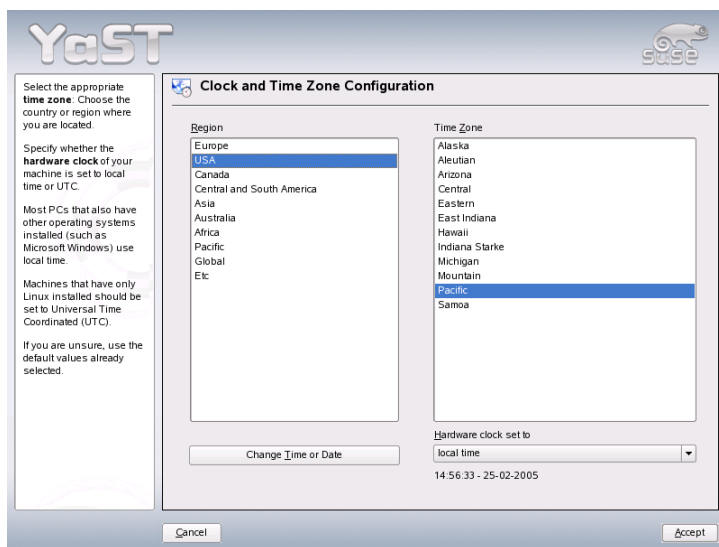


Figure 1.11: 选择时区

**no** root 用户的语言设置不受语言选择的影响。将取消所有 locale 变量的设置。

某些系统管理员不希望 root 帐户通过 UTF-8 多语言支持运行。如果这样，请取消选中‘使用 UTF-8 编码’。

可以在对话框下部的列表中选择要安装的其他语言。对于在此列表中选择的所有语言，YaST 检查当前软件选择中的任何包是否有任何语言特定的包。如果有，则将安装这些包。

单击‘接受’完成配置。单击‘取消’撤消您的更改。

## 1.5.9 启动安装

在指定所有安装设置后，在建议窗口中单击‘下一步’开始安装。在打开的对话框中单击‘是’进行确认。根据系统性能和所选的软件，安装通常需要 15 到 30 分钟的时间。在安装了所有包后，YaST 会立即引导新的 Linux 系统，此后您就可以配置硬件和设置系统服务了。

## 1.6 完成安装

在完成基本系统设置和所有所选软件包的安装后，请为系统管理员帐户（root 用户）提供一个口令。然后便可以配置自己的因特网访问和网络连接了。利用有效的因特网连接，您可以将系统更新作为安装的一部分来执行。您还可以配置一个鉴定服务器，以便在本地网络中集中管理用户。最后，配置连接到计算机的硬件设备。

### 1.6.1 根口令

root 是超级用户（即系统管理员）的名称。普通用户不一定有在系统上执行某些操作的权限，而 root 用户则有权在系统上执行任何操作：更改系统配置、安装程序以及设置新硬件。如果用户忘记他们的口令或遇到其他有关系统的问题，root 用户可以提供帮助。root 帐户应只用于系统管理、维护和修复工作。以 root 身份登录来进行日常工作风险较高：一个错误就会导致许多系统文件丢失而且无法挽回。

为了进行校验，必须两次输入 root 用户的口令，如图 1.12 “设置根口令” 所示。切勿忘记 root 口令。一旦输入，就不能在系统中检索此口令。

#### Warning

##### root 用户

用户 root 拥有更改系统所需的所有权限。要执行这样的任务，需要有 root 口令。没有此口令就不能执行任何管理任务。

#### Warning

### 1.6.2 网络配置

您现在可以配置用于连接到外部世界的任何网络设备，如网卡、调制解调器以及 ISDN 或 DSL 硬件。如果有这样的设备，最好现在就配置它们，因为因特网连接允许 YaST 检索任何可用的 SUSE LINUX 更新程序并将它们包括在安装中。

要在此阶段配置您的网络硬件，请参考第 22.4 节 “使用 YaST 配置网络连接”。否则，请选择“跳过配置”，然后单击“下一步”。网络硬件的配置工作也可以在系统安装完毕后进行。

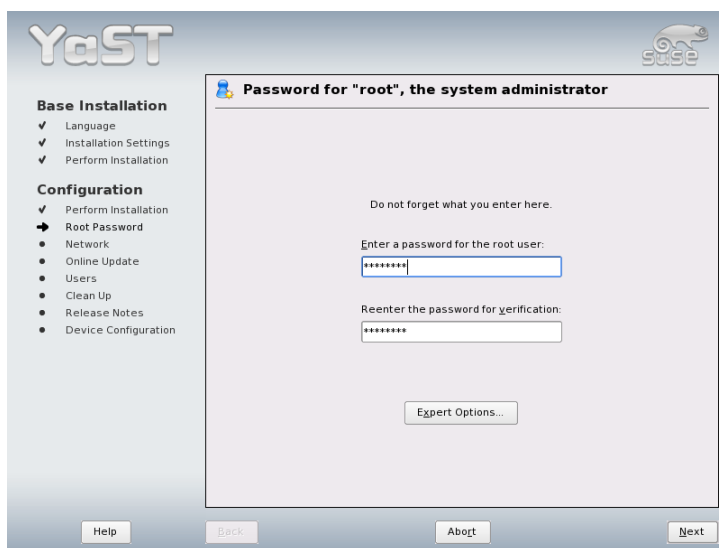


Figure 1.12: 设置根口令

### 1.6.3 防火墙配置

当您连接到网络时，将自动在已配置的接口上启动防火墙。防火墙设置显示在网络配置对话框中。每次修改接口或服务的配置时，都将自动更新防火墙的配置建议。要根据您的偏好调整自动设置，请单击‘更改’→‘防火墙’。在新对话框中，确定是否应启动防火墙。如果不希望启动防火墙，请选择适当的选项并退出对话框。要启动和配置防火墙，请单击‘下一步’进入类似于第34.1.4节“用 YaST 进行配置”中所介绍的一系列对话框。

### 1.6.4 测试因特网连接

如果已经配置了因特网连接，现在就可以测试它。为此，YaST 建立了与 SUSE 服务器的连接，并检查是否有可用于您的版本的 SUSE LINUX 产品更新。如果存在这样的更新，则可以将其包括在安装中。同时下载最新的发行说明。您可以在安装结束时看到这些说明。

如果不希望此时测试连接，请选择‘跳过测试’，然后单击‘下一步’。这样还将跳过下载产品更新和发行说明。



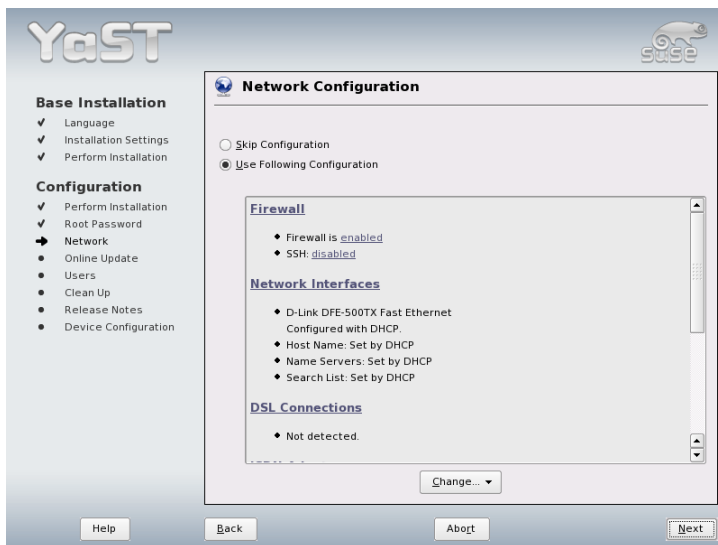


Figure 1.13: 配置网络设备

## 1.6.5 装载软件更新程序

如果 YaST 能够连接到 SUSE 服务器，请选择是否执行 YaST 联机更新。如果这些服务器上有任何增补程序包，请立即下载并安装它们，以修复已知错误或安全问题。

### Important

#### 下载软件更新程序

根据因特网连接的带宽和更新文件的大小，更新程序的下载可能需要一些时间。

### Important

要立即执行软件更新，请选择‘立即执行更新’，然后单击‘确定’。此时将打开 YaST 的联机更新对话框，其中显示可选择和装载的可用增补程序（如果存在）。有关详细信息，请参见第 2.3.3 节“YaST 联机更新”。此类更新可在安装后的任何时间执行。如果不希望立即更新，请选择‘跳过更新’，然后单击‘确定’。

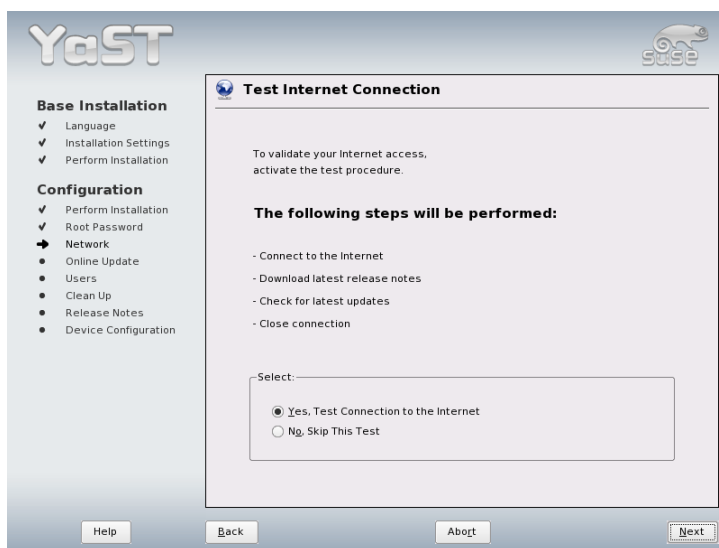


Figure 1.14: 测试因特网连接

## 1.6.6 用户鉴定

如果在前面的安装步骤中成功配置了网络访问，则现在可以选择 4 种方式来管理系统上的用户帐户。

**本地用户管理** 在已安装的主机上对用户进行本地管理。此选项适用于独立工作站。用户数据由本地文件 `/etc/passwd` 来管理。

**LDAP** 在 LDAP 服务器上对网络中的所有系统进行集中用户管理。

**NIS** 在 NIS 服务器上对网络中的所有系统进行集中用户管理。

**Samba** 在混合 Linux 和 Windows 网络中经常使用 SMB 鉴定。

如果满足所有要求，YaST 将会打开一个用于选择用户管理方法的对话框。图1.15 “用户鉴定”中显示了这一工具。如果没有必需的网络连接，则请创建本地用户帐户。

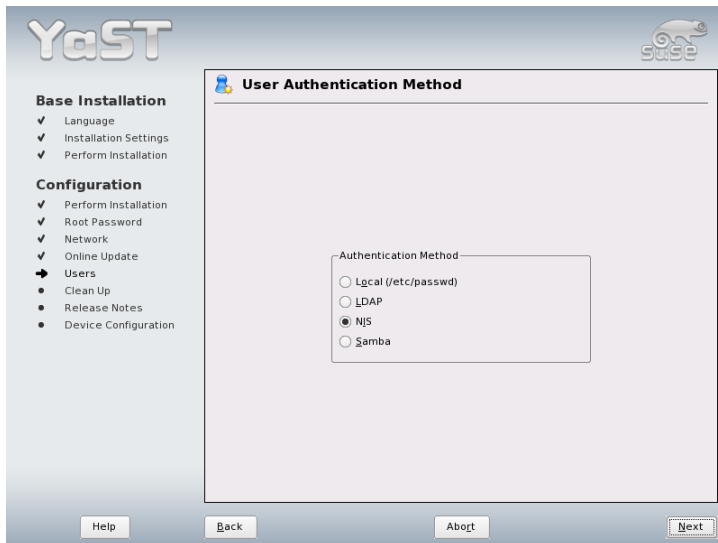


Figure 1.15: 用户鉴定

## 1.6.7 将主机配置为 NIS 客户机

要通过 NIS 实施用户管理，请在下一步配置 NIS 客户机。本节仅介绍客户端的配置。有关如何使用 YaST 配置 NIS 服务器的信息，请参见第 25 章 使用 NIS。

在以下对话框（如图 1.16 “NIS 客户机配置”所示）中，首先选择主机是具有静态 IP 地址，还是通过 DHCP 获得 IP 地址。如果选择 DHCP，就不能指定 NIS 域或 NIS 服务器地址，因为它们是由 DHCP 服务器提供的。有关 DHCP 的信息，请参见第 27 章 DHCP。如果使用静态 IP 地址，请手工指定 NIS 域和 NIS 服务器。

要搜索网络中的 NIS 服务器广播，请选中相关选项。您也可以指定多个 NIS 域并设置一个默认域。对于每个域，可选择“编辑”来指定多个服务器地址或按域启用广播功能。

在专家设置中，使用“仅回复本地主机”使其他网络主机不能查询您的客户机正在使用哪台服务器。如果激活“断开服务器”，则还将接受来自未授权端口上的服务器的响应。有关详细信息，请参考 `ypbind` 的手册页。

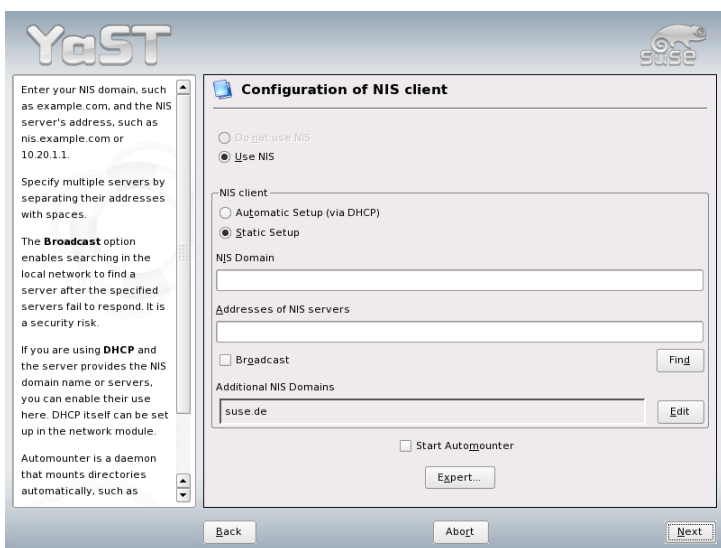


Figure 1.16: NIS 客户机配置

### 1.6.8 创建本地用户帐户

如果决定不使用鉴定服务器来进行用户鉴定，请创建本地用户。在已安装系统上储存和管理与用户帐户相关的任何数据（姓名、登录名和口令等）。

Linux 是一种允许多个用户同时同一系统上工作的操作系统。每个用户都需要用户帐户来登录系统。通过用户帐户，系统在安全性方面有很大提高。例如，普通用户不能更改或删除系统正常工作所需的文件。同时，其他用户也不能修改、查看或篡改给定用户的个人数据。用户可以设置自己的工作环境，而且这些环境在该用户以后登录时保持不变。

可以使用如图1.17“输入用户名和口令”所示的对话框创建用户帐户。在输入用户姓名后，指定一个用户名（登录名）。单击‘建议’，系统将自动生成一个用户名。

最后，输入此用户的口令。重新输入以进行确认（确保您的输入无误）。用户名向系统指示用户的身份，并使用口令对此进行校验。

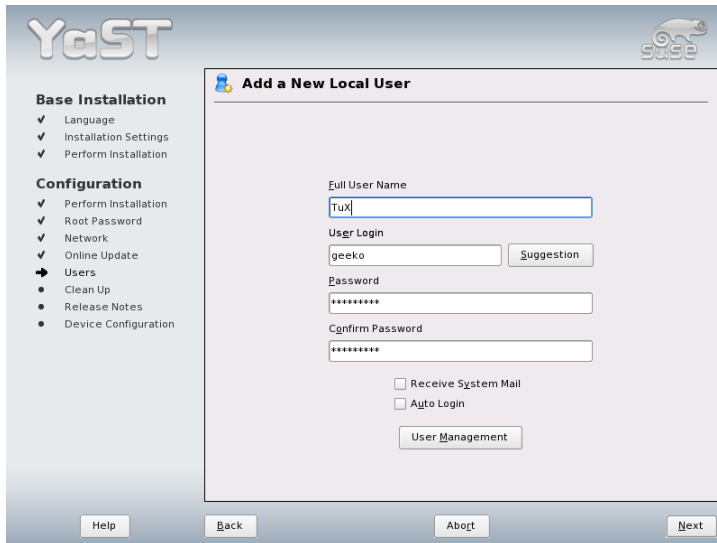


Figure 1.17: 输入用户名和口令

## Warning

用户名和口令

记住用户名和口令，因为每次登录系统都需要输入用户名和口令。

## Warning

为了提供有效的安全性，口令长度应在 5 到 8 个字符之间。口令的最大长度是 128 个字符。但是，如果没有装载特殊的安全性模块，则仅使用前 8 个字符来识别口令。口令区分大小写。不允许使用变音符号等特殊字符。允许使用其他特殊字符（7 位 ASCII）和数字 0 到 9。

本地用户有两个附加选项可用：

‘通过电子邮件接收系统讯息’ 如果选中此复选框，则将向您发送系统服务创建的讯息。这些讯息通常只发送到 root 用户（即系统管理员）。此选项对于最常用的帐户很有用，因为我们建议您仅在特殊情况下以 root 身份登录。

‘自动登录’此选项仅当 KDE 为默认桌面时才可用。如果选择此选项，系统在启动时就会自动将当前用户登录到系统。如果计算机只由一个用户操作，此选项很有用。

## Warning

### 自动登录

启用自动登录后，系统在引导时根本不进行鉴定就直接进入桌面。当您在系统上储存敏感数据时，如果其他用户也可以访问此计算机，则不应启用此选项。

## Warning

## 1.6.9 发行说明

完成用户鉴定设置后，YaST 即显示发行说明。建议您阅读这些发行说明，因为它们包含手册印刷时所没有的重要的最新信息。如果已经安装了更新软件包，则读到的就是最新版本的发行说明，与从 SUSE 的服务器获取的发行说明相同。

## 1.7 硬件配置

在安装结束时，YaST 会打开一个对话框，用于配置图形卡和连接到系统的其他硬件部件，例如打印机或声卡。单击相应部件来启动硬件配置。YaST 在很大程度上会自动检测和配置设备。

您可以跳过任何外围设备，并在以后配置它们。但您应立即配置图形卡。尽管一般情况下可以接受 YaST 自动配置的显示设置，但就分辨率、颜色深度以及其他图形功能而言，大多数用户都会有明显的个人偏好。要更改这些设置，请选择‘图形卡’。第 11.1 节“使用 SaX2 设置 X11”对此配置进行了说明。在 YaST 写入配置数据后，在最后的对话框中选择‘完成’来完成 SUSE LINUX 的安装。

## 1.8 图形登录

现在已安装了 SUSE LINUX。如果在本地用户管理模块中启用自动登录，就可以不经过登录而启动系统。如果没有，则将在屏幕上看到图形登录，如图 1.19 “KDM 的登录屏幕”所示。输入登录名和口令来登录系统。

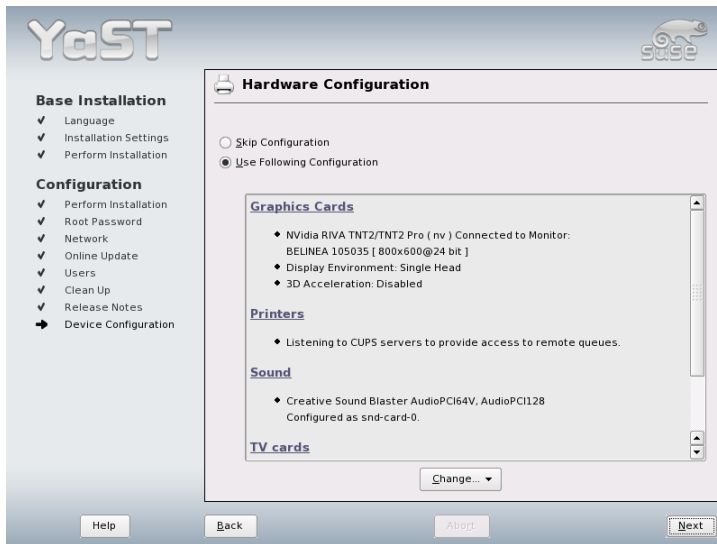
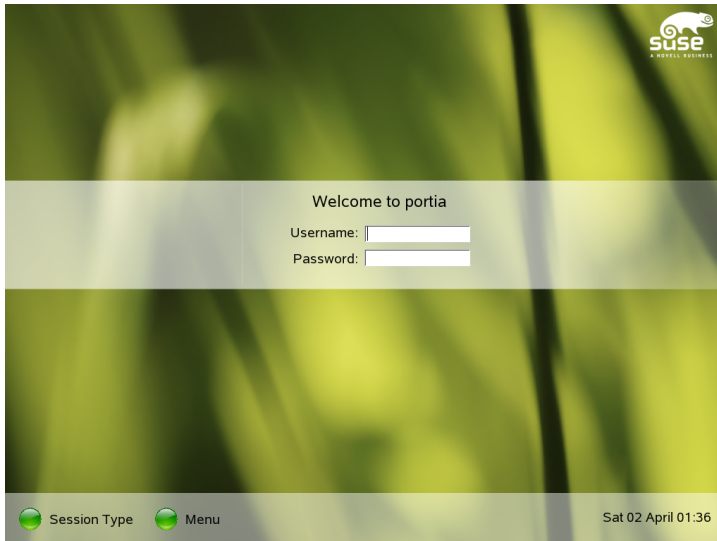


Figure 1.18: 配置系统部件



*Figure 1.19: KDM 的 登 录 屏 幕*



# 使用 YaST 进行系统配置

YaST 不仅是用于安装的安装工具，同时也是 SUSE LINUX 的配置工具。本章将介绍如何使用 YaST 配置您的系统。其中包括大部分硬件、图形用户界面、因特网访问、安全性设置、用户管理、软件安装、系统更新和系统信息。本章还介绍了如何在文本方式下使用 YaST。

2.1	启动 YaST . . . . .	32
2.2	YaST 控制中心 . . . . .	32
2.3	软件 . . . . .	33
2.4	硬件 . . . . .	44
2.5	网络设备 . . . . .	49
2.6	网络服务 . . . . .	50
2.7	安全性和用户 . . . . .	52
2.8	系统 . . . . .	56
2.9	其他 . . . . .	64
2.10	文本方式下的 YaST (ncurses) . . . . .	65
2.11	从命令行联机更新 . . . . .	69

## 2.1 启动 YaST

使用 YaST 进行系统配置是通过多个不同的 YaST 模块进行的。根据硬件平台和所安装的软件，将用不同的方法来访问已安装系统中的 YaST。

在 KDE 或 GNOME 中，从 SUSE 菜单启动 YaST 控制中心（‘系统’→‘YaST’）。此外，各个 YaST 配置模块被集成在 KDE 控制中心。在 YaST 启动之前，计算机将提示您输入根口令，因为 YaST 需要系统管理员权限来更改系统文件。

要从命令行启动 YaST，请输入命令 `su`（用于更改为 `root` 用户）和 `yast2`。要以文本方式启动 YaST，请输入 `yast` 而非 `yast2`。此外，也可以使用命令 `yast` 来从虚拟控制台启动此程序。

### Tip

要更改 YaST 的语言，请在 YaST 控制中心中选择‘系统’→‘选择语言’。选择语言，退出 YaST 控制中心并从系统中注销，然后再次登录。下次启动 YaST 时，新语言设置即开始生效。

### Tip

如果硬件平台不支持它们自己的显示设备，并要在其他主机上进行远程管理，应远程运行 YaST。首先，在主机上打开要显示 YaST 的控制台，然后输入命令 `ssh -X root@<system-to-configure>` 来登录到要配置 `root` 的系统，并将 X 服务器输出重定向到您的终端。在 SSH 成功登录后，输入 `yast2` 以图形方式启动 YaST。

要在另一系统上以文本方式启动 YaST，请使用 `ssh root@<system-to-configure>` 来打开连接。然后使用 `yast` 启动 YaST。

## 2.2 YaST 控制中心

以图形方式启动 YaST 时，会打开 YaST 控制中心，如图 2.1 “YaST 控制中心”所示。左框架中包含‘软件’、‘硬件’、‘系统’、‘网络设备’、‘网络服务’、‘安全性和用户’、‘系统’和‘其他’等类别。如果单击这些图标，就会在右侧列出其内容。然后选择所需元素。例如，如果选择‘硬件’并单击右侧的‘声卡’，就会打开声卡的配置对话框。各个项目的配置通常需要多个步骤。按‘下一步’继续进行下一步骤。

大多数模块的左框架显示帮助文本，用于介绍所需的项。要在模块中获得不带帮助框架的帮助，请按 **(F1)** 键或在菜单中选择‘帮助’。在完成所需设置后，请在最后一个配置对话框中单击‘完成’，结束配置过程。同时会保存所做的配置。

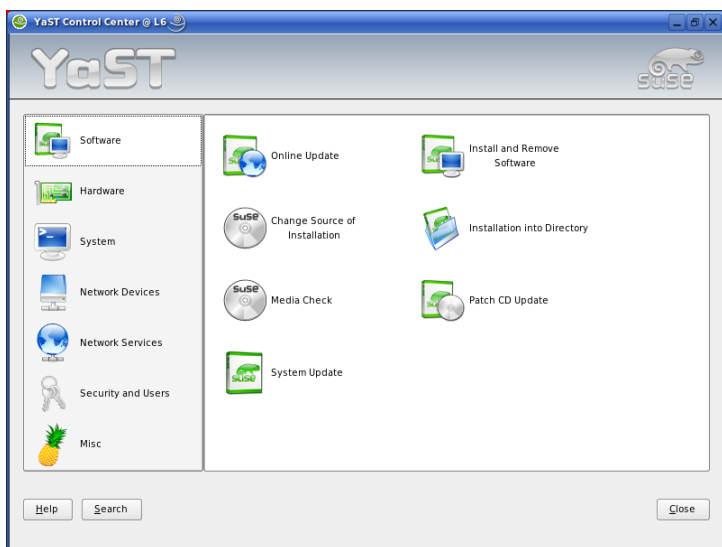


Figure 2.1: YaST 控制中心

## 2.3 软件

### 2.3.1 安装和去除软件

此模块用于在您的计算机上安装、卸载和更新软件。在 Linux 中，软件是以包的形式供用户使用的。通常情况下，包中包含一个程序所需的所有内容：程序本身、配置文件和文档。通常还提供一个包含程序源文件的包。这些源文件不是运行程序所需要的，但您可能需要安装这些源文件来编译程序的自定义版本。

某些包依赖于其他包。这意味着这些包中的软件只有在其他包已安装的情况下才能正常工作。此外，某些包只有在其他包已安装的情况下才能进行安装，原因可能是安装例程需要某些特定工具。因此必须以正确顺序安装这样的包。还有某些包具有相同或类似的功能。如果这些包使用相同的系统资源，就不应同时安装它们（包冲突）。在两个或多个包之间可能会存在依赖性 or 发生冲突，而且有时会非常复杂。为了实现顺畅交互，我们可能会需要特定的包版本，这会使得事情变得更为复杂。

在安装、卸载和更新软件时必须考虑所有这些因素。YaST 为解决这一问题提供了一种极为有效的工具：软件安装模块，通常称为包管理器。在启动包管理器

时，它会检查系统并显示已安装的包。如果选择其他的包进行安装，包管理器会自动检查依赖性并选择所需要的任何其他包（解决依赖性）。如果选择了存在冲突的包，包管理器会指出存在冲突并提供解决此问题的建议（解决冲突）。如果其他已安装的包所需要的包被标记为要进行删除，包管理器就会发出一个警报，提供详细信息和可选解决方案。

除了这些纯技术方面外，包管理器还为 SUSE LINUX 中的各种包提供了一个结构清晰的概述。这些包按主题来排列，并通过适当的过滤器来限制这些组的显示。

## 包管理器

要使用包管理器更改您的系统上的软件选择，请在 YaST 控制中心中选择‘安装或去除软件’。包管理器的对话框如图 2.2 “YaST 包管理器” 所示。这个窗口包含多个框架。通过单击并移动各区域的分隔线可修改框架大小。下面介绍这些框架的内容及其用途。

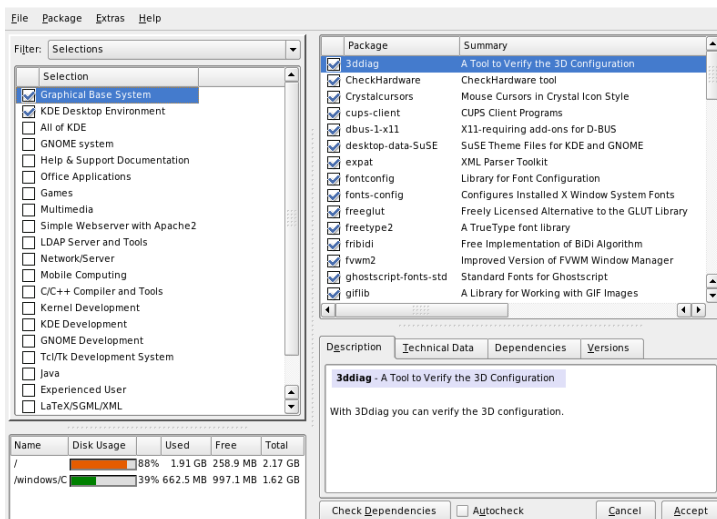


Figure 2.2: YaST 包 管 理 器

## 过滤器窗口

为了按类别排列包并限制所显示的包的数目，包管理器提供了多种过滤方法。过

过滤器窗口位于菜单栏的左下侧。它控制和显示多种不同的过滤方法。顶部的过滤器选择框决定着过滤器窗口下部将显示哪些内容。单击过滤器选择框，从可用过滤器列表中选择一个过滤器。

**选择过滤器** 在启动时，‘选择’过滤器处于活动状态。这个过滤器根据用途（如多媒体或办公应用程序）对程序包进行分组。在过滤器选择框下将列出多组‘选择’过滤器。这时将预先选择系统上已安装的包。单击行开头的状态框可切换选择的状态标志。通过鼠标右击选择和使用上下文菜单来直接选择状态。右侧的包列表窗口显示包含在当前选择中的包的列表，使您可选择和取消选择各包。

**包组过滤器** ‘包组’过滤器为各种包提供了一个更侧重技术方面的概述，适合熟悉 SUSE LINUX 的包结构的用户使用。这个过滤器按照主题（如应用程序、开发和硬件）在左侧以树结构对程序包进行排序。将分支展开得越深入，选择就越具体，在右侧包列表窗口中显示的包就越少。

此外，这个过滤器还提供了不进行分类而按字母顺序来显示所有包的功能。要做到这一点，请在顶级菜单中选择‘按字母顺序列出全部’。由于 SUSE LINUX 包含的包很多，因此可能需要一些时间来显示这个长列表。

**搜索功能** ‘搜索’功能是查找特定包最简单的方法。在指定多个搜索条件时，可能会过多地限制了过滤器，以致于在包列表窗口常常只显示一个包。输入搜索字符串，然后使用复选框来确定搜索该字符串的位置（即是在名称、说明还是包依赖性中）。高级用户可以使用通配符和常规表达式来定义特殊的搜索模式，以及在‘提供’和‘所需’字段中搜索包依赖性。例如，可以使用此功能来确定哪个包中包含某个特定库。

## Tip

### 快速搜索

除了‘搜索’过滤器外，包管理器的所有列表都具有快速搜索功能。只需输入一个字母，光标就会移动到列表中名称以此字母开头的第一个包。光标必须位于列表中（通过单击列表）。

## Tip

**语言** 对于 SUSE LINUX 中的某些包，提供了语言特定的包，例如程序的用户界面的翻译的文本、文档和字体。此过滤器在左边的窗口中显示 SUSE LINUX 支持的所有语言的列表。如果选择列出的语言之一，则右边的窗口显示此语言可用的所有包。在这些包之中，将自动标记应用于当前软件选择的所有包以进行安装。

## Note

因为语言特定的包可能依赖于其他的包，所以在某些情况下，包管理器将选择其他包进行安装。

## Note

**安装摘要** 在选择要安装、更新或删除的包后，可使用过滤器选择来查看安装摘要。它将显示当您单击‘接受’时会执行哪些操作。使用左侧的复选框来过滤要在包列表窗口中显示的包。例如，要查看已经安装了哪些包，请启动包管理器并取消‘保持’之外的所有复选框。

可以按常规方式更改包列表窗口中的包状态。但是，相应的包可能就不再满足搜索条件。要从列表中去除这样的包，请使用‘更新列表’来更新列表。

## 包列表窗口

如上所述，在包列表窗口右侧将显示各个包的列表。此列表的内容由当前选择的过滤器确定。例如，如果选择了‘选择’过滤器，则包列表窗口将显示当前选择的所有包。

在包管理器中，每个包都有一个状态，它决定着要对包执行的操作，如“安装”或“删除”。此状态通过位于行开头的状态框中的一个符号来显示。要切换某项目的状态，请右击此项目，然后从打开的菜单中单击或选择所需状态。根据当前情况，可能不能选择某些状态标志。例如，不能将尚未安装的包设置为“删除”。请使用‘帮助’→‘符号’来查看可用状态标志。

包管理器提供以下包状态标志：

**不要安装** 尚未安装且将不会安装此包。

**安装** 尚未安装但将会安装此包。

**保持** 已经安装此包，而且将不会更改它。

**更新** 已经安装此包，而且将用安装媒体上的版本来替换它。

**删除** 已经安装此包，而且将删除它。

**禁忌—切勿安装** 没有安装而且以后也永远不会安装此包。它将被视为不存在于任何安装媒体上。如果某个包会被自动选择用于解决依赖性，则可以通过将此包设置为“禁忌”来防止出现这种情况。但这样可能会导致必须手工解决某些不一致（依赖性检查）。因此，“禁忌”主要供专家用户使用。

**受保护** 已安装此包，而且不应修改它。计算机会为第三方包（没有 SUSE 签名的包）自动指派此状态，以防止安装介质上的更高版本将其覆盖。这样可能会导致必须手工解决某些包冲突。

**自动安装** 由于另一个包的需要，已自动选择此包来进行安装（解决包依赖性）。要取消选择这样的包，可能需要使用“禁忌”状态。

**自动更新** 已经安装此包。但由于另一个包需要此包的较新版本，系统将自动更新已安装版本。

**自动删除** 已经安装此包，但现存的包冲突要求删除此包。例如，在当前包已经被另一个包替换时就可能会发生这种情况。

**自动安装（选择后）** 由于此包是预定义的选择（如“多媒体”或“开发”）的一部分，已自动选择它来进行安装。

**自动更新（选择后）** 已经安装了此包，但安装媒体上存在此包的一个较新版本。此包是预定义的选择（如选择进行更新的“多媒体”或“开发”）的一部分，将被自动更新。

**自动删除（选择后）** 已经安装此包，但预定义的选择（如“多媒体”或“开发”）要求删除此包。这种情况不会经常发生。

此外，您可以确定是否安装包的源文件。这些信息可补充当前包状态，而且不能用鼠标进行切换或从上下文菜单直接选择。可以使用包行末尾的复选框来选择源包。也可在‘包’下面访问此选项。

**安装源** 同时安装源代码。

**不安装源** 不安装源代码。

包列表窗口中各个包所用的字体颜色提供了附加信息。安装媒体上存在有较新版本的已安装包显示为蓝色。版本号高于安装媒体上的版本的已安装包显示为红色。但是，因为包的版本编号不总是线性的，这些信息可能会不完整，但足以指出有问题的包。如有必要，可在信息窗口中检查版本号。

## 信息窗口

右下方框架中的选项卡提供了有关所选包的各种信息。所选包的说明自动处于活动状态。请单击其他选项卡来查看技术数据（包大小、组等）、此包所依赖的其他包的列表或版本信息。

## 资源窗口

在选择软件期间，左下方的资源窗口会显示所有已装入文件系统的未来用途。每次选择后，带颜色的条形图都会增长。只要它保持为绿色，就表明仍有足够的空间。随着不断接近磁盘空间上限，条柱的颜色会逐渐变为红色。如果选择安装的包过多，就会显示一个警报。

## 菜单栏

利用位于窗口左上部的菜单栏可以访问上述大多数功能，该菜单栏包含以下 4 个菜单：

**文件** 选择‘文件’→‘导出’将所有已安装包的列表保存在一个文本文件中。如果希望以后或在其他系统上复制某个特定的安装范围，建议您这样做。可以使用‘导入’来导入以这种方式生成的文件，同时生成与保存时相同的包选择。在这两种情况下，都要定义文件的位置或接受建议。

如果要退出包管理器而不将更改保存到包选择，请单击‘退出—放弃更改’。要保存更改，请单击‘退出—保存更改’。这样将应用所有的更改并终止程序。

**包** ‘包’菜单中的项目始终与包列表窗口中当前选择的包相关。虽然会显示所有的状态标志，但您只能选择当前包可用的状态标志。使用复选框来确定是否安装包的源文件。单击‘所有包状态标志列表’可打开一个子菜单，列出所有的包状态标志。但是，这些标志不仅影响当前包，而且影响此列表中的所有包。

**其他** ‘其他’菜单提供了用于处理包依赖性和冲突的选项。如果手工选择了要安装的包，请单击‘显示自动包更改’来查看包管理器自动选择用于解决依赖性的包列表。如果仍存在未解决的包冲突，则会显示警报并给出建议解决方案。

如果将包冲突设置为‘忽略’，则会将此信息永久地保存在系统中。否则在每次启动包管理器时，都需要将同一个包设置为‘忽略’。要取消忽略依赖性，请单击‘重设置被忽略的依赖性冲突’。

**帮助** ‘帮助’→‘概述’概要介绍了包管理器的功能。‘符号’下详细介绍了各种包标志。如果倾向于不使用鼠标来操作程序，请单击‘键’来查看快捷方式列表。

## 依赖性检查

‘检查依赖性’和‘自动检查’位于信息窗口下。如果单击‘检查依赖性’，包管理器将检查当前包选择是否会造成任何未解决的包依赖性 or 冲突。如果出现未解决的依



赖性，将自动选择所需的其他包。如果出现包冲突，包管理器将打开一个对话框来显示这些冲突，并给出解决问题的多种选择。

如果激活‘自动检查’，包状态的任何更改均会触发自动检查。这是一个很有用的功能，因为这样将永久地监视包选择的一致性。但这一进程会消耗资源并可能使包管理器运行速度下降。因此在默认情况下不激活自动检查。无论哪种情况，在单击‘接受’来确认您的选择时，均会执行一致性检查。

在以下示例中，可能无法同时安装 sendmail 和 postfix。图2.3 “包管理器的冲突管理”显示了冲突讯息，提示您作出决定。已经安装了 postfix。因此，您可以选择不安装 sendmail、去除 postfix 或冒险同时安装二者并忽略冲突。

### Warning

#### 处理包冲突

在处理包冲突时，建议您接受 YaST 的建议，因为不这样的话，您的系统的稳定性和功能就可能会受到现有冲突的影响。

Warning

## 2.3.2 更改安装源

YaST 可以管理多个不同的安装源。它使您可根据不同的安装或更新目的来选择它们。当此模块启动时，会显示先前注册的所有安装源的列表。从 CD 进行正常安装后，仅列出安装 CD。单击‘添加’将其他安装源包含在此列表中。与 CD 和 DVD 等可移动媒体一样，也可以添加如 NFS 和 FTP 服务器等网络安装源。甚至可以选择本地硬盘上的目录作为安装媒体。有关详细信息，请参见 YaST 帮助文本。

所有已注册安装源在列表的第一列都有一个激活状态。单击‘激活或取消’来激活或取消相应的安装源。在安装软件包或更新程序期间，YaST 会从已激活安装源列表中选择一项适当的项。选择‘关闭’退出此模块时，当前设置将被保存并应用到配置模块‘安装和去除软件’和‘系统更新’。

## 2.3.3 YaST 联机更新

YaST 联机更新 (YOU) 使您可安装重要的更新程序和改进程序。这些增补程序可从 SUSE FTP 服务器和多个镜像服务器下载。

在‘安装源’下选择多个服务器之一。当您选择服务器时，就会将其 URL 复制到输入字段，在其中可对它进行编辑。也可以采用 file:/my/path 或 /my/path

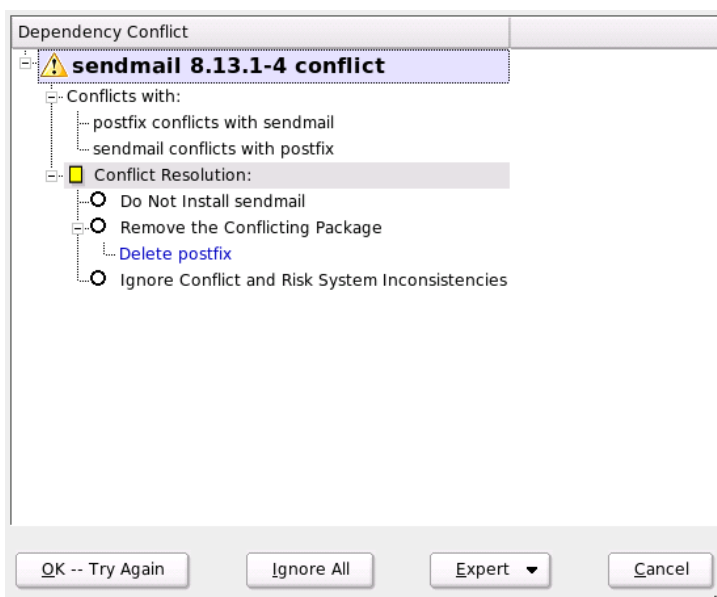


Figure 2.3: 包管理器的冲突管理

格式指定本地 URL。使用‘新建服务器’可用其他服务器扩展现有列表。单击‘编辑服务器’来修改当前所选服务器的设置。

在启动此模块时，‘手工选择增补程序’处于活动状态，用于选择要获取的增补程序。要应用所有可用的更新程序包，请取消此选项。但是，根据网络连接的带宽和要传送的数据量，下载时间可能会很长。

如果激活‘再次下载所有增补程序’，就会从服务器下载所有可用的增补程序、可安装包和说明。如果不激活它（即默认状态），则仅获取您的系统上尚未安装的增补程序。

此外，也可以自动更新系统。单击‘配置完全自动更新’，可配置定期自动查找并应用更新程序的进程。此过程完全是自动完成的。系统必须能够在安排的时间连接到更新服务器。

要执行更新，请单击‘下一步’。对于手工更新而言，这将装载所有可用增补程序的列表并启动包管理器，如第2.3.1节“安装和去除软件”中所述。在包管理器中，YOU 增补程序过滤器会被激活，使您可选择要安装的更新程序。在启动

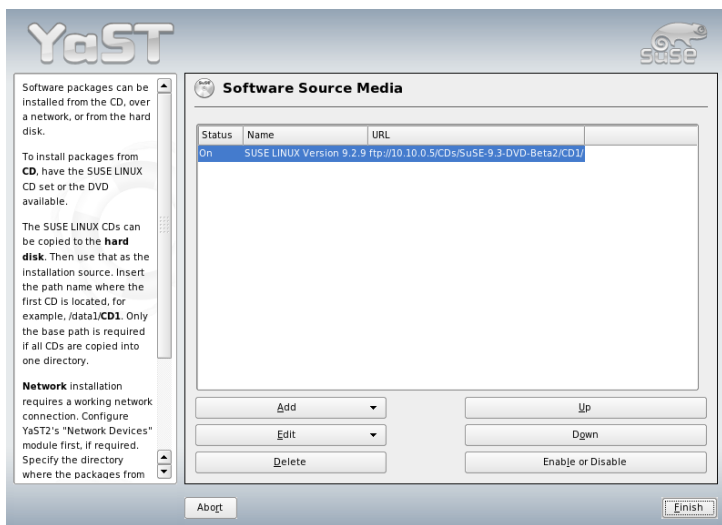


Figure 2.4: 更改安装源

时，如果在系统上已安装了相关的包，就会预先选择可用的安全性增补程序和建议增补程序。应接受此建议。

在完成选择后，在包管理器中单击‘接受’。然后从服务器下载所选的全部更新程序并将其安装在您的计算机上。根据连接速度和硬件性能，这可能会需要较长时间。所有错误都被显示在一个窗口中。如果需要，可跳过有问题的包。在安装前，某些增补程序会打开一个显示详细信息的窗口。

在下载和安装更新程序时，可以在日志窗口中跟踪所有操作。在成功安装所有增补程序后，单击‘完成’退出 YOU。如果在安装后不需要这些更新程序文件，请选择‘更新后去除源包’，这样就可以在更新后删除它们。最后执行 SuSEconfig 来根据需要调整系统配置。

### 2.3.4 增补程序 CD 更新

此选项将从 CD 而非 FTP 服务器安装增补程序。其优势在于使用 CD 可以更快地进行更新。插入增补程序 CD 后，将扫描 CD 上的所有增补程序并将其显示在对话框中。然后可以从增补程序列表中选择要安装的包。如果不存在增补程序

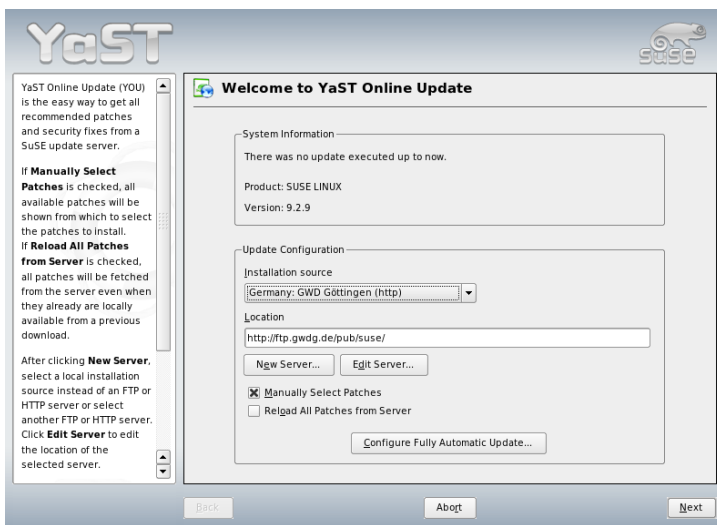


Figure 2.5: YaST 联机更新

CD，模块就会发出一条错误讯息。这时应插入增补程序 CD，然后重新启动此模块。

### 2.3.5 系统更新

此模块用于更新您的系统上已安装的版本。在操作期间，只能更新应用程序软件，而不能更新 SUSE LINUX 基础系统。要更新基础系统，请从安装媒体（如 CD）引导计算机。在 YaST 中选择安装方式时，应选择‘更新现有系统’而非‘全新安装’。

更新系统的过程与全新安装类似。最初，YaST 会检查系统，确定适当的更新策略，并将结果显示在建议对话框中。用鼠标单击各项来更改详细信息。某些项目（如‘语言’和‘键盘布局’）在介绍安装过程的一节中（请参见第 1.3 节“语言选择”）已有说明。本节仅介绍特定于更新的设置。

#### 选择要更新的分区

如果您的系统上安装有多个版本的 SUSE LINUX，可使用此项目从列表中选择—一个分区来进行更新。

## 更新选项

设置您的系统的更新方法。有两个选项可用。

**通过安装新软件更新** 要将整个系统更新到最新软件版本，请选择一个预先定义的选择。这些选择与安装过程提供的选择相同。它们可确保安装先前不存在的包。

**仅更新已安装的包** 此选项仅更新系统上已存在的包。将不会安装任何新功能。

此外，可以使用“删除过时的包”来去除新版本中不存在的包。默认情况下将预先选择此选项，以避免过时的包无谓地占用硬盘空间。

## 包

单击“包”来启动包管理器并选择或取消选择要更新的各个包。应使用一致性检查来解决任何包冲突。第2.3.1节“安装和去除软件”中详细介绍了包管理器的使用。

## 备份

在更新期间，某些包的配置文件可能会被替换为新版本的包的配置文件。因为您可能会修改当前系统中的某些文件，包管理器通常会保留被替换文件的备份副本。利用此对话框可确定这些备份的范围。

### Important

#### 备份的范围

这里的备份不包括软件。它只包括配置文件。

### Important

## 有关更新的重要信息

系统更新是一个非常复杂的过程。对于每个程序包，YaST 必须首先检查计算机上已安装的版本，然后确定需要执行哪些步骤来正确地以新版本替换旧版本。YaST 同时会尝试采用已安装包的任何个人设置。有些配置可能会产生问题，原因是旧的配置无法正确地处理新程序版本，或在不同配置之间出现意外的不一致。

现有的版本越旧，要更新的包的配置与标准配置的差别就越大，更新时就越可能会出现。有时无法正确采用旧的配置。在这种情况下就必须进行全新配置。在启动更新之前，应保存现有配置。

## 2.3.6 媒体检查

如果在使用 SUSE LINUX 安装媒体时遇到问题，则可以使用此模块检查 CD 或 DVD。在特殊情况下，某些设备在读取特定媒体时会出现问题。自己刻录的媒体较容易出现这种情况。要检查 SUSE LINUX CD 或 DVD 是否有错误，只要将该媒体插入驱动器中并运行此模块即可。单击‘启动’，YaST 将检查媒体的 MD5 校验和。这可能要花几分钟时间。如果检测到有任何错误，则不应使用此媒体进行安装。

## 2.4 硬件

必须首先按照供应商的说明安装或连接新硬件。打开外部设备（如打印机或调制解调器），然后启动相应的 YaST 模块。YaST 将自动检测大多数设备并显示其技术数据。如果自动检测失败，YaST 将提供一个设备列表（型号、供应商等），您可从中选择合适的设备。有关详细信息，请参考随硬件提供的文档。

### Important

#### 型号指定

如果您选择的型号未包括在设备列表中，可尝试使用具有类似指定的型号。但在某些情况下型号必须完全匹配，因为类似指定并不总是表示兼容性。

### Important

## 2.4.1 CD-ROM 和 DVD 驱动器

在安装范围内，所有被检测到的 CD-ROM 驱动器都通过文件 `/etc/fstab` 中的项而被集成在已安装的系统中。同时在 `/media` 中创建相应的子目录。使用此 YaST 模块来将其他驱动器集成到系统中。

在启动此模块时，会显示检测到的所有设备的列表。使用行开头的复选框来选择新驱动器，然后使用‘完成’来完成集成。这样就将新驱动器集成到系统中。

## 2.4.2 打印机

第12章 打印机操作 提供了有关在 Linux 中进行打印的详细信息，其中介绍了常见的打印问题。YaST 自动配置打印机或提供配置对话框帮助您手工设置打印机。然后，您可以从命令行进行打印或配置应用程序使用打印系统。有关在 YaST 中配置打印机的详细说明在第12.5.1节“本地打印机”中有所介绍。

### 2.4.3 硬盘控制器

通常情况下，YaST 会在安装期间配置系统的硬盘控制器。如果添加了控制器，请使用 YaST 模块将它们集成到系统中。您也可以修改现有配置，但通常没有必要这样做。

此对话框显示已检测到的硬盘控制器的列表，并使您可以使用特定参数指派合适的内核模块。在将当前设置永久保存在系统中之前，应使用‘测试内核装载’来检查它们是否正常工作。

#### Warning

##### 硬盘控制器的配置

这是一种专家工具。如果进行了错误设置，就可能再也无法引导您的系统了。如果进行了更改，请使用测试选项。

#### Warning

### 2.4.4 硬件信息

YaST 会检测硬件，以得到硬件部件的配置信息。检测到的技术数据将显示在一个单独的屏幕中。在您希望提交需要硬件信息的支持请求等情况下，这些数据就会特别有用。

### 2.4.5 IDE DMA 方式

使用此模块，可激活或取消已安装系统中的 IDE 硬盘、IDE CD 和 DVD 驱动器的 DMA 方式。此模块对 SCSI 设备没有任何作用。DMA 方式可大幅提高系统的性能和数据传送速度。

在安装期间，当前的 SUSE LINUX 内核会自动激活硬盘的 DMA 方式，但不激活 CD 驱动器的 DMA 方式，因为对所有驱动器均默认激活 DMA 方式常会造成 CD 驱动器出现问题。使用 DMA 模块来为您的驱动器激活 DMA 方式。如果设备支持 DMA 方式而没有任何问题，通过激活 DMA 可提高您的驱动器的数据传送速度。

#### Important

DMA（直接内存访问）意味着可以将您的数据不经处理器控制而直接传送到 RAM。

#### Important

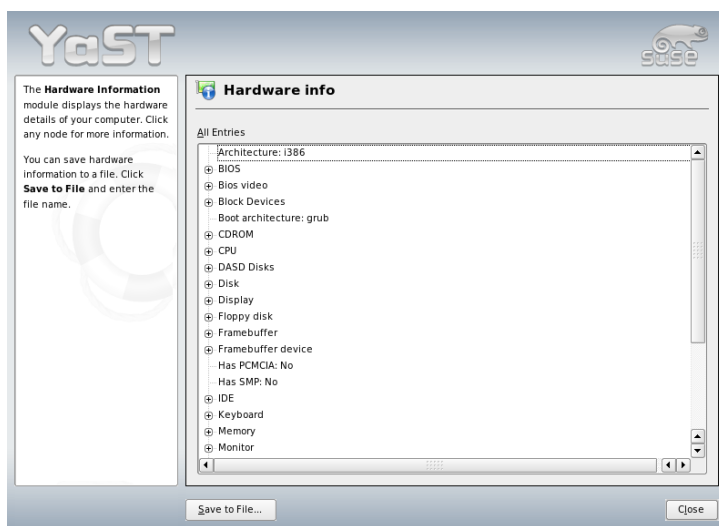


Figure 2.6: 显示硬件信息

## 2.4.6 扫描仪

如果已连接并打开了扫描仪，在启动此 YaST 模块时应自动检测到它。这时将出现扫描仪的安装对话框。如果没检测到任何扫描仪，就会出现手工配置对话框。如果已经安装了一台或多台扫描仪，就会显示一个可修改或删除的现有扫描仪的列表。按‘添加’来配置新设备。

接下来将使用默认设置执行安装。如果安装成功，就会显示一条相应的讯息。现在请插入一份文档并单击‘测试’来测试您的扫描仪。

### 未检测到扫描仪

只能自动检测所支持的扫描仪。无法检测连接到另一台网络主机上的扫描仪。手工配置要区分 3 种类型的扫描仪：USB 扫描仪、SCSI 扫描仪和网络扫描仪。

**USB 扫描仪** 指定供应商和型号。随后，YaST 将尝试装载 USB 模块。如果您的扫描仪非常新，就可能不会自动装载这些模块。这时将自动转到一个对话框，您可以在其中手工装载 USB 模块。有关详细信息，请参考 YaST 帮助文本。



**SCSI 扫描仪** 指定设备，如 `/dev/sg0`。不能在系统正在运行时连接或断开 SCSI 扫描仪。应首先关闭系统。

**网络扫描仪** 输入 IP 地址或主机名。要配置网络扫描仪，请参考支持数据库文章 *Linux 中的扫描* (<http://portal.suse.com/sdb/en/index.html>，关键字*扫描仪*)。

如果未检测到您的扫描仪，原因可能是不支持此设备。但有时也会检测不到支持的扫描仪。如果是这种情况，应手工选择扫描仪。如果能够在供应商和型号列表中找到您要使用的扫描仪，则将其选中。否则应选择‘取消’。有关可用于 Linux 的扫描仪的信息，请参考 <http://cdb.suse.de/> 和 <http://www.mostang.com/sane>。

### Warning

#### 手工指派扫描仪

只有在您完全有把握的情况下才能手工指派扫描仪。错误选择可能会损坏您的硬件。

### Warning

### 查错

无法检测到您的扫描仪的原因可能是以下之一：

- 不支持此扫描仪。请参考 <http://cdb.suse.de/> 查看与 Linux 兼容的设备的列表。
- 没有正确安装 SCSI 控制器。
- 您的 SCSI 端口存在终止问题。
- SCSI 数据线过长。
- 此扫描仪具有一个 Linux 不支持的 SCSI 指示灯控制器。
- 此扫描仪存在缺陷。

### Warning

不能在系统正在运行时连接或断开 SCSI 扫描仪。应首先关闭系统。

### Warning

有关扫描的详细信息，请参考 *用户指南* 中有关 `kooka` 的一章。

## 2.4.7 声卡

在启动声卡配置工具时，YaST 会尝试自动检测您的声卡。可配置一个或多个声卡。要使用多个声卡，请首先选择一个要配置的声卡。按‘配置’进入‘设置’对话框。选择‘编辑’打开一个对话框，在其中编辑先前配置的声卡。单击‘完成’来保存当前设置并完成声卡配置。

如果 YaST 无法自动检测到您的声卡，请在‘声卡配置’中选择‘添加声卡’来打开一个对话框，从中选择一个声卡和模块。有关详细信息，请参考声卡文档。`/usr/share/doc/packages/alsa/cards.txt` 和 <http://www.alsa-project.org/~goemon/> 中提供了 ALSA 所支持的声卡及其对应声卡模块的参考列表。在完成选择后，单击‘下一步’返回到‘设置’。

### 设置

在第一个设置屏幕中选择配置级别。采用‘自动快速设置’时，不要求您进一步执行配置和任何声卡测试。这时将自动配置声卡。采用‘正常设置’时，您能够调节输出音量并播放测试声音。采用‘高级设置’时，能够手工自定义声卡选项。

在此对话框中，还有用于游戏杆配置的一个快捷方式。单击相应的复选框。在以下对话框中选择游戏杆类型，然后单击‘下一步’。

### 声卡音量

在此测试屏幕中测试您的声卡配置。使用‘+’和‘-’来调节音量。请从总音量的 10% 开始，以免损坏您的扬声器或听力。在您按‘测试’后应听到一段测试声音。如果听不到任何声音，请增大音量。按‘继续’完成声卡配置。同时保存音量设置。

### 声卡配置

使用‘删除’来去除声卡。在文件 `/etc/modprobe.d/sound` 中取消已配置声卡的现有项。单击‘选项’打开一个对话框，在其中手工自定义声卡模块选项。在‘添加声卡’下，配置其他的声卡。如果 YaST 检测到另一个声卡，请继续单击‘配置声卡’。如果 YaST 没有检测到声卡，就会自动定向到‘手工声卡选择’。

如果使用的是 Creative Soundblaster Live 或 AWE 声卡，请单击‘安装声卡字体’来将 SF2 声音字体从原始 Soundblaster 驱动程序 CD-ROM 复制到您的硬盘。声卡字体保存在目录 `/usr/share/sfbank/creative/` 中。

要播放 MIDI 文件，请激活‘启动序列发生器’。这样就将装载用于提供序列发生器支持的模块和声卡模块。

在单击‘完成’后，就会保存所有已安装声卡的音量和配置。混音器设置将被保存在文件 `/etc/asound.conf` 中，同时 ALSA 配置数据被追加到文件 `/etc/modprobe.conf` 的末尾。

## 2.4.8 电视和无线电卡

在启动和初始化此 YaST 模块后，就会出现‘电视和无线电卡’对话框。如果自动检测到您的电视和无线电卡，就会将其在列表的顶部显示出来。在这种情况下，请用鼠标高亮显示此行，然后选择‘配置’。如果未检测到您的卡，请选择‘其他（未能识别）’。按‘配置’来进行手工选择，从供应商和型号列表中选择您的卡。

如果已经配置了电视或无线电卡，请使用‘更改’来修改现有配置。在这种情况下会有一个对话框来显示所有已配置的卡的列表。选择卡，然后单击‘编辑’来启动手工配置。

在自动检测硬件期间，YaST 会尝试向您的卡指派正确的调谐器。如果您没有把握，可保留‘默认（已识别）’并检查它是否正常工作。如果不能设置所有通道，可能是因为自动检测调谐器类型出现故障。在这种情况下，请单击‘选择调谐器’并高亮显示列表中正确的调谐器类型。

如果熟悉技术详细信息，可以使用专家对话框来指定电视或无线电卡的设置。请在此对话框中选择内核模块及其参数。此外还应检查您的电视卡驱动程序的所有参数。为此，应选择相应的参数并在参数行中输入新值。单击‘应用’确认新值，或单击‘重设置’来恢复默认值。

‘电视和无线电卡、音频’对话框用于将您的电视或无线电卡与已安装的声卡相连接。您必须使用电缆来将电视或无线电卡的输出与声卡的外部音频输入相连接。仅当已配置了声卡并且外部输入处于活动状态时，才能进行有效连接。如果尚未配置您的声卡，应按照第2.4.7节“声卡”中的介绍选择‘配置声卡’来转到相应的对话框。

如果您的电视或无线电卡具有扬声器插孔，就可以直接连接扬声器而无需配置声卡。有的电视卡没有任何声音功能（如用于 CCD 相机的那些电视卡），不需要对其进行音频配置。

## 2.5 网络设备

系统的所有网络设备必须先进行初始化才能被服务使用。这些设备的检测和配置是在模块组‘网络设备’中完成的。有关在 YaST 中配置任何支持的网络适配器类型的详细说明（包括有关连接到网络的背景信息）在第22.4节“使用 YaST 配置网络连接”中有所介绍。有关配置网络设备以进行无线通讯的信息在第17章无线通讯中有所介绍。

## 2.6 网络服务

此组包含在网络中配置各种服务的工具。这包括名称解析、用户鉴定和文件服务。

### 2.6.1 邮件传送代理

如果您使用 sendmail、postfix 或服务提供商的 SMTP 服务器来发送电子邮件，可使用此模块来配置您的邮件设置。您可以通过 fetchmail 程序获取邮件，还可以为此输入服务提供商的 POP3 服务器或 IMAP 服务器的详细信息。此外，还可使用自己选择的邮件程序（如 KMail 或 Evolution）来按照常规方式设置您的 POP 和 SMTP 访问数据（通过 POP3 接收邮件，通过 SMTP 发送邮件）。在这种情况下就不需要此模块。

要使用 YaST 配置您的邮件，请在电子邮件配置模块的第一个对话框中指定要使用的因特网连接类型。请选择以下选项之一：

‘永久’ 如果使用专线连接到因特网，请选择此选项。您的计算机将永久联机，所以不需要拨号。如果您的系统是具有中央电子邮件服务器的本地网络的一部分，请选择此选项来确保自己能够永久访问电子邮件。

‘拨号’ 此项适用于在家中有电脑、不在网络中但有时连接到因特网的用户。

无任何连接 如果既不能访问因特网，又不在某个网络中，就无法发送或接收电子邮件。

此外，通过激活相应的复选框，您可以使用 AMaViS 对进入和外发的电子邮件激活病毒扫描。在您激活电子邮件过滤功能时，将自动安装此包。在以下对话框中，指定外发邮件服务器（通常是您的服务提供商的 SMTP 服务器）和进入邮件的参数。如果使用拨号连接，指定不同用户用于接收邮件的多种 POP 或 IMAP 服务器。通过此对话框，您也可以指派别名、使用掩饰或设置虚拟域。单击‘完成’退出邮件配置。

### 2.6.2 其他可用服务

YaST 提供了许多其他网络模块。

**DHCP 服务器** YaST 只需几个步骤就可以设置自定义 DHCP 服务器。第 27 章 DHCP 提供了有关此主题的基本知识，并逐步介绍了 YaST 中的配置进程。

**DNS 服务器** 对于较大的网络，我们建议您配置 DNS 服务器来进行名称解析。有关使用 YaST 进行配置的详细信息，请参考第24.1节“用 YaST 进行配置”。第24章 域名系统 提供了有关 DNS 的背景信息。

**DNS 和主机名** 使用此模块配置主机名和 DNS。如果在配置网络设备时已完成了这些设置，则无需执行此操作。还可使用它来更改主机名和域名。如果已经正确配置了服务提供程序来实现 DSL、调制解调器或 ISDN 访问，名称服务器列表将包含自动从服务提供程序数据中抽取的项。如果您位于本地网络中，则可能会通过 DHCP 接收您的主机名，这种情况下不应修改此名称。

**HTTP 服务器** 要运行您自己的万维网服务器，请使用 YaST 配置 Apache。有关详细信息，请参考第30章 Apache 万维网服务器。

**主机名** 当在较小的网络中引导时，还可以使用此模块而不是使用 DNS 进行主机名解析。此模块中的项反映文件 `/etc/hosts` 的数据。有关详细信息，请参见第22.5.1节“`/etc/hosts`”。

**LDAP 客户机** LDAP 可取代 NIS 用于网络中的用户鉴定。有关 LDAP 的背景信息和使用 YaST 进行客户机配置的详细说明，请参考第29章 LDAP — 目录服务。

**NFS 客户机和 NFS 服务器** NFS 用于运行您的网络所有成员都能访问的文件服务器。使用此文件服务器可将一些应用程序、文件和储存空间供用户使用。在 `NFS 服务器` 模块中，可以将您的主机配置为 NFS 服务器，并确定要导出供网络用户共同使用的目录。具有适当权限的所有用户均可将这些目录装入他们自己的文件树中。第26章 通过 NFS 共享文件系统 介绍了 YaST 模块并提供了有关 NFS 的背景信息。

**NIS 客户机和 NIS 服务器** 如果您运行了多个系统，本地用户管理（使用文件 `/etc/passwd` 和 `/etc/shadow`）就变得不切实际，并需要很多的维护工作。在这种情况下，应在中央服务器上管理用户数据，然后从这里分发到客户机。可以采用 NIS、LDAP 和 Samba 等解决方案。有关 NIS 和使用 YaST 进行配置的详细信息，请参考第25章 使用 NIS。

**NTP 客户机** NTP（网络时间协议）是一个用于同步网络上硬件时钟的协议。有关 NTP 的背景信息和使用 YaST 进行配置的说明，请参考第28章 使用 `xntp` 同步时间。

**网络服务 (inetd)** 使用此工具来确定在引导 SUSE LINUX 时要启动的网络服务（如 `finger`、`talk` 和 `ftp`）。这些服务使外部主机可以连接到您的计算机。可以为每个服务配置多个参数。默认情况下，不启动管理个别服务的主服务（`inetd` 或 `xinetd`）。

启动此模块时，选择是否启动 `inetd` 或 `xinetd`。可以用标准服务选择启动所选的守护程序。此外，可以用‘添加’、‘删除’和‘编辑’来撰写自己的服务选择。

## Warning

### 配置网络服务 (`inetd`)

在系统上撰写和调整网络服务是一个复杂的过程，要求您全面了解 Linux 服务的概念。

## Warning

**代理** 使用此模块，您可以编辑系统范围的代理设置。有关代理的详细信息，请参见第33章 代理服务器 Squid。

**从远程主机进行管理** 要允许通过 VNC 连接从远程主机维护您的系统，请允许使用这个 YaST 模块来建立连接。请参考第3.2.2节 “用于 VNC 安装的客户程序”。

**路由选择** 如果您通过本地网络中的网关连接到因特网，则需要此工具。对于 DSL 而言，仅当配置网卡时才需要网关数据。但是，DSL 的这些项只是没有任何功能的虚设项。

**配置 Samba 服务器和客户机** 在包含 Linux 和 Windows 主机的异构网络中，Samba 控制着这两个世界之间的通讯。有关 Samba 以及客户机和服务器的配置的信息，请参考第32章 Samba。

## 2.7 安全性和用户

多用户功能是 Linux 的一个基本特点。因此，多个用户可以相互独立地在同一个 Linux 系统上工作。每个用户都有一个由登录名标识的用户帐户和一个用于登录系统的个人口令。所有用户都有自己的主目录，其中储存着他们的个人文件和配置。

### 2.7.1 用户管理

在选择编辑用户后，YaST 将提供系统中所有本地用户的概述。如果您是大型网络的一部分，请单击‘设置过滤器’来列出所有系统用户（例如 `root`）或 NIS 用户。您可以创建自定义过滤器设置。这里不是在各个用户组之间进行切换，而

是根据您的需要结合它们。要添加新用户，请在以下屏幕中所要求的空白中填入相应内容。随后，新用户就可以使用登录名和口令登录到主机。可以使用‘详细信息’对用户配置文件进行微调。您可以手工设置用户 ID、主目录和默认登录 shell。为特定组指派新用户。在‘口令设置’中配置口令的有效性。如有必要可随时单击‘编辑’来更改这些设置。要删除某用户，请从列表中将其选中，然后单击‘删除’。

对于高级网络管理，可使用‘专家选项’来定义用于创建新用户的默认设置。选择鉴定方法（NIS、LDAP、Kerberos 或 Samba）和用于口令加密的算法。这些设置适用于大型网络。

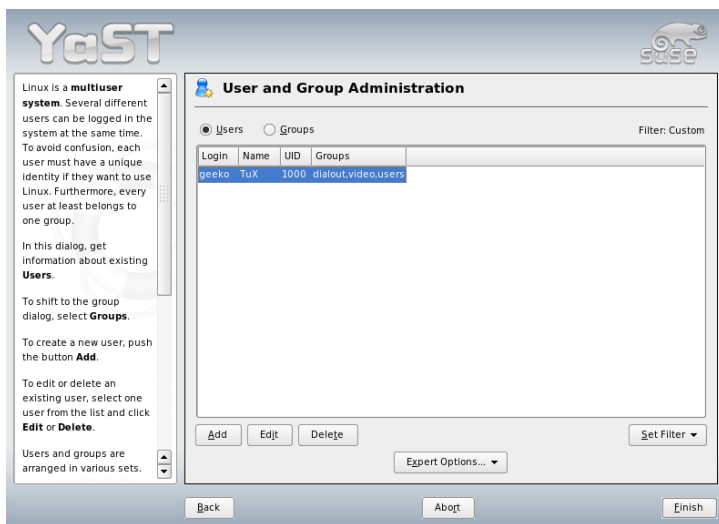


Figure 2.7: 用户管理

## 2.7.2 组管理

从 YaST 控制中心启动组管理模块，或单击用户管理中的‘组’。这两个对话框的功能相同，用于创建、编辑或删除组。

YaST 提供了所有组的列表。要删除某个组，请从列表中将其选中（所进行高亮显示为深蓝色），然后单击‘删除’。在‘添加’和‘编辑’下，在相应的 YaST 屏幕中输入名称、组 ID (gid) 和组的成员。如果需要，为此组设置一个更改口令。过滤器设置与‘用户管理’对话框中的相同。

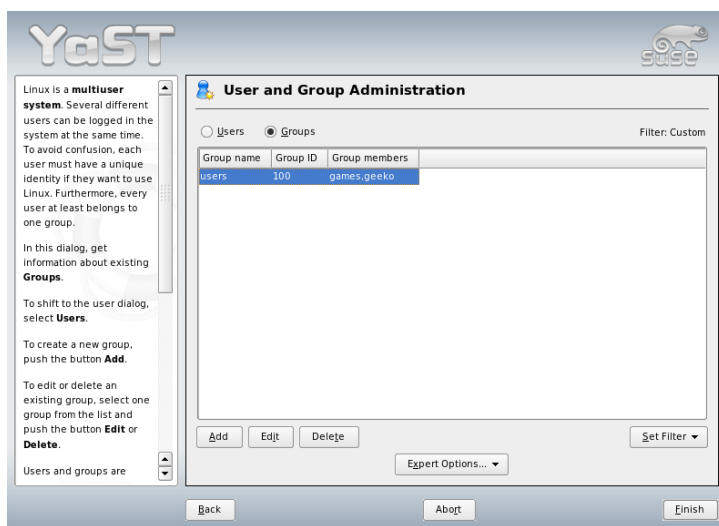


Figure 2.8: 组管理

### 2.7.3 安全性设置

在‘本地安全性配置’（可从‘安全性和用户’下访问）中，选择以下四个选项之一：级别 1 用于独立计算机（预配置）。级别 2 用于具有网络的工作站（预配置）。级别 3 用于具有网络的服务器（预配置）。“自定义设置”用于自己的配置。

如果单击前三项之一，应包含预配置系统安全性级别选项之一。要做到这一点，只需单击‘完成’。在‘详细信息’下，访问可以修改的各设置。如果选择‘自定义设置’，请单击‘下一步’进入另一个对话框。在这里可找到默认安装值。

‘口令设置’为了使系统在接受新口令之前对其进行检查，请选择‘检查新口令’和‘口令的合理性测试’。设置新创建用户口令的最小和最大长度。定义口令的有效期间以及提前多少天在用户登录文本控制台时发出口令过期警报。

‘引导设置’通过选择所需的操作来指定如何解释组合键 **(Ctrl)-(Alt)-(Del)**。通常情况下，这个组合键（在文本控制台中输入）会导致系统重引导。除非您的计算机或服务器是公共访问的，而您又担心有人会在没有授权的情况下执



行此操作，否则不要修改此设置。如果选择‘停止’，这个组合键将关闭系统。如果选择‘忽略’，将忽略此组合键。

通过授予从 KDE 显示管理器（KDE 的图像登录）关闭系统的权限来指定‘KDM 的关闭行为’。可向‘仅根用户’（系统管理员）、‘所有用户’、‘无人’或‘本地用户’授予权限。如果选择‘无人’，就只能通过文本控制台关闭系统。

‘登录设置’通常情况下，在登录尝试失败后，需要等数秒之后才能尝试重新登录。这样就使口令嗅探者难以登录。还可以选择激活‘记录失败登录尝试’和‘记录成功登录尝试’。如果您怀疑某人试图盗取您的口令，可检查 `/var/log` 中系统日志文件中的项。选择‘允许远程图形登录’，可允许其他用户通过网络访问您的图形登录屏幕。由于这一访问功能具有潜在的安全风险，所以在默认情况下它处于非活动状态。

‘添加用户设置’每个用户都有一个数字用户 ID 和一个字母用户 ID。二者之间的相互关系是通过文件 `/etc/passwd` 建立起来的，而且应尽可能地保持唯一性。使用此屏幕中的数据，可定义在添加新用户时指派给用户 ID 的数字部分的数字范围。对于用户来说，这一数字最小应为 500；自动生成的数字从 1000 开始。请以相同的方法进行组 ID 设置。

‘其他设置’‘文件权限的设置’有 3 个选择选项：‘简单’、‘安全’和‘高度警惕’。对大多数用户而言使用第一个选项就足够了。YaST 帮助文本提供了关于 3 个安全性级别的信息。‘高度警惕’设置的限制极为严格，并可以作为系统管理员设置的基本操作级别。如果选择‘高度警惕’，应注意某些程序可能不能工作或不能正常工作，原因是用户不再具有访问某些文件的权限。

在这个对话框中，还定义了哪个用户应启动 `updatedb` 程序。该程序每天定期自动运行或在引导后自动运行，并生成一个数据库 (`locatedb`)，其中储存着每个文件在您的计算机上的位置。如果选择‘无人’，则任何用户都只能在数据库中找到任何其他（未授权）用户均可看到的路径。如果选择 `root`，则将索引所有本地文件，原因是 `root` 是超级用户，可以访问所有目录。最后，请务必取消选中选项‘根路径中的当前目录’（默认设置）。

按‘完成’结束安全性配置。

## 2.7.4 防火墙

使用此模块来配置 `SuSEfirewall2`，以保护您的计算机免受来自因特网的攻击。有关 `SuSEfirewall2` 的详细信息，请参见第 34.1 节“伪装和防火墙”。



Figure 2.9: 安全性设置

## Tip

### 自动激活防火墙

YaST 会在每个已配置的网络接口上自动启动具有适当设置的防火墙。只有您希望使用自定义设置重配置防火墙或取消它时，才需要启动此模块。

## Tip

## 2.8 系统

### 2.8.1 系统区域的备份副本

YaST 备份模块使您能够创建系统备份。此模块创建的备份不包括整个系统。它仅保存有关已更改包的信息和关键储存区域和配置文件的副本。

定义要保存在备份中的数据类型。默认情况下，备份中包含有关自上次安装以来被更改的所有包的信息。它可能还包含不属于包本身的数据，如 /home 下的

目录或 `/etc` 中的许多配置文件。此外，备份中还包含您的硬盘上的重要储存区域，如分区表或主引导记录 (MBR)，它们在您尝试恢复系统时非常重要。

## 2.8.2 恢复系统

如图 2.10 “启动恢复模块的窗口” 所示，恢复模块用于从备份档案恢复系统。请按照 YaST 中的说明操作。按“下一步”进入相应的对话框。首先指定这些档案的位置（可移动媒体、本地硬盘或网络文件系统）。这时会显示各档案的说明和内容，帮助您确定要从档案中恢复哪些内容。

此外，还有一个对话框用于卸载自上次备份以来添加的包，另一个对话框用于重安装自上次备份以来删除的包。这两个步骤可使您将系统完全恢复到上次备份时的状态。

### Warning

#### 系统恢复

由于此模块通常安装、替换和卸载很多的包和文件，所以仅当您具有备份经验时才能使用它。否则可能会丢失数据。

### Warning

## 2.8.3 创建引导磁盘和应急磁盘

使用此 YaST 模块创建引导和应急磁盘。如果系统的引导配置被损坏，这些软盘就非常有用。如果根分区的文件系统被损坏，则需要使用应急磁盘。

有以下选项可用：

‘标准引导软盘’ 使用此选项来创建用于引导已安装系统的标准引导软盘。根据体系结构，引导磁盘的实际数目可能会有所变化，但您应该创建对话框中显示的所有引导磁盘，这是因为所有这些磁盘都是进行引导所必需的。启动应急系统也需要这些磁盘。

‘应急软盘’ 此磁盘包含一个特殊环境，使您能够在已安装系统中执行维护任务，如检查和修复文件系统以及更新引导装载程序。要启动应急系统，请用标准引导磁盘进行引导，然后选择‘手工安装’→‘启动安装或系统’→‘应急系统’。然后会提示您插入应急磁盘。

‘自定义软盘’ 使用此磁盘可将任何现有的软盘映像从硬盘写入到软盘。

‘下载软盘映像’ 选择它后，输入 URL 和鉴定数据以从因特网下载软盘映像。

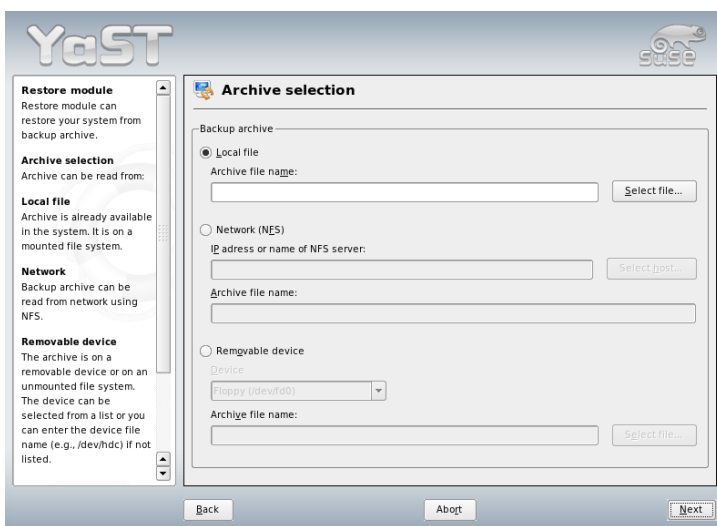


Figure 2.10: 启动恢复模块的窗口

要创建这些软盘之一，请选择相应的选项，然后单击‘下一步’。在系统提示时插入软盘。如果再次单击‘下一步’，就会创建软盘。

## 2.8.4 LVM

逻辑卷管理器 (LVM) 是一种利用逻辑驱动器对硬盘进行自定义分区的工具。有关 LVM 的详细信息，请参考第3.7节“LVM 配置”。

## 2.8.5 分区

使用如图2.12“YaST 专家分区工具”所示的专家对话框，可以手工修改一个或多个硬盘的分区。可以添加、删除和编辑分区。请仍从 YaST 模块中获取软 RAID、和 LVM 配置。

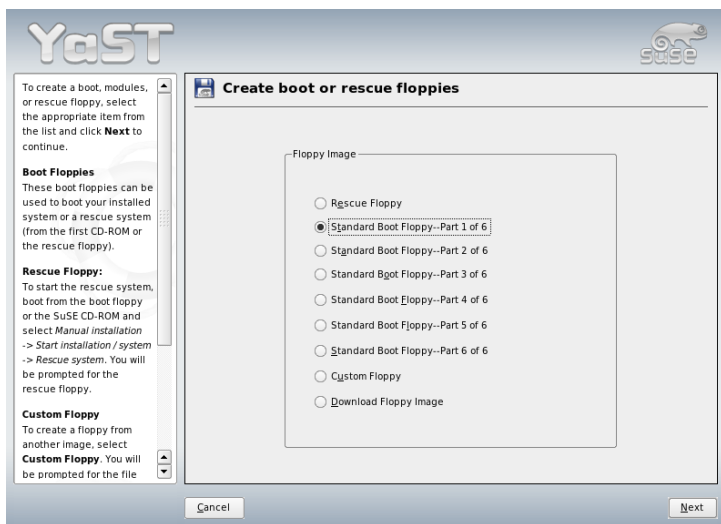


Figure 2.11: 创建引导磁盘和应急磁盘

## Warning

虽然可以在已安装系统中修改分区，但这些任务只应由专家来完成。否则一旦出错就很容易丢失数据。如果是对正在使用的硬盘重分区，完成任务后应立即重引导系统。比起对正在运行的系统进行重分区，使用应急系统更为安全。

## Warning

专家对话框中列出了所有已连接硬盘上的所有现有分区或建议分区。其中将整个硬盘作为不带编号的设备列出，如 `/dev/hda` 或 `/dev/sda`。将分区作为这些设备的部分列出，如 `/dev/hda1` 或 `/dev/sda1`。此外还显示硬盘的大小、类型、文件系统和装入点以及硬盘的分区。装入点描述将分区装入 Linux 文件系统树中的哪个位置。

如果在安装期间运行专家对话框，还会列出并自动选中所有可用硬盘空间。要为 SUSE LINUX 提供更多磁盘空间，请在列表中自下而上（从硬盘的最后一个分区向上到第一个分区）释放所需空间。例如，如果您有 3 个分区，则不能将第 2 个分区专用于 SUSE LINUX，而为其他操作系统保留第 3 个和第 1 个分区。

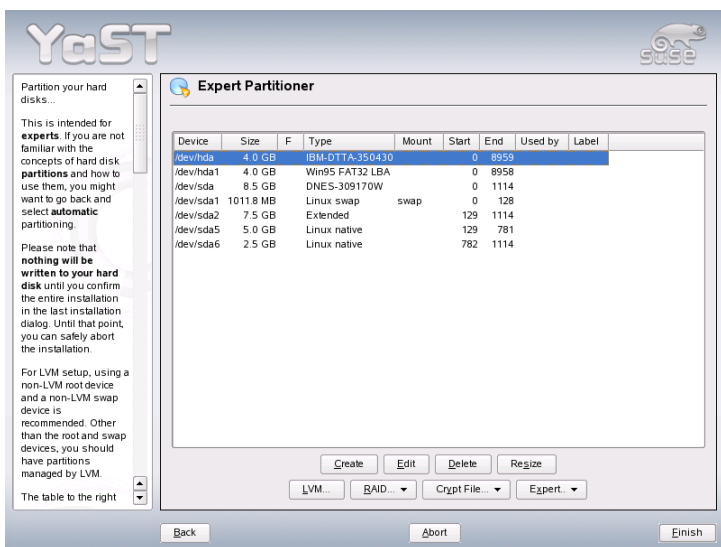


Figure 2.12: YaST 专家分区工具

## 创建分区

选择‘创建’。如果连接了多个硬盘，则会出现一个选择对话框，可以在其中选择要用于新分区的硬盘。然后，指定分区类型（即主分区还是扩展分区）。最多可以创建 4 个主分区或 3 个主分区和 1 个扩展分区。在扩展分区内，可以创建多个逻辑分区（请参见第 1.5.4 节“分区类型”）。

如有必要，选择要使用的文件系统和装入点。YaST 会为所创建的每个分区建议一个装入点。下一节将详细介绍这些参数。选择‘确定’即可应用您的更改。随后新分区就会被列在分区表中。如果单击‘下一步’，则采用当前值。若在安装期间，您随后会返回建议屏幕。

## 分区参数

在创建新分区或修改现有分区时，可以设置各种参数。对于新分区，YaST 会设置适当的参数，而且通常无需进行任何修改。要执行手工设置，请按以下步骤操作：

1. 选择分区。

## 2. ‘编辑’分区并设置以下参数：

**文件系统 ID** 即使不希望在此阶段格式化分区，仍需要为它指派一个文件系统 ID 来确保正确注册分区。可能值包括 ‘Linux’、‘Linux swap’、‘Linux LVM’ 和 ‘Linux RAID’。有关 LVM 和 RAID 的详细信息，请参考第3.7节“LVM 配置”和第3.8节“软 RAID 配置”。

**文件系统** 要在安装范围内立即格式化分区，请为分区指定以下文件系统之一：‘Swap’、‘Ext2’、‘Ext3’、‘ReiserFS’ 或 ‘JFS’。有关各种文件的详细信息，请参考第20章 Linux 中的文件系统。

交换是一种特殊格式，它允许将分区用作虚拟内存。ReiserFS 是 Linux 分区的默认文件系统。ReiserFS、JFS 和 Ext3 是日记文件系统。这些文件系统能够在系统崩溃后非常快地恢复系统，因为它们的操作期间记录了写进程。此外，ReiserFS 可以极快地处理很多小文件。Ext2 不是一种日记文件系统。但它坚如磐石并且很适合较小的分区，因为它不需要太多的磁盘空间来用于管理。

**文件系统选项** 在此可设置所选文件系统的各种参数。根据所用的文件系统，为专家提供了不同的选项。

**加密文件系统** 如果激活加密，则将所有数据以加密形式写入硬盘。这可以提高敏感数据的安全性，但会稍微降低系统速度，因为加密需要一些时间。有关文件系统加密的详细信息，请参见第34.3节“对分区和文件进行加密”。

**fstab 选项** 在此指定文件系统 (/etc/fstab) 的管理文件的各种参数。

**装入点** 指定应将分区装入文件系统树中的哪个目录。请从各个 YaST 建议中选择，或指定任何其他名称。

## 3. 选择‘下一步’激活分区。

如果手工分区，请创建一个至少为 256 MB 的交换分区。交换分区用于释放当前未使用数据的主存储器。这样可保持主存储器可用于最常用的重要数据。

### 专家选项

使用‘专家’可打开包含以下命令的菜单：

**重新读取分区表** 重新读取磁盘中的分区。例如，在文本控制台进行手动分区后需要此命令。

**删除分区表和磁盘标签** 此命令将完全覆盖以前的分区表。例如，如果非常规磁盘标签出现问题，则可以使用此命令。使用此方法，硬盘中的所有数据都将丢失。

## 更多分区提示

如果使用 YaST 执行分区且在系统中检测到其他分区，则也将这些分区输入到文件 `/etc/fstab` 中，以便能够方便地访问此数据。此文件包含系统中的所有分区及其属性，如文件系统、装入点和用户权限。

### 2.1: `/etc/fstab`: 分区数据

```
/dev/sda1    /data1    auto      noauto,user 0 0
/dev/sda5    /data2    auto      noauto,user 0 0
/dev/sda6    /data3    auto      noauto,user 0 0
```

这些分区（无论是 Linux 还是 FAT 分区）都指定了选项 `noauto` 和 `user`。这允许任何用户都可以根据需要装入或卸装这些分区。由于安全原因，YaST 不会自动在这里输入 `exec` 选项（当从此位置执行程序时需要此选项）。但是，如果要从那里运行程序，您可以手工输入此选项。如果出现“错误解释器”或“权限被拒绝”等系统讯息，则需要执行此操作。

## 分区和 LVM

从专家分区工具中，使用 ‘LVM’ 访问 LVM 配置（请参见第3.7节“LVM 配置”）。但是，如果系统中已经存在有效的 LVM 配置，当您在会话中首次进入 LVM 配置时将自动激活该配置。这种情况下，凡是包含属于激活卷组的分区的磁盘都无法进行重分区，因为只要硬盘上有任何活动分区，Linux 内核就无法重新读取该硬盘已经修改的分区表。不过，如果系统上已存在有效的 LVM 配置，则无需进行物理重分区，但需要更改逻辑卷的配置。

在物理卷 (PV) 的开始位置，将有关卷的信息写入到分区中。这样，PV 就会“知道”它属于哪个卷组。要将这样的分区重新用于 LVM 之外的其他用途，最好删除此卷开始位置的信息。例如，在 `system` 和 PV `/dev/sda2` 中，可以通过命令 `dd if=/dev/zero of=/dev/sda2 bs=512 count=1` 完成此操作。

### Warning

用于引导的文件系统

用于引导的文件系统（根文件系统或 `/boot`）不能储存在 LVM 逻辑卷上。而应将其储存在通常的物理分区中。

Warning



## 2.8.6 配置文件管理器 (SCPM)

SCPM (系统配置配置文件管理) 模块提供了创建、管理和切换系统配置的功能。它尤其适用于在不同位置 (在不同网络中) 和由不同用户使用的便携式计算机。这个功能对于固定计算机来说也同样有用, 因为它使您可使用多种不同的硬件部件或测试配置。有关 SCPM 基础知识和处理的详细信息, 请参考第15章 系统配置配置文件管理。

## 2.8.7 Systems Services (RunLevel)

可以在多个运行级别中操作 SUSE LINUX。默认情况下, 系统会引导到运行级别 5, 该运行级别提供了多用户方式、网络访问和图形用户界面 (X 窗口系统)。其他运行级别提供了有网络但没有 X 的多用户方式 (运行级别 3)、没有网络的多用户方式 (运行级别 2)、单用户方式 (运行级别 1 和 S)、系统暂停 (运行级别 0) 和系统重引导 (运行级别 6)。

如果在连接较高运行级别中的特定服务 (X 或网络) 时出现问题, 则可以使用其他运行级别。在这种情况下, 可能会将系统引导到较低的运行级别来修复此服务。许多服务器都在无图形用户界面的情况下进行操作, 所以必须在没有 X 的运行级别 (如运行级别 3) 进行引导。

通常情况下只需要标准运行级别 (5)。但是, 如果图形用户界面发生冻结, 您可以重新启动 X 窗口系统, 具体方法是使用组合键 **Ctrl-Alt-F1** 切换到文本控制台, 并以根用户身份登录, 然后使用命令 `init 3` 切换到运行级别 3。这样将关闭您的 X 窗口系统, 仅留下文本控制台可供使用。要重新启动图形系统, 请输入 `init 5`。

有关 SUSE LINUX 的运行级别的详细信息和 YaST 运行级别编辑器的介绍, 请参考第7章 引导和配置 Linux 系统。

## 2.8.8 Sysconfig 编辑器

目录 `/etc/sysconfig` 所包含的文件中具有 SUSE LINUX 最重要的设置。sysconfig 编辑器在一个排列整齐的表中显示了所有设置。可以修改这些值并将其保存到相应的配置文件中。通常情况下不需要手工编辑, 因为在安装包或配置服务时会自动调整这些文件。有关 `/etc/sysconfig` 和 YaST sysconfig 编辑器的详细信息, 请参考第7章 引导和配置 Linux 系统。

## 2.8.9 时区选择

在安装期间已经设置了时区，但可在这里进行更改。在列表中单击您所在的国家或地区，然后选择‘当地时间’或‘UTC’（世界协调时间，以前称为格林威治标准时间）。‘UTC’经常用在 Linux 系统中。使用 Microsoft Windows 等其他操作系统的计算机大多采用本地时间。

## 2.8.10 语言选择

在此选择您的 Linux 系统的语言。在 YaST 中选择的语言将应用于整个系统，包括 YaST 和桌面环境。

# 2.9 其他

## 2.9.1 提交支持请求

通过购买 SUSE LINUX，您即有权获得免费安装支持。有关支持范围、地址和电话号码的详细信息，请访问我们的网站，网址是 <http://www.novell.com/linux/suse/>。

YaST 提供了一种功能，使您可以将支持请求通过电子邮件直接发送到 SUSE 小组。首先要进行注册。输入所要求的数据—您的注册码位于 CD 封面的背面。对于查询，在以下窗口中选择问题类别，然后提供问题描述。请参见图 2.13 “提交支持请求”。另请参见 YaST 帮助文本，它解释了如何最准确地描述问题以便支持小组能够有效地提供帮助。

---

### Tip

如果需要高级支持，例如对特殊问题的支持，请参考 <http://support.novell.com/linux/> 了解详细信息。

---

### Tip

## 2.9.2 引导日志

引导日志 `/var/log/boot.msg` 中包含计算机启动时显示的屏幕讯息。使用这个 YaST 模块可查看此日志。例如，可检查所有服务和功能是否正常启动。

The screenshot shows the YaST SUSE Support interface. On the left, there is a 'Support Module' section with instructions: 'Enter your personal information as completely as possible in this form. This allows us to reach you if, for example, it is not possible to e-mail you. To avoid additional inquiries, review the support key entered.' The main form area is titled 'SUSE Support' and contains the following fields: 'Enter support data' with radio buttons for 'Mr.' and 'Ms.'; 'First name:' and 'Last name:' text boxes; 'Company:' text box; 'Street:' text box; 'ZIP:' and 'City:' text boxes; 'State:' and 'Country:' text boxes; 'E-mail:' text box; and 'Support Key:' text box. At the bottom, there are 'Back' and 'Next' buttons.

Figure 2.13: 提交支持请求

### 2.9.3 系统日志

系统日志将您的计算机操作记录到 `/var/log/messages`。这里还记录着内核讯息，并按照日期和时间进行排序。

### 2.9.4 装载供应商的驱动程序 CD

使用此模块可自动从包含 SUSE LINUX 驱动程序的 Linux 驱动程序 CD 来安装设备驱动程序。当从头开始安装 SUSE LINUX 时，在安装后应使用此 YaST 模块从供应商 CD 装载所需的驱动程序。

## 2.10 文本方式下的 YaST (ncurses)

本节所针对的读者主要是在其系统上不运行 X 服务器而依赖于基于文本的安装工具的系统管理员和专家。它提供了与以文本方式 (ncurses) 启动和操作 YaST 有关的基本信息。

当以文本方式启动 YaST 时，将首先出现 YaST 控制中心。请参见图2.14 “文本方式下 YaST 的主窗口”。该窗口包含三个区域。左框架有一个深白色边框，其中列出各个模块所属的类别。它使用有色背景来指示活动类别。右框架有一个浅白色边框，其中提供了活动类别中可用模块的概述。底部框架中包含‘帮助’和‘退出’按钮。

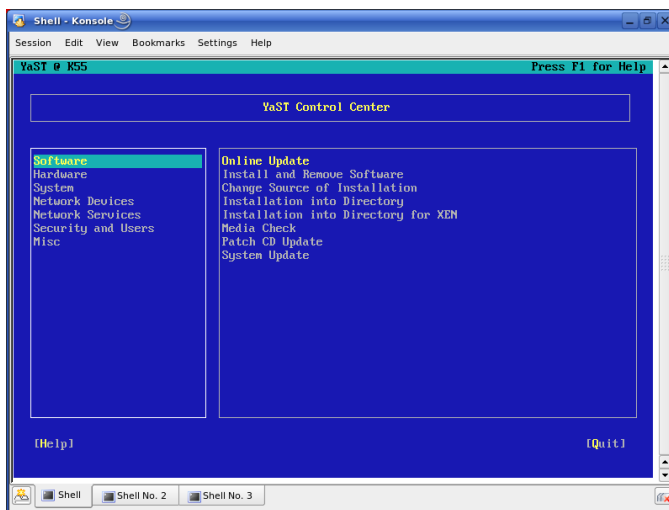


Figure 2.14: 文本方式下 YaST 的主窗口

启动 YaST 控制中心时，将自动选择‘软件’类别。使用  $\downarrow$  键和  $\uparrow$  键可更改类别。要从所选类别启动某个模块，请按  $\rightarrow$  键。模块选择此时显示有深色边框。使用  $\downarrow$  键和  $\uparrow$  键可选择所需模块。按住箭头键在可用模块列表中滚动。选中某个模块后，即会以彩色背景显示模块标题，同时在底部框架中显示简要说明。

按  $\text{Enter}$  键启动所需模块。模块中的各种按钮和选择字段中包含一个具有不同颜色（默认为黄色）的字母。使用  $\text{Alt}-(\text{yellow\_letter})$  可直接选择按钮，而无需使用  $\text{Tab}$  键导航到那里。要退出 YaST 控制中心，可按‘退出’按钮，也可在类别概述中选择‘退出’并按  $\text{Enter}$  键。

### 2.10.1 在模块中导航

下面在介绍 YaST 模块中的控制元素时，均假定所有功能键和  $\text{Alt}$  组合键都

可用并且没有被指派不同的全局功能。有关可能出现的异常的信息，请参见第2.10.2节“组合键的限制”。

在按钮和选择列表中导航 使用 **Tab** 和 **Alt+Tab** 或 **Shift+Tab** 可在按钮和包含选择列表的框架中导航。

在选择列表中导航 使用箭头键 (**↑** 和 **↓**) 可在包含选择列表的活动框架中的各个元素之间导航。如果框架内的项超出了框架宽度，请使用 **Shift+→** 或 **Shift+←** 来左右水平滚动。也可以使用 **Ctrl+E** 或 **Ctrl+A**。如果使用 **→** 键或 **←** 键会导致更改活动框架或当前选择列表（像在控制中心中那样），则可以使用此组合键。

按钮、单选按钮和复选框 要选择带空方括号（复选框）或空圆括号（单选按钮）的按钮，请按 **Space** 或 **Enter** 键。也可以直接使用 **Alt+yellow\_letter** 来选择单选按钮和复选框。在这种情况下，无需使用 **Enter** 键进行确认。如果使用 **Tab** 键导航到某个项目，请按 **Enter** 键执行所选操作或激活相应的菜单项。

功能键 使用各功能键 (**F1** 到 **F12**) 可快速访问多个按钮。功能键和按钮的实际映射关系取决于活动 YaST 模块，因为不同的模块提供不同的按钮（详细信息、信息、添加、删除等）。可以将 **F10** 键用作‘确定’、‘下一步’和‘完成’。按 **F1** 键可访问 YaST 帮助，其中显示了与各个 F 键对应的功能。

## 2.10.2 组合键的限制

如果您的窗口管理器使用全局 **Alt** 组合键，则 YaST 中的 **Alt** 组合键可能无效。像 **Alt** 或 **Shift** 这样的键也可能被终端设置占用。

使用 **Esc** 代替 **Alt** **Esc** 可以代替 **Alt** 来执行 **Alt** 快捷键。例如，**Esc+H** 可代替 **Alt+H**。

使用 **Ctrl+F** 和 **Ctrl+B** 执行向后和向前导航

如果 **Alt** 和 **Shift** 组合键被窗口管理器或终端占用，可改用组合键 **Ctrl+F**（向前）和 **Ctrl+B**（向后）。

功能键的限制 功能键也可用于执行多种功能。某些功能键可能会被终端占用而不能用于 YaST。但 **Alt** 组合键和功能键应该始终在纯文本控制台上完全可用。

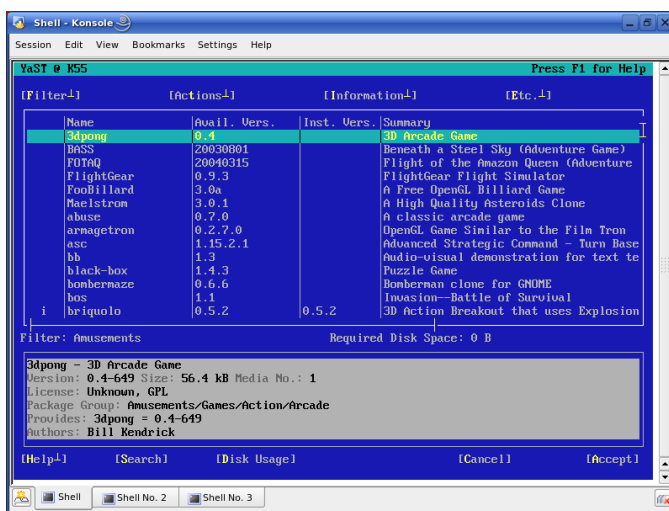


Figure 2.15: 软件安装模块

### 2.10.3 启动单个模块

为了节省时间，可以直接启动单个 YaST 模块。要启动某个模块，请输入 `yast <module_name>`。例如，要启动网络模块，请输入 `yast lan`。要查看系统上所有可用模块名称的列表，请使用 `yast -l` 或 `yast --list`。

### 2.10.4 YOU 模块

与其他任何 YaST 模块一样，可以从命令行以 root 用户身份启动 YaST 联机更新 (YOU) 模块：

```
yast online_update .url <url>
```

`yast online_update` 将启动相应的模块。可以使用选项 `url` 来指定 YOU 要从中获得所有信息和增补程序的服务器（本地服务器或因特网上的服务器）。如果在启动此模块时未指定服务器，请在 YaST 对话框中选择服务器或目录。通过‘配置完全自动更新’可配置 cron 作业执行自动更新。

## 2.11 从命令行联机更新

使用命令行工具 `online.update` 可以自动更新系统（例如，通过脚本）。例如，您可能希望自己的系统搜索特定服务器以获得更新程序，以及定期在指定时间下载增补程序和增补程序信息。但您可能不希望自动安装这些增补程序。而是想查看这些增补程序并选择要在以后安装的增补程序。

要使用此工具，请首先配置一个 `cron` 作业，由它执行以下命令：

```
online_update -u <URL> -g <type_specification>
```

`-u` 引入要从中下载增补程序的目录树的基础 URL。所支持的协议有 `http`、`ftp`、`smb`、`nfs`、`cd`、`dvd` 和 `dir`。`-g` 将增补程序下载到本地目录但不安装它们。也可以通过指定以下类型来过滤增补程序：`security`、`recommended` 或 `optional`。如果未指定任何过滤器，`online.update` 将下载所有新的 `security` 和 `recommended` 增补程序。

可以不检查各个增补程序而立即安装已下载的包。`online.update` 将增补程序保存在目录 `/var/lib/YaST2/you/mnt` 中。要安装这些增补程序，请执行以下命令：

```
online_update -u /var/lib/YaST2/you/mnt/ -i
```

参数 `-u` 指定了要安装的增补程序的本地 URL。`-i` 启动安装过程。

要在安装之前检查已下载的增补程序，请启动 YOU 对话框：

```
yast online_update .url /var/lib/YaST2/you/mnt/
```

YOU 启动和使用包含已下载增补程序的本地目录而不是因特网上的远程目录。选择要安装的增补程序，方法与在包管理器中安装包时相同。

可以使用命令行参数控制 YaST 联机更新的行为。语法是 `online.update [command-line parameter]`。下面列出了可能使用的参数及其功能。

**-u URL** 要从中下载增补程序的目录树的基础 URL。

**-g** 仅下载增补程序。不安装。

**-i** 安装已下载的增补程序。不下载。

**-k** 检查是否有新的增补程序可用。

- c 显示当前配置。不执行任何操作。
- p **product** 要为其下载增补程序的产品。
- v **version** 要为其下载增补程序的产品版本。
- a **architecture** 要为其下载增补程序的产品的基础体系结构。
- d 排演。下载增补程序并模拟安装（系统保持不变，仅用于测试目的）。
- n 不对已下载文件进行签名检查。
- s 显示可用增补程序列表。
- v 冗长方式。
- D 供专家和查错使用的调试方式。

有关 `online.update` 的详细信息，请输入 `online.update -h`。



# 特殊安装过程

SUSE LINUX 有多种安装方式。既可以选择图形化的快速安装，也可以选择支持多项手工调整的基于文本的安装。以下各节说明了各种安装过程以及不同安装源（包括 CD-ROM 和 NFS）的用法。本章还特别介绍了如何解决安装过程中遇到的问题的相关信息，还专有一节对分区进行了详细说明。

3.1	linuxrc . . . . .	72
3.2	通过 VNC 进行安装 . . . . .	73
3.3	通过 YaST 进行基于文本的安装 . . . . .	74
3.4	启动 SUSE LINUX . . . . .	76
3.5	提示和技巧 . . . . .	77
3.6	为 SCSI 设备指派永久设备文件名 . . . . .	81
3.7	LVM 配置 . . . . .	81
3.8	软 RAID 配置 . . . . .	87

## 3.1 linuxrc

每台计算机都有一些在启动时执行的特殊 BIOS 例程，用于初始化硬件。在实际引导进程中，这些例程装载由计算机执行的映像，并且必须控制随后的引导进程。该映像通常是一个引导管理程序，支持用户选择已安装的系统或要安装的系统。选择安装 SUSE LINUX 时会装载一个引导映像，其中包含一个内核和一个名为 linuxrc 的程序。

linuxrc 程序会分析和初始化系统，为实际的安装进程做准备。默认情况下，该程序的运行无需用户参与，运行完毕后即启动 YaST。如果需要向某个模块提供特殊参数，或者硬件检测失败，则最好启动手工安装以交互方式运行 linuxrc。

linuxrc 并不仅限于在安装中使用。您还可以将它用作已安装系统甚至基于独立 RAM 磁盘的应急系统的引导工具。有关详细信息，请参考第 5.4 节“SUSE 应急系统”。

如果系统使用初始 RAM 磁盘 (initrd)，引导时将有一个同样名为 linuxrc 的 shell 脚本来装载模块。此脚本由脚本 /sbin/mkinitrd 动态生成。它与用于安装的 linuxrc 程序完全不同，不应将两者混淆。

### 3.1.1 将参数传递给 linuxrc

可以向 linuxrc 传递参数，用来改变启动行为。linuxrc 会在软盘上或在 /info 的 initrd 中查找一个信息文件。随后，linuxrc 在内核提示符处装载参数。您可以在文件 /linuxrc.config 中编辑默认值。但建议您在信息文件中实施更改。

#### Tip

可以在手工方式下运行 linuxrc。为此，需要在安装提示符处使用参数“manual=1”。

#### Tip

信息文件由关键字和值组成，格式为 key: value。也可以在安装媒体提供的引导提示符处使用格式 key=value 输入这些关键字和值对。文件 /usr/share/doc/packages/linuxrc/linuxrc.html 中提供了所有关键字的列表。下面的列表列出了一些最重要的关键字（带有示例值）：

**Install:URL (nfs, ftp, hd, etc.)** 将安装源指定为 URL。可能的协议包括 cd、hd、nfs、smb、ftp、http 和 tftp。URL 语法符合万维网浏览器所采用的常见形式，例如：

- `nfs://<server>/<directory>`
- `ftp://[user[:password]@]<server>/<directory>`

**Netdevice;**`jeth0` Netdevice: 关键字指定 linuxrc 应使用的接口（如果安装主机上有多个 Ethernet 接口可用）。

**HostIP;**`j10.10.0.2` 指定主机的 IP 地址。

**Gateway;**`j10.10.0.128` 此关键字指定网关，如果安装服务器不在主机的子网内，则可以通过此网关访问安装服务器。

**Proxy;**`j10.10.0.1` Proxy: 关键字定义了 FTP 和 HTTP 协议的代理。

**ProxyPort;**`j3128` 此关键字指定代理使用的端口（如果代理不使用默认端口）。

**Textmode;**`j0—1` 此关键字支持在文本方式下启动 YaST。

**VNC;**`j0—1` VNC 参数通过 VNC 控制安装进程，对于没有图形控制台的主机而言，采用 VNC 可以更方便地进行安装。如果启用了此参数，则在安装主机上激活相应的服务。另请参见 `VNCPassword` 关键字。

**VNCPassword;**`jpassword` 此关键字为 VNC 安装设置口令以控制对会话的访问。

**UseSSH;**`j0—1` 当在文本方式下使用 YaST 执行安装时，此关键字支持通过 SSH 访问 linuxrc。

**SSHPassword;**`jpassword` 此关键字设置用户 `root` 访问 linuxrc 所用的口令。

**Insmod;**`jmodule parameters` 此关键字指定内核应装载的模块和此模块所需的任何参数。必须用空格分隔模块参数。

**AddSwap;**`j0—3—/dev/hda5` 如果设置为 0，则系统不尝试激活交换分区。如果设置为正数，则将与此数字对应的分区作为交换分区激活。也可以指定分区的完整设备名。

## 3.2 通过 VNC 进行安装

VNC（虚拟网络计算）是一个客户机/服务器解决方案，允许通过简单易用的瘦客户机访问远程 X 服务器。此客户程序可用于多种操作系统，包括 Microsoft Windows、Apple 的 MacOS，以及 Linux。

VNC 客户程序 `vncviewer` 用于确保安装进程中的图形显示和 YaST 的处理。在引导系统进行安装之前，对远程计算机进行准备以便它可以访问要通过网络安装的系统。

### 3.2.1 准备进行 VNC 安装

要执行 VNC 安装，请将某些参数传递到内核。这必须在启动内核前完成。要执行此操作，请在引导提示符处输入以下命令：

```
vnc=1 vncpassword=<xyz> install=<source>
```

`vnc=1` 指示应该在安装系统上启动 VNC 服务器。`vncpassword` 是稍后要使用的口令。安装源 (`install`) 可以手工指定（输入相关目录的协议和 URL），也可以包含指令 `slp:/`。在后一种情况中，安装源由 SLP 查询自动确定。第23章网络中的 SLP 服务中包含有关 SLP 的信息。

### 3.2.2 用于 VNC 安装的客户程序

与安装计算机的连接和运行在其上的 VNC 服务器是通过 VNC 客户程序建立的。在 SUSE LINUX 下，使用 `vncviewer`。这是 `xorg-x11-Xvnc` 包的一部分。若要从 Windows 客户机建立与安装系统的连接，则在 Windows 系统上安装 `tightvnc` 程序。此程序在第一张 SUSE LINUX CD 的 `/dosutils/tightvnc` 目录中。

启动所选的 VNC 客户程序。随后，在收到提示时，输入安装系统的 IP 地址和 VNC 口令。

也可以使用支持 Java 的浏览器建立 VNC 连接。要执行此操作，请在浏览器的地址字段中输入以下内容：

```
http://<IP address of the installation system>:5801/
```

一旦建立好连接，YaST 便会启动并开始安装。

## 3.3 通过 YaST 进行基于文本的安装

除了借助图形界面进行安装之外，还可以借助 YaST 的文本版本（控制台方式）来安装 SUSE LINUX。在此文本方式中，也可以使用所有 YaST 模块。如果您不

需要图形界面（例如对于服务器系统），或者如果 X 窗口系统不支持图形卡，则文本方式将非常有用。使用这种安装方式，视觉有障碍的用户可以借助合适的输出设备安装 SUSE LINUX。

首先，在 BIOS 中将引导顺序设置为支持从 CD-ROM 驱动器引导。在驱动器中插入 DVD 或 CD 1，然后重引导计算机。几秒钟后将显示启动屏幕。

在 10 秒钟内使用 **↑** 和 **↓** 键选择‘手工安装’，以防止自动启动已安装的系统。如果您的硬件需要特殊参数（通常不会出现这种情况），请在引导选项中输入这些参数。如果您选择键盘的语言作为安装语言，则键盘布局将是正确的。这将简化参数的输入。

使用 **F2** 键（‘视频方式’）设置安装时屏幕的分辨率。如果您认为图形卡会导致安装过程中出问题，则选择‘文本方式’。然后按 **(Enter)** 键。出现一个带有进度条的对话框，显示 Loading Linux kernel（正在装载 Linux 内核）。内核引导，linuxrc 启动。使用 linuxrc 的菜单继续安装。

通常可以使用内核参数避免其他引导问题。如果是 DMA 导致了问题，请使用启动选项‘安装 — 安全设置’。如果 ACPI（高级配置和电源接口）出现问题，则可以使用以下内核参数。

**acpi=off** 此参数禁用计算机上的整个 ACPI 子系统。如果您的计算机根本不能处理 ACPI 或如果您认为是计算机中的 ACPI 导致问题的产生，则可以使用此参数。

**acpi=oldboot** 关闭 ACPI 中除进行引导必需的部分之外的部分。

**acpi=force** 始终启用 ACPI，即使计算机使用的是日期为 2000 年以前的较早的 BIOS。如果除了 **acpi=off** 之外还设置了此参数，则此参数可以启用 ACPI。

**pci=noacpi** 禁用新 ACPI 系统的 PCI IRQ 路由。

在此连接中，使用关键字“acpi”搜索 <https://portal.suse.com> 中的支持数据库文章。

如果在装载内核或安装过程中出现无法解释的错误，则在引导菜单中选择‘内存测试’以检查内存。Linux 需要硬件符合较高的标准，这意味着必须正确设置内存及其计时。支持数据库中关键字“memtest86”下提供了更多信息。如果可能，请运行整夜的内存测试。

## 3.4 启动 SUSE LINUX

安装之后，决定如何引导 Linux 进行日常操作。以下概述介绍了引导 Linux 的多种选择。最合适的方法取决于预期的目的。

**Linux 引导装载程序** 功能最全面、技术上最讲究的引导系统的方法是使用 Linux 引导管理器，例如 GRUB (Grand Unified Bootloader) 或 LILO (Linux 装载程序)，这些引导管理器允许在引导前选择多种操作系统。可以在安装过程中或稍后借助 YaST 配置引导装载程序。

**引导磁盘** 您可以从引导磁盘引导 Linux。只有当您的系统有软盘驱动器时才可以使用此方法。可以通过 YaST 创建引导磁盘。请参见第 2.8.3 节“创建引导磁盘和应急磁盘”。

如果您在配置其他功能时遇到困难或者如果您要推迟做出有关最终引导机制的决定，则引导磁盘是有用的临时解决方案。引导磁盘可能也是用于 OS/2 或 Windows NT 的合适解决方案。

### Warning

有一些 BIOS 变体会检查引导扇区 (MBR) 的结构，并在安装 GRUB 或 LILO 后错误地显示病毒警告。可以通过进入 BIOS 并查找对应的可调整设置解决此问题。例如，关闭“病毒防护”。您可以稍后再打开此选项。但是，如果您只使用 Linux 操作系统，则没有必要执行此操作。

### Warning

第 8 章 引导装载程序中介绍了不同引导方法的详细讨论。

### 3.4.1 图形 SUSE 屏幕

从 SUSE LINUX 7.2 开始，如果将选项“vga=*jvaluej*”用作内核参数，则图形 SUSE 屏幕会显示在第一个控制台上。如果您使用 YaST 进行安装，则将依照所选的分辨率和图形卡自动激活此选项。

### 3.4.2 禁用 SUSE 屏幕

可以通过三种方法禁用 SUSE 屏幕：

在必要时禁用 SUSE 屏幕。在命令行上输入命令 `echo 0 >/proc/splash` 以禁用图形屏幕。要将其再次激活，请输入 `echo 1 >/proc/splash`。

**默认禁用 SUSE 屏幕** 将内核参数 `splash=0` 添加到您的引导装载程序配置中。第8章 引导装载程序 提供了有关此内容的详细信息。但是，如果您倾向于使用文本方式（这是早期版本中的默认方式），请设置 `vga=normal`。

**完全禁用 SUSE 屏幕** 编译新内核并禁用‘帧缓冲支持’中的选项‘使用启动屏幕而不是引导徽标’。

#### Tip

在内核中禁用帧缓冲支持也会自动禁用启动屏幕。如果您使用自定义内核运行 SUSE，则它不能为系统提供任何支持。

#### Tip

## 3.5 提示和技巧

在某些计算机上，没有可用的 CD-ROM 驱动器，但有可引导软盘驱动器。要安装这样一个系统，必须创建引导磁盘，然后用它引导系统。

需要经过格式化的 3.5" HD 软盘来从提供的映像中创建可引导软盘。CD 1 上的 `boot` 目录包含许多磁盘映像。使用合适的实用程序，可以将这些映像复制到软盘上。以这种方式制作的软盘称为“引导磁盘”。

磁盘映像还包括装载程序 `SYSLINUX` 和程序 `linuxrc`。`SYSLINUX` 支持在引导过程中选择内核以及指定所使用的硬件所需的任何参数。程序 `linuxrc` 支持为您的硬件装载内核模块并随后启动安装。

### 3.5.1 使用 `rawwritewin` 创建引导磁盘

在 Windows 中，可以使用图形实用程序 `rawwritewin` 创建引导磁盘。可以在 CD 1 上的目录 `dosutils/rawwritewin` 中找到该实用程序。

启动时，指定映像文件。映像文件位于 CD 1 上的 `boot` 目录中。您至少需要映像 `bootdisk` 和 `modules1`。若要在文件浏览器中列出这些映像，请将文件类型设置为 `all files`（所有文件）。然后将软盘插入软盘驱动器中并单击‘写入’。

可以以相同的方式创建其他磁盘映像（`modules1`、`modules2`、`modules3` 和 `modules4`）。如果有 USB 或 SCSI 设备或要在安装过程中寻址的网卡或 PCMCIA 卡，则需要这些软盘。如果在安装过程中使用特殊的文件系统，则还需要模块磁盘。

## 3.5.2 使用 rawrite 创建引导磁盘

DOS 实用程序 `rawrite.exe`（在 CD 1 上目录 `dosutils/rawrite` 中）可用于创建 SUSE 引导和模块磁盘。若要使用此实用程序，您需要操作系统为 DOS（例如 FreeDOS）或 Windows 的计算机。

在 Windows XP 中，按以下步骤操作：

1. 插入 SUSE LINUX CD 1。
2. 打开 DOS 窗口（在“开始”菜单中，选择‘附件’→‘命令提示符’）。
3. 通过指定 CD 驱动器的正确路径运行 `rawrite.exe`。本示例假设您当前的位置是硬盘 C: 上的 Windows 目录，您的 CD 驱动器是 D:。

```
d:\dosutils\rawrite\rawrite
```

4. 启动时，该实用程序将询问要复制的文件的源和目标。引导磁盘的映像位于 CD 1 上的 `boot` 目录中。文件名是 `bootdisk`。记住指定 CD 驱动器的路径。

```
d:\dosutils\rawrite\rawrite
RaWrite 1.2 - Write disk file to raw floppy diskette
```

```
Enter source filename: d:\boot\bootdisk
Enter destination drive: a:
```

在输入目标驱动器 `a:` 之后，`rawrite` 提示您插入经过格式化的软盘并按 `(Enter)` 键。随后，将显示复制操作的进度。可以使用 `(Ctrl)-C` 终止该进程。若要创建多个软盘，请重复相同的步骤。

## 3.5.3 在 UNIX-Type 系统中创建引导磁盘

在 UNIX 或 Linux 系统上，您需要 CD-ROM 驱动器和多张经过格式化的软盘。按以下步骤创建引导磁盘：

1. 如果需要先格式化磁盘，请使用：

```
fdformat /dev/fd0u1440
```

此命令还检查软盘是否有错误。请不要使用有错误的媒体。



2. 将 CD 1 插入到 CD-ROM 驱动器中，然后转至此 CD 上的 boot 目录：在当前 SUSE 版本中，不需要再装入 CD。

```
cd /media/cdrom/boot
```

3. 用以下命令创建引导磁盘：

```
dd if=bootdisk1 of=/dev/fd0 bs=8k
```

4. 对映像 bootdisk2 和 bootdisk3 重复此过程。

boot 目录中的 README 文件提供有关软盘映像的详细信息。使用 more 或 less 阅读这些文件。

可以以相同的方式创建其他磁盘映像（modules1、modules2、modules3 和 modules4）。如果有 USB 或 SCSI 设备或要在安装过程中寻址的网卡或 PCMCIA 卡，则需要这些软盘。如果在安装过程中使用特殊的文件系统，则还需要模块磁盘。

模块磁盘的创建并非无关紧要。/usr/share/doc/packages/yast2-installation/vendor.html 提供了如何构建模块磁盘的详细说明。

### 3.5.4 从软盘引导 (SYSLINUX)

引导磁盘用于处理特殊的安装要求（例如，如果 CD-ROM 驱动器不可用）。引导过程是由引导装载程序 SYSLINUX（包 syslinux）启动的。当引导系统时，SYSLINUX 运行最小硬件检测，主要由以下步骤组成：

1. 该程序检查 BIOS 是否提供符合 VESA 2.0 的帧缓冲支持并相应地引导内核。
2. 读取监视数据（DDC 信息）。
3. 读取第一个硬盘的第一个块 (MBR) 以在引导装载程序配置过程中将 BIOS ID 映射到 Linux 设备名。程序将尝试通过 BIOS 的 lba32 功能读取块以确定 BIOS 是否支持这些功能。

## Tip

如果在 SYSLINUX 启动时按住 (Shift) 键，则将跳过所有这些步骤。出于查错的目的，请将行

```
verbose 1
```

插入 `syslinux.cfg` 中，以便引导装载程序显示当前正在执行哪个操作。

## Tip

如果不能从软盘引导计算机，则可能需要将 BIOS 中的引导顺序更改为 A, C, CDROM。

### ► x86

在 x86 系统上，CD 2 也是可引导的。与 CD 1（使用可引导的 ISO 映像）相比，CD 2 是通过 2.88 MB 磁盘映像引导的。如果确定可以从 CD 引导，但使用 CD 1 不起作用，则使用 CD 2（后备解决方案）。◀

## 3.5.5 不支持的 CD-ROM 驱动器

支持大多数 CD-ROM 驱动器。如果 CD-ROM 驱动器发生引导问题，请尝试引导 CD 集的 CD 2。

如果系统既无 CD-ROM 又无软盘，仍可使用外部 CD-ROM（通过 USB 连接）、FireWire 或 SCSI 来引导系统。这主要取决于 BIOS 和使用硬件的交互。如果遇到问题，有时执行 BIOS 更新可能会有用。

## 3.5.6 Installation from a Network Source

Sometimes a standard installation using a CD-ROM drive is not possible. For example, your CD-ROM may not be supported because it is an older proprietary drive. A secondary machine, like a laptop, might not have a CD-ROM drive at all, but only an ethernet adapter. SUSE LINUX offers the possibility of performing the installation on machines without a CD-ROM drive over a network connection. Usually this is done by means of NFS or FTP over ethernet.

No installation support is available for this approach. Therefore, the following procedure should only be attempted by experienced computer users.

To install SUSE LINUX from a network source, two steps are necessary:

1. The data required for the installation (CDs, DVD) must be made available on a machine that will serve as the installation source.
2. The system to install must be booted from floppy disk, CD, or the network and the network must be configured.

The installation source can be made available over various protocols, such as NFS and FTP. See 第3.1节 “linuxrc” for information about the actual installation.

## 3.6 为 SCSI 设备指派永久设备文件名

当引导系统时，将以一种相对动态的方式为 SCSI 设备指派设备文件名。只要设备的数目或配置不发生更改，这便不会构成问题。但是，如果添加了新 SCSI 硬盘并且内核在检测到旧硬盘之前先检测到了新硬盘，则将为旧硬盘指派一个新名称，于是便导致装入表 `/etc/fstab` 中的项不匹配。

要避免此问题，可以使用系统启动脚本 `boot.scsidev`。使用 `/sbin/insserv` 启用此脚本并在 `/etc/sysconfig/scsidev` 中为其设置参数。脚本 `/etc/rc.d/boot.scsidev` 在引导过程中处理 SCSI 设备的设置并在 `/dev/scsi/` 下输入永久设备名。这些名称随后可用在 `/etc/fstab` 中。另外，`/etc/scsi.alias` 可用于定义 SCSI 配置的永久名称。`/etc/scsi` 中设备的命名方案在 `man scsidev` 中有所介绍。

在运行级别编辑器的专家方式中，为级别 B 激活 `boot.scsidev`。于是将在 `/etc/init.d/boot.d` 中创建引导过程生成名称所需的链接。

### Tip

#### 设备名和 udev

对于 SUSE LINUX，虽然仍支持 `boot.scsidev`，但创建永久设备名的首选方式是使用 `udev` 在 `/dev/by-id/` 中创建具有永久名称的设备节点。

### Tip

## 3.7 LVM 配置

本节简要介绍 LVM 的原理及其基本功能，这些功能使 LVM 在许多情况下都很有用。在第3.7.2节 “用 YaST 配置 LVM” 中，将学习如何用 YaST 设置 LVM。

## Warning

使用 LVM 可能会增加一些风险，例如数据丢失。这些风险还包括应用程序崩溃、电源故障及有问题的命令。在实施 LVM 或重配置卷前，请保存数据。决不要在没有备份的情况下工作。

## Warning

### 3.7.1 逻辑卷管理器

逻辑卷管理器 (LVM) 支持在多个文件系统上灵活分配硬盘空间。开发逻辑卷管理器是因为有时只有在安装过程中初始分区完成后才需要更改硬盘空间的分段。因为在运行的系统中修改分区比较困难，LVM 提供了内存空间的虚拟池（卷组，简称 VG），如果需要，可以从中生成逻辑卷 (LV)。操作系统访问这些逻辑卷而不是物理分区。卷组可以跨多个磁盘，这样多个磁盘或部分磁盘可以构成一个 VG。LVM 以这种方式提供了一种对物理磁盘空间的抽象，从而能够以比物理分区更方便、更安全的方式更改硬盘空间的分段。第 1.5.4 节“分区类型”和第 2.8.5 节“分区”中提供了有关物理分区的背景信息。

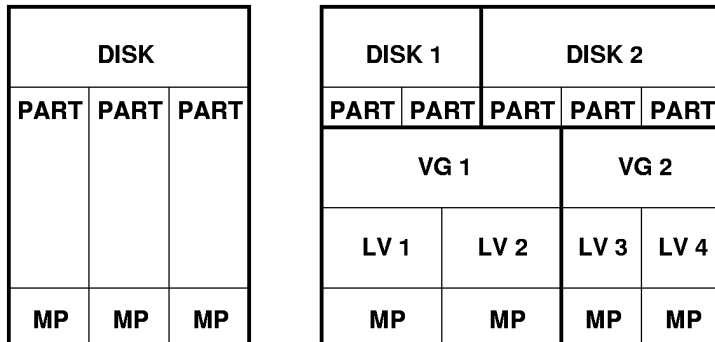


Figure 3.1: 物理分区与 LVM

图 3.1 “物理分区与 LVM” 比较物理分区（左）和 LVM 分段（右）。在左侧，将一个磁盘分成 3 个物理分区 (PART)，每个分区指派了一个装入点 (MP)，以便操作系统可以访问它们。在右侧，有两个磁盘，一个磁盘分为 2 个物理分区，另一个磁盘分为物理 3 个分区。定义了两个 LVM 卷组 (VG 1 和 VG 2)。VG 1 包含 DISK 1 中的 2 个分区和 DISK 2 中的 1 个分区。VG 2 包含 DISK 2 中剩余的

2 个分区。在 LVM 中，将卷组中包含的物理磁盘分区称为物理卷 (PV)。在卷组中，定义了 4 个逻辑卷（从 LV 1 到 LV 4），操作系统可通过相关的装入点使用这些逻辑卷。不同逻辑卷之间的边界不一定是任何分区边界。请参见本示例中 LV 1 和 LV 2 之间的边界。

LVM 功能：

- 可以将多块硬盘或多个分区合并为一个较大的逻辑卷。
- 如果配置合适，当可用空间用完后，可以扩大 LV（例如 /usr）。
- 使用 LVM，甚至可以在正在运行的系统中添加硬盘或 LV。但这需要能执行此类操作的可热插拔的硬件。
- 可以激活将逻辑卷的数据流分布在多个物理卷上的“分带方式”。如果这些物理卷驻留在不同的磁盘上，则可以提高读写性能，这与 RAID 0 类似。
- 使用快照功能可以在正在运行的系统中执行一致的备份（尤其适合服务器）。

通过这些功能，使用 LVM 还对频繁使用的家用 PC 或小型服务器有用。如果您的数据储存量（如数据库、音乐档案或用户目录）不断增长，则 LVM 正是您所需要的工具。此工具支持您使用大于物理硬盘的文件系统。LVM 的另一个优点是最多可以添加 256 个 LV。但是，请记住，使用 LVM 与使用传统的分区截然不同。位于 <http://tldp.org/HOWTO/LVM-HOWTO/> 的官方 LVM HOWTO 提供了有关配置 LVM 的说明和详细信息。

从内核版本 2.6 开始，您便可以使用 LVM 版本 2 了，该版本向下兼容以前的 LVM，从而使您能继续管理以前的卷组。在创建新卷组时，决定是使用新格式还是使用向下兼容的版本。LVM 2 不需要任何内核增补程序。它使用内核 2.6 中集成的设备映射器。此内核只支持 LVM 版本 2。因此，当谈及 LVM 时，本节始终指 LVM 版本 2。

### 3.7.2 用 YaST 配置 LVM

YaST 专家分区程序完成 YaST LVM 配置（请参见第 2.8.5 节“分区”）。此专业分区工具用于编辑和删除现有分区并创建用于 LVM 的新分区。在此，首先单击‘创建’→‘不格式化’创建 LVM 分区，然后选择‘0x8E Linux LVM’作为分区标识符。在创建了 LVM 使用的所有分区后，请单击‘LVM’启动 LVM 配置。

## 创建卷组

如果系统上仍无卷组存在，则系统将提示您添加一个卷组（请参见图3.2“创建卷组”）。可以通过‘添加组’创建附加组，但通常一个卷组就足够了。建议将 `system` 作为卷组的名称，SUSE LINUX 系统文件位于这个卷组中。物理区域大小定义卷组中物理块的大小。卷组中的所有磁盘空间都是按此大小的区块来处理的。通常将这个值设置为 4 MB，并允许物理卷和逻辑卷的最大大小采用 256 GB。如果要设置大于 256 GB 的逻辑卷，则只应增加物理区域大小（例如，增加到 8、16 或 32 MB）。

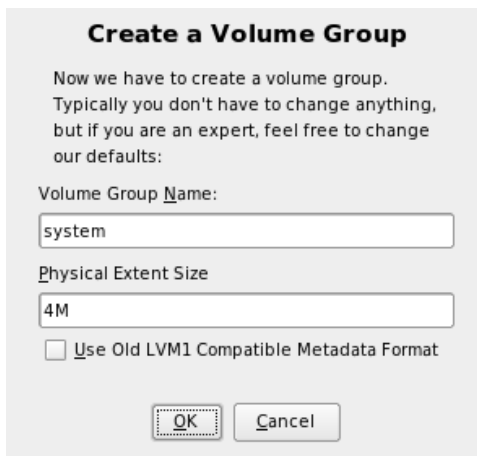


Figure 3.2: 创建卷组

## 配置物理卷

创建了卷组之后，以下对话框将列出类型为“Linux LVM”或“Linux native”的所有分区。未显示交换分区或 DOS 分区。如果已将某个分区指派给卷组，则在列表中显示此卷组的名称。用“--”表示未指派的分区。

如果存在多个卷组，请在选择框的左上角设置当前卷组。使用右上角的按钮可以创建其他卷组和删除现有的卷组。只能删除没有指派任何分区的卷组。指派给卷组的所有分区还被称为物理卷 (PV)。

要将以前未指派的分区添加到所选的卷组中，请先单击该分区，然后单击‘添加卷’。此时，卷组的名称就被输入到所选分区的旁边。将为 LVM 预留的所有分区指派给卷组。否则，分区中的空间仍处于未使用状态。在退出对话框前，必须为

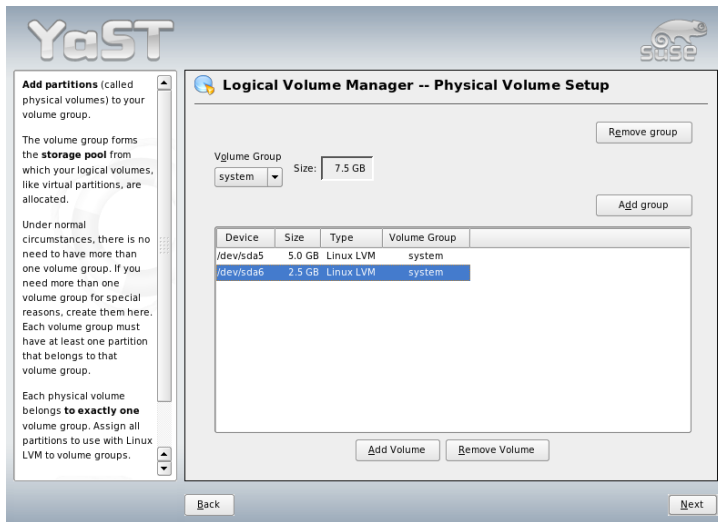


Figure 3.3: 物理卷组

每个卷组指派至少一个物理卷。在指派所有物理卷后，单击‘下一步’继续逻辑卷的配置。

### 配置逻辑卷

在用物理卷填充了卷组后，请定义操作系统应在下一个对话框中使用的逻辑卷。在选择对话框的左上角设置当前卷组。接着，显示当前卷组中的可用空间。下面的列表包含该卷组中的所有逻辑卷。这里列出了为其指派装入点的所有普通 Linux 分区、所有交换分区和所有现有的逻辑卷。根据需要，‘添加’、‘编辑’和‘去除’逻辑卷，直到卷组中的所有空间都用完为止。请为每个卷组至少指派一个逻辑卷。

要创建新逻辑卷，请单击‘添加’并填写打开的弹出窗口。对于分区，可以输入大小、文件系统和装入点。通常，文件系统（如 reiserfs 或 ext2）是在逻辑卷上创建的，然后为其指定装入点。在已安装系统的这个装入点可以找到存储在此逻辑卷中的文件。另外，可以在多个物理卷（分带）之间分布逻辑卷中的数据流。如果这些物理卷驻留在不同的硬盘上，则通常会提高读写性能（与 RAID 0 类似）。但是，只有可以将 LV 所需的硬盘空间平均分配给  $n$  个物理卷，才能正确创建有  $n$  个分带的分带 LV。例如，如果只有两个物理卷可用，则不可能存在

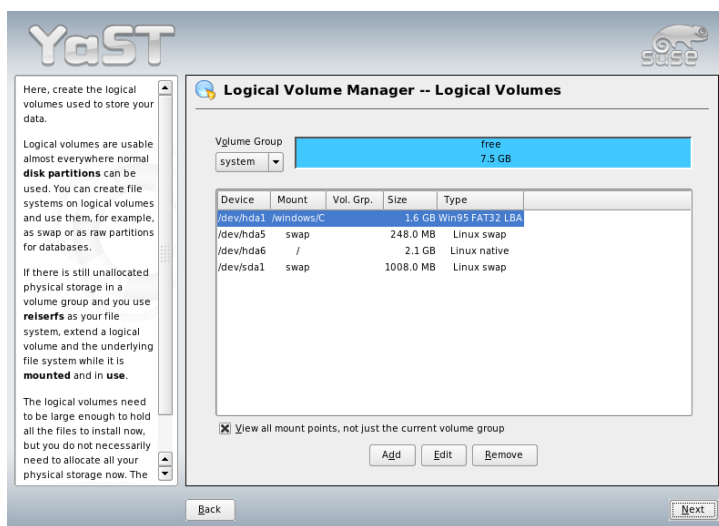


Figure 3.4: 逻辑卷管理

有三个分带的逻辑卷。

### Warning

#### 分带

在此，YaST 无法校验有关分带项的正确性。这里所犯的任何错误只有以后在磁盘中实施 LVM 时才能显现。

### Warning

如果您已在系统上配置了 LVM，则可以立即输入现有的逻辑卷。在继续前，将适当的装入点指派给这些逻辑卷。通过‘下一步’，返回到 YaST 专家分区程序并在此完成工作。

### 直接 LVM 管理

如果您已配置了 LVM 并希望更改某些设置，则可采用替代方法来完成这一工作。在 YaST 控制中心，选择‘系统’→‘LVM’。基本上，此对话框允许执行如上所述相同的操作，但不允许执行物理分区操作。此对话框在两个列表中显示了现有的物理卷和逻辑卷，并且可以使用已介绍的方法来管理 LVM 系统。



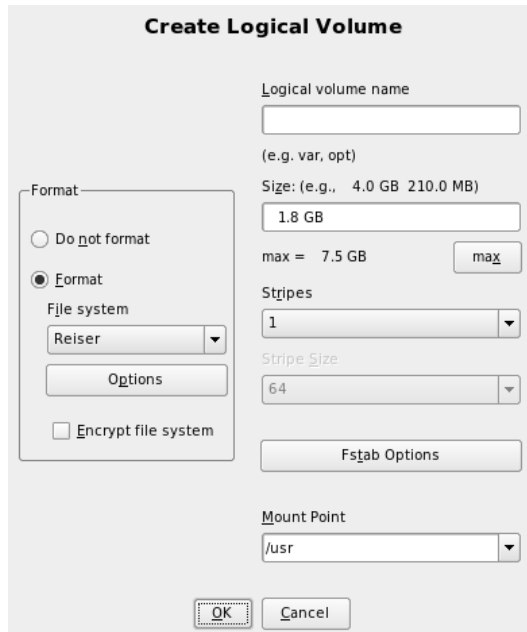


Figure 3.5: 创建逻辑卷

## 3.8 软 RAID 配置

RAID（廉价磁盘冗余阵列）的用途是将多个硬盘分区合并成一个大的虚拟硬盘，以便优化性能和/或数据安全性。但使用此方法是牺牲一个优点来换取另一个优点。大多数 RAID 控制器使用 SCSI 协议，因为此协议可以采用一种比 IDE 协议更有效的方式对较大数目的硬盘进行寻址，并且更适合命令的并行处理。还有一些支持 IDE 或 SATA 硬盘的 RAID 控制器。请参考位于 <http://cdb.suse.de> 的硬件数据库。

### 3.8.1 软 RAID

与 RAID 控制器（通常非常昂贵）一样，软 RAID 也能执行这些任务。借助于 YaST—硬件 RAID 的一个非常合理的替代解决方案，SUSE LINUX 可以将多块硬盘合并成一个软 RAID 系统。RAID 暗示将多块硬盘合成一个 RAID 系统的多

种策略，这些策略的目标、优点及特点各不相同。这些变化形式通常称作 RAID 级别。

常用的 RAID 级别如下：

**RAID 0** 此级别通过将每个文件按块分放到多个磁盘驱动器上，提高了数据访问性能。这实际上并不是真正的 RAID，因为它未提供数据备份，但 RAID 0 已成为这种类型的系统的标准名称。使用 RAID 0，可以将两块或多块硬盘组合在一起。这样性能固然很好，但如果有任何一块硬盘出现故障，都将损坏 RAID 系统并且会丢失数据。

**RAID 1** 此级别为数据提供了充分的安全性，因为它将数据按 1:1 复制到另一块硬盘上。这种方法称为硬盘镜像。如果一块磁盘损坏，则可以使用另一块磁盘上的内容副本。在所有这些硬盘中，只要有一块硬盘没有损坏，您的数据就不会丢失。与使用单个磁盘访问时相比，写性能在复制进程中稍有损失（慢 10% 到 20%），但读访问的速度要大大快于任何一块普通物理硬盘，原因是数据进行了复制，从而可以并行扫描它们。一般来讲，使用级别 1 读事务的速率几乎是使用单个磁盘时的两倍，而写事务的速率与使用单个磁盘时相差无几。

**RAID 2 和 RAID 3** 这些不是典型的 RAID 实现。级别 2 在位一级而不是块一级对数据进行分带。级别 3 则利用专用的校验磁盘在字节一级进行分带，但不能同时处理多个请求。这两种级别都极少使用。

**RAID 4** 级别 4 与级别 0 一样，也是在块一级进行分带，只是结合使用了专用的校验磁盘。当数据盘发生故障时，则可以利用奇偶校验数据来制作一块替代盘。不过，这块校验磁盘可能造成写访问的瓶颈。尽管如此，有时仍使用级别 4。

**RAID 5** RAID 5 是级别 0 和级别 1 在性能和冗余方面经优化后的折衷方案。硬盘空间等于使用的磁盘数减 1。数据分布在这些硬盘上，这一点与 RAID 0 相同。但出于安全原因，在其中一个分区上创建了奇偶校验块。这些块通过 XOR 互相链接，在系统出现故障时，可由相应的奇偶校验块通过 XOR 重新构建内容。对于 RAID 5，在同一时间只能有一块硬盘出现故障。如果一块硬盘出现故障，则必须尽快将其更换，以防止丢失数据。

**其他 RAID 级别** 其他多种 RAID 级别也已开发出来（RAID n、RAID 10、RAID 0+1、RAID 30、RAID 50 等），其中某些级别属于硬件厂商创建的专有实施方法。由于这些级别并不是很普及，所以在此不再赘述。

### 3.8.2 使用 YaST 配置软 RAID

可以通过 YaST 专家分区工具（请参见第2.8.5节“分区”）访问 YaST 软 RAID 配置。此专业分区工具用于编辑和删除现有分区并创建用于软 RAID 的新分区。利用该工具可创建 RAID 分区，方法是首先单击‘创建’→‘不格式化’，然后选择‘0xFD Linux RAID’作为分区标识符。对于 RAID 0 和 RAID 1，至少需要两个分区——对于 RAID 1，通常只需要两个分区。如果使用 RAID 5，则至少需要 3 个分区。建议只采用相同大小的分区。应将 RAID 分区储存在多块硬盘上，以降低由于某块硬盘出现问题而导致数据丢失的风险（RAID 1 和 5），同时还可以优化 RAID 0 的性能。创建了所有用于 RAID 的分区后，请单击‘RAID’→‘创建 RAID’开始 RAID 配置。

在下一个对话框中选择 RAID 级别 0、1 和 5（有关详细信息，请参见第3.8.1节“软 RAID”）。单击‘下一步’后，随即显示的对话框将列出类型为“Linux RAID”或“Linux native”的所有分区（请参见图3.6“RAID 分区”）。未显示交换分区或 DOS 分区。如果已将某个分区指派给 RAID 卷，则在列表中显示此 RAID 设备的名称（例如，/dev/md0）。用“--”表示未指派的分区。

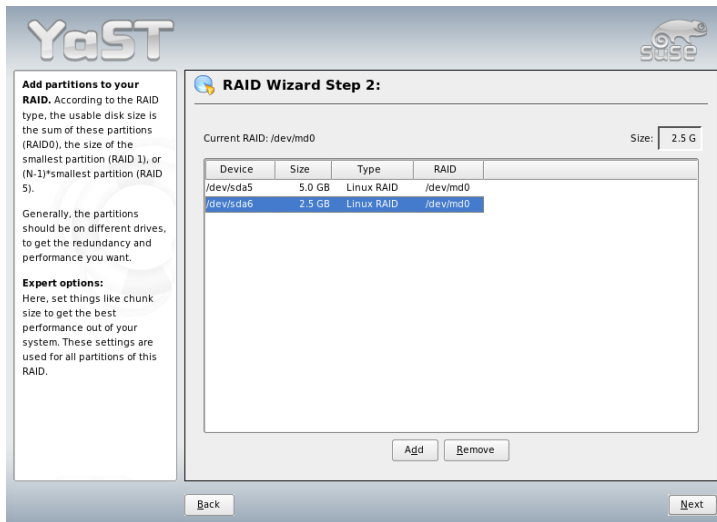


Figure 3.6: RAID 分区

要将以前未指派的分区添加到所选的 RAID 卷中，请先单击该分区，然后单击‘添加’。此时，RAID 设备的名称就被输入到所选分区的旁边。指派所有为

RAID 保留的分区。否则，分区中的空间仍处于未使用状态。指派了所有分区后，单击‘下一步’进入设置对话框，从中对性能进行微调（请参见图3.7“文件系统设置”）。

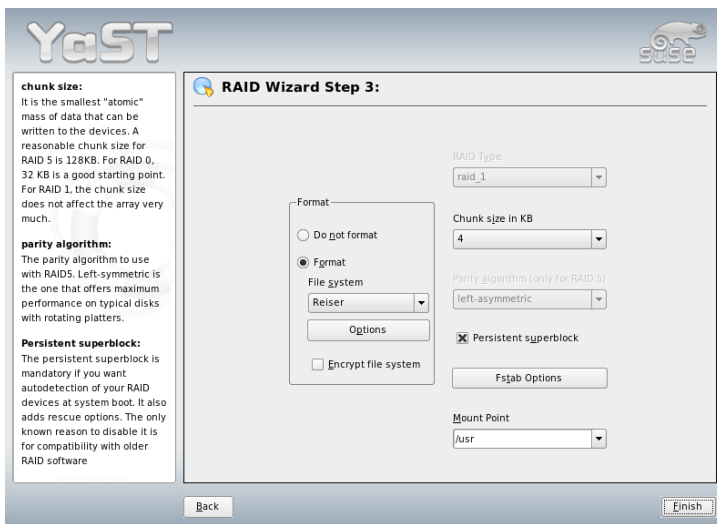


Figure 3.7: 文件系统设置

与传统的分区一样，设置所用的文件系统，以及 RAID 卷的加密方法和装入点。选中‘持久性超级块’确保在引导时以这种方式识别 RAID 分区。单击‘完成’完成配置后，请查看 `/dev/md0` 设备和专家分区工具中指示为 RAID 的其他设备。

### 3.8.3 查错

查看文件 `/proc/mdstats` 以确定 RAID 分区是否受损。如果系统出现故障，请关闭 Linux 系统，然后使用以相同方式分区的新硬盘替换有问题的硬盘。然后重新启动您的系统并输入命令 `mdadm /dev/mdX --add /dev/sdX`。当然，您必须将“X”换成您的特定设备的标识符。此命令会自动将该硬盘集成到 RAID 系统并完全重新构建它。

### 3.8.4 参考信息

位于下列位置的 HOWTO 文档提供了软 RAID 的配置说明和详细信息：

- `/usr/share/doc/packages/raidtools/Software-RAID.HOWTO.html`
- `http://en.tldp.org/HOWTO/Software-RAID-HOWTO.html`

另外还可参考邮件列表，如 `http://www.mail-archive.com/linux-raid@vger.rutgers.edu`。



# 更新系统和包管理

SUSE LINUX 使您可以更新现有系统而不用完全重安装系统。更新有两种类型：**更新个别的软件包和更新整个系统**。还可以使用包管理器 RPM 手工安装包。

4.1	更新 SUSE LINUX . . . . .	94
4.2	版本间的软件更改 . . . . .	95
4.3	RPM — 包管理器 . . . . .	110

## 4.1 更新 SUSE LINUX

从旧版本到新版本，软件的大小有增长的趋势。因此，在进行更新之前，请使用 `df` 查看可用分区空间。如果您怀疑磁盘空间不足，请在进行更新和重分区系统前保护好您的数据。对于每个分区应该具有多少空间，没有一般的经验可以借鉴。空间需求取决于特定的分区配置文件、所选的软件和 SUSE LINUX 的版本号。

### Important

阅读 README 文件（在 DOS 或 Windows 中，阅读 CD 上的 README.DOS 文件）。此文件包含出版此手册后进行的所有更改。

### Important

### 4.1.1 准备工作

在进行更新之前，将以前的配置文件复制在单独的媒体上（例如 Streamer、可移动硬盘或 ZIP 驱动器）以保护数据。这主要适用于储存在 `/etc` 中的文件以及 `/var` 和 `/opt` 中的一些目录和文件。最好将 `/home`（HOME 目录）中的用户数据也写入备份媒体。以 `root` 用户的身份备份此数据。只有 `root` 用户具有读取所有本地文件的权限。

在开始更新之前，记录必要的根分区信息。命令 `df /` 可以列出根分区的设备名。在例 4.1 “使用 `df -h` 列示信息”中，要记录的根分区是 `/dev/hda2`（作为 `/` 装入）。

#### 4.1: 使用 `df -h` 列示信息

```
Filesystem      Size  Used Avail Use% Mounted on
/dev/hda1       1.9G  189M  1.7G  10%  /dos
/dev/hda2       8.9G   7.1G  1.4G  84%  /
/dev/hda5       9.5G   8.3G  829M  92%  /home
```

### 4.1.2 可能的问题

#### 检查 `/etc` 中的 `passwd` 和 `group`

在更新系统之前，确保 `/etc/passwd` 和 `/etc/group` 不包含任何语法错误。为此，以 `root` 用户身份启动校验实用程序 `pwck` 和 `grpck` 并消除任何报告的错误。



## PostgreSQL

在更新 PostgreSQL (postgres) 之前，先转储数据库。请参见 `pg_dump` 的手册页。只有当实际上是在更新之前使用了 PostgreSQL 时才需要执行此操作。

### 4.1.3 使用 YaST 进行更新

完成了第4.1.1节“准备工作”中介绍的准备过程后可以开始更新系统了：

1. 像进行安装时那样引导系统，如第1.1节“系统启动以进行安装”中所述。在 YaST 中，选择语言并选择‘更新现有系统’。不要选择‘全新安装’。
2. YaST 确定是否有多个根分区。如果只有一个根分区，则继续下一步。如果有多个根分区，则选择正确的分区并选择‘下一步’进行确认（在第4.1.1节“准备工作”的示例中，选择的是 `/dev/hda2`）。YaST 读取此分区上旧的 `fstab` 以分析并装入列出的文件系统。
3. 随后便可以在更新过程中制作系统文件的备份副本。此选项将降低更新进程的速度。如果没有最近的系统备份，请使用此选项。
4. 在随后的对话框中，选择只更新已安装的软件或将新软件部件添加到系统中（升级方式）。最好接受建议的组合，例如，‘默认系统’。以后可以使用 YaST 进行调整。

### 4.1.4 更新个别的包

您总是可以更新个别的包，而不考虑整体已更新的环境。但是，从此开始您需要确保系统保持一致。<http://www.novell.com/linux/download/updates/> 中提供了更新建议。

根据您的需要从 YaST 包选择列表中选择部件。如果选择的是对系统整体操作至关重要的包，则 YaST 将发出一个警告。只能在更新方式中更新此类包。例如，许多包包含共享库。如果在正在运行的系统中更新这些程序和应用程序，则系统可能会出现故障。

## 4.2 版本间的软件更改

以下内容详细介绍了版本间各方面进行的更改。本摘要指出了是否已完全重配置基本设置、是否已将配置文件移动到其他位置以及是否对常用应用程序进行了重

大更改。此处将介绍在用户级别或管理员级别进行的影响系统日常使用的重大更改。

我们将及时联机发布所确定的各个版本的问题和特殊问题。请参见下面列出的链接。使用 YaST 联机更新 (YOU) 可以在 <http://www.novell.com/products/linuxprofessional/downloads/> 中访问各个包的重要更新——请参见第 2.3.3 节“YaST 联机更新”。

## 4.2.1 从 8.1 到 8.2

问题和特殊问题：<http://portal.suse.com/sdb/en/2003/04/bugs82.html>

- 对 nVidia-based 图形卡的 3D 支持（更改）：不再包括 RPM `NVIDIA-GLX/NVIDIA_kernel`（包含脚本 `switch2nvidia_glx`）。从 nVidia 万维网站点 (<http://www.nvidia.com>) 为 Linux IA32 下载 nVidia 安装程序，通过此安装程序安装驱动程序，然后使用 SaX2 或 YaST 激活 3D 支持。
- 在新安装过程中，将安装 `xinetd` 而不是 `inetd`，并用安全值进行配置。请参见目录 `/etc/xinetd.d`。但是，在系统更新过程中将保留 `inetd`。
- PostgreSQL 的版本是 7.3。当从版本 7.2.x 进行转换时，请使用 `pg_dump` 执行 `dump/restore`。如果应用程序查询系统目录，则需要进行额外调整，因为版本 7.3 中引入了纲要。有关详细信息，请参见 [http://www.ca.postgresql.org/docs/momjian/upgrade\\_tips\\_7.3](http://www.ca.postgresql.org/docs/momjian/upgrade_tips_7.3)。
- `stunnel` 的版本 4 不再支持任何命令行选项。但是，附带的脚本 `/usr/sbin/stunnel3_wrapper` 可以将命令行选项转换为适合 `stunnel` 的配置文件并在程序启动时使用（用您的选项替换 `OPTIONS`）：

```
/usr/sbin/stunnel3_wrapper stunnel OPTIONS
```

所生成的配置文件将被输出到默认输出，从而可以使用这些指定生成永久配置文件。

- `openjade` (`openjade`) 是当前使用的 DSSSL 引擎，而不是运行 `db2x.sh` (`docbook-toys`) 时使用的 `jade` (`jade_dsl`)。出于兼容性原因，所提供的各程序不带前缀 `o`。

如果您自己的应用程序依赖目录 `jade_dsl` 和以前安装在此处的文件，则对它们进行调整使之适合新目录 `/usr/share/sgml/openjade` 或用下列命令以 `root` 用户身份创建链接：

```
cd /usr/share/sgml
rm jade_dsl
ln -s openjade jade_dsl
```

为避免与 `rszsz` 冲突，命令行工具 `sx` 继续称为 `s2x`、`sgml2xml` 或 `osx`。

## 4.2.2 从 8.2 到 9.0

问题和特殊问题：<http://portal.suse.com/sdb/en/2003/07/bugs90.html>

- 目前提供 RPM 包管理器的版本 4。构建包的功能已转移到单独的程序 `rpmbuild`。`rpm` 继续用于安装、更新和数据库查询。请参见第 4.3 节“RPM — 包管理器”。
- 目前，提供包 `foomatic-filters` 用于打印。此内容是从 `cups-drivers` 拆分的，因为它甚至可用于在没有安装 CUPS 的情况下进行打印。这样，YaST 支持独立于打印系统（CUPS 或 LPRng）的配置。此包的配置文件是 `/etc/foomatic/filter.conf`。
- 对于 LPRng 和 `lpdfilter`，目前还需要包 `foomatic-filters` 和 `cups-drivers`。
- 可以通过 `/etc/xml/suse-catalog.xml` 中的项访问附带软件包的 XML 资源。不应使用 `xmlcatalog` 对此文件进行编辑，因为这将导致删除进行正确更新所需的结构注释。通过 `/etc/xml/catalog` 中的 `nextCatalog` 语句对 `/etc/xml/suse-catalog.xml` 进行访问，从而使 XML 工具（例如 `xmllint` 或 `xsltproc`）可以自动查找本地资源。

## 4.2.3 从 9.0 到 9.1

请参考位于 <http://portal.suse.com> 的 SUSE 支持数据库中关键字特殊功能下的文章“SUSE LINUX 9.1 中的已知问题和特殊功能”。我们为每个 SUSE LINUX 版本发布这些文章。

## 升级到内核 2.6

SUSE LINUX 目前完全基于内核 2.6。不能再使用以前的版本 2.4，因为附带的应用程序不能用于内核 2.4。请注意以下详细信息：

- 装载模块是通过文件 `/etc/modprobe.conf` 进行配置的。不再使用文件 `/etc/modules.conf`。YaST 将尝试转换文件（另请参见脚本 `/sbin/generate-modprobe.conf`）。
- 模块具有后缀 `.ko`。
- 刻录 CD 不再需要模块 `ide-scsi`。
- ALSA 声卡模块选项已去除了前缀 `snd_`。
- `sysfs` 现在补充 `/proc` 文件系统。
- 电源管理（特别是 ACPI）已得到改进，可以通过 YaST 模块进行配置。

## 装入 VFAT 分区

当装入 VFAT 分区时，必须将参数 `code` 更改为 `codepage`。如果装入 VFAT 分区出现问题，请检查文件 `/etc/fstab` 是否包含以前的参数名。

## 通过 ACPI 执行待机和暂停

内核 2.6 支持通过 ACPI 执行待机和暂停。此功能仍处于试验阶段，某些硬件部件可能不支持此功能。要使用此功能，您需要 `powersave` 包。`/usr/share/doc/packages/powersave` 中提供有关此包的信息。`kpowersave` 包中提供图形前端。

## 输入设备

有关与输入设备相关的更改，请参考上面介绍的 Portal 文章“SUSE LINUX 9.1 中的已知问题和特殊功能”（在位于 <http://portal.suse.com> 的支持数据库中关键字特殊功能下）。

## 本机 POSIX 线程库和 glibc 2.3.x

依据 NGPT（下一代 POSIX 线程）链接的应用程序不能用于 glibc 2.3.x。所有受影响的应用程序如果未随 SUSE LINUX 附带，则必须使用 linuxthreads 或 NPTL（本机 POSIX 线程库）进行编译。建议使用 NPTL，因为它是将来的标准。

如果使用 NPTL 出现问题，则可以使用较早的 linuxthreads 实施，方法是设置以下环境变量（将 `<kernel-version>` 替换为各自内核的版本号）：

```
LD_ASSUME_KERNEL=kernel-version
```

可以使用以下版本号：

2.2.5 (i386、i586) : linuxthreads (不带浮点堆栈)

2.4.1 (AMD64、i586、i686) : linuxthread (带有浮点堆栈)

有关内核和 linuxthreads（带有浮点堆栈）的说明：使用 `errno`、`h_errno` 和 `_res` 的应用程序必须通过 `#include` 包括标题文件（`errno.h`、`netdb.h` 和 `resolv.h`）。对于使用线程取消、具有多线程支持的 C++ 程序，必须使用环境变量 `LD_ASSUME_KERNEL=2.4.1` 提示使用 linuxthreads 库。

## 本机 POSIX 线程库的调整

NPTL 以线程包的形式包括在 SUSE LINUX9.1 中。NPTL 与以前的 linuxthreads 库是二进制兼容的。但是，对于 linuxthreads 不符合 POSIX 标准的方面，需要进行 NPTL 调整。其中包括：信号处理、在所有线程中返回相同值的 `getpid` 以及向 `pthread_atfork` 注册的线程处理程序在使用 `vfork` 的情况下不工作。

## 网络接口配置

网络接口的配置已更改。以前，在配置不存在的接口后对硬件进行初始化。现在，系统将搜索新硬件并立即对其进行初始化，从而配置新网络接口。

已为配置文件引入了新名称。由于网络接口的名称是动态生成的并且热插拔设备的使用不断增加，像 `eth0`、`eth1` 这样的名称不再适用于配置目的。出于此原因，转而使用唯一指定（例如 MAC 地址或 PCI 插槽）对接口配置进行命名。您可以在接口名出现后立即使用它们。仍然可以使用像 `ifup eth0` 或 `ifdown eth0` 这样的命令。

设备配置位于 `/etc/sysconfig/hardware` 中。这些设备提供的接口通常位于 `/etc/sysconfig/network` 中（具有不同的名称）。请参见 `/usr/share/doc/packages/sysconfig/README` 中的详细说明。

## 声卡配置

更新之后，必须重配置声卡。这可以通过 YaST 声卡模块完成。以 root 用户身份输入 `yast2 sound`。

## 顶级域 `.local` 作为“链路本地”域

解析程序库将顶级域 `.local` 视为“链路本地”域，并向多路广播地址 `224.0.0.251` 端口 `5353` 发送多路广播 DNS 查询，而不是常规 DNS 查询。这是一个不兼容的更改。如果域 `.local` 已用在名称服务器配置中，则使用另一个域名。有关多路广播 DNS 的详细信息，请参见 <http://www.multicastdns.org>。

## 系统范围的 UTF-8 编码

系统的默认编码是 UTF-8。这样，当执行标准安装时，将使用 UTF-8 编码进行区域设置，如 `en_US.UTF-8`。有关详细信息，请参见 <http://www.suse.de/~mfabian/suse-cjk/locales.html>。

## 将文件名转换为 UTF-8

（除非特别指定）以前创建的文件系统中的文件不将 UTF-8 编码用于文件名。如果这些文件名包含非 ASCII 字符，则这些字符会变为乱码。要更正此问题，请使用 `convmv` 脚本，该脚本将文件名编码转换为 UTF-8。

## 与 POSIX 标准 2001 兼容的 Shell 工具

在默认设置中，`coreutils` 包（`tail`、`chown`、`head`、`sort` 等）中的 shell 工具不再符合 POSIX 标准 1992 但符合 POSIX 标准 2001 (*Single UNIX Specification, version 3 == IEEE Std 1003.1-2001 == ISO/IEC 9945:2002*)。可以使用以下环境变量强制执行以前的行为：

```
_POSIX2_VERSION=199209
```

新值为 `200112`，用作 `_POSIX2_VERSION` 的默认值。可以在 <http://www.unix.org> 查看 SUS 标准（免费但需要注册）。

*Table 4.1: POSIX 1992 与 POSIX 2001 的比较*

POSIX 1992	POSIX 2001
<code>chown tux.users</code>	<code>chown tux:users</code>

<code>tail +3</code>	<code>tail -n 3</code>
<code>head -1</code>	<code>head -n 1</code>
<code>sort +3</code>	<code>sort -k 4</code>
<code>nice -10</code>	<code>nice -n 10</code>
<code>split -10</code>	<code>split -l 10</code>

### Tip

第三方软件可能尚不符合新标准。在这种情况下，按上面的介绍设置环境变量。

### Tip

## `/etc/gshadow` 不再使用

因为 `/etc/gshadow` 多余，已将其放弃并去除，原因如下：

- 不受 `glibc` 支持。
- 此文件没有正式接口。即使 `Shadow Suite` 也不包含这一接口。
- 大多数检查组口令的工具不支持该文件，出于上述原因，这些工具忽略该文件。

## OpenLDAP

因为数据库格式已更改，必须重新生成数据库。更新过程中，系统会尝试自动执行此转换。但是，肯定会有转换失败的情况。

纲要检查进行了重大改进。因此，现在不再能执行一些可以在以前的 LDAP 服务器上执行的不符合标准的操作。

考虑到 ACL，对配置文件的语法进行了部分更改。安装之后，可以在文件 `/usr/share/doc/packages/openldap2/README.update` 中获得有关更新的信息。

## Apache 1.3 替换为 Apache 2

Apache 万维网服务器（版本 1.3）已被替换为 Apache 2。可以在万维网网页 <http://httpd.apache.org/docs-2.0/en/> 获得版本 2.0 的详细文档。在具有 HTTP 服务器安装的系统上，更新程序去除 Apache 包并安装 Apache 2。随后，必须通过 YaST 或手工对系统进行调整。`/etc/httpd` 中的配置文件现在位于 `/etc/apache2` 中。

可以选择线程或进程来处理多个并行查询。已将进程管理移动到一个独立的模块，即多重处理模块 (MPM)。因此，Apache 2 需要 `apache2-prefork` 包（建议使用，更稳定）或 `apache2-worker` 包。根据 MPM 的不同，Apache 2 以不同的方式响应查询。这会影响到性能以及模块的使用。第 30.4 节“线程”中对这些特性进行了详细介绍。

Apache 2 现在支持下一代因特网协议 IPv6。

实施一种机制，使模块程序员可以指定所需的模块装载顺序，用户便无需执行此任务。执行模块的顺序非常重要，以前是通过装载顺序确定的。例如，必须先装载只允许鉴定用户访问某些特定资源的模块，以防止不具有访问权限的用户看到这些页面。

可以使用过滤器处理对 Apache 的查询和来自 Apache 的响应。

## 从 Samba~2.x 到 Samba~3.x

在完成从 Samba~2.x 到 Samba~3.x 的更新后，不再提供 `winbind` 鉴定。仍然可以使用其他鉴定方法。出于此原因，以下程序已被去除：

```
/usr/sbin/wb_auth  
/usr/sbin/wb_ntlmauth  
/usr/sbin/wb_info_group.pl
```

另请参见 <http://www.squid-cache.org/Doc/FAQ/FAQ-23.html#ss23.5>。

## OpenSSH 更新（版本 3.8p1）

`gssapi` 支持已被替换为 `gssapi-with-mic` 以防止潜在的 MITM 攻击。这两个版本不兼容。这意味着不能使用以前分发中的 Kerberos 票证进行鉴定，因为现在使用其他鉴定方法。



## SSH 和终端应用程序

当从远程主机（特别是通过 SSH、Telnet 和 RSH）建立版本 9（激活了 UTF-8 的标准配置）与较早系统（SUSE LINUX 9.0 和以前的版本）的连接时，终端应用程序可能会显示有问题的字符。

这是因为 OpenSSH 不能转发本地设置。因此，将使用可能与远程终端设置不匹配的默认系统设置。这会影响到以文本方式运行的 YaST 和以普通用户（不是 root 用户）身份从远程主机执行的应用程序。只有当用户更改了 root 的标准区域设置（只有 LC\_CTYPE 是默认设置的）时，才会影响到 root 用户启动的应用程序。

### 放弃使用 libiodbc

FreeRADIUS 的用户现在必须链接 unixODBC，因为现在已放弃使用 libiodbc。

### /usr/share/xml 中的 XML 资源

FHS（请参见第 A 节“标准和规范”）现在要求在 /usr/share/xml 中安装 XML 资源（DTD、样式表等）。因此，/usr/share/sgml 中不再提供某些目录。如果遇到问题，请修改脚本和 Makefile 或使用正式编目（特别是 /etc/xml/catalog 或 /etc/sgml/catalog）。

### 具有 subfs 的可移动媒体

目前，可移动媒体集成有 subfs。不再需要通过 mount 手工装入媒体。要装入媒体，只需在 /media 中更改为相应的设备目录即可。只要程序在访问媒体，就不能弹出该媒体。

## 4.2.4 从 9.1 到 9.2

请参考位于 <http://portal.suse.com> 的 SUSE 支持数据库中关键字特殊功能下的文章“SUSE LINUX 9.2 中的已知问题和特殊功能”。

### 安装过程中在建议对话框中激活防火墙

为提高安全性，安装结束时在建议对话框中激活附带的防火墙解决方案 SuSEFirewall2。这意味着最初将关闭所有端口，如果需要，可以在建议对话框中将其打开。默认情况下，不能从远程系统登录。此外，还会影响到网络浏览和多路广播应用程序，如 SLP、Samba（“网络邻居”）以及某些游戏程序。您可以使用 YaST 对防火墙设置进行微调。

如果在安装或配置服务过程中需要进行网络访问，则相应的 YaST 模块将为所有内部和外部接口打开所需的 TCP 和 UDP 端口。如果不想这样，用户可以在 YaST 模块中关闭端口或指定其他具体的防火墙设置。

*Table 4.2: 重要服务使用的端口*

服务	端口
HTTP 服务器	根据 “list” 语句设置防火墙（仅限 TCP）
邮件 (postfix)	smtp 25/TCP
Samba 服务器	netbios-ns 137/TCP; netbios-dgm 138/TCP; netbios-ssn 139/TCP; microsoft-ds 445/TCP
DHCP 服务器	bootpc 68/TCP
DNS 服务器	domain 53/TCP; domain 53/UDP
DNS 服务器	对 SuSEFirewall2 中端口映射器的附加特殊支持
端口映射器	sunrpc 111/TCP; sunrpc 111/UDP
NFS 服务器	nfs 2049/TCP
NFS 服务器	附加端口映射器
NIS 服务器	激活端口映射
TFTP	tftp 69/TCP
CUPS (IPP)	ipp 631/TCP; ipp 631/UDP

## KDE 和 IPv6 支持

默认情况下，不能在 KDE 中启用 IPv6 支持。您可以使用 YaST 的 `/etc/sysconfig` 编辑器启用该支持。禁用此功能的原因在于 IPv6 地址得不到所有因特网服务提供者的充分支持，从而导致浏览万维网时出现错误讯息，并且显示万维网网页时出现延迟。

## YaST 联机更新和“增量包”

YaST 联机更新现支持一种特殊的 RPM 包，这种包仅储存与给定基础包不同的二进制内容。这项技术显著减少了包的大小并缩短了下载时间。不过，因需要重组最终包，致使 CPU 负载加重。在 `/etc/sysconfig/onlineupdate` 中，配置 YOU 是否应使用这些“增量包”。有关技术细节，请参见 `file:///usr/share/doc/packages/deltarpm/README`。

## 打印系统配置

安装结束时（建议对话框），必须在防火墙配置中打开打印系统所需的端口。CUPS 需要端口 631/TCP 和端口 631/UDP，要执行一般操作，不应关闭这两个端口。要通过 LPD 或 SMB 进行打印，还应该打端口 515/TCP（用于以前的 LPD 协议）和 Samba 使用的端口。

## 更改为 X.Org

兼容性链接简化了从 XFree86 到 X.Org 的更改，这些链接支持访问使用以前名称的重要文件和命令。

Table 4.3: 命令

XFree86	X.Org
XFree86	Xorg
xf86config	xorgconfig
xf86cfg	xorgcfg

Table 4.4: */var/log* 中的日志文件

XFree86	X.Org
XFree86.0.log	Xorg.0.log
XFree86.0.log.old	Xorg.0.log.old

在更改为 X.Org 的过程中，将包的名称从 XFree86\* 重命名为 xorg-x11\*。

## 适用于 X11 的终端模拟器

我们已去除了多个终端模拟器，原因是不再进行维护或者在默认环境下不能使用，尤其是不支持 UTF-8。SUSE LINUX 提供各种标准终端，如 xterm、KDE 终端、GNOME 终端和 mlterm（适用于 X 的多语言终端模拟器，有可能代替 aterm 和 eterm）。

## powersave 包中的更改

*/etc/sysconfig/powersave* 中的配置文件已更改：

*Table 4.5: 拆分 /etc/sysconfig/powersave 中的配置文件*

拆分前	现在拆分为
/etc/sysconfig/powersave/common	common
	cpufreq
	events
	battery
	sleep
	thermal

/etc/powersave.conf 已不再使用。现有变量已移至表4.5 “拆分 /etc/sysconfig/powersave 中的配置文件” 中列出的文件。如果更改 /etc/powersave.conf 中的“event”变量，则必须在 /etc/sysconfig/powersave/events 中进行调整。

休眠状态的名称从：

- 暂停（ACPI S4、APM 暂停）
- 待机（ACPI S3、APM 待机）

更改为：

- 暂停到磁盘（ACPI S4、APM 暂停）
- 暂停到 RAM（ACPI S3、APM 暂停）
- 待机（ACPI S1、APM 待机）

### **OpenOffice.org (OOo)**

目录： OOo 现在安装在 /usr/lib/ooo-1.1 而不是 /opt/OpenOffice.org 中。用户设置的默认目录现在是 ~/.ooo-1.1 而不是 ~/OpenOffice.org1.1。

包装程序： 一些新包装程序可用于启动 OOo 部件。新名称显示在表4.6 “包装程序” 中。

Table 4.6: 包装程序

旧名称	新名称
/usr/X11R6/bin/OOo-calc	/usr/bin/oocalc
/usr/X11R6/bin/OOo-draw	/usr/bin/oodraw
/usr/X11R6/bin/OOo-impress	/usr/bin/ooimpress
/usr/X11R6/bin/OOo-math	/usr/bin/oomath
/usr/X11R6/bin/OOo-padmin	/usr/sbin/oopadmin
/usr/X11R6/bin/OOo-setup	-
/usr/X11R6/bin/OOo-template	/usr/bin/oofromtemplate
/usr/X11R6/bin/OOo-web	/usr/bin/ooweb
/usr/X11R6/bin/OOo-writer	/usr/bin/oowriter
/usr/X11R6/bin/OOo	/usr/bin/ooffice
/usr/X11R6/bin/OOo-wrapper	/usr/bin/ooo-wrapper

包装程序现在支持使用选项 `--icons-set` 在 KDE 和 GNOME 图标之间进行切换。不再支持以下选项：`--default-configuration`、`--gui`、`--java-path`、`--skip-check`、`--lang`（现在通过区域设置确定语言）、`--messages-in-window` 和 `--quiet`。

**KDE 和 GNOME 支持：** 可以在 `OpenOffice_org-kde` 和 `OpenOffice_org-gnome` 包中获得 KDE 和 GNOME 扩展。

### 混音器 `kmix`

预设混音器 `kmix` 作为默认设置。对于高端硬件，还提供其他混音器，如 `QAMix/KAMix`、`envy24control`（仅限 `ICE1712`）或 `hdspmixer`（仅限 `RME Hammerfall`）。

### DVD 刻录

以往，我们需要从 `cdrecord` 包中应用 `cdrecord` 二进制文件的增补程序以支持 DVD 刻录功能。而现在安装的新二进制文件 `cdrecord-dvd` 即提供这个增补程序。

dvd+rw-tools 包中的 growisofs 程序现在支持刻录所有 DVD 介质（DVD+R、DVD-R、DVD+RW、DVD-RW、DVD+RL）。建议用此程序来代替经过修补的 cdrecord-dvd。

### 多个内核

可以同时安装多个内核。此功能旨在支持管理员从一个内核升级到另一个内核，方法是安装新内核，确认新内核能按预期运行后，卸载旧内核。尽管 YaST 尚不支持此功能，仍可以使用 `rpm -i <package>.rpm` 通过 shell 轻松地安装和卸载内核。有关通过命令行管理包的信息，请参考第 4.3 节“RPM — 包管理器”。

默认的引导装载程序菜单中包含一个内核项。在安装多个内核前，为这些加装的内核添加一个对应项很有用，这样便于选择这些内核。在安装新内核前处于活动状态的内核可以作为 `vmlinuz.previous` 和 `initrd.previous` 进行访问。通过创建类似于默认项的引导装载程序项，并令此项指向 `vmlinuz.previous` 和 `initrd.previous` 而不是 `vmlinuz` 和 `initrd`，可以访问先前活动的内核。另外，GRUB 和 LILO 支持带通配符的引导装载程序项。有关详细信息，请参考 GRUB 信息页 (`info grub`) 和 `lilo.conf` (5) 手册页。

## 4.2.5 从 9.2 到 9.3

请参考位于 <http://portal.suse.com> 的 SUSE 支持数据库中关键字特殊功能下的文章“SUSE LINUX 9.3 中的已知问题和特殊功能”。

### 在内核提示符处启动手工安装

‘手工安装’方式已不再出现在引导装载程序屏幕上。不过，您仍可以在引导提示符处使用 `manual=1` 令 `linuxrc` 转换为手工方式。通常不必这样做，因为您可以在内核提示符处直接设置安装选项，例如 `textmode=1` 或将 URL 设为安装源。

### 用于网络鉴定的 Kerberos

Kerberos 已代替 heimdal 成为默认的网络鉴定方法。不能自动转换现有的 heimdal 配置。在系统更新过程中，配置文件的备份副本将按表 4.7 “备份文件” 中所示创建。

Table 4.7: 备份文件

旧文件	备份文件
/etc/krb5.conf	/etc/krb5.conf.heimdal
/etc/krb5.keytab	/etc/krb5.keytab.heimdal

客户机配置 (/etc/krb5.conf) 与 heimdal 的配置很相似。如果没有任何特殊配置，完全可以将 `kpasswd.server` 参数替换为 `admin.server`

不可能控制与服务器 (kdc/kadmind) 相关的数据。系统更新后，旧的 heimdal 数据库在 /var/heimdal 下仍可用；MIT kerberos 在 /var/lib/kerberos/krb5kdc 下维护其数据库。

### X.Org 配置文件

配置工具 SaX2 将 X.Org 配置设置写入 /etc/X11/xorg.conf。在从头安装过程中，不会创建从 XF86Config 到 xorg.conf 的任何兼容性链接。

### PAM 配置

**common-auth** auth 部分的默认 PAM 配置

**common-account** 帐户部分的默认 PAM 配置

**common-password** 更改口令的默认 PAM 配置

**common-session** 会话管理的默认 PAM 配置

您应该在您的应用程序特有的配置文件中包括这些默认的配置，因为修改和维护一个配置文件而不是系统中以往有的约 40 个文件要容易些。如果您稍后安装某个应用程序，该程序将继承已应用的更改，而且管理员也不必记着调整该配置。

所做的更改很简单：如果您使用下面的配置文件（应该是多数应用程序的默认配置）：

```
#%PAM-1.0
auth      required      pam_unix2.so
account  required      pam_unix2.so
password required      pam_pwcheck.so
password required      pam_unix2.so   use_first_pass use_authtok
#password required    pam_make.so     /var/yp
session  required      pam_unix2.so
```

可以将其改为：

```
##PAM-1.0
auth      include      common-auth
account   include      common-account
password  include      common-password
session   include      common-session
```

## 4.3 RPM — 包管理器

在 SUSE LINUX 中，RPM (Red Hat 包管理器) 用于管理软件包。其主要程序是 rpm 和 rpmbuild。用户、系统管理员和包构建人员可以查询强大的 RPM 数据库以获得有关已安装软件的详细信息。

本质上，rpm 具有 5 种方式：安装、卸载或更新软件包；重建 RPM 数据库；查询 RPM 库或个别的 RPM 档案；包的完整性检查和包签名。rpmbuild 可用于从最初的源代码构建可安装包。

用特殊的二进制格式对可安装 RPM 档案进行打包。这些档案由要安装的程序文件和某些元信息组成，这些元信息供 rpm 在安装过程中配置软件包使用或者储存在 RPM 数据库中进行存档。RPM 档案通常具有扩展名 .rpm。

rpm 可用于管理符合 LSB 的包。有关 LSB 的信息，请参考第 A 节“标准和规范”。

### Tip

对于许多包，已将软件开发所需的部件（库、标题、包含文件等）放入单独的包中。只有当您自己要自己编译软件时才需要这些开发包（例如最新的 GNOME 包）。可以通过扩展名 -devel 确定这些开发包，例如包 alsa-devel、gimp-devel 和 kdelibs-devel。

Tip

### 4.3.1 校验包真实性

SUSE LINUX RPM 包具有 GnuPG 签名。包括指纹的密钥是：

```
1024D/9C800ACA 2000-10-19 SuSE Package Signing Key <build@suse.de>
Key fingerprint = 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA
```



命令 `rpm --checksig apache-1.3.12.rpm` 可用于校验 RPM 包的签名以确定它确实来自 SUSE 还是其他可信机构。特别建议对来自因特网的更新包使用此命令。SUSE 公共包签名密钥通常驻留在 `/root/.gnupg/` 中。该密钥还位于目录 `/usr/lib/rpm/gnupg/` 中，允许一般用户校验 RPM 包的签名。

### 4.3.2 管理包：安装、更新和卸载

通常，RPM 档案的安装非常简单：`rpm -i <package>.rpm`。使用此命令可以安装包，但前提是满足其依赖性并且不与其他包冲突。如果出现错误讯息，`rpm` 将请求那些需要安装的包以满足依赖性要求。在后台，RPM 数据库确保不出现冲突——一个特定文件只能属于一个包。通过选择不同的选项，您可以强制 `rpm` 忽略这些默认设置，但这只供专家用户使用。否则将影响系统的完整性并可能使系统无法更新。

选项 `-U`（即 `--upgrade`）和 `-F`（即 `--freshen`）可用于更新包。例如，`rpm -F <package>.rpm`。此命令将去除旧版本的文件并立即安装新文件。两个版本之间的差别是 `-U` 安装系统中以前不存在的包，但 `-F` 只更新以前安装的包。更新时，`rpm` 使用以下策略小心更新配置文件：

- 如果配置文件未被系统管理员更改，则 `rpm` 将安装适当文件的新版本。系统管理员无需执行任何操作。
- 如果更新前配置文件已被系统管理员更改，则 `rpm` 将以扩展名 `.rpmorig` 或 `.rpmsave`（备份文件）保存更改的文件并安装新包中的版本，但前提是原先安装的文件和较新的版本不同。如果是这种情况，则将备份文件（`.rpmorig` 或 `.rpmsave`）与新安装的文件进行比较，并在新文件中再次进行更改。随后，确保删除所有 `.rpmorig` 和 `.rpmsave` 文件以避免以后的更新出现问题。
- 如果配置文件已存在并且 `.spec` 文件中指定了 `noreplace` 标签，则出现 `.rpmnew` 文件。

更新后，在使用 `.rpmsave` 和 `.rpmnew` 文件进行比较后应将它们去除，从而防止它们阻碍以后的更新。如果 RPM 数据库以前未能识别文件，则将为其指派扩展名 `.rpmorig`。

否则，将使用 `.rpmsave`。换句话说，`.rpmorig` 是从异系统格式更新为 RPM 的结果。而 `.rpmsave` 是从较早的 RPM 更新为较新的 RPM 的结果。`.rpmnew` 不提供任何有关系统管理员是否对配置文件进行了任何更改的信息。`/var/adm/rpmconfigcheck` 中提供这些文件的列表。不覆盖某些配置文件（如 `/etc/httpd/httpd.conf`）以允许继续进行操作。

-U 开关不仅仅是使用 -e 选项进行卸载并使用 -i 选项进行安装的等效项。只要可能，就可以使用 -U。

要去掉包，请输入 `rpm -e <package>`。rpm 只有在没有未解决的依赖性的情况下才会删除该包。例如，只要有其他程序需要 Tcl/Tk，理论上就不能删除它。即使是在这种情况下，RPM 也会向数据库寻求帮助。如果出于任何原因或在任何特殊情况下不能进行这一删除操作（即使不存在任何其他依赖性），则最好使用 `--rebuilddb` 选项重建 RPM 数据库。

### 4.3.3 RPM 和增补程序

为了确保系统的操作安全性，必须时常在系统中安装更新包。以前，包中的错误只能通过替换整个包来消除。这样，对只有小文件中存在错误的较大的包进行替换时就很容易产生大量数据。不过 SUSE RPM 提供了一项功能，支持在包中安装增补程序。

以下使用 pine 的示例中对最重要的考虑事项进行了说明：

增补程序 RPM 是否适合我的系统？要对此进行检查，请先查询包的已安装版本。对于 pine，可以通过以下命令完成：

```
rpm -q pine pine-4.44-188
```

然后检查增补程序 RPM 是否适合此版本的 pine：

```
rpm -qp --basedon pine-4.44-224.i586.patch.rpm
pine = 4.44-188
pine = 4.44-195
pine = 4.44-207
```

此增补程序适用于 pine 的三个不同的版本。还列出示例中已安装的版本，因此可以安装增补程序。

增补程序将替换哪些文件？在增补程序 RPM 中可以方便地找到受增补程序影响的文件。rpm 参数 -P 允许选择特殊的增补程序功能。使用以下命令显示文件列表：

```
rpm -qpPl pine-4.44-224.i586.patch.rpm
/etc/pine.conf
/etc/pine.conf.fixed
/usr/bin/pine
```

或者，如果已安装增补程序，则使用以下命令：

```
rpm -qPl pine
/etc/pine.conf
/etc/pine.conf.fixed
/usr/bin/pine
```

如何在系统中安装增补程序 RPM？ 增补程序 RPM 的使用与普通 RPM 相同。唯一的区别就是必须已安装合适的 RPM。

系统中已安装了哪些增补程序，用于哪些包版本？

使用命令 `rpm -qPa` 可以显示系统中已安装的所有增补程序的列表。如果新系统中只安装了一个增补程序（如在本例中），则列表如下所示：

```
rpm -qPa pine-4.44-224
```

如果以后要了解最初安装了哪个包版本，则可以在 RPM 数据库中获得此信息。对于 pine，可以通过以下命令显示此信息：

```
rpm -q --basedon pine pine = 4.44-188
```

`rpm` 和 `rpmbuild` 的手册页中提供了详细信息（包括有关 RPM 的增补程序功能的信息）。

### 4.3.4 增量 RPM 包

“增量 RPM”包中包含新旧版本的 RPM 包之间的差异（即“增量”）。将增量 RPM 应用到旧 RPM 将生成全新的 RPM，但您不必保存旧 RPM 的副本，因为增量 RPM 同样适用于已安装的 RPM。deltarpm 包甚至比增补程序 RPM 还小；在必须通过因特网传送更新包时，这是一个优点。缺点是，涉及增量 RPM 的更新操作与使用纯粹 RPM 或增补程序 RPM 进行更新的情况相比，占用的 CPU 周期要长得多。要令 YaST 在 YOU 会话中使用增量 RPM 包，请在 `/etc/sysconfig/onlineupdate` 中将 `YOU.USE_DELTAS` 设置为“yes”。

`prepdeltarpm`、`writedeltarpm` 和 `applydeltarpm` 二进制文件是 `deltarpm` 套件的一部分，可帮助您创建和应用增量 RPM 包。使用以下命令可以创建一个名为 `new.delta.rpm` 的增量 RPM（假设 `old.rpm` 和 `new.rpm` 同时存在）：

```
prepdeltarpm -s seq -i info old.rpm >
old.cpio
prepdeltarpm -f new.rpm > new.cpio

xdelta delta -0 old.cpio new.cpio delta

writedeltarpm new.rpm delta info new.delta.rpm
rm old.cpio new.cpio delta
```

使用 `applydeltarpm` 可以重新构建新的 RPM；可以从文件系统构建（如果已安装旧软件包）：

```
applydeltarpm new.delta.rpm new.rpm
```

或者使用 `-r` 选项构建（如果要从旧 RPM 生成新 RPM 而不访问文件系统）：

```
applydeltarpm -r old.rpm new.delta.rpm new.rpm
```

有关技术细节，请参见 `file:///usr/share/doc/packages/deltarpm/README`。

### 4.3.5 RPM 查询

使用 `-q` 选项，`rpm` 将初始化查询，使查看 RPM 档案（通过添加选项 `-p`）并查询已安装包的 RPM 数据库成为可能。可以使用多个开关指定所需信息的类型。请参见表 4.8 “最重要的 RPM 查询选项”。

Table 4.8: 最重要的 RPM 查询选项

<code>-i</code>	包信息
<code>-l</code>	文件列表
<code>-f FILE</code>	查询包含文件 <code>&lt;FILE&gt;</code> 的包（必须使用 <code>&lt;FILE&gt;</code> 指定完整路径）
<code>-s</code>	带有状态信息的文件列表（间接指定 <code>-l</code> ）
<code>-d</code>	仅列出文档文件（间接指定 <code>-l</code> ）
<code>-c</code>	仅列出配置文件（间接指定 <code>-l</code> ）
<code>--dump</code>	带有完整详细信息的文件列表（将用于 <code>-l</code> 、 <code>-c</code> 或 <code>-d</code> ）
<code>--provides</code>	列出包中可被另一个包通过 <code>--requires</code> 请求的功能
<code>--requires, -R</code>	包需要的功能
<code>--scripts</code>	安装脚本（预安装、后安装、卸载）

---

例如，命令 `rpm -q -i wget` 显示例4.2 “`rpm -q -i wget`” 中所示的信息。

#### 4.2: rpm -q -i wget

```
Name           : wget                               Relocations: (not relocatable)
Version        : 1.9.1                             Vendor: SUSE LINUX AG, Nuernberg, Germany
Release       : 50                                 Build Date: Sat 02 Oct 2004 03:49:13 AM CEST
Install date: Mon 11 Oct 2004 10:24:56 AM CEST    Build Host: f53.suse.de
Group         : Productivity/Networking/Web/Utilities Source RPM: wget-1.9.1-50.src.rpm
Size          : 1637514                             License: GPL
Signature     : DSA/SHA1, Sat 02 Oct 2004 03:59:56 AM CEST, Key ID a84edae89c800aca
Packager      : http://www.suse.de/feedback
URL           : http://wget.sunsite.dk/
Summary       : A tool for mirroring FTP and HTTP servers
Description   :
Wget enables you to retrieve WWW documents or FTP files from a server.
This can be done in script files or via the command line.
[...]
```

只有当您指定带有完整路径的完整文件名时，选项 `-f` 才起作用。根据需要提供任意多个文件名。例如，以下命令

```
rpm -q -f /bin/rpm /usr/bin/wget
```

产生：

```
rpm-4.1.1-191
wget-1.9.1-50
```

只要知道部分文件名，就可以使用 shell 脚本，如例4.3 “搜索包的脚本” 所示。当运行所显示的脚本时，将部分文件名以参数的形式传递给该脚本。

#### 4.3: 搜索包的脚本

```
#!/bin/sh
for i in $(rpm -q -a -l | grep $1); do
    echo "\"$i\" is in package:"
    rpm -q -f $i
    echo ""
done
```

命令 `rpm -q --changelog rpm` 显示有关特定包信息（更新、配置、修改等）的详细列表。此示例显示有关包 `rpm` 的信息。但是，只列出 RPM 数据库中的最后 5 个更改项。包本身中包括所有项（日期向后推两年）。只有当在 `/media/cdrom` 装入了 CD1 时，此查询才起作用：

```
rpm -qp --changelog /media/cdrom/suse/i586/rpm-4*.rpm
```

借助已安装的 RPM 数据库，可以进行校验检查。使用 `-V`、`-y` 或 `--verify` 对其进行初始化。使用此选项，`rpm` 显示安装后已被更改的包中的所有文件。`rpm` 使用 8 个字符符号给出有关以下更改的一些提示：

*Table 4.9: RPM 校验选项*

5	MD5 校验和
S	文件大小
L	符号链接
T	修改时间
D	主要和次要设备编号
U	拥有者
G	组
M	方式（权限和文件类型）

对于配置文件，将输出字母 `c`。例如，对于 `/etc/wgetrc` (`wget`) 的更改：

```
rpm -V wget
S.5....T c /etc/wgetrc
```

RPM 数据库的文件被放置在 `/var/lib/rpm` 中。如果分区 `/usr` 的大小为 1 GB，则此数据库可能会占用将近 30 MB，特别是在完全更新之后。如果数据库比预期大得多，则最好使用选项 `--rebuilddb` 重建数据库。在执行此操作之前，制作旧数据库的备份。`cron` 脚本 `cron.daily` 每天制作数据库的副本（用 `gzip` 打包）并将这些副本储存在 `/var/adm/backup/rpmdb` 中。副本的数目是由 `/etc/sysconfig/backup` 中的变量 `MAX_RPMDDB_BACKUPS`（默认值为 5）控制的。对于 1 GB 的 `/usr`，单个备份的大小大约为 3 MB。

### 4.3.6 安装和编译源包

SUSE LINUX 的所有源包都带有 `.src.rpm` 扩展名（源 RPM）。

**Tip**

就像任何其他包一样，可以使用 YaST 安装源包。但是，在包管理器中，它们不会被标记为已安装 ([i])。这是因为不在 RPM 数据库中输入源包。当安装源包时，只将源代码添加到系统中。必须编译软件本身。只有已安装的操作系统软件列在 RPM 数据库中。

**Tip**

以下目录必须可用于 `/usr/src/packages` 中的 `rpm` 和 `rpmbuild`（除非在诸如 `/etc/rpmrc` 这样的文件中指定自定义设置）：

**SOURCES** 代表原始源（`.tar.bz2` 或 `.tar.gz` 文件等）和特定于发布版本的调整（多为 `.diff` 或 `.patch` 文件）

**SPECS** 代表 `.spec` 文件，类似于元 Makefile，该文件控制构建进程

**BUILD** 在此目录中解包、增补和编译所有源

**RPMS** 储存完成的二进制包的位置

**SRPMS** 这里是源 RPM

当您使用 YaST 安装源包时，将在 `/usr/src/packages` 中安装所有需要的部件：源和调整在 **SOURCES**，相关的 `.spec` 文件在 **SPECS** 中。

**Warning**

不要对系统部件（`glibc`、`rpm`、`sysvinit` 等）进行试验，因为这将会影响系统的可操作性。

**Warning**

下面的示例使用 `wget.src.rpm` 包。在使用 YaST 安装此包之后，您应该具有与以下所列相似的文件：

```
/usr/src/packages/SOURCES/nops_doc.diff
/usr/src/packages/SOURCES/toplev_destdir.diff
/usr/src/packages/SOURCES/wget-1.9.1+ipvmisc.patch
/usr/src/packages/SOURCES/wget-1.9.1-brokentime.patch
/usr/src/packages/SOURCES/wget-1.9.1-passive_ftp.diff
/usr/src/packages/SOURCES/wget-LFS-20040909.tar.bz2
/usr/src/packages/SOURCES/wget-wrong_charset.patch
/usr/src/packages/SPECS/wget.spec
```

`rpmbuild -b <X> /usr/src/packages/SPECS/wget.spec` 启动编译。`<X>` 是通配符，代表构建进程的不同阶段（有关详细信息，请参见 `--help` 的输出或 RPM 文档）。以下内容只是简要说明：

**-bp** 在 `/usr/src/packages/BUILD` 中准备源：解包和增补。

**-bc** 执行与 `-bp` 相同的操作，但还进行编译。

**-bi** 执行与 `-bp` 相同的操作，但还安装生成的软件。警告：如果包不支持 BuildRoot 功能，则可能会覆盖配置文件。

**-bb** 执行与 `-bi` 相同的操作，但还创建二进制包。如果编译成功，二进制包应该在 `/usr/src/packages/RPMS` 中。

**-ba** 执行与 `-bb` 相同的操作，但还创建源 RPM。如果编译成功，二进制包应该在 `/usr/src/packages/SRPMS` 中。

**--short-circuit** 跳过某些步骤。

现在可以使用 `rpm -i` 或最好使用 `rpm -U` 来安装创建的二进制 RPM。使用 `rpm` 进行安装使它显示在 RPM 数据库中。

### 4.3.7 使用 build 编译 RPM 包

许多包存在的风险是构建进程中会将许多不需要的文件添加到正在运行的系统中。为防止发生这种情况，请使用 `build`，它将创建构建包的确定环境。要建立这一 `chroot` 环境，`build` 脚本必须和完整的包树结构一起提供。可以通过 NFS 在硬盘上或者从 DVD 提供这一树结构。使用 `build --rpms <directory>` 可指定相应的位置。与 `rpm` 不同，`build` 命令在源目录中查找 SPEC 文件。要用系统中 `/media/dvd` 下装入的 DVD 重新构建 `wget`（就像在上面的示例中），请以 `root` 用户的身份使用以下命令：

```
cd /usr/src/packages/SOURCES/  
mv ../SPECS/wget.spec .  
build --rpms /media/dvd/suse/ wget.spec
```

随后，将在 `/var/tmp/build-root` 建立一个最小的环境。在此环境中构建包。完成后，生成的包位于 `/var/tmp/build-root/usr/src/packages/RPMS` 中。

`build` 脚本提供多个附加选项。例如，使脚本优先选择您自己的 RPM、忽略构建环境的初始化或者将 `rpm` 命令限制在上述阶段之一。使用 `build --help` 并通过阅读 `build` 手册页来访问更多信息。



### 4.3.8 用于 RPM 档案和 RPM 数据库的工具

Midnight Commander (mc) 可以显示 RPM 档案的内容并复制部分内容。它将档案表示为虚拟文件系统，提供 Midnight Commander 所有常用的菜单选项。使用 **(F3)** 键显示 HEADER。使用光标键和 **(Enter)** 键查看档案结构。使用 **(F5)** 键复制部分档案。

KDE 提供 kpackage 工具，作为 rpm 的前端。还提供作为 YaST 模块的功能齐全的包管理器（请参见第2.3.1节“安装和去除软件”）。



# 系统修复

除了提供大量的 YaST 模块用于系统安装和配置之外，SUSE LINUX 还提供了修复已安装系统的功能。本章介绍多种类型的系统修复及其步骤。通过 SUSE 应急系统可以访问分区。有经验的系统管理员可以通过应急系统来修复受损系统。

5.1	自动修复 .....	122
5.2	用户定义的修复 .....	123
5.3	专家工具 .....	124
5.4	SUSE 应急系统 .....	124

由于不能假定损坏的系统可以通过自身进行引导并且不能方便地修复正在运行的系统，所以应该像全新安装时那样引导以修复系统。按照第1章使用 YaST 进行安装中介绍的步骤进入提供多个安装选项的对话框，然后选择‘修复已安装系统’。

### Important

使用适当的安装媒体

要使修复系统正常运行，用于引导系统的安装媒体应该与已安装系统所用的安装媒体完全一致。

### Important

在下一步中，选择应该如何执行系统修复。可以使用自动修复、自定义修复和专家工具，本章对这些工具进行了介绍。

## 5.1 自动修复

此方法适合修复由于未知原因而损坏的系统。选择它将对已安装系统进行广泛的分析，因为要进行大量测试和检查，所以将花相当长时间。屏幕的底部使用两个进度条显示此过程的进度。上面的进度条显示当前正在运行的测试的进度。下面的进度条显示分析进程的总体进度。上面的日志窗口用于跟踪当前正在运行的测试及测试结果。请参见图5.1“自动修复方式”。每次运行都会执行以下主要测试。这些测试又包含许多单独的子测试。

**所有硬盘的分区表** 检查所有检测到的硬盘的分区表的有效性和一致性。

**交换分区** 检测并测试已安装系统的交换分区，并在合适的情况下建议激活交换分区。应该接受这一建议以实现更高的系统修复速度。

**文件系统** 所有检测到的文件系统都需要进行文件系统特定的检查。

**文件 `/etc/fstab` 中的项** 检查文件中项的完整性和一致性。将装入所有有效的分区。

**引导装载程序配置** 检查已安装系统（GRUB 或 LILO）的引导装载程序配置的完整性和一致性。检查引导和引导设备，并检查 `initrd` 模块的可用性。

**包数据库** 这将检查执行最小安装的操作所需的所有包是否存在。虽然还可以分析基础包，但因为基础包数量太大，将花费很长时间。

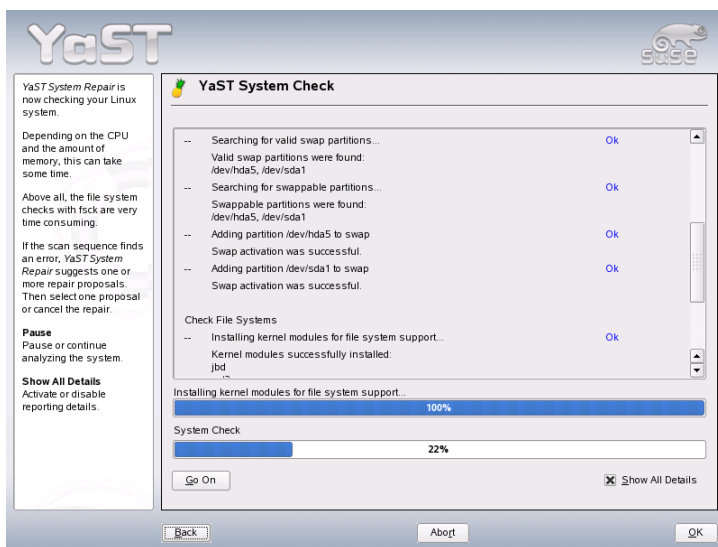


Figure 5.1: 自动修复方式

当出现错误时，过程将停止并打开一个对话框，对话框中提供详细信息和可能的解决方案。这里不可能介绍所有这些情况。仔细阅读屏幕上的讯息并从列表选项中选择所需的操作。如果拿不准，也可以拒绝执行建议的修复操作。如果是这样，系统将保持不变，在未提示用户的情况下不会自动执行任何修复。

## 5.2 用户定义的修复

上面一节中介绍的自动修复执行所有测试。这在系统损坏情况未知时很有用。但是，如果您已经知道系统的哪一部分受到影响，则可以缩小所应用测试的范围。选择‘用户定义的修复’将显示一组测试，这些测试最初都被标记为准备执行。这些测试的总范围和自动修复的测试范围一致。如果您清楚哪些方面没有损坏，则取消对应测试的标记。单击‘继续’将启动一个范围相对较小的测试过程，可能将显著缩短运行时间。

并不是所有的测试组都单独适用。fstab 项的分析总是和文件系统（包括现有的交换分区）的检查一起执行。YaST 将通过选择最小数目的必需测试自动满足这种依赖性。

## 5.3 专家工具

如果您熟悉 SUSE LINUX 并已经非常清楚需要对系统的哪些方面进行修复，则可以选择‘专家工具’直接应用进行修复所需的工具。

**安装新的引导装载程序** 这将启动 YaST 引导装载程序配置模块。有关详细信息，请参见 第8.4节 “使用 YaST 配置引导装载程序”。

**运行分区工具** 这将启动 YaST 中的专家分区工具。有关详细信息，请参见 第2.8.5节 “分区”。

**修复文件系统** 这将检查已安装系统的文件系统。首先将向您提供所有检测到的分区的选择，您可以在其中选择要检查的分区。

**恢复丢失的分区** 可以尝试重新构建损坏的分区表。首先将显示检测到的硬盘的列表以供选择。单击‘确定’开始检查。这可能要花一段时间，具体取决于处理能力和硬盘的大小。

### Important

#### 重新构建分区表

重新构建分区表相当复杂。YaST 尝试通过分析硬盘的数据扇区识别丢失的分区。在识别出来之后，会添加丢失的分区以重新构建分区表。但是，此操作不能保证在所有可能的情况下都成功。

### Important

**将系统设置保存到磁盘** 此选项将重要的系统文件保存到软盘上。如果这些文件中的某个文件被损坏，可以从磁盘恢复该文件。

**检查安装的软件** 这将检查包数据库的一致性和最重要包的可用性。使用此工具可以重安装任何损坏的已安装包。

## 5.4 SUSE 应急系统

SUSE LINUX 包含的应急系统可以在发生紧急情况时从外部访问 Linux 分区。可以从 CD、网络或 SUSE FTP 服务器装载应急系统。应急系统包括多个帮助程序。使用这些程序，您可以解决不可访问的硬盘、配置错误的配置文件等大问题或其他类似的问题。

应急系统的另一个部件是 Parted，它可用于调整分区的大小。如果不想使用 YaST 中集成的调整大小的程序，则可以在应急系统中启动此程序。有关 Parted 的信息，请参见 <http://www.gnu.org/software/parted/>。

### 5.4.1 启动应急系统

像在安装时那样引导您的系统。从引导菜单中选择‘应急系统’。应急系统随后进行解压缩，作为新的根文件系统装载到一个 RAM 磁盘上，然后装入并启动。

### 5.4.2 使用应急系统

在 **(Alt)-(F1)** 到 **(Alt)-(F3)** 下，应急系统提供三个虚拟控制台。以 root 用户身份，无需口令就可以登录这些控制台。按 **(Alt)-(F10)** 进入显示内核和 syslog 信息的系统控制台。

可以在 `/bin` 目录下找到 shell 和许多其他有用的实用程序，如 `mount` 程序。`sbin` 目录包含重要的用于查看和修复文件的文件和网络实用程序，包括 `reiserfsck` 和 `e2fsck`。此目录还包含用于系统维护的最重要的二进制文件，如 `fdisk`、`mkfs`、`mkswap`、`mount`、`mount`、`init` 和 `shutdown`，以及用于维护网络的 `ifconfig`、`route` 和 `netstat`。目录 `/usr/bin` 包含 `vi` 编辑器、`grep`、`find`、`less` 和 `telnet`。

#### 访问您的正常系统

要使用应急系统装入 SUSE LINUX 系统，请使用装入点 `/mnt`。您还可以使用或创建另一个目录。下面的示例说明了某个系统的过程，`/etc/fstab` 的详细信息显示在例 5.1 “示例 `/etc/fstab`” 中。

5.1: 示例 `/etc/fstab`

```
/dev/sdb5    swap    swap    defaults    0    0
/dev/sdb3    /        ext2    defaults    1    1
/dev/sdb6    /usr    ext2    defaults    1    2
```

## Warning

注意下面一节中介绍的装入不同设备的步骤的顺序。

## Warning

要访问整个系统，使用下面的命令将其逐步装入 /mnt 目录中：

```
mount /dev/sdb3 /mnt
mount /dev/sdb6 /mnt/usr
```

现在，访问整个系统并更正配置文件（例如 /etc/fstab、/etc/passwd 和 /etc/inittab）中的错误。配置文件现在位于 /mnt/etc 目录而不是 /etc 中。在使用 fdisk 程序通过简单地设置丢失的分区对它们进行恢复之前，生成 /etc/fstab 和 fdisk -l 输出的打印件。

### 修复文件系统

对于应急系统，修复损坏的文件系统是比较复杂的问题。通常，不能在正在运行的系统上修复文件系统。如果遇到严重问题，您甚至都无法装入根文件系统，系统引导可能以显示 kernel panic 结束。在这种情况下，唯一的方法是使用应急系统从外部修复系统。

SUSE LINUX 应急系统包含实用程序 reiserfsck、e2fsck 和 dumpe2fs（用于诊断）。这些实用程序可以解决大多数问题。在紧急情况下，通常无法获得手册页。出于此原因，这些实用程序的说明包含在此手册的第B节“reiserfsck的手册页”和第B节“e2fsck手册页”中。

如果由于无效的超级块导致装入 ext2 文件系统失败，则 e2fsck 程序可能也会失败。如果是这种情况，您的超级块可能也已损坏。每 8192 个块（8193、16385 等）中就有超级块的副本。如果您的超级块损坏，可以尝试改用这些副本中的一个。可以通过输入命令 e2fsck -f -b 8193 /dev/damaged\_partition 完成此操作。-f 选项强制文件系统检查并覆盖 e2fsck 的错误，因为超级块副本未经改动，所以上操作可以修复所有问题。



**Part II**

系 統



# 64 位系统环境中的 32 位和 64 位应用程序

SUSE LINUX 可用于多种 64 位平台。这并不意味着所包括的所有应用程序已被植入 64 位平台中。SUSE LINUX 支持在 64 位系统环境中使用 32 位应用程序。本章简单介绍了如何在 64 位 SUSE LINUX 平台上实现这种支持。它解释了如何执行 32 位应用程序（运行时支持）以及应该如何编译 32 位应用程序以使它们既可以在 32 位系统环境中运行，又可以在 64 位系统环境中运行。另外，您还可以了解有关内核 API 的信息和 32 位应用程序如何在 64 位内核下运行的解释。

6.1	运行时支持 . . . . .	130
6.2	软件开发 . . . . .	130
6.3	Biarch 平台上的软件编译 . . . . .	131
6.4	内核规范 . . . . .	132

用于 64 位平台 AMD64 和 EM64T 的 SUSE LINUX 所进行的设计可以让现有的 32 位应用程序无需进行额外设置便可以在 64 位环境下运行。这种支持意味着您可以继续使用所需的 32 位应用程序，而无需等待对应的 64 位端口可用。

## 6.1 运行时支持

### Important

应用程序版本之间的冲突

如果某个应用程序在 32 位和 64 位环境中都可用，则两个版本的并行安装必定会导致出现问题。在这种情况下，在两个版本中选一个，然后安装并使用这一版本。

### Important

若要正确执行，每个应用程序都需要一系列库。不巧的是，这些库的 32 位和 64 位版本的名称是相同的。必须通过另一种方法对它们加以区分。

为了保持与 32 位版本的兼容性，这些库在系统中的储存位置与在 32 位环境中相同。在 32 位和 64 位环境中，`libc.so.6` 的 32 位版本都位于 `/lib/libc.so.6` 下。

所有 64 位库和对象文件都位于名为 `lib64` 的目录中。通常可以在 `/lib`、`/usr/lib` 和 `/usr/X11R6/lib` 下找到的 64 位对象文件现在可以在 `/lib64`、`/usr/lib64` 和 `/usr/X11R6/lib64` 下找到。这意味着 `/lib`、`/usr/lib` 和 `/usr/X11R6/lib` 下有储存 32 位库的空间，因此两个版本的文件名都可以保持不变。

如果对象目录的数据内容不取决于词大小，则不移动该对象目录的任何子目录。例如，X11 字体仍位于 `/usr/X11R6/lib/X11/fonts` 下的常规位置。此方案符合 LSB（Linux 标准库）和 FHS（文件系统层次标准）。

## 6.2 软件开发

Biarch 开发工具链允许生成 32 位和 64 位对象。默认为编译 64 位对象。通过使用特殊标志也可以生成 32 位对象。对于 GCC，此特殊标志是 `-m32`。

必须以一种独立于体系结构的形式编写所有标题文件。安装的 32 位和 64 位库必须具有与所安装的标题文件匹配的 API（应用程序编程接口）。普通 SUSE 环境是根据此原则设计的。如果是手工更新的库，请自行解决此问题。

## 6.3 Biarch 平台上的软件编译

若要在 Biarch 体系结构上为其他体系结构开发二进制代码，则必须另外安装用于第二个体系结构的各个库。这些包被称为 `rpmname-32bit`。您还需要 `rpmname-devel` 包中各自的标题和库以及 `rpmname-devel-32bit`。

大多数开放源代码程序使用基于 `autoconf` 的程序配置。若要使用 `autoconf` 配置第二个体系结构的程序，请通过运行带有附加环境变量的 `configure` 脚本覆盖 `autoconf` 的常规编译器和链接器设置。

以下示例涉及 `x86` 为第二个体系结构的 AMD64 或 EM64T 系统：

1. 设置 `autoconf` 使用 32 位编译器：

```
CC="gcc -m32"
```

2. 指示链接器处理 32 位对象：

```
LD="ld -m elf64_i386"
```

3. 设置组装机生成 32 位对象：

```
AS="gcc -c -m32"
```

4. 确定 `libtool` 等的库是否来自 `/usr/lib`：

```
LDFLAGS="-L/usr/lib"
```

5. 确定库是否储存在 `lib` 子目录中：

```
--libdir=/usr/lib
```

6. 确定是否使用了 32 位 X 库：

```
--x-libraries=/usr/X11R6/lib/
```

并不是每个程序都需要这些变量。根据各个程序对这些变量进行调整。

```
CC="gcc -m64" \
LDFLAGS="-L/usr/lib64;" \
    .configure \
    --prefix=/usr \
    --libdir=/usr/lib64
```

```
make
```

```
make install
```

## 6.4 内核规范

AMD64 和 EM64T 的 64 位内核提供 64 位和 32 位内核 ABI（应用程序二进制接口）。后者与对应的 32 位内核的 ABI 相同。这意味着 32 位应用程序可以以与 32 位内核交流的相同方式与 64 位内核进行交流。

64 位内核系统调用的 32 位仿真不支持系统程序使用的某些 API。这取决于平台。出于此原因，少数应用程序（例如 `lspci` 或 LVM 管理程序）被编译为 64 位程序才能正常运行。

64 位内核只能装载专门为此内核编译的 64 位内核模块。不能使用 32 位内核模块。

### Tip

某些应用程序需要单独的内核可装载模块。如果要在 64 位系统环境中使用这种 32 位应用程序，请与此应用程序和 SUSE 的提供商联系以确保该内核可装载模块的 64 位版本和内核 API 的 32 位编译版本可用于此模块。

Tip

# 引导和配置 Linux 系统

引导 Linux 系统是一个复杂的过程。其中涉及许多不同的组件，而且它们之间需要无错交互。本章简要解释基本原理和所涉及的组件。本章还将讨论运行级别的概念和通过 `sysconfig` 执行的 SUSE 系统配置。

7.1	Linux 引导进程	134
7.2	init 程序	136
7.3	运行级别	137
7.4	更改运行级别	138
7.5	Init 脚本	139
7.6	Systems Services (RunLevel)	142
7.7	SUSEconfig 和 /etc/sysconfig	143
7.8	YaST sysconfig 编辑器	145

## 7.1 Linux 引导进程

Linux 引导进程包括多个阶段，每个阶段由一个组件来代表。下表概要总结了引导进程并介绍了所涉及的所有主要组件。

### 1. BIOS

在打开计算机后，BIOS 将初始化屏幕和键盘并测试主存储器。直到这一阶段，计算机不访问任何大容量储存媒体。随后，将从 CMOS 值（CMOS 设置）装载有关当前日期、时间和最重要的外设的信息。当识别出第一块硬盘及其空间之后，系统控制将从 BIOS 传递到引导装载程序。

### 2. 引导装载程序

第一块硬盘的前 512 个字节的物理数据扇区将被装载到主存储器中，位于此扇区开始位置的引导装载程序将接管系统控制。引导装载程序执行的命令决定了引导进程剩余的部分。因此，第一块硬盘的前 512 个字节被称为主引导记录 (MBR)。引导装载程序随后将控制传递到实际的操作系统（在本例中即 Linux 内核）。有关 Linux 引导装载程序 GRUB 的详细信息，请参见第 8 章 引导装载程序。

### 3. 内核和 initrd

为了传递系统控制，引导装载程序将内核和初始 RAM 磁盘 (initrd) 装载到内存中。Linux 内核包含一个选项，即将一个小文件系统装载到 RAM 磁盘，然后在装入实际根文件系统前运行程序。内核然后解压缩 initrd 并将其作为临时根文件系统装入。initrd 的内容是一个最小的 Linux 系统，其中包含一个称为 linuxrc 的可执行文件。这个可执行文件在装入真实根文件系统之前执行。如果可能，内核将在 linuxrc 成功终止后释放 initrd 占用的内存并启动 init。有关 initrd 的详细信息，请参见第 7.1.1 节 “initrd”。

### 4. linuxrc

这个程序执行装入正确的根文件系统所需的所有操作，如为所需的文件系统提供内核功能以及为大容量储存控制器提供设备驱动程序。一旦成功装入实际根文件系统，linuxrc 就会停止执行，同时内核启动 init 程序。有关 linuxrc 的详细信息，请参考第 7.1.2 节 “linuxrc”。

### 5. init

init 通过提供不同功能的多个不同的级别来处理系统的实际引导。有关 init 的介绍，请参见第 7.2 节 “init 程序”。



### 7.1.1 initrd

initrd 是一个较小的（通常经过压缩的）文件系统，内核可将其装载到 RAM 磁盘并作为临时根文件系统装入。它提供了一个最小的 Linux 环境，可在装入实际根文件系统之前执行程序。这个最小的 Linux 环境由 BIOS 例程装入内存，而且除了需要足够的内存外没有特别的硬件要求。initrd 必须始终提供一个名为 linuxrc 的可执行文件，而且该文件需要无错误退出。

在能够装入实际的根文件系统并启动实际的操作系统之前，内核需要相应的驱动程序来访问根文件系统所在的设备。这些驱动程序可能包括用于特定类型硬盘的特殊驱动程序，甚至还可能包括访问网络文件系统所需的网络驱动程序（请参见）。内核还必须包含读取 initrd 的文件系统所需的代码。linuxrc 可能会装载根文件系统所需的模块。

使用脚本 mkinitrd 创建一个 initrd。在 SUSE LINUX 中，由 /etc/sysconfig/kernel 中的变量 INITRD\_MODULES 指定要装载的模块。安装后，系统会自动将此变量设置为正确的值（安装 linuxrc 保存已装载了哪些模块）。将严格按照这些模块在 INITRD\_MODULES 中出现的顺序来装载它们。这一点在使用多个 SCSI 驱动程序时尤为重要，因为如果不这样，硬盘的名称将更改。严格地讲，仅装载访问根文件系统所需的那些驱动程序就足够了。但是，会通过 initrd 装载安装所需的所有 SCSI 驱动程序，因为稍后装载可能会出现问題。

#### Important

##### 更新 initrd

引导装载程序装载 initrd 的方式与内核相同。在更新 initrd 后无需重装 GRUB，因为 GRUB 会在引导时搜索目录以获得正确的文件。

Important

### 7.1.2 linuxrc

linuxrc 的主要用途是准备真实根文件系统的装入和访问。根据实际系统配置的不同，linuxrc 负责以下任务。

**装载内核模块** 根据硬件配置的不同，可能需要一些特殊的驱动程序来访问计算机的硬件部件（最重要的部件是硬盘）。要访问最终的根文件系统，内核需要装载适当的文件系统驱动程序。

**管理 RAID 和 LVM 设置** 如果将系统配置为在 RAID 或 LVM 下保存根文件系统，则 linuxrc 将设置 LVM 或 RAID 以支持稍后对根文件系统的访问。有

关 RAID 的信息，请参见第3.8节“软 RAID 配置”。有关 LVM 的信息，请参见第3.7节“LVM 配置”。

**管理网络配置** 如果对系统进行配置以使用通过网络装入的根文件系统（通过 NFS 装入），则 linuxrc 必须确保装载了适当的网络驱动器，并确保对其进行设置以允许对根文件系统的访问。

在初始引导期间调用 linuxrc 时（安装进程一部分），要执行的任务将与前面提到的任务不同：

**查找安装媒体** 启动安装进程时，计算机将通过安装媒体中的 YaST 安装程序装载一个安装内核和一个特殊的 initrd。YaST 安装程序在 RAM 文件系统中运行，它需要有关安装媒体实际位置的信息以访问安装媒体并安装操作系统。

**启动硬件识别并装载适当的内核模块** 如第7.1.1节“initrd”中所述，引导进程从可用于大多数硬件配置的一组最小的驱动程序启动。linuxrc 将启动初始硬件扫描进程，以确定适合您的硬件配置的一组驱动程序。这些值稍后将被写入到 /etc/sysconfig/kernel 中的 INITRD\_MODULES 中，以便随后的任何引导进程能够使用自定义 initrd。在安装进程中，linuxrc 将装载这组模块。

**装载安装系统或应急系统** 一旦正确地识别出硬件并装载了适当的驱动程序后，linuxrc 就会启动安装系统，其中包含实际的 YaST 安装程序或应急系统。

**启动 YaST** 最后，linuxrc 将启动 YaST，由后者启动包安装和系统配置。

### 7.1.3 参考信息

有关详细信息，请参见 /usr/src/linux/Documentation/ramdisk.txt、/usr/src/linux/Documentation/initrd.txt 以及 page initrd(4) 和 mkinitrd(8) 的手册页。

## 7.2 init 程序

程序 init 是进程编号为 1 的进程，负责以要求的方式初始化系统。所有其他进程均为 init 的子进程或其子进程之一。init 具有特殊作用。它由内核直接启动，

并且信号 9（该信号通常会停止进程）对它不起作用。所有其他程序由 `init` 直接启动，或由它的其中一个子进程启动。

`init` 在 `/etc/inittab` 文件中进行集中配置。在这里定义运行级别（请参见第 7.3 节“运行级别”）。它还指定了在每个级别有哪些服务和守护程序可用。根据 `/etc/inittab` 中的项，`init` 将运行若干个脚本。为了清楚起见，这些脚本都位于目录 `/etc/init.d` 中。

启动和关闭系统的整个过程是由 `init` 维护的。从这一点来看，可以将内核视为一个后台进程，其任务是维护所有其他进程，以及根据其他程序的请求来调整 CPU 时间和硬件访问。

## 7.3 运行级别

在 Linux 中，运行级别定义了系统如何启动以及正在运行的系统中有哪些服务可用。在引导后，系统会按照 `/etc/inittab` 中的 `initdefault` 行所定义的方式启动。通常是 3 或 5。请参见表 7.1“可用运行级别”。也可以选择引导时指定运行级别（例如，在引导提示符处指定）。任何不直接由内核本身求值的参数均将被传递给 `init`。

要在系统运行时更改运行级别，请输入 `init` 和作为参数的相应数字。只有系统管理员才有权这样做。`init 1`（或 `shutdown now`）会导致系统更改为单用户方式，可用于进行系统维护和管理。在完成操作后，管理员可以切换回正常运行级别，方法是输入 `init 3`，这将启动所有必要的程序，并允许一般用户登录没有 X 的系统并进行操作。要启用图形环境（如 GNOME、KDE）或任何其他窗口管理器，应改用 `init 5`。`init 0` 或 `shutdown -h now` 将导致系统暂停。`init 6` 或 `shutdown -r now` 将导致系统先关闭然后重引导。

### Important

#### 运行级别 2 与通过 NFS 装入的 `/usr` 分区

如果您的系统通过 NFS 装入了 `/usr` 分区，则不应使用运行级别 2。`/usr` 目录含有系统正常工作所需的重要程序。由于 NFS 服务在运行级别 2（没有远程网络的本地多用户方式）中不可用，系统会在许多方面受到严重限制。

### Important

Table 7.1: 可用运行级别

运行级别	说明
0	系统暂停
S	单用户方式；从引导提示符进入，仅限 US 键盘映射
1	单用户方式
2	没有远程网络的本地多用户方式（例如 NFS）
3	有网络的完全多用户方式
4	未使用
5	有网络和 X 显示管理器（KDM（默认）、GDM 或 XDM）的完全多用户方式
6	系统重引导

运行级别 5 是所有 SUSE LINUX 标准安装中的默认运行级别。系统提示用户直接在图形界面下登录。如果默认运行级别是 3，必须按照第 11 章 X 窗口系统中的说明正确配置 X 窗口系统，才能将运行级别切换为 5。完成切换后，请通过输入 `init 5` 来检查系统是否以预期方式运行。如果一切合乎预期，就可以使用 YaST 将默认运行级别设置为 5。

### Warning

#### 修改 `/etc/inittab`

如果 `/etc/inittab` 损坏，则可能无法正常引导系统。因此，在编辑 `/etc/inittab` 时应极其小心，并始终保留一份未修改版本的备份。要修复损坏，可尝试在引导提示符处内核名称后输入 `init=/bin/sh` 以直接引导到 shell。此后，使用命令 `mount -o remount,rw /` 使根文件系统可写，并使用 `cp` 将 `/etc/inittab` 替换为备份版本。为避免出现文件系统错误，应在重引导之前将根文件系统更改为只读：  
`mount -o remount,ro /。`

### Warning

## 7.4 更改运行级别

通常情况下，更改运行级别时会发生两件事情。首先是启动当前运行级别的停止脚本，同时关闭当前运行级别必需的一些程序。然后启动新运行级别的启动脚

本。在大多数情况下，这时会启动多个程序。例如，将运行级别从 3 更改到 5 时会发生以下情况：

1. 通过输入 `init 5`，管理员 (root) 指示 `init` 更改为另一个运行级别。
2. `init` 根据它的配置文件 (`/etc/inittab`) 来确定它是否应该以新运行级别作为参数来启动 `/etc/init.d/rc`。
3. `rc` 现在调用当前运行级别的所有停止脚本，但仅限新运行级别中没有启动脚本的那些停止脚本。在本例中，这些就是位于 `/etc/init.d/rc3.d` (旧的运行级别是 3) 中以 `K` 开头的脚本。`K` 后面的数字指定启动的顺序，因为这时要考虑某些依赖性。
4. 最后要启动的是新运行级别的启动脚本。在本例中，这些是位于 `/etc/init.d/rc5.d` 中以 `S` 开头的脚本。有关它们的启动顺序的相同过程在这里也适用。

当更改为与当前运行级别相同的运行级别时，`init` 仅检查 `/etc/inittab` 的更改，并启动相应的步骤（例如，在另一个界面上启动 `getty` 所需的步骤）。

## 7.5 Init 脚本

`/etc/init.d` 中有两种类型的脚本：

由 `init` 直接执行的脚本 仅在引导进程中或在立即关闭系统（电源故障或用户按下了 `(Ctrl)-(Alt)-(Del)`）时才会发生这种情况。这些脚本的执行是在 `/etc/inittab` 中定义的。

由 `init` 间接执行的脚本 这些脚本在更改运行级别时运行并始终调用主脚本 `/etc/init.d/rc`，后者能够确保相关脚本以正确顺序运行。

所有脚本位于 `/etc/init.d` 中。用于更改运行级别的脚本也在这里，但是它们是通过符号链接从一个子目录 (`/etc/init.d/rc0.d` 到 `/etc/init.d/rc6.d`) 进行调用的。这仅仅是为了清楚起见，并避免在多个运行级别使用时出现重复脚本。因为每个脚本既可以作为启动脚本也可以作为停止脚本来执行，这些脚本必须理解 `start` 和 `stop` 参数。这些脚本还必须理解 `restart`、`reload`、`force-reload` 和 `status` 选项。表 7.2 “可能的 `init` 脚本选项” 对这些不同的选项进行了解释。由 `init` 直接运行的脚本没有这些链接。需要时，可以从运行级别独立运行它们。

Table 7.2: 可能的 *init* 脚本选项

选项	说明
start	启动服务。也可以在不执行任何操作的情况下成功启动运行中的服务。
stop	停止服务。
restart	如果服务正在运行，则首先将其停止，然后重新启动。如果服务未在运行，则启动服务。
reload	在不停止和重新启动服务的情况下重载配置。
force-reload	如果服务支持，则重载配置。否则，要执行的步骤与指定 <code>restart</code> 时相同。
status	显示服务的当前状态。

每个特定于运行级别的子目录中的链接使将脚本与不同的运行级别相关联成为可能。在安装或卸载包时，在程序 `insserv`（或使用 `/usr/lib/lsb/install_initd`，它是调用此程序的一个脚本）的帮助下可添加和去除这些链接。有关详细信息，请参见手册页 `insserv(8)`。下面分别简要介绍最先或最后启动的引导和停止脚本，并对脚本的维护进行了说明。

**boot** 在使用 `init` 直接启动系统时执行。它与选择的运行级别无关，而且仅执行一次。这时将装入 `proc` 和 `pts` 文件系统，并激活 `blogd`（引导日志记录守护程序）。如果在更新或安装后首次引导系统，则会启动初始系统配置。

`blogd` 守护程序是由 `boot` 和 `rc` 启动的第一个服务。它在完成以上脚本触发的操作（例如运行很多子脚本）后停止运行。`blogd` 将任何屏幕输出写入日志文件 `/var/log/boot.msg`，但前提是必须以可读写方式装入 `/var`。否则，`blogd` 将缓冲所有屏幕数据，直到 `/var` 可用。有关 `blogd` 的详细信息，请参见手册页 `blogd(8)`。

脚本 `boot` 还负责启动 `/etc/init.d/boot.d` 中名称以 `s` 开头的脚本。在这里，将检查文件系统并根据需要配置回路设备。同时设置系统时间。如果在自动检查和修复文件系统时出错，系统管理员可以在输入根口令后进行干预。最后执行的是脚本 `boot.local`。

**boot.local** 在这里，输入引导时在更改为某个运行级别之前执行的其他命令。这类似于 DOS 系统上的 `AUTOEXEC.BAT`。

`boot.setup` 在从单用户方式更改为任何其他运行级别时均执行该脚本，它负责许多基本设置，如键盘布局和虚拟控制台的初始化。

`halt` 仅当更改为运行级别 0 或 6 时执行该脚本。它在这里作为 `halt` 或 `reboot` 来执行。是关闭系统还是重引导系统取决于调用 `halt` 的方式。

`rc` 此脚本调用当前运行级别的相应停止脚本和新选择的运行级别的启动脚本。

## 7.5.1 添加 `init` 脚本

您可以创建自己的脚本并方便地将它们集成到上面描述的方案中。有关格式化、命名和组织自定义脚本的说明，请参考 LSB 的规范以及手册页 `init(8)`、`init.d(7)` 和 `insserv(8)`。另请参考手册页 `startproc(8)` 和 `killproc(8)`。

### Warning

#### 创建自己的 `init` 脚本

有问题的 `init` 脚本可能会冻结您的计算机。应认真编辑这些脚本，如果可能，应在多用户环境中对它们进行严格测试。第 7.3 节“运行级别”中提供了有关 `init` 脚本的一些有用信息。

### Warning

要为给定程序或服务创建自定义 `init` 脚本，请使用文件 `/etc/init.d/skeleton` 作为模板。以新名称保存此文件的备份，然后根据需要编辑相关程序和文件名、路径及其他详细信息。您可能还需要用自己的部分来增强此脚本，以便 `init` 过程可以触发正确的操作。

位于顶部的 `INIT INFO` 块是脚本的一个必需部分，应进行编辑。请参见例 7.1“最小的 `INIT INFO` 块”。

#### 7.1: 最小的 `INIT INFO` 块

```
### BEGIN INIT INFO
# Provides:          FOO
# Required-Start:    $syslog $remote_fs
# Required-Stop:     $syslog $remote_fs
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Description:       Start FOO to allow XY and provide YZ
### END INIT INFO
```

在 INFO 块第一行上 Provides: 后, 指定由此 init 脚本控制的程序或服务的名称。在 Required-Start: 和 Required-Stop: 行中, 指定在启动或停止服务本身之前需要启动或停止的所有服务。这些信息稍后用于生成脚本名的编号 (可以在运行级别目录中找到)。在 Default-Start: 和 Default-Stop: 下, 指定应自动启动或停止的服务所在的运行级别。最后, 在 Description: 下, 提供对相关服务的简短说明。

要创建从运行级别目录 (/etc/init.d/rc?.d/) 到 /etc/init.d/ 中相应脚本的链接, 请输入命令 `insserv <new-script-name>`。insserv 程序对 INIT INFO 标题进行求值, 以便为运行级别目录 (/etc/init.d/rc?.d/) 中的启动和停止脚本创建必要的链接。此程序还负责保证每个运行级别的启动和停止顺序正确无误, 方法是在这些链接的名称中包含必要的数字。如果要使用图形工具来创建这样的链接, 请按照第 7.6 节 “Systems Services (RunLevel)” 中说明的方法使用 YaST 提供的运行级别编辑器。

如果应将已存在于 /etc/init.d/ 中的脚本集成到现有运行级别方案中, 请立即通过 insserv 或启用 YaST 的运行级别编辑器中的相应服务在运行级别目录中创建链接。您的更改将在下次重引导时生效 — 新服务将自动启动。

不要手工设置这些链接。如果 INFO 块中出错, 则在稍后为其他服务运行 insserv 时将会出现问题。

## 7.6 Systems Services (RunLevel)

在启动此 YaST 模块后, 它将显示一个概要, 列出所有可用服务和每个服务的当前状态 — 即它们是否已被启用。确定是以‘简单方式’还是以‘专家方式’使用此模块。默认的‘简单方式’足以完成大多数操作。左边的列显示服务的名称, 中间的列指示其当前状态, 而右边的列则给出简短说明。窗口下部提供了对所选服务的更为详细的说明。若要启用某个服务, 请首先在表中选定它, 然后选择‘启用’。同样的步骤可用于禁用服务。

要对所启动或停止的服务所在运行级别进行更具体的控制, 或者更改默认运行级别, 请先选择‘专家方式’。在此方式下, 对话框将在顶部显示当前默认运行级别或 “initdefault” (默认情况下将系统引导到的运行级别)。通常情况下, SUSE LINUX 系统的默认运行级别是 5 (有网络和 X 的完全多用户方式)。运行级别 3 (有网络的完全多用户方式) 是合适的替代选择。

此 YaST 对话框用于选择一个运行级别 (如表 7.1 “可用运行级别” 中所列) 作为新的默认运行级别。此外, 可使用此窗口中的表来启用或禁用各个服务和守护程序。此表列出可用的服务和守护程序, 显示它们当前是否已在您的系统上启用, 如果已启用, 则指示它们用于哪些运行级别。用鼠标选择其中的一行后, 请单击表示运行级别 (‘B’、‘0’、‘1’、‘2’、‘3’、‘5’、‘6’ 和 ‘S’) 的复选框来确定所选



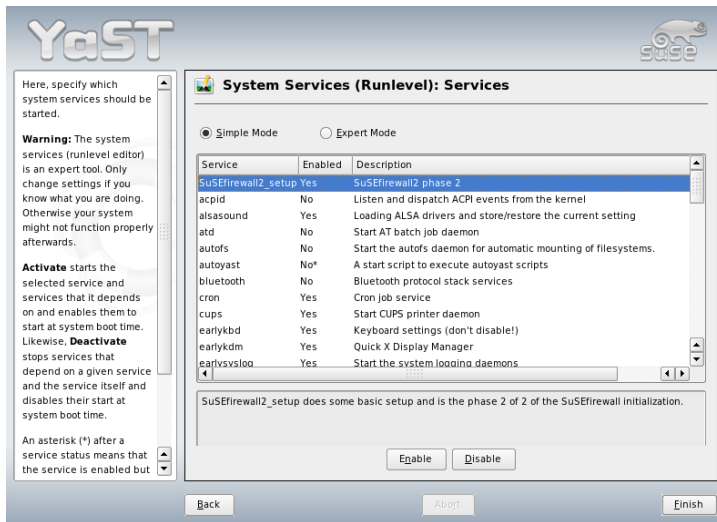


Figure 7.1: Systems Services (RunLevel)

服务或守护程序的运行级别。最初，未对运行级别 4 进行定义，目的是供用户创建自定义运行级别。表概要下方提供了当前所选服务或守护程序的简要说明。

用‘启动、停止或刷新’来确定是否应激活某服务。‘刷新状态’用来检查当前状态。‘设置或重设置’用于选择是将更改应用到系统，还是恢复启动运行级别编辑器之前存在的设置。选择‘完成’即可将已更改的设置保存到磁盘。

### Warning

#### 更改运行级别设置

有问题的运行级别设置可能会导致系统无法使用。在应用您的更改之前，请确保您清楚这些设置可能产生的结果。

Warning

## 7.7 SuSEconfig 和 /etc/sysconfig

SUSE LINUX 的主配置可以通过 /etc/sysconfig 中的配置文件来完成。只有与 /etc/sysconfig 中的各个文件相关的脚本才会读取它们。这样

有很多好处，例如确保了网络设置只需要由与网络相关的脚本来分析。根据 `/etc/sysconfig` 中的设置生成了许多其他系统配置文件。此任务是由 `SuSEconfig` 执行的。例如，如果更改网络配置，`SuSEconfig` 也可能会更改文件 `/etc/host.conf`，因为它是与网络配置相关的文件之一。

如果手工对这些文件进行了某些更改，随后应运行 `SuSEconfig` 来确保在所有相关的位置都进行了全部必要的更改。如果使用 `YaST sysconfig` 编辑器更改了配置，则将自动应用所有更改，因为 `YaST` 会自动启动 `SuSEconfig` 以根据需要更新配置文件。

这一概念使您能够对配置进行基本更改，而无需重引导系统。由于某些更改相当复杂，所以必须重新启动某些程序才能使更改生效。例如，对网络配置的更改可能要求重新启动相关的网络程序。这可以通过输入命令 `rcnetwork stop` 和 `rcnetwork start` 实现。

建议执行以下步骤来更改系统配置：

1. 使用 `init 1` 将系统转入单用户方式（运行级别 1）。
2. 根据需要更改配置文件。要做到这一点，可以使用您选择的编辑器，或使用 `YaST` 的 `sysconfig` 编辑器（请参考第 7.8 节“`YaST sysconfig` 编辑器”）。

---

### Warning

#### 手工更改系统配置

如果不使用 `YaST` 来更改 `/etc/sysconfig` 中的配置文件，则要确保将空变量值用两个引号表示 (`KEYTABLE=""`)，并将含有空白的值用引号括起来。只包括一个单词的值不需要用引号括起来。

---

### Warning

3. 执行 `SuSEconfig` 来确保更改生效。如果已经使用 `YaST` 更改了配置文件，则将自动完成此操作。
4. 使用 `init 3` 等命令将系统切换回先前的运行级别（用先前的运行级别替换 3）。

这一过程主要用于更改整个系统范围的配置，例如网络配置。若要进行较小的更改，不一定要切换到单用户方式，但这样做可以完全确保正确重新启动所有相关的程序。

**Tip****配置自动系统配置**

要禁用 SuSEconfig 进行自动系统配置，请将 `/etc/sysconfig/suseconfig` 中的变量 `ENABLE_SUSECONFIG` 设置为 `no`。如果要使用 SUSE 安装支持，请不要禁用 SuSEconfig。也可以部分禁用自动配置。

**Tip**

## 7.8 YaST sysconfig 编辑器

储存了最重要的 SUSE LINUX 设置的文件位于 `/etc/sysconfig` 目录中。sysconfig 编辑器以易读的方式显示各选项。可以对这些值进行修改，并随后将它们添加到此目录的各个配置文件中。但通常情况下不需要手工编辑它们，因为在安装包或配置服务时会自动调整这些文件。

**Warning****修改 `/etc/sysconfig/*` 文件**

如果没有足够的经验和知识，切勿修改 `/etc/sysconfig` 文件。否则可能会对您的系统造成巨大损害。`/etc/sysconfig` 中的文件包含对每个变量的简短注释，解释了这些变量的实际作用。

**Warning**

YaST sysconfig 对话框分为三个部分。对话框左边的部分显示了一个树视图，其中列出了所有可配置变量。当您选择某个变量时，右边的部分会显示当前选择和此变量的当前设置。在下部的第三个窗口中，简要说明了变量的用途、可能的值、默认值以及作为此变量来源的实际配置文件。此对话框还提供了有关更改变量后将执行哪些配置脚本，以及作为更改的结果将启动哪些新服务等信息。YaST 将提示您确认更改，并通知您在选择“完成”退出对话框后将执行哪些脚本。在这里还可以选择需要现在跳过而在以后启动的服务和脚本。

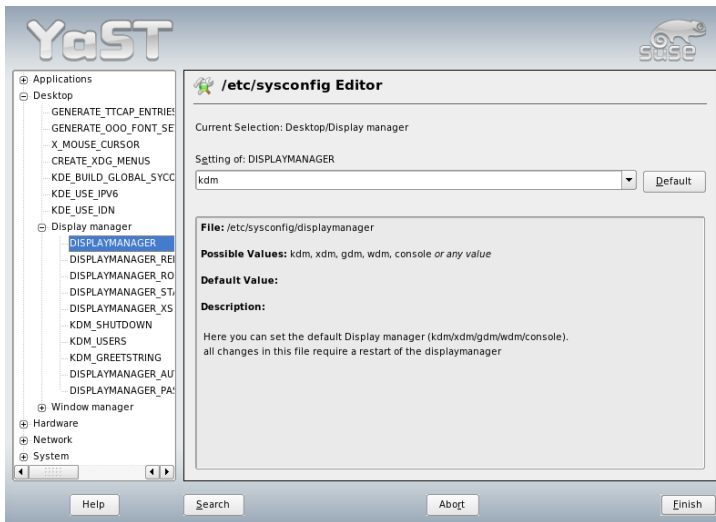


Figure 7.2: 使用 `sysconfig` 编辑器进行系统配置

# 引导装载程序

本章介绍如何配置 GRUB（在 SUSE LINUX 中使用的引导装载程序）。一个特殊的 YaST 模块可用于执行所有设置。如果您不熟悉在 Linux 中进行引导的相关内容，请阅读下面几节获得一些背景信息。本章还介绍了使用 GRUB 进行引导时经常遇到的一些问题和它们的解决方案。

8.1	引导管理	148
8.2	选择引导装载程序	149
8.3	通过 GRUB 引导	149
8.4	使用 YaST 配置引导装载程序	157
8.5	卸载 Linux 引导装载程序	160
8.6	创建引导 CD	160
8.7	图形 SUSE 屏幕	161
8.8	查错	162
8.9	参考信息	163

本章主要介绍引导装载程序 GRUB 的引导管理和配置。第7章 引导和配置 Linux 系统中将引导过程作为一个整体进行了介绍。引导装载程序代表计算机 (BIOS) 和操作系统 (SUSE LINUX) 之间的接口。引导装载程序的配置确定要启动的操作系统及其选项。

本章经常出现以下术语，可能需要进行解释：

**主引导记录** MBR 的结构是由独立于操作系统的约定确定的。前 446 个字节为程序代码保留。它们通常保存有引导装载程序，在本例中为 GRUB。随后的 64 个字节为最多包含 4 项的分区表提供空间（请参见第1.5.4节“分区类型”）。分区表包含有关硬盘分区和文件系统类型的信息。操作系统需要使用此表来处理硬盘。MBR 的最后两个字节包含静态“幻数” (AA55)。包含不同值的 MBR 被 BIOS 和所有计算机操作系统视为是无效的。

**引导扇区** 引导扇区是硬盘分区（除扩展分区之外）上的前几个扇区，扩展分区只充当其他分区的“容器”。引导扇区具有 512 字节的空间，引导扇区储存用于引导安装在各个分区上的操作系统的代码。这适用于经过格式化的 DOS、Windows 和 OS/2 分区的引导扇区，这些扇区还包含文件系统的一些重要的基本数据。相比之下，Linux 分区的引导扇区在设置文件系统之后最初是空的。因此，即使 Linux 分区包含内核和有效的根文件系统，它也不能通过自身进行引导。储存了引导系统的有效代码的引导扇区具有与 MBR 中的最后两个字节 (AA55) 相同的幻数。

## 8.1 引导管理

在最简单的情况中——如果计算机上只安装了一个操作系统——则按上面所介绍的步骤进行引导管理。如果计算机上安装了多个操作系统，则可以使用以下选项：

**从外部媒体引导其他系统** 从硬盘引导其中一个操作系统。其他操作系统是通过安装在外部媒体（软盘、CD、USB 储存媒体）上的引导管理器进行引导的。因为 GRUB 能够引导所有其他操作系统，所以不需要使用任何外部引导装载程序。

**在 MBR 中安装引导管理器** 引导管理器使在计算机上并行安装多个系统以及选择使用这些系统成为可能。用户可以在引导进程中选择要引导的系统。要更改为另一个系统，必须重引导计算机。上述内容的前提是所选的引导管理器必须和所安装的操作系统的兼容。GRUB 是 SUSE LINUX 中使用的引导管理器，它能够引导所有常用的操作系统。默认情况下，SUSE LINUX 在 MBR 中安装所选的引导管理器。

## 8.2 选择引导装载程序

默认情况下，引导装载程序 GRUB 用于 SUSE LINUX 中。但是，在某些情况下以及对于特殊的硬件和软件，使用 LILO 可能更合适。如果您更新较早的 SUSE LINUX 版本（该版本使用 LILO），则将安装 LILO。对于新的安装，将安装 GRUB，除非根分区安装在以下系统上：

- 依赖于 CPU 的 RAID 控制器（例如许多 Promise 或 Highpoint 控制器）
- 软件 RAID
- LVM

有关安装和配置 LILO 的信息，可以在支持数据库中关键字 *LILO* 下获得。

## 8.3 通过 GRUB 引导

GRUB (Grand Unified Bootloader) 由两段组成。stage1 包含 512 个字节，将被写入硬盘分区的 MBR 或引导扇区，或者软盘中。随后，装载 stage2。此段包含实际程序代码。第一段的唯一任务就是装载引导装载程序的第二段。

stage2 可以访问文件系统。当前，支持 Ext2、Ext3、ReiserFS、Minix，以及 Windows 使用的 DOS FAT 文件系统。在某种程度上还支持 BSD 系统使用的 JFS、XFS、UFS 和 FFS。从版本 0.95 开始，GRUB 还能够从包含 ISO 9660 标准文件系统、符合“El Torito”规范的 CD 或 DVD 进行引导。即使是在引导系统之前，GRUB 也可以访问支持的 BIOS 磁盘设备（BIOS 检测到的软盘或硬盘、CD 驱动器和 DVD 驱动器）的文件系统。因此，对 GRUB 配置文件 (*menu.lst*) 进行更改不再要求重安装引导管理器。当引导系统时，GRUB 重装菜单文件以及内核或初始 RAM 磁盘 (*initrd*) 的有效路径和分区数据，并对这些文件进行定位。

GRUB 的实际配置是基于三个文件进行的，下面对这三个文件进行介绍：

**`/boot/grub/menu.lst`** 此文件包含有关可通过 GRUB 进行引导的分区或操作系统的信息。没有这些信息就无法将系统控制传递到操作系统。

**`/boot/grub/device.map`** 此文件将 GRUB 和 BIOS 符号中的设备名转换为 Linux 设备名。

**`/etc/grub.conf`** 此文件包含 GRUB shell 正确安装引导装载程序所需的参数和选项。

可以通过多种方式控制 GRUB。可以在图形菜单（启动屏幕）中选择现有配置的引导项。配置是从文件 `menu.lst` 装载的。

在 GRUB 中，在引导前可以更改所有引导参数。例如，可以通过这种方式更正编辑菜单文件时出现的错误。还可以通过输入提示以交互的方式输入引导命令（请参见第 8.3.1 节“在引导过程中编辑菜单项”）。GRUB 提供在引导前确定内核和 `initrd` 的位置的可能性。通过这种方式，您甚至可以引导在引导装载程序配置中不存在任何项的已安装操作系统。

*GRUB shell* 在已安装系统中提供 GRUB 的仿真。它可用于安装 GRUB 或在应用新设置前对这些设置进行测试。请参见第 8.3.4 节“GRUB Shell”。

### 8.3.1 GRUB 引导菜单

带有引导菜单的图形启动屏幕基于 GRUB 配置文件 `/boot/grub/menu.lst`，该文件包含有关可以通过菜单引导的所有分区或操作系统的所有信息。

每次引导系统时，GRUB 都从文件系统装载菜单文件。出于此原因，不必每次更改文件后都重安装 GRUB。使用 YaST 引导装载程序修改 GRUB 配置，如第 8.4 节“使用 YaST 配置引导装载程序”中所述。

菜单文件中包含命令。语法非常简单。每行都包含一条命令，后跟可选参数，可选参数之间用空格隔开，就像在 shell 中一样。出于历史原因，某些命令允许在第一个参数前使用 `=`。注释以井号 (`#`) 开头。

若要在菜单概述中标识菜单项，请为每项指定一个 `title`。关键字 `title` 后的文本（包括任何空格）显示为菜单中的可选择选项。当选择此菜单项时，将执行下一个 `title` 前的所有命令。

最简单的情况是重定向到其他操作系统的引导装载程序。命令是 `chainloader`，参数通常是 GRUB 中另一个分区的引导块 `block notation`。例如：

```
chainloader (hd0,3)+1
```

GRUB 中的设备名在第 8.3.1 节“硬盘和分区的命名约定”中有所解释。上面的示例指定第一个硬盘第四个分区中的第一个块。

使用命令 `kernel` 指定内核映像。第一个参数是指向分区中内核映像的路径。命令行上的其他参数将被传递到内核。

如果内核不具有访问根分区的内置驱动程序，则必须用单独的 GRUB 命令指定 `initrd`，该命令的唯一参数便是指向 `initrd` 文件的路径。因为 `initrd` 的装载地址会被写入装载的内核映像中，所以 `initrd` 命令必须紧接在 `kernel` 命令之后。



命令 `root` 简化了内核和 `initrd` 文件的指定。`root` 的唯一参数是一个 GRUB 设备或 GRUB 设备上的分区。此设备用于所有内核、`initrd` 或下一个 `root` 命令前未显式指定设备的其他文件路径。此命令不用于安装过程中生成的 `menu.lst` 文件。它仅仅使手工编辑更容易。

每个菜单项的末尾都间接指定 `boot` 命令，因此无需将其写入菜单文件中。但是，如果以交互方式使用 GRUB 进行引导，则必须在最后输入 `boot` 命令。该命令本身没有参数。它只引导装载的内核映像或指定的链装载程序。

在写入所有菜单项之后，将其中一项定义为 `default` 项。否则，将使用第一项（项 0）。您还可以指定在一段时间后引导默认项的超时值（以秒为单位）。`timeout` 和 `default` 通常在各菜单项前面。示例文件在第 8.3.1 节“示例菜单文件”中有所介绍。

### 硬盘和分区的命名约定

GRUB 用于硬盘和分区的命名约定不同于普通 Linux 设备使用的命名约定。在 GRUB 中，分区的编号从 0 开始。因此，`(hd0,0)` 是第一块硬盘的第一个分区。在普通台式机上，作为 Primary Master（第一个 IDE 控制器上的主设备）连接的硬盘所对应的 Linux 设备名为 `/dev/hda1`。

4 个可能的主分区所分配的分区号为 0 到 3。逻辑分区的编号从 4 开始：

```
(hd0,0)  first primary partition of the first hard disk
(hd0,1)  second primary partition
(hd0,2)  third primary partition
(hd0,3)  fourth primary partition (usually an extended partition)
(hd0,4)  first logical partition
(hd0,5)  second logical partition
```

GRUB 不区分 IDE、SCSI 和 RAID 设备。BIOS 或其他控制器识别的所有硬盘将按照 BIOS 中显示的引导顺序进行编号。

不过，GRUB 不能将 Linux 设备名准确映射为 BIOS 设备名。它借助某种算法生成这一映射并将其保存到文件 `device.map` 中，可以根据需要对该文件进行编辑。有关文件 `device.map` 的信息在第 8.3.2 节“文件 `device.map`”中有所介绍。

完整的 GRUB 路径包含写在括号中的设备名和指向指定分区的文件系统中文件的路径。路径以斜线开头。例如，在具有一个 IDE 硬盘（该硬盘的第一个分区中包含 Linux）的系统上，可以按如下方式指定可引导内核：

```
(hd0,0)/boot/vmlinuz
```

## 示例菜单文件

以下示例说明了 GRUB 菜单文件的结构。该示例安装包括 /dev/hda5 下的 Linux 引导分区、/dev/hda7 下的引导分区和 /dev/hda1 下的 Windows 安装。

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8

title linux
    kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd

title windows
    chainloader(hd0,0)+1

title floppy
    chainloader(fd0)+1

title failsafe
    kernel (hd0,4)/vmlinuz.shipped root=/dev/hda7 ide=nodma \
    apm=off acpi=off vga=normal nosmp maxcpus=0 3
    initrd (hd0,4)/initrd.shipped
```

第一块定义了启动屏幕的配置：

**gfxmenu (hd0,4)/message** 背景图像 message 位于 /dev/hda5 中。

**color white/blue black/light-gray** 颜色方案：白色（前景色）、蓝色（背景色）、黑色（所选内容）、浅灰色（所选内容的背景）。颜色方案对启动屏幕没有任何影响，它只影响通过按 **(Esc)** 键退出启动屏幕后所访问的可自定义的 GRUB 菜单。

**default 0** 第一个菜单项 title linux 是默认情况下引导的对象。

**timeout 8** 如果 8 秒钟后无任何用户输入，GRUB 将自动引导默认项。

第二块（也就是最大的块）列出了各个可引导的操作系统。各个操作系统的不同部分由 title 引出。

- 第一项 (`title linux`) 负责引导 SUSE LINUX。内核 (`vmlinux`) 位于第一块硬盘的第一个逻辑分区（引导分区）。内核参数（例如引导分区和 VGA 方式）也被追加在此处。引导分区是根据 Linux 命名约定 (`/dev/hda7/`) 指定的，这是因为此信息将被内核读取而与 GRUB 无关。`initrd` 也位于第一块硬盘的第一个逻辑分区中。
- 第二项负责装载 Windows。Windows 将从第一块硬盘的第一个分区 (`hd0,0`) 引导。命令 `chainloader +1` 将导致 GRUB 读取并执行指定分区的第一个扇区。
- 下一项支持从软盘进行引导，而无需修改 BIOS 设置。
- 引导选项 `failsafe` 用一组内核参数启动 Linux，这些参数使 Linux 甚至可以在有问题的系统上引导。

随时可以根据需要更改菜单文件。GRUB 会在下次引导时使用修改后的设置。使用 YaST 或所选的编辑器对文件进行永久编辑。或者，使用 GRUB 的编辑功能可以以交互方式进行临时更改。请参见第 8.3.1 节“在引导过程中编辑菜单项”。

### 在引导过程中编辑菜单项

在图形 GRUB 引导菜单中，使用箭头键选择要引导的操作系统。如果选择 Linux 系统，则可以在引导提示符处输入其他引导参数。若要直接编辑个别菜单项，请按 `(Esc)` 键退出启动屏幕并按 `(E)` 键。通过这种方式进行的更改仅适用于当前引导过程，不会被永久采用。

## Important

### 引导过程中的键盘布局

US 键盘布局是引导时唯一可用的键盘布局。

## Important

在激活编辑方式后，可以使用箭头键选择要编辑其配置的菜单项。若要使配置可以编辑，请再次按 `(E)` 键。通过这种方式，可以编辑不正确的分区或路径指定，从而防止它们对引导进程产生负面影响。按 `(Enter)` 键退出编辑方式并返回菜单。随后按 `(B)` 键引导此项。可以进行的进一步操作显示在底部的帮助文本中。

若要永久输入更改的引导选项并将它们传递到内核，则以 `root` 用户身份打开文件 `menu.lst` 并将相应的内核参数追加到现有的行上，用空格分隔：

```
title linux
kernel (hd0,0)/vmlinux root=/dev/hda3 additional parameter
initrd (hd0,0)/initrd
```

GRUB 会在下次引导系统时自动采用新参数。或者，还可以通过 YaST 引导装载程序模块进行此更改。将新参数追加到现有的行上，用空格分隔。

### 使用通配符选择引导内核

特别是在开发或使用自定义内核时，您需要更改 `menu.lst` 中的项或编辑命令行来反映当前内核和 `initrd` 文件名。为简化这一过程，请使用通配符动态更新 GRUB 的内核列表。然后，符合特定模式的所有内核映像都被自动添加到可引导映像的列表中。但请注意，系统不对这一功能提供支持。

通过在 `menu.lst` 中输入附加菜单项激活通配符选项。为了可用，所有内核和 `initrd` 映像都必须具有常见基本名称和标识符，该标识符使内核与其关联的 `initrd` 匹配。以下面的设置为例：

```
initrd-default
initrd-test
vmlinuz-default
vmlinuz-test
```

此本例中，可以在一个 GRUB 配置中添加两个引导映像。要获得菜单项 `linux-default` 和 `linux-test`，需要 `menu.lst` 中的以下项：

```
title linux-*
    wildcard (hd0,4)/vmlinuz-*
    kernel (hd0,4)/vmlinuz-* root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd-*
```

在本例中，GRUB 在分区 (hd0,4) 中搜索与通配符匹配的项。这些项用于生成新的 GRUB 菜单项。在上一个示例中，GRUB 所表现出来的行为方式就好像以下项存在于 `menu.lst` 中一样：

```
title linux-default
    wildcard (hd0,4)/vmlinuz-default
    kernel (hd0,4)/vmlinuz-default root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd-default
title linux-test
    wildcard (hd0,4)/vmlinuz-test
    kernel (hd0,4)/vmlinuz-test root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd-test
```

如果没有以一致的方式使用文件名，或如果其中一个扩展文件（例如 `initrd` 映像）丢失，此配置都可能出现問題。

### 8.3.2 文件 device.map

文件 `device.map` 将 GRUB 设备名映射为 Linux 设备名。在包含 IDE 和 SCSI 硬盘的混合系统中，GRUB 必须通过特殊过程尝试确定引导顺序，因为 GRUB 不能访问 BIOS 上有关引导顺序的信息。GRUB 会将此分析的结果保存在文件 `/boot/grub/device.map` 中。对于 BIOS 中引导顺序设置为 IDE 在 SCSI 之前的系统，文件 `device.map` 如下所示：

```
(fd0) /dev/fd0
(hd0) /dev/hda
(hd1) /dev/sda
```

因为 IDE、SCSI 和其他硬盘的顺序取决于不同的因素，并且 Linux 无法标识映射，所以可以在 `device.map` 文件中手工设置顺序。如果在引导时遇到问题，则检查此文件中的顺序是否对应于 BIOS 中的顺序，如果需要，使用 GRUB shell（如第 8.3.4 节“GRUB Shell”中所述）对其进行临时修改。一旦引导了 Linux 系统之后，便可以使用 YaST 引导装载程序模块或所选的编辑器对文件 `device.map` 进行永久编辑。

在手工更改 `device.map` 之后，请执行以下命令重安装 GRUB。此命令导致重装载文件 `device.map` 并且执行 `grub.conf` 中列出的命令：

```
grub --batch < /etc/grub.conf
```

### 8.3.3 文件 /etc/grub.conf

除了 `menu.lst` 和 `device.map` 之外，第三个重要的 GRUB 配置文件就是 `/etc/grub.conf`。此文件包含 `grub` 命令正确安装引导装载程序所需的参数和选项：

```
root (hd0,4)
  install /grub/stage1 d (hd0) /grub/stage2 0x8000 (hd0,4)/grub/menu.lst
quit
```

各个项的含义：

**root (hd0,4)** 此命令指示 GRUB 将以下命令应用到第一块硬盘的第一个逻辑分区（引导文件的位置）。

**install parameter** 运行 grub 命令应该使用参数 `install`。引导装载程序的 `stage1` 应该安装在第一块硬盘的 MBR 中 (`/grub/stage1d (hd0)`)。 `stage2` 应该被装载到内存地址 `0x8000` (`/grub/stage2 0x8000`)。最后一项 (`(hd0,4)/grub/menu.lst`) 指示 GRUB 查找菜单文件的位置。

### 8.3.4 GRUB Shell

GRUB 实际上以两种形式存在：作为引导装载程序，以及作为 `/usr/sbin/grub` 中的普通 Linux 程序。此程序被称为 *GRUB shell*。将 GRUB 作为引导装载程序安装在硬盘或软盘上的功能以 `install` 和 `setup` 命令的形式集成在 GRUB 中。当装载了 Linux 后在 GRUB shell 中可用。

但是，在启动 Linux 之前的引导过程中也可以使用 `setup` 和 `install` 命令。这简化了无法再进行引导的有问题系统的修复工作，因为可以通过手工输入参数规避引导装载程序中有问题的配置文件。在引导过程中手工输入参数还可用于测试新设置而避免损坏本机系统。只要用类似于 `menu.lst` 中的语法输入试验配置文件即可。然后，测试此项的功能，而无需更改现有的配置文件。例如，要测试一个新内核，请输入命令 `kernel` 和指向新内核的路径。如果引导过程失败，则可以在下次引导时继续使用未修改的 `menu.lst`。同样，通过输入正确的参数，不管 `menu.lst` 文件是否有问题，都可以使用命令行界面引导系统。在运行的系统中，可以在 `menu.lst` 中输入正确的参数使系统永远可引导。

只有当将 GRUB shell 作为 Linux 程序运行时（通过输入 `grub`，如第 8.3.2 节“文件 `device.map`”中所述），将 GRUB 设备映射为 Linux 设备名才有意义。出于此目的，程序将读取文件 `device.map`。有关详细信息，请参见第 8.3.2 节“文件 `device.map`”。

### 8.3.5 设置引导口令

即使是在引导操作系统之前，GRUB 也支持对文件系统的访问。没有根权限的用户可以访问 Linux 系统中的文件，而一旦引导系统后，他们将无权访问这些文件。若要阻止这种访问或防止用户引导某些操作系统，可以设置引导口令。

以 `root` 用户身份按如下步骤设置引导口令：

1. 在根提示符处，输入 `grub`。
2. 在 GRUB shell 中对口令进行加密：

```
grub> md5crypt
Password: ****
Encrypted: $1$1S2dv/$JOYcdxIn7CJk9xShzzJVw/
```

3. 将经过加密的字符串粘贴到 menu.lst 文件的全局部分：

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8
password --md5 $1$1S2dv/$JOYcdxIn7CJk9xShzzJVw/
```

现在，只有在按 **P** 键并输入口令后，才可以在引导提示符处执行 GRUB 命令。但是，用户仍可以从引导菜单引导所有操作系统。

4. 要防止从引导菜单引导一个或多个操作系统，请将项 lock 添加到 menu.lst 中不输入口令就不能引导的每个部分。例如：

```
title linux
kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
initrd (hd0,4)/initrd
lock
```

在重引导系统并从引导菜单中选择 Linux 项后，将显示以下错误讯息：

```
Error 32: Must be authenticated
```

按 **Enter** 键进入该菜单。然后按 **P** 键，系统将提示您输入口令。在输入口令并按 **Enter** 键之后，将引导所选的操作系统（在本例中为 Linux）。

### Important

引导口令和启动屏幕

如果对 GRUB 使用引导口令，则不显示通常的启动屏幕。

Important

## 8.4 使用 YaST 配置引导装载程序

在您的 SUSE LINUX 系统中配置引导装载程序最简单的方式就是使用 YaST 模块。在 YaST 控制中心，选择‘系统’→‘引导装载程序配置’。即会显示系统的当前引导装载程序配置，您可以在这里进行任何所需的更改。请参见图 8.1 “使用 YaST 配置引导装载程序”。

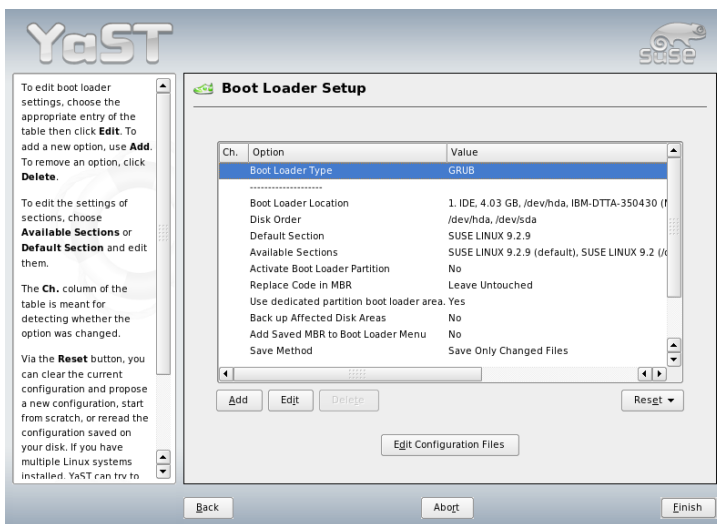


Figure 8.1: 使用 YaST 配置 引导 装载 程序

## 8.4.1 主窗口

列出配置数据的表由三列组成。在‘已更改’（左侧）下，有标志标记出中间列中所列出的选项中已更改的选项。若要添加某个选项，请单击‘添加’。若要更改现有选项的值，请通过鼠标单击将其选中，然后单击‘编辑’。如果根本不想使用某个现有选项，则将其选中，然后单击‘删除’。‘重置’提供了以下选项：

**建议新配置** 生成新配置建议。引导菜单中包括较早的 Linux 版本或其他分区上的其他操作系统，使您可以引导 Linux 或其以前的引导装载程序。后者会将您带入另一个引导菜单。

**从头开始** 用于从头开始创建整个配置。不生成任何建议。

**重新从磁盘读取配置** 如果您执行了某些更改但对结果不满意，则可以使用此选项重装当前配置。

**建议并与现有的 GRUB 菜单合并** 如果其他分区中安装有另一个操作系统和较早的 Linux 版本，则将生成一个菜单，该菜单包括用于新 SUSE LINUX 的一项，用于其他系统的一项和以前引导装载程序菜单的所有项。执行此过程可能要花较长的时间。如果使用 LILO，则这是不可能的。



恢复硬盘的 MBR 重新写入保存在硬盘上的备份 MBR。

使用‘编辑配置文件’在编辑器中编辑相关的配置文件。若要编辑某个文件，则通过选择字段装载该文件。单击‘确定’保存您的更改。若要退出引导装载程序配置，请单击‘取消’。单击‘后退’返回主窗口。

## 8.4.2 引导装载程序配置选项

使用 YaST 进行配置比直接编辑文件简单得多。选择一个选项，然后单击‘编辑’打开一个对话框，可以在该对话框中根据需要更改设置。单击‘确定’确认更改并返回主菜单，您可以在主菜单中编辑其他选项。可用的选项取决于所使用的引导装载程序。以下列表介绍了引导装载程序 GRUB 的一些选项：

**引导装载程序类型** 使用此选项在 GRUB 和 LILO 间切换。继续另一个对话框，可以在该对话框中指定执行此更改的方式。例如，将当前的 GRUB 配置转换为类似的 LILO 配置。但是，如果没有等效选项可用，某些设置将丢失。您还可以从头开始创建新配置或生成并编辑配置建议。

如果在正在运行的系统中启动引导装载程序配置，则可以从硬盘装载配置。如果决定返回原始引导装载程序，则可以通过最后一个选项装载其配置。但是，只有在您不关闭引导装载程序模块的情况下才存在这种可能。

**引导装载程序位置** 使用此对话框定义安装引导装载程序的位置：在主引导记录 (MBR) 中、在引导分区的引导扇区中（如果可用）、在根分区的引导扇区中、在软盘上。使用‘其他’指定其他位置。

**磁盘顺序** 如果您的计算机有多个硬盘，则按照计算机的 BIOS 设置中定义的顺序指定磁盘的引导顺序。

**默认部分** 使用此选项，设置默认情况下应引导的内核或操作系统。将在超时后引导所选的系统。在此菜单中，使用‘编辑’按钮获取所有引导菜单项的列表。从列表中选择一项，然后单击‘设为默认’。此时，您还可以使用‘编辑’修改任意项。

**可用部分** 引导菜单的现有项列在主窗口中此选项下。如果您选择此选项，然后单击‘编辑’，则将打开一个对话框，该对话框与‘默认项’对话框相同。

**使引导装载程序分区成为活动分区** 使用此选项激活其引导扇区中包含引导装载程序的分区，独立于储存了具有引导装载程序的帮助程序文件的目录的分区（/boot 或根目录 /）。

**替换 MBR 中的代码** 如果 GRUB 安装在 MBR 中，或者您正在新硬盘上安装系统但不想将 GRUB 安装在 MBR 中，则可以使用此选项将通用引导代码恢复到 MBR。

备份文件和部分硬盘 备份更改的硬盘区域。

将保存的 MBR 添加到引导装载程序菜单

将保存的 MBR 添加到引导装载程序菜单中。

使用‘超时’定义在引导默认系统之前引导装载程序应该等待键盘输入多少秒。可以通过‘添加’指定其他选项。有关可能选项的详细信息，请参考各自的手册页（grub(8) 或 lilo(8)）和位于 <http://www.gnu.org/software/grub/manual/> 的联机文档。

## 8.5 卸载 Linux 引导装载程序

YaST 可用于卸载 Linux 引导装载程序并将 MBR 恢复为安装 Linux 之前的状态。在安装过程中，YaST 自动创建原始 MBR 的备份副本并根据请求进行恢复，从而覆盖 GRUB。

若要卸载 GRUB，请启动 YaST 引导装载程序模块（‘系统’→‘引导装载程序配置’）。在第一个对话框中，选择‘重设置’→‘恢复硬盘的 MBR’，然后单击‘完成’退出对话框。在 MBR 中，将用原始 MBR 的数据覆盖 GRUB。

## 8.6 创建引导 CD

如果使用引导管理器引导系统时出现问题或如果不能将引导管理器安装在硬盘的 MBR 或软盘上，那么还可以创建包含所有必需的 Linux 启动文件的可引导 CD。这需要您的系统中安装有 CD 刻录机。

用 GRUB 创建可引导 CD-ROM 只需要特殊形式的 *stage2*（名为 *stage2\_eltorito*）以及自定义的 *menu.lst*（可选）。不需要标准文件 *stage1* 和 *stage2*。

例如，用 `cd /tmp` 和 `mkdir iso` 创建一个目录（将在该目录中创建 ISO 映像）。并且用 `mkdir -p iso/boot/grub` 为 GRUB 创建子目录。将文件 *stage2\_eltorito* 复制到目录 *grub* 中：

```
cp /usr/lib/grub/stage2_eltorito iso/boot/grub
```

另外，将内核（`/boot/vmlinuz`）、`initrd`（`/boot/initrd`）和文件 `/boot/message` 复制到 `iso/boot/` 中：

```
cp /boot/vmlinuz iso/boot/  
cp /boot/initrd iso/boot/  
cp /boot/message iso/boot/
```

若要使它们可用于 GRUB，请将文件 `menu.lst` 复制到 `iso/boot/grub` 并调整路径项使它们指向 CD-ROM 设备。执行此操作的方法是将路径名中硬盘的设备名（以 `(hd*)` 格式列出）替换为 CD-ROM 驱动器的设备名（即 `(cd)`）：

```
gfxmenu (cd)/boot/message  
timeout 8  
default 0  
  
title Linux  
    kernel (cd)/boot/vmlinuz root=/dev/hda5 vga=794 resume=/dev/hda1  
splash=verbose showopts  
    initrd (cd)/boot/initrd
```

最后，用以下命令创建 ISO 映像：

```
mkisofs -R -b boot/grub/stage2_eltorito -no-emul-boot \  
-boot-load-size 4 -boot-info-table -o grub.iso iso
```

这会使用您选择的实用程序将最终文件 `grub.iso` 刻录到 CD 上。

## 8.7 图形 SUSE 屏幕

从 SUSE LINUX 7.2 开始，如果将选项“`vga=|value|`”用作内核参数，则图形 SUSE 屏幕会显示在第一个控制台上。如果您使用 YaST 进行安装，则将依照所选的分辨率和图形卡自动激活此选项。可以根据需要通过三种方法禁用 SUSE 屏幕：

**在必要时禁用 SUSE 屏幕。** 在命令行上输入命令 `echo 0 >/proc/splash` 以禁用图形屏幕。要将其再次激活，请输入 `echo 1 >/proc/splash`。

**默认禁用 SUSE 屏幕。** 将内核参数 `splash=0` 添加到您的引导装载程序配置中。第 8 章 引导装载程序 提供了有关此内容的详细信息。但是，如果您倾向于使用文本方式（这是早期版本中的默认方式），请设置 `vga=normal`。

**完全禁用 SUSE 屏幕。** 编译新内核并禁用‘帧缓冲支持’中的选项‘使用启动屏幕而不是引导徽标’。

## Tip

在内核中禁用帧缓冲支持也会自动禁用启动屏幕。如果您使用自定义内核运行 SUSE，则它不能为系统提供任何支持。

## Tip

## 8.8 查错

本节列出使用 GRUB 进行引导的一些常见问题并提供可能解决方案的简短说明。一些问题在位于 <http://portal.suse.de/sdb/en/index.html> 的支持数据库的文章中有所介绍。如果某个特定问题未包括在此列表中，请使用位于 <https://portal.suse.com/PM/page/search.pm> 的支持数据库的搜索对话框搜索 *GRUB*、*boot*（引导）、和 *boot loader*（引导装载程序）等关键字。

**GRUB 和 XFS** XFS 未在分区引导块中为 *stage1* 预留任何空间。因此，不要指定 XFS 分区作为引导装载程序的位置。此问题可以通过创建单独的引导分区（不使用 XFS 进行格式化）得到解决。

**GRUB 和 JFS** 虽然在技术上可以实现，但将 GRUB 和 JFS 结合使用经常会出现问题。在这种情况下，创建单独的引导分区 (*/boot*) 并用 Ext2 对其进行格式化。在此分区中安装 GRUB。

**GRUB 报告 GRUB Geom 错误** 当引导系统时，GRUB 将检查连接的硬盘的磁盘空间。有时，BIOS 将返回不一致的信息，GRUB 将报告 GRUB Geom 错误。如果出现这种情况，请使用 LILO 或更新 BIOS。有关安装、配置和维护 LILO 的详细信息，可以在支持数据库中关键字“LILO”下获得。

如果将 Linux 安装在未在 BIOS 中注册的其他硬盘上，GRUB 也会返回此错误讯息。找到并正确装载了引导装载程序的 *stage1*，但未找到 *stage2*。可以通过在 BIOS 中注册新硬盘解决此问题。

**包含 IDE 和 SCSI 硬盘的系统未引导** 在安装过程中，YaST 可能会错误地确定硬盘的引导顺序（并且您未对此进行更正）。例如，GRUB 可能将 */dev/hda* 视为 *hd0* 并将 */dev/sda* 视为 *hd1*，虽然 BIOS 中的引导顺序是相反的（SCSI 先于 IDE）。

在这种情况下，在引导进程中借助 GRUB 命令行对硬盘进行更正。在引导系统后，编辑文件 *device.map* 永久应用新映射。然后，检查 */boot/grub/menu.lst* 和 */boot/grub/device.map* 文件中的 GRUB 设备名，并使用以下命令重安装引导装载程序：

```
grub --batch < /etc/grub.conf
```

从第二块硬盘引导 **Windows** 某些操作系统（例如 Windows）只能从第一块硬盘进行引导。如果这样的操作系统安装在第一块硬盘之外的硬盘上，您可以影响相应菜单项的逻辑更改。

```
...
title windows
map (hd0) (hd1)
map (hd1) (hd0)
chainloader(hdl,0)+1
...
```

在此示例中，将从第二块硬盘启动 Windows。出于此目的，请使用 `map` 更改硬盘的逻辑顺序。此更改不会影响 GRUB 菜单文件中的逻辑。因此，必须为 `chainloader` 指定第二块硬盘。

## 8.9 参考信息

有关 GRUB 的大量信息可以在 <http://www.gnu.org/software/grub/> 处获得。如果计算机上安装有 `texinfo`，则可以通过输入 `info grub` 在 shell 中查看 GRUB 信息页。您还可以在位于 <http://portal.suse.de/sdb/en/index.html> 的支持数据库中搜索关键字“GRUB”获得有关特殊问题的信息。



# Linux 内核

内核管理着每个 Linux 系统的硬件并使其可用于各种进程。虽然本章提供的信息不会使您成为一个内核方面的高手，但您可以了解如何执行内核更新以及如何编译并安装自定义内核。如果遵照本章中的说明，先前的内核仍将保持有效，并可在需要时对其进行引导。

9.1	内核更新	166
9.2	内核源	166
9.3	内核配置	166
9.4	内核模块	168
9.5	编译内核	170
9.6	安装内核	170
9.7	在编译后清理硬盘	171

可以配置安装在 `/boot` 目录中的内核用于众多硬件。通常情况下没有必要编译自定义内核，除非您希望测试试用功能和驱动程序。

通常，可以通过内核参数来修改已安装内核的行为。例如，参数 `desktop` 为安排器设置较短的时间片，从而产生系统的主观加速。假定已安装了包 `kernel-source`，目录 `/usr/src/linux/Documentation` 中的内核文档中提供相应的信息。

随内核提供有多个 `Makefiles`，用于自动化进程。选择硬件设置和其他内核功能。因为您需要非常了解自己的计算机系统才能做出正确的选择，所以建议您开始时先尝试修改现有的和正在使用的配置文件。

## 9.1 内核更新

要安装正式的 SUSE 更新内核，请使用 YaST 的联机更新功能。在内核更新后，必须重引导系统，因为仍在运行的旧内核将找不到正确的模块来提供所需的功能。请参见位于第 2.3.3 节“YaST 联机更新”的有关 YaST 联机更新的详细信息。

在运行更新时，将出现一个弹出窗口，它解释所有需要的操作。请使用这些命令来维护一致的系统。

## 9.2 内核源

要构建一个内核，就必须安装包 `kernel-source`。YaST 将自动选择安装其他的包，例如 C 编译器（包 `gcc`）、GNU Binutils（包 `binutils`）和 C 编译器的包含文件（包 `glibc-devel`），并且必须安装这些包。

在安装后，内核源位于 `/usr/src/linux-<kernel-version>` 中。如果要试用不同的内核，应将它们解包在不同的子目录中，并创建一个到当前内核源的符号链接。因为存在依赖位于 `/usr/src/linux` 中的源的软件包，所以应将此目录作为一个到当前内核源的符号链接进行维护。YaST 会自动完成此操作。

## 9.3 内核配置

当前内核的配置储存在文件 `/proc/config.gz` 中。要修改此配置，请以 `root` 身份转到目录 `/usr/src/linux`，然后执行以下命令：



```
zcat /proc/config.gz > .config
make oldconfig
```

命令 `make oldconfig` 使用文件 `/usr/src/linux/.config` 作为当前内核配置的模板。这时将查询当前内核源的所有新选项。如果文件 `.config` 不存在，就会使用包含在内核源中的默认配置。

本文不可能详细介绍所有内核配置选项。请利用有关内核配置的大量帮助文本。在 `/usr/src/linux/Documentation` 中总可以找到最新的内核文档。

### 9.3.1 用命令行进行配置

要配置内核，可转到 `/usr/src/linux` 并输入命令 `make config`。选择希望由内核支持的功能。通常情况下存在着 2 个或 3 个选项：**(Y)**、**(N)** 和 **(M)**。**(M)** 意味着此设备不应直接编译入内核，而应作为一个模块来装载。引导系统所需的驱动程序必须使用 **(Y)** 集成到内核中。按 **(Enter)** 键来确认从文件 `.config` 读取的默认设置。按任何其他键可查看各选项的简要帮助文本。

### 9.3.2 用文本方式进行配置

`menuconfig` 是一种更方便的配置内核的方式。如果需要，使用 YaST 安装 `ncurses-devel`。使用命令 `make menuconfig` 启动内核配置。

要在配置中进行小的更改，不必经过整个过程，直接使用菜单来访问相关部分即可。这时将从文件 `.config` 装载默认设置。要装载不同的配置，请选择‘装载备用配置文件’，然后输入文件名。

### 9.3.3 在 X 窗口系统中进行配置

如果已安装并配置 X 窗口系统（包 `xorg-x11`）和 QT 开发包（`qt3-devel`），则可以使用命令 `make xconfig` 访问图形用户界面以进行配置。如果没有以 `root` 身份登录 X 窗口系统，请输入命令 `sux` 来获得具有此显示访问权限的 `root-shell`。这时将从文件 `.config` 装载默认设置。由于使用 `make xconfig` 进行的配置不像其他配置功能一样得到适当维护，所以在使用这种配置方法后应运行命令 `make oldconfig`。

## 9.4 内核模块

PC 硬件部件各种各样。要正常地使用硬件，就需要一个“驱动程序”，以便操作系统（在 Linux 中就是内核）可以通过它访问此硬件。将驱动程序集成到您的系统时，基本上可采用以下两种方式：

- 可以将驱动程序直接编译入内核。这样的内核（“一个整体”）被称为整体内核。某些驱动程序仅在这种形式下可用。
- 也可以根据需要将驱动程序装载到内核中。这样的内核被称为模块化内核。它的优势是只装载真正需要的驱动程序，从而使得内核中不包含任何不需要的东西。

内核配置中定义了要将哪些驱动程序编译入内核，以及要将哪些驱动程序作为运行时模块加载。基本上应将引导系统时不需要的部件构建为模块。这样可确保内核不会变得太大而难以由 BIOS 或引导装载程序来装载。应将用于 ext2 的驱动程序、基于 SCSI 系统的 SCSI 驱动程序以及类似驱动程序编译到内核中。而 isofs、msdos 或 sound 等项目在启动您的计算机系统时不需要，所以应明确地将其构建为模块。

### Tip

甚至可以将引导系统所需的驱动程序构建为模块。在这种情况下，最初的 RAM 磁盘用于在引导时装载这些模块。

### Tip

内核模块位于 `/lib/modules/<version>` 中。`version` 代表当前内核版本。

### 9.4.1 通过 `hwinfo` 进行硬件检测

`hwinfo` 可以检测系统的硬件并选择运行此硬件所需的驱动程序。使用 `hwinfo --help` 可获得对此命令的简要介绍。例如，如果您需要关于 SCSI 设备的信息，可使用命令 `hwinfo --scsi`。所有这些信息也可在 YaST 的硬件信息模块中获得。

### 9.4.2 处理模块

包 `module-init-tools` 中提供了将模块装载到内核所需的实用程序。有以下命令可用：

**insmod** insmod 在 `/lib/modules/<version>` 的一个子目录中搜索所请求的模块，然后将其装载。但是，最好使用 `modprobe` 而不是 `insmod`，因为 `modprobe` 还检查模块的依赖性。

**rmmmod** 卸载所请求的模块。仅当不再需要此模块时才能这样做。例如，不能在未取出 CD 时卸载 `isofs` 模块。

**depmod** 在 `/lib/modules/<version>` 中创建文件 `modules.dep`，由它定义所有模块之间的依赖性。这一步骤是必需的，目的是确保装载所选择模块依赖的全部模块。如果此文件不存在，则将在启动系统后构建它。

**modprobe** 装载或卸载给定模块，同时考虑此模块的依赖性。此命令非常强大，可用于很多方面，包括探测给定类型的所有模块，直到成功装载其中的一个模块。与 `insmod` 不同，`modprobe` 检查 `/etc/modprobe.conf`，使其成为装载模块的首选方法。有关此主题的详细信息，请参考相应的手册页。

**lsmod** 显示当前装载了哪些模块以及有多少其他模块正在使用它们。由内核守护程序启动的模块标记有 `autoclean`。这个标签表示这些模块一旦超出空闲时间限制就会被自动去除。

**modinfo** 显示模块信息。由于这些信息是从模块本身抽取的，所以只能显示由驱动程序开发人员构建在其中的信息。这些信息可能包括作者、说明、许可证、模块参数、依赖性和别名。

### 9.4.3 /etc/modprobe.conf

这些模块的装载受文件 `/etc/modprobe.conf` 和 `/etc/modprobe.conf.local` 以及目录 `/etc/modprobe.d` 的影响。请参见 `man modprobe.conf`。必须在此文件中输入直接访问硬件的模块的参数。这些模块（如 CD-ROM 驱动程序或网络驱动程序）可能需要特定于系统的选项。在内核源中将说明这里使用的参数。安装包 `kernel-source` 并阅读目录 `/usr/src/linux/Documentation` 中的文档。

### 9.4.4 Kmod — 内核模块装载程序

内核模块装载程序是使用模块最合适的方式。`Kmod` 执行背景监视，并确保 `modprobe` 在内核中需要某功能时装载所需的模块。

要使用 `Kmod`，请激活内核配置中的‘内核模块装载程序’(`CONFIG_KMOD`) 选项。`Kmod` 不是设计用来自动卸载模块的。考虑到现在的 RAM 容量，潜在的内存节省已经变得不再重要。

## 9.5 编译内核

### ► x86, AMD64, EM64T

建议编译“bzImage”。作为一条规则，这样可避免出现内核变得过大的问题。如果选择过多功能并创建“zImage”，就很容易发生这种问题。这时您将收到如内核过大或系统过大等错误讯息。◀

在自定义内核配置后（如第9.3节“内核配置”所述），要启动编译，应输入以下内容（切记应首先转至目录 /usr/src/linux）：

```
make clean
make bzImage
```

这两个命令可以作为一个命令行输入：

```
make clean bzImage
```

在成功编译后，压缩内核位于 /usr/src/linux/arch/<arch>/boot 中。内核映像（包含内核的文件）被称为 bzImage。

如果无法找到此文件，在内核编译时就可能会出错。在 Bash shell 中，输入以下命令来再次启动内核编译，并将输出写到文件 kernel.out：

```
make bzImage V=1 2>&1 | tee kernel.out
```

如果已经配置内核的某些部分作为模块来装载，请启动模块编译。要使用的命令是 make modules。

## 9.6 安装内核

在编译完内核后，必须将其安装后才能引导它。必须将内核安装在目录 /boot 中。请使用以下命令完成该操作：

```
INSTALL_PATH=/boot make install
```

现在需要安装已编译的模块。输入 make modules\_install 来将它们复制到 /lib/modules/<version> 中的正确目标目录。如果内核版本相同，则会覆盖旧模块。但是，仍可从 CD 与内核一起重新安装原始模块。

**Tip**

为了避免出现意外问题，应确保从 `/lib/modules/<version>` 去除其功能现在可能已被直接编译入内核的那些模块。这也是强烈建议没有经验的用户不要编译内核的原因之一。

**Tip**

为了使 GRUB 能够引导旧内核（现在是 `/boot/vmlinuz.old`），请将标签 `Linux.old` 作为引导映像添加到文件 `/boot/grub/menu.lst` 中。第8章 引导装载程序中详细说明了此步骤。不需要重新安装 GRUB。

文件 `/boot/System.map` 包含模块所需的内核符号，用于确保成功启动内核功能。此文件取决于当前内核。因此，在编译和安装了内核后，应立即将 `/usr/src/linux/System.map` 复制到目录 `/boot`，每次编译内核时均会重新生成此文件。如果收到如“`System.map` 与当前内核不匹配”等错误讯息，就极可能是您在编译内核后忘记将 `System.map` 复制到 `/boot`。

## 9.7 在编译后清理硬盘

如果硬盘空间有限，可使用 `make clean` 删除内核编译时在 `/usr/src/linux` 目录中生成的对象文件。如果磁盘空间充裕并计划定期重配置内核，则可以跳过此步骤。重编译内核要快得多，因为实际上将只重编译受更改影响的部分。



# SUSE LINUX 的特殊功能

本章提供有关各种软件包、虚拟控制台和键盘布局的信息。随后是有关语言和国家/地区特定设置（I18N 和 L10N）的内容。

10.1	特殊软件包的相关信息	174
10.2	虚拟控制台	181
10.3	键盘映射	181
10.4	语言和国家/地区特定的设置	182

## 10.1 特殊软件包的相关信息

### 10.1.1 包 `bash` 和 `/etc/profile`

下面的列表列出了 `Bash` 被用作登录 shell 时读取的所有 init 文件。Bash 按照这些文件在列表中出现的顺序处理它们。

1. `/etc/profile`
2. `~/.profile`
3. `/etc/bash.bashrc`
4. `~/.bashrc`

用户可以在 `~/.profile` 或 `~/.bashrc` 中进行个人输入。要确保正确处理这些文件，需要将基本设置从 `/etc/skel/.profile` 或 `/etc/skel/.bashrc` 复制到用户的主目录中。建议在更新后从 `/etc/skel` 复制这些设置。执行以下 shell 命令以避免个人调整丢失：

```
mv ~/.bashrc ~/.bashrc.old
cp /etc/skel/.bashrc ~/.bashrc
mv ~/.profile ~/.profile.old
cp /etc/skel/.profile ~/.profile
```

然后将个人调整从文件 `*.old` 复制回来。

### 10.1.2 `cron` 包

`cron` 表位于 `/var/spool/cron/tabs` 中。`/etc/crontab` 用作整个系统范围的 `cron` 表。输入用户的名称，该用户应按照时间表直接运行命令。在例 10.1 “`/etc/crontab` 中的项的示例”中，输入的是 `root`。位于 `/etc/cron.d` 中的包特定的表具有相同的格式。请参见 `man cron`。

#### 10.1: `/etc/crontab` 中的项的示例

```
1-59/5 * * * * root test -x /usr/sbin/atrun && /usr/sbin/atrun
```



不能使用 `crontab -e` 来处理 `/etc/crontab`。必须直接将其装载到编辑器中，对其进行修改，然后保存。

许多包将 shell 脚本安装到目录 `/etc/cron.hourly`、`/etc/cron.daily`、`/etc/cron.weekly` 和 `/etc/cron.monthly` 中，它们的指令是由 `/usr/lib/cron/run-crons` 控制的。`/usr/lib/cron/run-crons` 每隔 15 分钟在主表 (`/etc/crontab`) 中运行一次。这样可以确保在适当的时间运行可能被忽略的进程。

为了清楚起见，已经将日常系统维护作业分发给了多个脚本。这些脚本包含在包 `aaa_base` 中。例如，`/etc/cron.daily` 包含部件 `backup-rpmdb`、`clean-tmp` 或 `clean-vi`。

### 10.1.3 日志文件：包 `logrotate`

有一些系统服务（守护程序）和内核本身一起定期将系统状态和特定事件记录到日志文件中。这样，管理员可以定期检查系统在某一时刻的状态，发现错误或有问题的功能，并极为精确地解决这些问题。这些日志文件通常储存在 FHS 指定的 `/var/log` 中，文件大小每天都会增长。`logrotate` 包可以帮助控制这些文件的生长。

#### 配置

用文件 `/etc/logrotate.conf` 配置 `logrotate`。特别要指出的是，`include` 指定主要配置要读取的其他文件。SUSE LINUX 确保各个包在 `/etc/logrotate.d` 中安装文件（例如 `syslog` 或 `yast`）。

#### 10.2: `/etc/logrotate.conf` 的示例

```
# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
#compress

# RPM packages drop log rotation information into this directory
include /etc/logrotate.d
```

```
# no packages own lastlog or wtmp - we'll rotate them here
#/var/log/wtmp {
#   monthly
#   create 0664 root utmp
#   rotate 1
#}

# system-specific logs may be also be configured here.
```

通过 cron 控制 logrotate，并通过 `/etc/cron.daily/logrotate` 每天对其进行调用。

### Important

使用 `create` 选项可以读取管理员在 `/etc/permissions*` 中进行的所有设置。确保没有因个人修改而引起的冲突。

Important

## 10.1.4 手册页

对于某些 GNU 应用程序（如 `tar`），已不再保留手册页。对于这些命令，可使用 `--help` 选项查看快速概述或信息页，其中提供更多深入的说明。`info` 是 GNU 的超文本系统。通过输入 `info info`，就可以看到此系统的介绍。通过输入 `emacs -f info` 可使用 Emacs 查看信息页，也可以在控制台中使用 `info` 直接查看信息页。还可以使用 `tkinfo`、`xinfo` 或 SUSE 帮助系统来查看信息页。

## 10.1.5 locate 命令

安装软件的标准安装范围不包括用于快速查找文件的 `locate`。如有必要，请安装包 (`find-locate`)。 `updatedb` 进程将在每天晚上或引导系统约 15 分钟后自动启动。

## 10.1.6 ulimit 命令

使用 `ulimit` (*user limits*) 命令，可以对系统资源的使用设置限制并将这些信息显示出来。`ulimit` 尤其适用于限制应用程序可用的内存。使用此命令，可以防止某个应用程序自己占用太多内存，这可能导致系统停顿。

可以对 `ulimit` 使用多个选项。要限制使用内存，请使用表 10.1 “`ulimit`: 为用户设置资源” 中列出的选项。

Table 10.1: ulimit: 为用户设置资源

-m	物理内存的最大大小
-v	虚拟内存的最大大小
-s	堆栈的最大大小
-c	核心文件的最大大小
-a	显示限制设置

可以在 `/etc/profile` 中进行整个系统范围的设置。在这里可以创建编程人员进行调试所需的核心文件。普通用户不能增加系统管理员在 `/etc/profile` 中指定的值，但可以在他自己的 `~/.bashrc` 中进行特殊输入。

### 10.3: ulimit: `~/.bashrc` 中的设置

```
# Limits of physical memory:
ulimit -m 98304

# Limits of virtual memory:
ulimit -v 98304
```

必须以 KB 为单位指定内存量。有关详细信息，请参见 `man bash`。

#### Important

并非所有 shell 都支持 `ulimit` 指令。如果您依赖于这些限制的内存设置，则 PAM（例如 `pam_limits`）提供了全面的调整功能。

#### Important

## 10.1.7 free 命令

如果您的目的是查看当前使用了多少 RAM，则 `free` 命令可能会令人产生误解。`/proc/meminfo` 中提供了相关信息。目前，使用 Linux 等现代操作系统的用户实际上无需过多地担心内存。可用 RAM 的概念可追溯到统一内存管理之前。可用内存不是好的内存这种说法非常适用于 Linux。因此，Linux 一直在平衡超速缓存方面下功夫，不允许实际上存在可用或未使用的内存。

内核基本上不直接管理任何应用程序或用户数据。而是在一个页超速缓存中管理应用程序和用户数据。如果内存不足，它的某些部分会被写入交换分区或文件中，借助于 `mmap` 命令，可以最先从这些交换分区或文件中读取这些部分（请参见 `man mmap`）。

此外，内核还包含其他超速缓存，如 `slab` 超速缓存，其中储存了用于网络访问的超速缓存。这也许能够解释 `/proc/meminfo` 中计数器之间的差异。通过 `/proc/slabinfo` 可以访问大多数（但并非全部）`slab` 超速缓存。

### 10.1.8 文件 `/etc/resolv.conf`

系统通过文件 `/etc/resolv.conf` 处理域名解析。请参考第24章 域名系统。此文件仅由脚本 `/sbin/modify_resolvconf` 进行更新，任何其他程序都没有直接修改 `/etc/resolv.conf` 的权限。只有实施这条规则才能确保系统的网络配置和相关文件保持一致。

### 10.1.9 GNU Emacs 的设置

GNU Emacs 是一个复杂的工作环境。有关详细信息，请参见 <http://www.gnu.org/software/emacs/>。下面几节介绍当启动 GNU Emacs 时处理的配置文件。

启动时，Emacs 读取包含用户、系统管理员和经销商的设置的多个文件以进行自定义或预配置。初始化文件 `~/.emacs` 被安装到 `/etc/skel` 中各个用户的主目录中。`.emacs` 又会读取文件 `/etc/skel/.gnu-emacs`。要自定义程序，请将 `.gnu-emacs` 复制到主目录（通过 `cp /etc/skel/.gnu-emacs ~/.gnu-emacs`）并在那里进行所需的设置。

`.gnu-emacs` 将文件 `~/.gnu-emacs-custom` 定义为 `custom-file`。如果用户通过 `customize` 选项进行设置，则会将这些设置保存到 `~/.gnu-emacs-custom` 中。

通过 SUSE LINUX，`emacs` 包将文件 `site-start.el` 安装在目录 `/usr/share/emacs/site-lisp` 中。文件 `site-start.el` 在初始化文件 `~/.emacs` 之前进行装载。除其它作用之外，`site-start.el` 确保自动装载通过 Emacs 扩充包分发的特殊配置文件（如 `psgml`）。此类型的配置文件也位于 `/usr/share/emacs/site-lisp` 中，总是以 `suse-start-` 开头。本地系统管理员可以在 `default.el` 中指定整个系统范围的设置。

初始化文件下的 Emacs 信息文件中提供了有关这些文件的详细信息：`info:/emacs/InitFile`。此位置还提供了有关如何禁止装载这些文件（如果需要）的信息。

Emacs 的部件被分成多个包：

- 基础包 `emacs`。
- `emacs-x11`（通常已安装）：带有 X11 支持的程序。
- `emacs-nox`：不带 X11 支持的程序。
- `emacs-info`：信息格式的联机文档。
- `emacs-el`：Emacs Lisp 中未编译的库文件。运行时不需要这些库文件。
- 如果需要，可以安装多个扩充包：`emacs-auctex`（用于 LaTeX）、`psgml`（用于 SGML 和 XML）、`gnuserv`（用于客户机和服务器操作）等。

### 10.1.10 vi 简介

文本编辑器仍用于执行许多系统管理任务和编程。在 Unix 世界中，vi 是一款很好的编辑器，它提供了便于使用的编辑功能，而且比许多具有鼠标支持的编辑器更符合人体工程学。

#### 操作方式

vi 基本上采用三种操作方式：*插入方式*、*命令方式*和*扩展方式*。根据操作方式，各按键具有不同的功能。启动时，vi 通常被设置为命令方式。首先需要了解如何在这些方式之间进行切换：

**命令方式切换到插入方式** 此时有许多选择，其中使用 `@` 可以进行追加，使用 `I` 可以进行插入，使用 `O` 可以在当前行下创建一个新行。

**插入方式切换到命令方式** 按 `(Esc)` 键退出插入方式。不能在插入方式下终止 vi，所以一定要习惯于按 `(Esc)` 键。

**命令方式切换到扩展方式** 通过输入冒号 `(:)` 可以激活 vi 的扩展方式。扩展或 *ex* 方式就像是一个独立的面向行的编辑器，可用于多种简单和较复杂的任务。

**扩展方式切换到命令方式** 在扩展方式下执行命令后，编辑器将自动返回命令方式。如果决定不在扩展方式下执行任何命令，请使用 `(=)` 键删除冒号。编辑器即返回到命令方式。

必须先从插入方式切换到命令方式，之后才能切换到扩展方式。

vi 与其他编辑器一样，也有自己的终止程序的过程。您不能在插入方式下终止 vi。首先，按 **(Esc)** 键退出插入方式。接下来有两种选择：

1. **退出而不进行保存：**要终止编辑器而不保存更改，请在命令方式下输入 **(:)** **(q)!**。感叹号 **!** 使 vi 忽略任何更改。
2. **保存并退出：**保存您的更改并终止编辑器可以采用多种方法。在命令方式下，使用 **(Z)Z)**。在扩展方式下，要保存所有更改并退出程序，请输入 **(:)** **(w)(q)**。在扩展方式下，**(w)** 代表“写”，**(q)** 代表“退出”。

## 使用 vi

vi 可用作常规编辑器。在插入方式下，可以输入文本，也可以使用 **(←)**Backspace 键和 **(Del)**Delete 键删除文本。使用箭头键可以移动光标。

但这些控制键经常会出现问题，因为有许多终端类型使用特殊键代码。这时就要使用命令方式。按 **(Esc)** 键从插入方式切换到命令方式。在命令方式下，使用 **(H)**、**(J)**、**(K)** 和 **(L)** 键移动光标。这些键具有以下功能：

- (H)** 左移一个字符
- (J)** 下移一行
- (K)** 上移一行
- (L)** 右移一个字符

在命令方式下，允许命令采用多种变化形式。要多次执行一个命令，只需要在输入实际命令之前输入重复次数即可。例如，输入 **(5)(L)** 可将光标右移 5 个字符。

## 参考信息

vi 支持多种不同的命令。它支持使用宏、快捷方式、命名缓冲区和许多其他有用的功能。本手册不提供各个选项的详细说明。SUSE LINUX 附带有 vim (vi 改进版)，这是经过改进的 vi 版本。此应用程序有许多信息源：

- vimtutor 是 vim 的交互式教程。
- 在 vim 中，输入命令 `:help` 可以获得有关许多主题的帮助。

- <http://www.truth.sk/vim/vimbook-OPL.pdf> 上联机提供了一本有关 vim 的参考书。
- 位于 <http://www.vim.org> 上的 vim 项目万维网网页提供了各类新闻、邮件列表和其他文档。
- 因特网上有许多 vim 信息源：<http://www.selinux.org/selinux/html/vim.html>、<http://www.linuxgazette.com/node/view/9039>、[http://www.apmaths.uwo.ca/~xli/vim/vim\\_tutorial.html](http://www.apmaths.uwo.ca/~xli/vim/vim_tutorial.html)。有关指向各教程的链接，请参见 <http://linux-universe.com/HOWTO/Vim-HOWTO/vim-tutorial.html>。

## Important

### VIM 许可证

vim 是一款“慈善软件”，这意味着作者不对此软件收取任何费用，但鼓励您进行捐款以支持一项非营利计划。此计划鼓励人们为乌干达的贫困儿童提供帮助。有关详细信息，请访问 <http://iccf-holland.org/index.html>、<http://www.vim.org/iccf/> 和 <http://www.iccf.nl/>。

Important

## 10.2 虚拟控制台

Linux 是一个多用户和多任务的系统。即使是在独立计算机系统上也可以感受到这些功能的好处。在文本方式下，提供了 6 个虚拟控制台。可以使用 **(Alt)-(F1)** 到 **(Alt)-(F6)** 在这些控制台间切换。第七个控制台为 X 保留。可以通过修改文件 `/etc/inittab` 控制指派控制台的数目。

要从 X 切换到控制台而不将其关闭，请使用 **(Ctrl)-(Alt)-(F1)** 到 **(Ctrl)-(Alt)-(F6)**。使用 **(Alt)-(F7)** 可以返回到 X。

## 10.3 键盘映射

为了标准化程序的键盘映射，对以下文件进行了更改：

```
/etc/inputrc
/usr/X11R6/lib/X11/Xmodmap
/etc/skel/.Xmodmap
/etc/skel/.exrc
/etc/skel/.less
/etc/skel/.lesskey
/etc/csh.cshrc
/etc/termcap
/usr/lib/terminfo/x/xterm
/usr/X11R6/lib/X11/app-defaults/XTerm
/usr/share/emacs/<VERSION>/site-lisp/term/*.el
```

这些更改只影响使用 `terminfo` 项的应用程序或其配置文件被直接更改（`vi`、`less` 等）的应用程序。不是 SUSE LINUX 附带的应用程序应该根据这些默认设置进行调整。

在 X 下，可以使用 `(Ctrl)-(Shift)`（右边的）访问组合键 (multikey)。同时可以在 `/usr/X11R6/lib/X11/Xmodmap` 中看到对应的项。

可以通过“X 键盘扩展” (XKB) 进行其他设置。桌面环境 GNOME (`gswitchit`) 和 KDE (`kxkb`) 也使用此扩展。有关 XKB 的信息，请参见 `/etc/X11/xkb/README` 和那里列出的文档。

有关中文、日文和韩文 (CJK) 输入的详细信息，请参见 Mike Fabian 的网页：<http://www.suse.de/~mfabian/suse-cjk/input.html>。

## 10.4 语言和国家/地区特定的设置

SUSE LINUX 在很大程度上进行了国际化，可通过灵活的方式进行修改以满足本地需要。换句话说，国际化 (*I18N*) 允许特定的本地化 (*L10N*)。I18N 和 L10N 这两个缩写词使用原单词的第一个和最后一个字母，中间的数字表示省略的字母数。

设置是通过文件 `/etc/sysconfig/language` 中定义的 `LC_` 变量进行的。这不仅指本地语言支持，还指讯息（语言）、字符集、排序顺序、日期和时间、数字和货币等类别。这些类别中的每个类别都可以使用其自己的变量直接定义或使用文件 `language` 中的主变量间接定义（请参见手册页 `man locale`）。

**RC\_LC\_MESSAGES、RC\_LC\_CTYPE、RC\_LC\_COLLATE、RC\_LC\_TIME、RC\_LC\_NUMERIC、RC\_LC\_MONETARY**

将这些变量不带 `RC_` 前缀的形式传递到 shell，它们控制上述类别。下面列出了相关文件。可以使用命令 `locale` 显示当前设置。

**RC\_LC\_ALL** 此变量（如果设置）覆盖上述变量的值。



**RC.LANG** 如果未设置上面介绍的任何变量，则这是后备变量。默认情况下，SUSE LINUX 只设置 `RC.LANG`。这便于用户输入他们自己的值。

**ROOT\_USES\_LANG** `yes` 或 `no` 变量。如果将其设置为 `no`，则 `root` 用户始终在 POSIX 环境中工作。

可以通过 YaST `sysconfig` 编辑器设置其他变量。这一变量的值包含语言代码、国家/地区代码、编码和修饰键。各部分之间通过特殊字符连接：

```
LANG=<language>[_<COUNTRY>].<Encoding>[@<Modifier>]]
```

### 10.4.1 一些示例

语言和国家/地区代码始终应该一起设置。语言设置遵循标准 ISO 639 (<http://www.evertype.com/standards/iso639/iso639-en.html> 和 <http://www.loc.gov/standards/iso639-2/>)。国家/地区代码列在 ISO 3166 中 ([http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/en\\_listp1.html](http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/en_listp1.html))。只有设置可以在 `/usr/lib/locale` 中找到其可用说明文件的值才有意义。可以使用命令 `localedef` 基于 `/usr/share/i18n` 中的文件创建更多说明文件；说明文件是 `glibc-i18ndata` 软件包的一部分。可以使用以下命令创建 `en_US.UTF-8`（用于英国英语和美国英语）的说明文件：

```
localedef -i en_US -f UTF-8 en_US.UTF-8
```

**LANG=en\_US.UTF-8** 如果在安装过程中选择的是英语，则这是默认设置。如果选择的是另一种语言，则将启用该语言，但仍将 UTF-8 用作字符编码。

**LANG=en\_US.ISO-8859-1** 这会将语言设置为英语，将国家/地区设置为美国，将字符集设置为 ISO-8859-1。此字符集不支持欧元符号，但它有时可用于尚未进行更新以支持 UTF-8 的程序。随后，Emacs 等程序将对定义字符集的字符串（在本例中为 ISO-8859-1）进行求值。

**LANG=en\_IE@euro** 上例将欧元符号显式包含在语言设置中。严格来说，此设置目前已过时，因为 UTF-8 也包含欧元符号。该设置仅在应用程序不支持 UTF-8 但支持 ISO-8859-15 时才有用。

SuSEconfig 读取 `/etc/sysconfig/language` 中的变量并将必需的更改写入 `/etc/SuSEconfig/profile` 和 `/etc/SuSEconfig/csh.cshrc`。`/etc/SuSEconfig/profile` 被 `/etc/profile` 读取或用作其数据的来源。`/etc/`

SuSEconfig/csh.cshrc 被用作 /etc/csh.cshrc 的数据来源。这使设置在  
整个系统范围内可用。

用户可以通过相应地编译他们的 ~/.bashrc 覆盖系统默认值。例如，如果不想  
将整个系统范围的 en\_US 用于程序讯息，请包括 LC\_MESSAGES=es\_ES，这样讯  
息将以西班牙语显示。

## 10.4.2 语言支持的设置

讯息类别中的文件通常只储存在对应的语言目录（例如 en）中以保留后备。如  
果您将 LANG 设置为 en\_US 并且 /usr/share/locale/en\_US/LC\_MESSAGES  
中的讯息文件不存在，则它将使用 /usr/share/locale/en/LC\_MESSAGES。  
还可以定义后备链，例如，定义法国布列塔尼地区的语言的后备为法语，定义加  
利西亚语的后备为西班牙语，第二后备为葡萄牙语：

```
LANGUAGE="br_FR:fr_FR"
```

```
LANGUAGE="gl_ES:es_ES:pt_PT"
```

如果需要，改用挪威语变体 *nynorsk* 和 *bokml*（将其他后备设置为 no）：

```
LANG="nn_NO"
```

```
LANGUAGE="nn_NO:nb_NO:no"
```

或

```
LANG="nb_NO"
```

```
LANGUAGE="nb_NO:nn_NO:no"
```

请注意，在挪威语中，LC\_TIME 的处理方式也有所不同。

### 可能的问题

不识别千分位逗号。LANG 可能被设置为 en，但 glibc 使用的说明位于 /usr/  
share/lib/en\_US/LC\_NUMERIC 中。例如，必须将 LC\_NUMERIC 设置为 en\_  
US。

### 参考信息

- GNU C Library 参考手册中的“区域设置和国际化”一章；包括在  
glibc-info 中。
- Markus Kuhn 编写的 *Unix/Linux 的 UTF-8 和 Unicode 常见问题解答*，当前  
位于 <http://www.cl.cam.ac.uk/~mgk25/unicode.html>。
- Bruno Haible 编写的 *Unicode-Howto*：file:/usr/share/doc/howto/  
en/html/Unicode-HOWTO.html。

# X 窗口系统

X 窗口系统 (X11) 是 UNIX 中图形用户界面的实际标准。X 是基于网络的，可以使在一个主机上启动的应用程序显示在通过任何类型的网络（LAN 或 Internet）连接的另一个主机上。

本章介绍在 SUSE LINUX 中所用字体的设置、优化选项、背景信息，以及 OpenGL 和 3D 的配置。

11.1	使用 SaX2 设置 X11 . . . . .	186
11.2	优化 X 配置 . . . . .	194
11.3	安装和配置字体 . . . . .	199
11.4	OpenGL — 3D 配置 . . . . .	204

## 11.1 使用 SaX2 设置 X11

图形用户界面（即 X 服务器）处理硬件和软件之间的通讯。桌面（例如 KDE 和 GNOME）和多种窗口管理器都使用 X 服务器与用户进行交互。

在安装期间对图形用户界面进行了初始配置。要在以后更改这些设置，请运行 SaX2。这样将保存当前设置，您以后可随时重设置为这些设置。此时将显示当前值供修改：屏幕分辨率、颜色深度、刷新率以及您的监视器的厂商和类型（如果已自动检测出）。

如果刚安装了新的图形卡，则会出现一个小对话框，询问您是否激活该图形卡的 3D 加速。单击‘编辑’。此时将在一个单独的窗口中启动输入和显示设备的配置工具 SaX2。此窗口如图 11.1 “SaX2 的主窗口” 所示。

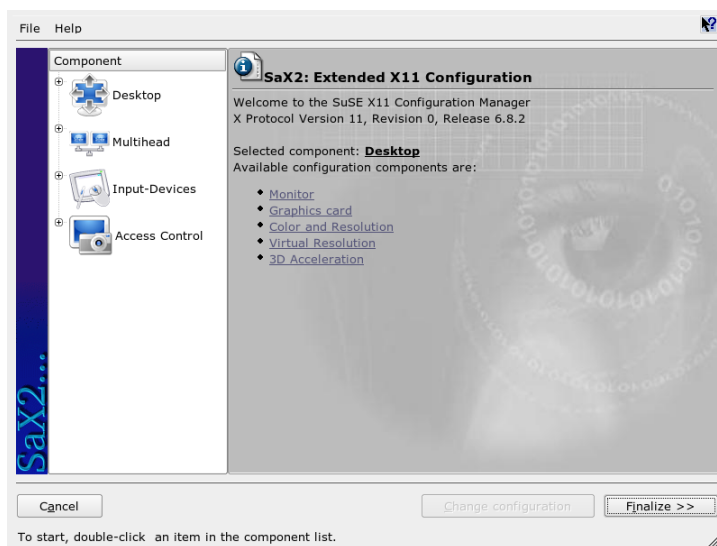


Figure 11.1: SaX2 的主窗口

在左侧导航栏中，有以下 4 个主要项目：‘桌面’、‘多头’、‘输入设备’和‘AccessX’。在‘桌面’下可配置您的监视器、图形卡、颜色深度、分辨率和屏幕的位置和大小。在‘输入设备’下可配置键盘、鼠标、触摸屏监视器和图形输入板。使用‘多头’可配置多个屏幕（请参见第 11.1.7 节“多头”）。‘AccessX’是一种使用数字小键盘上的键来控制鼠标指针的有用工具。

选择所用监视器和图形卡。通常，系统会自动检测监视器和图形卡。如果未自动检测到您的监视器，则将自动转到监视器选择对话框。请从厂商和设备列表中选择您的监视器，或手工输入监视器手册中指定的监视器值。或者，选择预配置的 VESA 方式之一。

在完成对监视器和图形卡的设置后，在主窗口中单击‘完成’，然后测试您的设置。这可以确保您的配置适合自己的设备。如果图像不稳定，请立即按 **(Esc)** 键终止测试，然后降低刷新率或分辨率和颜色深度。无论您是否运行测试，所有修改均在您重新启动 X 服务器后才会被激活。

### 11.1.1 桌面

选择‘编辑配置’→‘属性’后，将出现一个窗口，其中包含‘监视器’、‘频率’和‘专家’选项卡。

‘监视器’在窗口的左侧选择厂商。在右侧选择型号。如果您有监视器的 Linux 驱动程序光盘，请单击‘制造商磁盘’来安装它们。

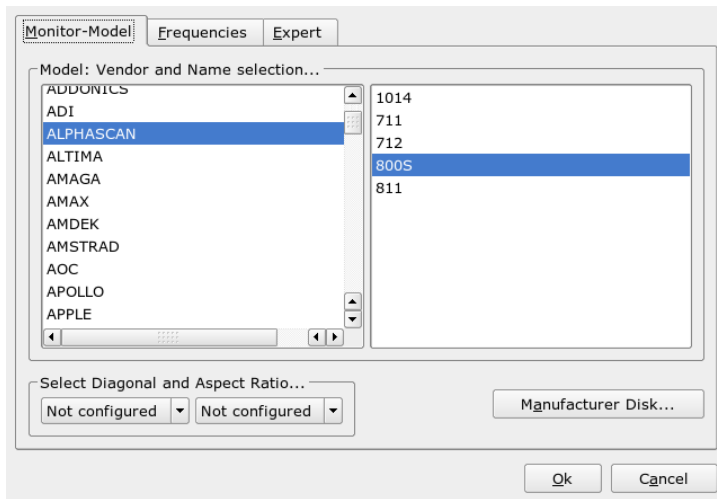


Figure 11.2: 监视器选择

‘频率’在此输入屏幕的水平和垂直频率。垂直频率是图像刷新率的另一个名称。通常，从型号读取可接受的值范围并在此输入。一般不需要进行修改。

‘专家’ 在此输入您的屏幕的一些选项。在上部的选择字段中，定义用于计算屏幕分辨率和屏幕几何属性的方法。除非监视器寻址错误且显示不稳定，否则不要更改这里的任何设置。此外，您可以更改所显示图像的大小并激活省电方式 DPMS。

## Warning

### 配置监视器频率

虽然提供有安全机制，但在手工更改所允许的频率时仍要非常小心。不正确的值可能损坏监视器。如果存在疑问，请参考监视器的手册。

Warning

## 11.1.2 图形卡

图形卡对话框有两个选项卡：‘常规’和‘专家’。在‘常规’选项卡中，可在左侧选择图形卡的厂商，在右侧选择其型号。

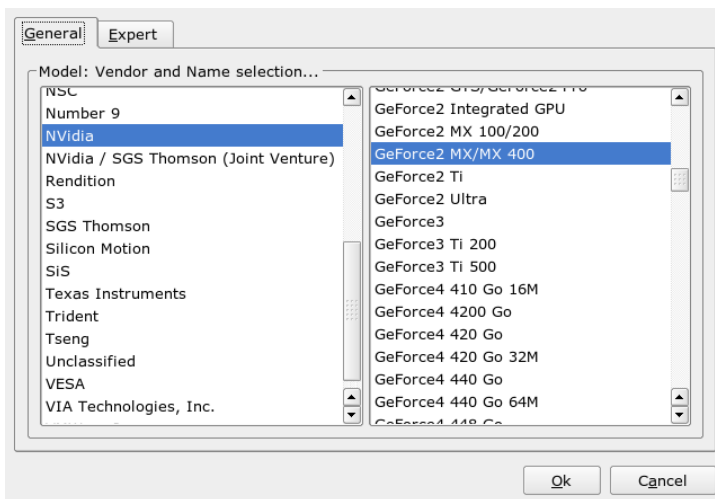


Figure 11.3: 选择图形卡

‘专家’选项卡提供了更高级的配置功能。在右侧，将您的屏幕向左旋转或旋转到垂直位置（适用于某些可旋转的 TFT 屏幕）。只有当您操作多个屏幕时才需要

使用有关 BusID 的项。通常无需在这里进行任何更改。除非您有这方面的经验并清楚这些选项的作用，否则切勿修改这些卡选项。如有必要，请查看图形卡的文档。

### 11.1.3 颜色和分辨率

这里提供了 3 个选项卡：‘颜色’、‘分辨率’和‘专家’。

‘颜色’ 根据所使用的硬件，选择颜色深度 16、256、32768、65536 或 16.7 百万色（4、8、15、16 或 24 位）。要实现可接受的显示质量，至少应设置 256 色。

‘分辨率’ 此模块提供了您的硬件可以正确显示的所有分辨率和颜色深度组合。这样就极大降低了 SUSE LINUX 中错误设置损坏您的硬件的危险。如果手工更改分辨率，请参考硬件文档以确保监视器能够使用您设置的值进行显示。

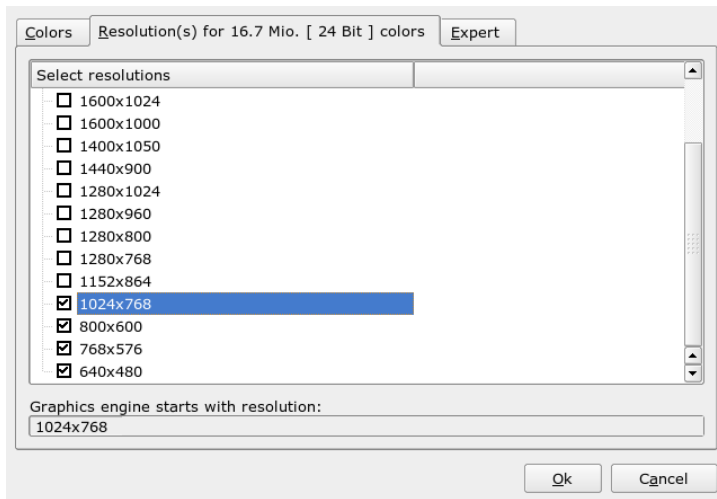


Figure 11.4: 配置分辨率

‘专家’ 除了上一个选项卡中提供的分辨率之外，还可以使用此选项卡添加您自己的分辨率，这些分辨率随后将包括在选项卡中以供选择。

### 11.1.4 虚拟分辨率

每个桌面都有一个特定的监视器全屏显示分辨率。此外，还可以设置一个大于屏幕可见区域的分辨率。如果将鼠标移到桌面的边缘之外，则会在屏幕上显示桌面的虚拟部分。这样就增加了可用工作空间。

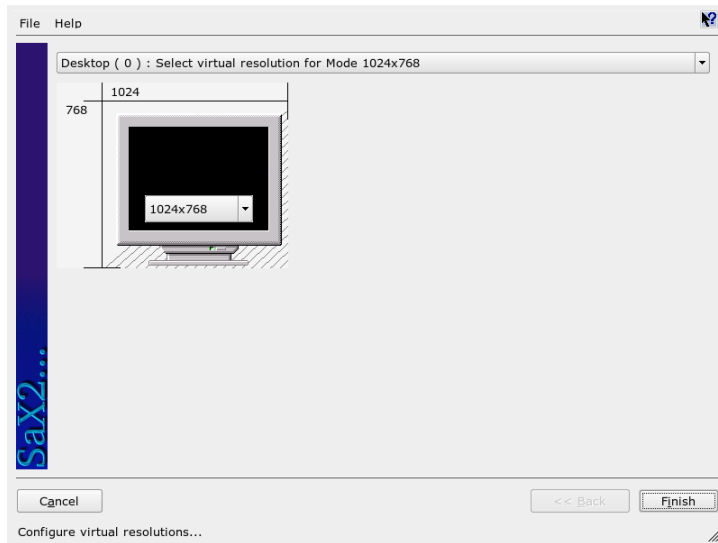


Figure 11.5: 配置虚拟分辨率

可以通过两种不同的方法来设置虚拟分辨率。要使用‘通过拖放’进行设置，请将鼠标指针在监视器图像上移动直到变为十字线。在按住鼠标左键的同时移动鼠标来放大对应于虚拟分辨率的光栅图像。如果对桌面上需要多少虚拟空间不是很确定，则最好采用此方法。

如果使用‘通过从弹出菜单中选择’进行设置，则光栅图像中间的弹出菜单将显示当前使用的虚拟分辨率。要使用默认虚拟分辨率之一，从菜单中选择一个即可。

### 11.1.5 3D 加速

如果在最初的安装期间未激活 3D 加速或安装了新的图形卡，则可以在此处执行此操作。



### 11.1.6 图像位置和大小

在这两个选项卡下，可使用箭头键精确调整图像的大小和位置。请参见图11.6“调整图像几何属性”。如果是多头环境（有多个屏幕），请使用‘下一个屏幕’切换到其他监视器来调整其大小和位置。按‘保存’即可保存您的设置。

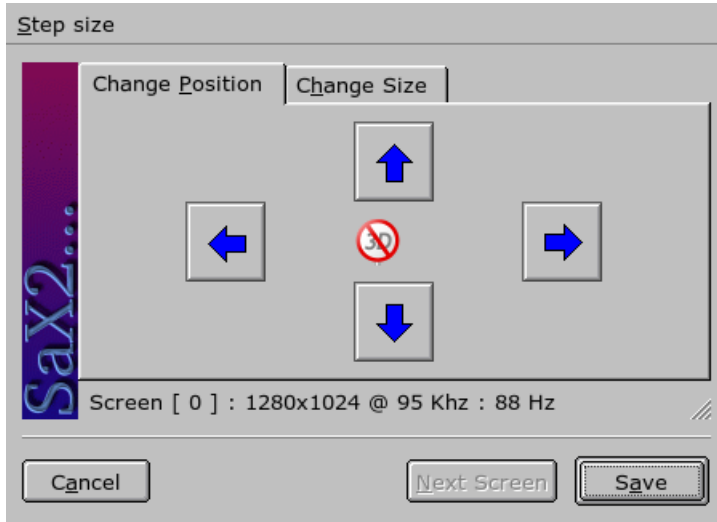


Figure 11.6: 调整图像几何属性

### 11.1.7 多头

如果在计算机中安装了多个图形卡，或安装了一个具有多个输出的图形卡，则可以为系统连接多个屏幕。如果使用两个屏幕，则称为双头。两个以上则称为多头。SaX2可自动检测到系统中的多个图形卡并相应地准备配置。在多头对话框中可以设置多头方式和屏幕排列。有三种方式可供选择：‘传统’（默认）、‘单屏(Xinerama)’和‘克隆方式’。

**传统多头** 每个监视器都代表一个单独的单元。鼠标指针可以在这些屏幕间切换。

**克隆多头** 在此方式下，所有监视器均显示相同的内容。鼠标仅在主屏幕上可见。

**Xinerama 多头** 所有屏幕组合起来构成一个大屏幕。可以将程序窗口放在所有屏幕上，也可以调整为填满多个监视器的某一大小。

多头环境的布局描述了各个屏幕的排列以及它们之间的关系。默认情况下，SaX2 将配置符合所检测出的图形卡的顺序的标准布局，将所有屏幕从左到右排列成一行。在多头工具的‘布局’对话框中，可以通过使用鼠标在网格中移动屏幕符号来确定监视器的排列方式。完成布局对话框后，请单击‘测试’来校验新配置。

Linux 目前尚未提供对 Xinerama 多头环境的 3D 支持。在这种情况下，SaX2 将取消 3D 支持。

### 11.1.8 输入设备

**鼠标** 如果自动检测失败，则可以使用此对话框来手工配置鼠标。请参考鼠标文档查看其型号。从受支持的鼠标类型列表中选择所用鼠标的型号，然后在数字小键盘上按 **⑤** 确认。

**键盘** 使用此对话框顶部的选择字段来指定要使用的键盘的种类。然后选择键盘布局（国家/地区特定的按键位置）的语言。使用测试字段来检查是否能正确显示特殊字符。

用于激活和取消重音字母输入的复选框的状态取决于相应的语言，不需要进行更改。单击‘完成’即可将新设置应用于您的系统。

**触摸屏** 目前，X.Org 只支持 Microtouch 和 Elo TouchSystems 触摸屏。SaX2 只能自动检测监视器，而不能自动检测触摸屏。触摸屏被作为输入设备对待。

要配置触摸屏，请启动 SaX2 并选择‘输入设备’→‘触摸屏’。单击‘添加’添加触摸屏。单击‘完成’即可保存配置。无需对此配置进行测试。

触摸屏有很多选项，而且通常必须首先进行校准。遗憾的是，Linux 中没有执行此操作的常规工具。标准配置中包含触摸屏尺寸的合适默认值。通常，无需进行其他配置。

**图形输入板** 目前，X.Org 只支持数目有限的图形输入板。使用 SaX2 可配置连接到 USB 端口或串行端口的图形输入板。从配置的角度看，图形输入板是像鼠标一样的输入设备。

启动 SaX2 并选择‘输入设备’→‘图形输入板’。单击‘添加’，从出现的对话框中选择厂商，然后从选择列表中添加一个图形输入板。如果已连接了画笔或橡皮擦，则请选中右侧的复选框。如果将输入板连接到串行端口，请校验此端口。`/dev/ttyS0` 指第 1 个串行端口。`/dev/ttyS1` 指第 2 个串行端口。其他端口使用类似的表示法。单击‘完成’即可保存配置。

### 11.1.9 AccessX

如果计算机上未使用鼠标，请启动 SaX2 并激活 AccessX，以便能够使用数字小键盘上的按键来控制鼠标指针。表 11.1 “AccessX — 使用数字小键盘操作鼠标” 说明了不同按键的功能。使用滑块来设置按键时鼠标指针移动的速度。

Table 11.1: AccessX — 使用数字小键盘操作鼠标

键	说明
⊘	选择鼠标左键
⊗	选择鼠标中键
⊖	选择鼠标右键
⑤	调用上次选择的鼠标键的单击事件。如果未选择其他鼠标键，则预设鼠标左键。在此事件后，系统会将选择重设置为默认值。
⊕	与 ⑤ 类似，只不过这是一个双击事件
⓪	与 ⑤ 类似，只不过这是一个单击并按住事件
Ⓚ	释放先前使用 ⑪ 调用的单击并按住事件
⑦	将光标向左上方移动
⑧	将光标垂直向上移动
⑨	将光标向右上方移动
④	将光标向左移动
⑥	将光标向右移动
①	将光标向左下方移动
②	将光标垂直向下移动
③	将光标向右下方移动

### 11.1.10 参考信息

有关 X 窗口系统及其属性的详细信息，请参考 第 11 章 X 窗口系统。

### 11.1.11 游戏杆

使用此模块，可通过从所显示的列表中选择制造商和型号来配置游戏杆。使用“测试”来检查游戏杆是否能正确响应。测试对话框中显示 3 个图标来表示游戏杆的各模拟轴和 4 个标准按钮的标志。当您移动游戏杆或按按钮时，在测试对话框中应看到有反应。由于游戏杆通常被连接到声卡，您也可以从声卡配置访问此模块。

### 11.1.12 键盘布局选择

所需要的键盘布局通常与所选择的语言相匹配，但也可以自由选择而无需考虑是什么语言。使用测试字段来检查管道符 | 等特殊字符能否正确显示。

### 11.1.13 鼠标

使用此 YaST 模块来配置您的鼠标。在介绍鼠标的安装时已经说明了选择鼠标的过程，请参考第 1.5.3 节“鼠标”。

## 11.2 优化 X 配置

X.Org 是 X 窗口系统的开放源代码实施。X.Org Foundation 对 X.Org 进行了进一步开发，它还负责开发 X 窗口系统的新技术和标准。

要以最佳方式使用可用的硬件（包括鼠标、图形卡、监视器和键盘），可以手工优化配置。下面介绍这种优化的一些方面。有关配置 X 窗口系统的详细信息，请查看目录 `/usr/share/doc/packages/Xorg` 中的各个文件和 `man xorg.conf`。

### Warning

配置 X 窗口系统时要非常小心。在完成配置前，切勿启动 X 窗口系统。错误配置的系统可能会对您的硬件造成不可修复的损坏（对于固定频率的监视器来说尤其如此）。本书的作者和 SUSE LINUX AG 不对这种损坏负任何责任。这里提供的信息已经仔细斟酌，但不能保证所提供的所有方法均正确且不会对您的硬件造成任何损坏。

Warning

默认情况下，程序 SaX2 和 xf86config 在 `/etc/X11` 中创建文件 `xorg.conf`。这是 X 窗口系统的主配置文件。请在此查找与图形卡、鼠标和监视器有关的所有设置。

下面几段介绍配置文件 `/etc/X11/xorg.conf` 的结构。每个部分都以关键字 `Section <designation>` 开头，以 `EndSection` 结尾。下面大致介绍最重要部分。

`xorg.conf` 由多个部分组成，每个部分处理配置的某个特定方面。各部分始终采用相同的形式：

```
Section designation
    entry 1
    entry 2
    entry n
EndSection
```

表11.2 “`/etc/X11/xorg.conf` 中的部分” 中列出了可用的部分类型。

*Table 11.2: /etc/X11/xorg.conf 中的部分*

类型	含义
Files	此部分描述用于字体和 RGB 颜色表的路径。
ServerFlags	在这里设置通用开关。
InputDevice	此部分配置输入设备，如键盘和特殊输入设备（触摸板、游戏杆等）。此部分的重要参数有 <code>Driver</code> 以及定义 <code>Protocol</code> 和 <code>Device</code> 的选项。
监视器	描述所用的监视器。此部分的各个元素是名称（稍后在 <code>Screen</code> 定义中引用）、 <code>bandwidth</code> 和同步频率限制（ <code>HorizSync</code> 和 <code>VertRefresh</code> ）。这些设置采用的单位为 MHz、kHz 和 Hz。通常，服务器拒绝不符合监视器规格的任何方式行。这样可防止意外地将过高的频率发送到监视器。
Modes	这里储存了特定屏幕分辨率的方式行参数。可以根据用户给出的值由 SaX2 计算出这些参数，并且通常无需更改这些参数。您可以在此时进行手工干预，例如当要连接固定频率监视器时。HOWTO 文件 <code>/usr/share/doc/howto/en/XFree86-Video-Timings-HOWTO.gz</code> 中提供了各个数字值含义的详细信息。

Device	此部分定义特定的图形卡。系统通过图形卡的描述性名称对其进行引用。
Screen	此部分将 Monitor 和 Device 放在一起以组成 X.Org 的所有必要设置。在 Display 子部分中，指定虚拟屏幕 (Virtual) 的大小、ViewPort 以及此屏幕所用的 Modes。
ServerLayout	此部分定义单头配置或多头配置的布局。此部分将输入设备 InputDevice 和显示设备 Screen 绑定在一起。

---

下面详细介绍 Monitor、Device 和 Screen。X.Org 和 xorg.conf 的手册页提供了有关其他部分的详细信息。

xorg.conf 中可以存在多个不同的 Monitor 和 Device 部分。甚至可以存在多个 Screen 部分。随后的 ServerLayout 部分确定使用哪个部分。

## 11.2.1 Screen 部分

首先深入研究一下 Screen 部分，此部分将 Monitor 部分与 Device 部分结合起来并确定要使用的分辨率和颜色深度。Screen 部分与例11.1 “文件 /etc/X11/xorg.conf 的 Screen 部分” 类似。

### 11.1: 文件 /etc/X11/xorg.conf 的 Screen 部分

```
Section "Screen"
    DefaultDepth 16
    SubSection "Display"
        Depth 16
        Modes "1152x864" "1024x768" "800x600"
        Virtual 1152x864
    EndSubSection
    SubSection "Display"
        Depth 24
        Modes "1280x1024"
    EndSubSection
    SubSection "Display"
        Depth 32
        Modes "640x480"
    EndSubSection
    SubSection "Display"
        Depth 8
        Modes "1280x1024"
    EndSubSection
    Device "Device[0]"
```

```
Identifier    "Screen[0]"
Monitor      "Monitor[0]"
EndSection
```

行 `Identifier` (这里是 `Screen[0]`) 为此部分指定一个定义的名称, 在随后的 `ServerLayout` 部分中可以使用此名称唯一引用这个部分。行 `Device` 和 `Monitor` 指定属于此定义的图形卡和监视器。这些行仅仅是通过 `Device` 和 `Monitor` 部分的相应名称或标识符指向这些部分的链接。下面详细讨论这些部分。

除非启动服务器时就使用特定的颜色深度, 否则请使用 `DefaultDepth` 设置来选择服务器应使用的颜色深度。每个颜色深度都有一个 `Display` 子部分。关键字 `Depth` 指派对此子部分有效的颜色深度。`Depth` 的可能值包括 8、15、16 和 24。并非所有 X 服务器模块都支持所有这些值。

在颜色深度后, 在 `Modes` 部分中设置分辨率列表。X 服务器从左到右检查此列表。对于每个分辨率, X 服务器均会在 `Modes` 部分中搜索合适的 `Modeline`。`Modeline` 取决于监视器和图形卡的功能。`Monitor` 设置确定最终的 `Modeline`。

找到的第一个分辨率是 `Default mode`。使用 `(Ctrl)-(Alt)+(` (在数字小键盘上) 向右切换到列表中的下一个分辨率。使用 `(Ctrl)-(Alt)-` (在数字小键盘上) 向左切换。这使您能够在 X 正在运行时改变分辨率。

包含 `Depth 16` 的 `Display` 子部分的最后一行指出了虚拟屏幕的大小。虚拟屏幕的最大可能大小取决于图形卡中安装的内存量和所需的颜色深度, 而不取决于监视器的最大分辨率。因为目前的图形卡都具有大量视频内存, 所以您可以创建非常大的虚拟桌面。但是, 如果您将大部分视频内存用于虚拟桌面, 则可能不能再使用 3D 功能了。例如, 如果图形卡有 16 MB 视频内存, 则当采用 8 位颜色深度时, 虚拟屏幕最多可以有 4096x4096 个像素。但建议不要将所有内存用于虚拟屏幕, 因为图形卡上的内存还要用于多种字体和图形超速缓存, 对于加速卡而言尤其如此。

## 11.2.2 Device 部分

`Device` 部分描述特定的图形卡。您可以在 `xorg.conf` 中包含任意多个设备项, 只要使用关键字 `Identifier` 对这些项的名称进行区分即可。如果您安装了多个图形卡, 通常按顺序对这些部分进行编号。第一个设备称为 `Device[0]`, 第二个设备称为 `Device[1]`, 依此类推。以下文件是从安装有 `Matrox Millennium PCI` 图形卡的计算机的 `Device` 部分摘出的一段:

```
Section "Device"
    BoardName      "MGA2064W"
    BusID          "0:19:0"
    Driver         "mga"
    Identifier     "Device[0]"
    VendorName     "Matrox"
    Option         "sw_cursor"
EndSection
```

如果您使用 SaX2 进行配置，则 Device 部分应与上面的示例类似。Driver 和 BusID 均取决于计算机上安装的硬件并由 SaX2 自动检测。BusID 定义安装图形卡的 PCI 或 AGP 插槽。它与使用命令 `lspci` 显示的 ID 相匹配。X 服务器需要采用十进制形式的详细信息，但 `lspci` 以十六进制形式显示这些信息。

通过 Driver 参数，指定要用于此图形卡的驱动程序。如果此卡是 Matrox Millennium，则将驱动程序模块称为 `mga`。然后，X 服务器通过 `drivers` 子目录的 Files 部分中定义的 ModulePath 进行搜索。在标准安装中，这是目录 `/usr/X11R6/lib/modules/drivers`。然后将 `_drv.o` 添加到名称中。因此，对于 `mga` 驱动程序，将装载驱动程序文件 `mga_drv.o`。

还可以通过其他选项影响 X 服务器或驱动程序的行为。在 Device 部分中设置的选项 `sw_cursor` 就是这方面的一个示例。此选项取消激活硬件鼠标光标并使用软件显示鼠标光标。根据驱动程序模块，有不同的选项可用，它们位于目录 `/usr/X11R6/lib/X11/doc` 中驱动程序模块的说明文件中。通常还可以在手册页（`man xorg.conf` 和 `man X.Org`）中查看有效的选项。

### 11.2.3 Monitor 部分和 Modes 部分

与 Device 部分类似，Monitor 和 Modes 部分分别描述一个监视器。配置文件 `/etc/X11/xorg.conf` 可以包含任意多个 Monitor 部分。服务器布局部分指定相关的 Monitor 部分。

只有有经验的用户才可以设置监视器定义。方式行组成了 Monitor 部分的重要组成部分。方式行设置相应分辨率的水平定时和垂直定时。Monitor 部分储存有监视器属性（特别是所允许的频率）。

#### Warning

除非您对监视器和图形卡功能有深入了解，否则不要更改方式行中的任何内容，因为这可能会对监视器造成严重损坏。

Warning



如果您要创建自己的监视器描述，则应非常熟悉 `/usr/X11/lib/X11/doc` 中的文档。介绍视频方式的内容非常值得专门阅读一下。它详细介绍了硬件的工作原理和如何创建方式行。

现在，很少需要手工指定方式行。如果您使用的是最新的多频同步监视器，则通常由 X 服务器通过 DDC 直接从监视器中读取允许的频率和最佳分辨率，如 SaX2 配置一节所述。如果由于某种原因无法执行此操作，请使用 X 服务器附带的 VESA 方式之一。这种方式可用于几乎所有图形卡和监视器组合。

## 11.3 安装和配置字体

在 SUSE LINUX 中安装附加字体非常简单。只需要将字体复制到位于 X11 字体路径中的任何目录即可（请参见第 11.3.2 节“X11 核心字体”）。为了使这些字体能够在新的 xft 字体显示系统下使用，安装目录应该是 `/etc/fonts/fonts.conf` 中配置的目录的子目录（请参见第 11.3.1 节“Xft”）。

可以（以 root 用户身份）将字体文件手工复制到适当的目录中，如 `/usr/X11R6/lib/X11/fonts/truetype`。或者，可以使用 KDE 控制中心中的 KDE 字体安装程序来执行此任务。结果是一样的。

您还可以创建符号链接，而不复制实际字体。例如，如果已装入的 Windows 分区上有许可字体并希望使用它们，则可以创建符号链接。随后，运行 `SuSEconfig --module fonts`。

`SuSEconfig --module fonts` 执行脚本 `/usr/sbin/fonts-config`，此脚本处理字体的配置。要了解此脚本执行的操作，请参考此脚本的手册页 (`man fonts-config`)。

上面的过程同样适用于位图字体、TrueType 和 OpenType 字体以及 Type1 (PostScript) 字体。可以将所有这些字体类型安装在任何目录中。只有 CID-keyed 字体的处理过程稍有不同。有关详细信息，请参见第 11.3.3 节“CID-Keyed 字体”。

X.Org 包含两个完全不同的字体系统：较早的 X11 核心字体系统和新设计的 *Xft* 和 *fontconfig* 系统。下面几节简要介绍这两种系统。

### 11.3.1 Xft

从一开始，Xft 的编程人员就确保该系统可以很好地支持可缩放字体（包括反锯齿处理）。如果使用 Xft，则是由使用字体的应用程序显示字体，而不是像 X11 核心字体系统中由 X 服务器显示字体。采用这种方式，相应的应用程序能够访

问实际字体文件并完全控制如何显示字形。这就为正确显示多种语言的文本奠定了基础。直接访问字体文件对于用于打印的嵌入字体非常有用，因为这样可以确保打印输出与屏幕输出看上去完全一样。

在 SUSE LINUX 中，两个桌面环境 KDE 和 GNOME、Mozilla 和许多其他应用程序均已默认使用 Xft。使用 Xft 的应用程序在数目上已经超过了使用以前的 X11 核心字体系统的应用程序。

Xft 使用 fontconfig 库来查找字体并影响字体的显示方式。fontconfig 的属性由全局配置文件 `/etc/fonts/fonts.conf` 和用户特定的配置文件 `~/.fonts.conf` 来控制。所有这些 fontconfig 配置文件的开头必须是

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
```

并且结尾必须是

```
</fontconfig>
```

要添加用于搜索字体的目录，请追加类似下面内容的一行：

```
<dir>/usr/local/share/fonts/</dir>
```

但通常没有必要这样做。默认情况下，已经在 `/etc/fonts/fonts.conf` 中输入了用户特定的目录 `~/.fonts`。因此，要安装附加字体，只需将它们复制到 `~/.fonts` 即可。

您还可以插入用来确定字体外观的规则。例如，输入

```
<match target="font">
  <edit name="antialias" mode="assign">
    <bool>>false</bool>
  </edit>
</match>
```

来禁用所有字体的反锯齿处理，或输入

```
<match target="font">
  <test name="family">
    <string>Luxi Mono</string>
    <string>Luxi Sans</string>
  </test>
  <edit name="antialias" mode="assign">
    <bool>>false</bool>
  </edit>
</match>
```

来禁用特定字体的反锯齿处理。

默认情况下，大多数应用程序使用字体名称 `sans-serif`（或等效的 `sans`）、`serif` 或 `monospace`。它们不是真正的字体，而只是可解析为合适的字体（取决于语言设置）的别名。

用户可以方便地将规则添加到 `~/.fonts.conf` 中，以将这些别名解析为他们喜欢的字体：

```
<alias>
  <family>sans-serif</family>
  <prefer>
    <family>FreeSans</family>
  </prefer>
</alias>
<alias>
  <family>serif</family>
  <prefer>
    <family>FreeSerif</family>
  </prefer>
</alias>
<alias>
  <family>monospace</family>
  <prefer>
    <family>FreeMono</family>
  </prefer>
</alias>
```

因为几乎所有应用程序都默认使用这些别名，所以这几乎影响到整个系统。这样，您可以方便地在几乎所有位置都使用自己喜欢的字体，而无需在各个应用程序中修改字体设置。

使用 `fc-list` 命令可以查看已安装了哪些字体以及哪些字体可用。例如，`fc-list ""` 命令返回所有字体的列表。要查看可用的可缩放字体（`:outline=true`）中有哪些包含希伯来语（`:lang=he`）所需的所有字形、它们的字体名称（`family`）、字型（`style`）、粗细（`weight`）以及包含这些字体的文件的名称，请输入以下命令：

```
fc-list ":lang=he:outline=true" family style weight file
```

此命令的输出如下所示：

```
/usr/X11R6/lib/X11/fonts/truetype/FreeSansBold.ttf: FreeSans:style=Bold:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeMonoBoldOblique.ttf: FreeMono:style=BoldOblique:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeSerif.ttf: FreeSerif:style=Medium:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSerifBoldItalic.ttf: FreeSerif:style=BoldItalic:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeSansOblique.ttf: FreeSans:style=Oblique:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSerifItalic.ttf: FreeSerif:style=Italic:weight=80
```

```

/usr/X11R6/lib/X11/fonts/truetype/FreeMonoOblique.ttf: FreeMono:style=Oblique:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeMono.ttf: FreeMono:style=Medium:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSans.ttf: FreeSans:style=Medium:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSerifBold.ttf: FreeSerif:style=Bold:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeSansBoldOblique.ttf: FreeSans:style=BoldOblique:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeMonoBold.ttf: FreeMono:style=Bold:weight=200

```

可以使用 `fc-list` 查询的重要参数包括：

*Table 11.3: fc-list 的参数*

参数	含义和可能值
family	字体系列的名称，如 FreeSans。
foundry	字体的制造商，如 urw。
style	字型，如 Medium、Regular、Bold、Italic、Heavy。
lang	字体支持的语言，例如 de 表示德语，ja 表示日语，zh-TW 表示繁体中文，zh-CN 表示简体中文。
weight	字体粗细，例如 80 表示常规粗细，200 表示粗体。
slant	倾斜，通常 0 表示不倾斜，100 表示斜体。
file	包含字体的文件的名称。
outline	true 表示外框字体，false 表示其他字体。
scalable	true 表示可缩放字体，false 表示其他字体。
bitmap	true 表示位图字体，false 表示其他字体。
pixelsize	以像素为单位表示的字体大小。与 <code>fc-list</code> 一起使用时，此选项仅对位图字体有意义。

### 11.3.2 X11 核心字体

目前，X11 核心字体系统不仅支持位图字体，还支持可缩放字体，例如 Type1 字体、TrueType 和 OpenType 字体以及 CID-keyed 字体。此外，支持 Unicode 字体也有很长时间了。1987 年，为 X11R1 开发了 X11 核心字体系统，最初的目的是为了处理单色位图字体。上面提到的所有扩展都是在以后添加的。

X11 核心字体系统只在没有反锯齿处理和子像素显示的情况下支持可缩放字体，并且装载许多语言具有字形的大型可缩放字体可能需要较长的时间。使用 Unicode 字体的速度也较慢，并且需要较多内存。

X11 核心字体系统带有一些固有缺陷。它已经过时，而且不再能以有意义的方式实现扩展。虽然为了实现向后兼容而不得不保留 X11 核心字体系统，但应尽可能使用更先进的 Xft 和 fontconfig 系统。

为了执行相应的操作，X 服务器需要知道它可使用的字体以及在系统中的哪些位置可找到这些字体。这由 FontPath 变量来处理，该变量包含所有有效系统字体目录的路径。在其中每个目录中，一个名为 fonts.dir 的文件会列出此目录中的可用字体。FontPath 由 X 服务器在启动时生成。它将在配置文件 /etc/X11/xorg.conf 的每个 FontPath 项中搜索有效的 fonts.dir 文件。这些项位于 Files 部分。使用 `xset q` 可显示实际的 FontPath。运行时也可以使用 `xset` 更改该路径。要添加其他路径，请使用 `xset +fp <path>`。要去除不需要的路径，请使用 `xset -fp <path>`。

如果 X 服务器已经处于活动状态，则可以使用命令 `xset fp rehash` 使装入的目录中新安装的字体可用。通过 `SuSEconfig --module fonts` 执行此命令。因为命令 `xset` 需要访问正在运行的 X 服务器，所以只有当从可以访问正在运行的 X 服务器的 shell 启动 `SuSEconfig --module fonts` 时，此命令才能发挥作用。实现此目的最简单的方法是通过输入 `sux` 和根口令来获得 root 用户权限。`sux` 会将启动 X 服务器的用户的访问权限传送到根 shell。要检查是否正确安装了字体以及是否可以通过 X11 核心字体系统使用字体，请使用命令 `xlsfonts` 列出所有可用字体。

默认情况下，SUSE LINUX 使用 UTF-8 区域设置。因此，应首选 Unicode 字体（`xlsfonts` 输出中以 `iso10646-1` 结尾的字体名称）。可以使用 `xlsfonts | grep iso10646-1` 列出所有可用的 Unicode 字体。几乎所有在 SUSE LINUX 中可用的 Unicode 字体都至少包括欧洲语言所需的字形（以前编码为 `iso-8859-*`）。

### 11.3.3 CID-Keyed 字体

与其他字体类型不同的是，您不能将 CID-keyed 字体随意安装在任何目录中。必须将 CID-keyed 字体安装在 `/usr/share/ghostscript/Resource/CIDFont` 中。这与 Xft 和 fontconfig 无关，但对 Ghostscript 和 X11 核心字体系统来说则是必需的。

#### Tip

有关 X11 下的字体的详细信息，请参见 <http://www.xfree86.org/current/fonts.html>。

#### Tip

## 11.4 OpenGL — 3D 配置

### 11.4.1 硬件支持

SUSE LINUX 包含若干用于 3D 硬件支持的 OpenGL 驱动程序。表 11.4 “支持的 3D 硬件” 对此进行了概述。

*Table 11.4: 支持的 3D 硬件*

OpenGL 驱动程序	支持的硬件
nVidia	nVidia 芯片: 除 Riva 128(ZX) 之外的所有芯片
DRI	3Dfx Voodoo Banshee、 3Dfx Voodoo-3/4/5、 Intel i810/i815/i830M、 Intel 845G/852GM/855GM/865G、915、 Matrox G200/G400/G450/G550、 ATI Rage 128(Pro)/Radeon (最高 9250)

如果是首次使用 YaST 进行安装，只要 YaST 检测到 3D 支持，即可在安装期间激活 3D 加速。对于 nVidia 图形芯片，必须首先安装 nVidia 驱动程序。为此，应在 YOU (YaST 在线更新) 中选择 nVidia 驱动程序增补程序。受许可证限制，发布的版本中未包含 nVidia 驱动程序。

如果执行的是更新而不是全新安装，或者需要安装 3Dfx 附加图形适配器 (Voodoo Graphics 或 Voodoo-2)，配置 3D 硬件支持的过程将有所不同。这取决于使用哪种 OpenGL 驱动程序。下一节对此进行了详细说明。

### 11.4.2 OpenGL 驱动程序

使用 SaX2 可以轻松配置 OpenGL 驱动程序 nVidia 和 DRI。对于 nVidia 适配器，必须首先安装 nVidia 驱动程序。输入命令 `3Ddiag`，检查 nVidia 或 DRI 的配置是否正确。

出于安全考虑，只有属于 `video` 组的用户才有权访问 3D 硬件。因此，应确保所有本地用户都是该组的成员。否则，将对 OpenGL 应用程序使用运行较慢的 OpenGL 驱动程序的软件呈现后备程序。使用命令 `id` 检查当前用户是否属于 `video` 组。如果不属于，请使用 YaST 将该用户添加到该组中。

### 11.4.3 诊断工具 3Ddiag

诊断工具 3Ddiag 用于对 SUSE LINUX 中的 3D 配置进行校验。这是一个必须从终端启动的命令行工具。输入 `3Ddiag -h` 可列出 3Ddiag 的可能选项。

为校验 X.Org 配置，该工具会检查是否已安装 3D 支持所需的包，以及是否使用了正确的 OpenGL 库和 GLX 扩展。在收到已失败讯息时按 3Ddiag 的说明操作。如果一切正常，您只会看到屏幕上显示已完成讯息。

### 11.4.4 OpenGL 测试实用程序

使用 `glxinfo` 命令检查是否已激活 3D；若已激活，命令输出中应包含直接呈现：是这行文字。

要测试 OpenGL，`glxgears` 程序以及 `tuxracer` 和 `armagetron` 之类的游戏（程序包具有相同的名称）会十分有用。如果已激活 3D 支持，则应该可以在一台较新的计算机上顺利运行这些游戏。如果没有 3D 支持，这些游戏的运行速度会非常慢（类似幻灯片的演示效果）。使用 `glxinfo` 命令检查是否已激活 3D；若已激活，命令输出中应包含直接呈现：是这行文字。

### 11.4.5 查错

如果 OpenGL 3D 测试结果不佳（无法顺利运行游戏），请使用 3Ddiag 确保配置正确无误（已失败讯息）。如果纠正了这些错误也无济于事，或是未出现已失败讯息，请查看 X.Org 日志文件。

通常，您会发现 X.Org `/var/log/Xorg.0.log` 文件中有这样一行文字：DRI 已禁用。只有仔细检查该日志文件才能发现根本原因——而执行这项任务需要具备一定的经验。

这种情况下不存在任何配置错误，配置错误早已通过 3Ddiag 检测出来。因此，此时只能使用 DRI 驱动程序软件呈现后备程序，该程序不提供 3D 硬件支持。如果出现 OpenGL 表示错误或运行不稳定，也应该在禁用 3D 支持的情况下进行检测。使用 `SaX2` 可以完全禁用 3D 支持。

### 11.4.6 安装支持

除 DRI 驱动程序软件呈现后备程序之外，Linux 中的所有 OpenGL 驱动程序均处于开发阶段，因此均应视为试用驱动程序。这些驱动程序之所以包含在发布的版本中是因为 Linux 对 3D 硬件加速具有较高的要求。考虑到 OpenGL 驱动程

序的试用状态，SUSE 无法提供任何针对配置 3D 硬件加速的安装支持，也不能为相关问题提供任何进一步的帮助。图形用户界面（X 窗口系统）的基本配置不包含 3D 硬件加速配置。如果遇到 3D 硬件加速的问题，建议完全禁用 3D 支持。

### 11.4.7 其他联机文档

有关 DRI 的信息，请参考 `/usr/X11R6/lib/X11/doc/README.DRI (xorg-x11-doc)`。有关 nvidia 驱动程序安装的详细信息，请参见 <http://ftp.suse.com/pub/suse/i386/supplementary/X/nvidia-installer-HOWTO.html>。



# 打印机操作

本章提供有关操作打印机的一般信息并帮助确定在网络中操作打印机的合适解决方案。本文将重点介绍 CUPS 操作。详细查错部分概述了打印机操作中最常见的错误并介绍了避免这些错误相应的方法。

12.1	准备工作和其他考虑事项 . . . . .	208
12.2	打印系统工作流程 . . . . .	209
12.3	连接打印机的方法和协议 . . . . .	209
12.4	安装软件 . . . . .	210
12.5	配置打印机 . . . . .	210
12.6	为应用程序进行配置 . . . . .	215
12.7	SUSE LINUX 中的特殊功能 . . . . .	216
12.8	查错 . . . . .	221

## 12.1 准备工作和其他考虑事项

CUPS 是 SUSE LINUX 中的标准打印系统。CUPS 高度面向用户。在很多情况下，它与 LPRng 兼容或者可以相对方便地进行调整。出于兼容性的原因，LPRng 仅包括在 SUSE LINUX 中。

可以根据接口（例如 USB 或网络）以及打印机语言对打印机进行区分。当购买打印机时，确保打印机具有硬件所支持的接口和合适的打印机语言。可以按照以下三类打印机语言对打印机进行分类：

**PostScript 打印机** Linux 和 Unix 中的内部打印系统使用 PostScript 这种打印机语言生成并处理大部分打印作业。这种语言已经有很长的历史并且非常有效。如果打印机可以直接处理 PostScript 文档而不需要在打印系统中通过附加步骤转换这些文档，则可以降低可能出现的错误的数目。因为 PostScript 打印机购买许可证要花费大量的成本，所以购买这些打印机的花费比不带 PostScript 解释器的打印机要高得多。

**标准打印机（PCL 和 ESC/P 等语言）**

虽然这些打印机语言有相当长的历史，但它们仍在进行扩展以处理打印机中的新功能。对于已知的打印机语言，打印系统可以借助 Ghostscript 将 PostScript 作业转换为相应的打印机语言。这一处理阶段被称为解释。最有名的语言有 PCL（主要是 HP 打印机及其克隆产品使用）和 ESC/P（Epson 打印机使用）。这些打印机语言通常受 Linux 支持，可以生成相当好的打印效果。Linux 可能不能处理非常新以及非常特别的打印机的一些功能，原因是开放源代码的开发人员可能仍在开发这些功能的代码。除了 HP 开发的 `hpijs` 驱动程序之外，当前尚没有其他打印机制造商开发 Linux 驱动程序并在开发源代码许可证下将这些驱动程序提供给 Linux 经销商。这些打印机中大多数价格适中。

**专用打印机（通常是 GDI 打印机）** 通常，只有一个或几个 Windows 驱动程序可用于专用打印机。这些打印机不支持任何常用打印机语言，而且每次发布某个型号的新版本时，它们所使用的打印机语言都可能会更改。有关详细信息，请参见第 12.8.1 节“打印机没有标准打印机语言支持”。

在您购买新打印机之前，请参考以下资源以了解您要购买的打印机的支持情况：

- <http://cdb.suse.de/>— SUSE LINUX 打印机数据库
- <http://www.linuxprinting.org/>— LinuxPrinting.org 打印机数据库
- <http://www.cs.wisc.edu/~ghost/>— Ghostscript 万维网网页

- `/usr/share/doc/packages/ghostscript/catalog.devices`—包括的驱动程序

联机数据库总是显示最新的 Linux 支持状态。但是，Linux 分发只能集成生产时可用的驱动程序。因此，在最新的 SUSE LINUX 版本发布时，当前标为“完全支持”的打印机不一定具有此状态。这样，数据库不一定可以指出正确的状态，只是提供大致估计而已。

## 12.2 打印系统工作流程

用户创建一个打印作业。该打印作业包含有要打印的数据以及假脱机程序的信息，例如打印机的名称或打印机队列的名称，还可能包括过滤器的信息，例如打印机特定的选项。

每台打印机都有一个专用的打印机队列。假脱机程序储存着队列中的打印作业，直到所需打印机已做好接收数据的准备。打印机准备就绪后，假脱机程序通过过滤器和后端将数据发送到打印机。

过滤器将用户要打印的数据（ASCII、PostScript、PDF、JPEG 等）转换为打印机特定的数据（PostScript、PCL、ESC/P 等）。PPD 文件中描述了打印机的功能。PPD 文件包含打印机特定的选项以及在打印机上启用这些选项所需的参数。过滤器系统用于确保用户选择的选项被启用。

如果使用的是 PostScript 打印机，则过滤器系统将数据转换为打印机特定的 PostScript。这样做不需要打印机驱动程序。如果使用的是非 PostScript 打印机，则过滤器系统会使用 Ghostscript 将数据转换为打印机特定的数据。这样做需要一个适合您的打印机的 Ghostscript 打印机驱动程序。后端从过滤器接收打印机特定的数据，然后将其传递到打印机。

## 12.3 连接打印机的方法和协议

可以通过多种方法将打印机连接到系统。CUPS 打印系统的配置不能区分本地打印机和通过网络连接到系统的打印机。在 Linux 中，必须按照打印机制造商提供的手册中所说明的方法连接本地打印机。CUPS 支持串口、USB、并口和 SCSI 连接。有关打印机连接的更多信息，请阅读位于 <http://portal.suse.com> 的支持数据库中的文章 *CUPS in a Nutshell*。通过在搜索对话框中输入 *cups* 可以找到这篇文章。

## Warning

与计算机的电缆连接

当将打印机连接到计算机时，一定不要忘记操作期间只能插入或拔出 USB 设备。当更改其他类型的连接时，必须先关闭系统。

## Warning

## 12.4 安装软件

PPD (PostScript 打印机描述) 是描述属性 (例如, 分辨率) 和选项 (例如, 双面打印单位的可用性) 的计算机语言。这些描述对于使用 CUPS 中的各个打印机选项是必需的。如果没有 PPD 文件, 打印数据将被以“原始”状态转发到打印机, 通常这不是希望出现的情况。在 SUSE LINUX 安装过程中, 将预安装多个 PPD 文件, 使系统甚至可以使用不带 PostScript 支持的打印机。

要配置 PostScript 打印机, 最佳的方法是获得一个合适的 PPD 文件。包 `manufacturer-PPDs` 中提供许多 PPD 文件, 标准安装会自动安装此包。请参见第 12.7.4 节“多种包中的 PPD 文件”和第 12.8.2 节“没有合适的 PPD 文件可用于 PostScript 打印机”。

可以将新 PPD 文件储存在目录 `/usr/share/cups/model/` 中或使用 YaST 添加到打印系统中 (请参见第 12.5.1 节“手工配置”)。随后, 可以在安装过程中选择 PPD 文件。

如果打印机制造商要求您除修改配置文件之外安装整个软件包, 则一定要注意。首先, 这种安装将导致丢失 SUSE LINUX 提供的支持; 其次, 打印命令将以不同的方式工作, 系统可能不再能处理其他制造商的设备。出于此原因, 不建议安装制造商软件。

## 12.5 配置打印机

在将打印机连接到计算机并安装软件之后, 在系统中安装打印机。应该使用 SUSE LINUX 提供的工具执行此操作。因为 SUSE LINUX 非常注重安全性, 所以第三方工具常在安全方面受到限制, 所导致的复杂问题比带来的好处还要多。

## 12.5.1 本地打印机

如果在您登录时检测到未配置的本地打印机，则将启动 YaST 对其进行配置。这将使用与以下配置说明相同的对话框。

要配置打印机，请在 YaST 控制中心中选择‘硬件’→‘打印机’。这将打开打印机主配置窗口，窗口上部列出了检测到的设备。下部列出了目前已配置的所有队列。如果未检测到您的打印机，请手工配置它。

### Important

如果 YaST 控制中心中没有可用的‘打印机’项，则最可能的原因是未安装 `yast2-printer` 包。要解决此问题，请安装 `yast2-printer` 包并重新启动 YaST。

### Important

#### 自动配置

如果能够自动设置并行或 USB 端口并检测到所连接的打印机，则 YaST 能够自动配置打印机。打印机数据库还必须包含 YaST 在自动硬件检测期间检索到的打印机的 ID 字符串。如果硬件 ID 与型号指定不同，则手工选择型号。

为了确保一切工作正常，应使用 YaST 的打印测试功能对每项配置进行检查。YaST 测试页还提供了与被测试配置有关的重要信息。

#### 手工配置

如果没有满足自动配置的要求，或希望进行自定义设置，请手工配置打印机。根据自动检测的实现程度，以及在数据库中找到的有关打印机型号的信息的多少，YaST 也许能够自动确定正确的设置，或至少可以作出合理的预选择。

必须配置以下参数：

**硬件连接（端口）** 硬件连接的配置取决于 YaST 是否能够在硬件检测期间找到打印机。如果 YaST 能够自动检测到打印机型号，则可以认为打印机连接可以在硬件级别工作，而且不需要在此方面更改任何设置。如果 YaST 未能自动检测到所用打印机型号，则硬件级别上的连接就可能存在问题。在这种情况下，可能需要一些人工干预来配置连接。

**队列的名称** 在发布打印命令时需要使用队列名。此名称应相对简短，并应仅包含小写字母和数字。

**打印机型号和 PPD 文件** 所有打印机特定的参数（例如要使用的 Ghostscript 驱动程序和驱动程序的打印机过滤器参数）储存在 PPD（PostScript 打印机描述）文件中。有关 PPD 文件的详细信息，请参见第 12.4 节“安装软件”。

对于许多打印机型号来说，都有多个 PPD 文件可用，例如当多个 Ghostscript 驱动程序用于给定的型号时。当您选择制造商和型号时，YaST 会选择与打印机对应的 PPD 文件。如果所选型号有多个 PPD 文件可用，YaST 会默认采用其中之一（通常是标记为 *recommended* 的那个文件）。选择“编辑”后即可更改默认 PPD 文件。

对于非 PostScript 型号，所有打印机特定的数据都是由 Ghostscript 驱动程序生成的。因此，驱动程序配置是决定输出质量的最重要的因素。打印输出同时受所选择的 Ghostscript 驱动程序（PPD 文件）和为其指定的选项的影响。如有必要，可在选择“编辑”后更改其他选项（PPD 文件使这些选项可用）。

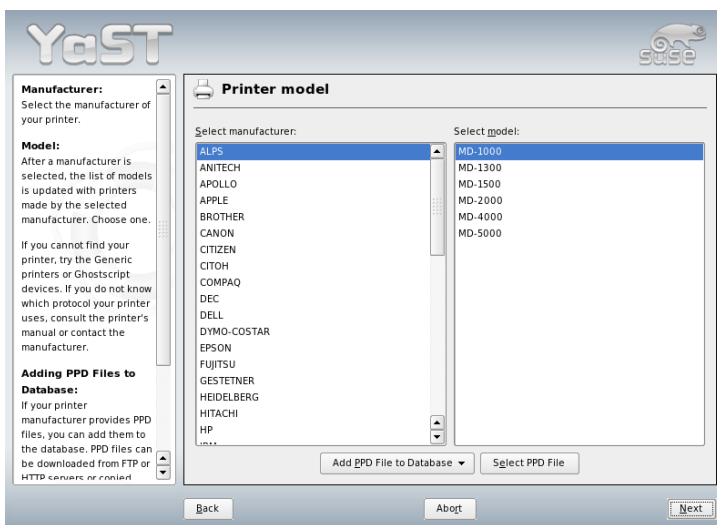


Figure 12.1: 选择打印机型号

应始终通过打印测试页来检查您的设置是否正常工作。如果输出不正常（例如有多页几乎是空白的），则应首先取出所有纸张，然后从 YaST 停止测试，这样便可以停止打印机。

如果打印机数据库不包括所用型号的项，您可以通过选择‘向数据库添加 PPD 文件’来添加新的 PPD 文件，也可以使用一组通用 PPD 文件来使打印机可以使用标准打印机语言之一。为此，请选择‘未知制造商’作为您的打印机制造商。

高级设置 通常无需更改这些设置。

### 使用命令行工具配置打印机

要使用命令行工具（如第 12.5.3 节“使用命令行工具进行配置”中所述）手工配置打印机，您需要一个设备 URI（统一资源标识符），URI 由一个后端（例如“usb”）和多个参数（例如“/dev/usb/lp1”）组成。例如，完整的 URI 可能是 `parallel:/dev/lp0`（连接到第一个并行端口的打印机）或 `usb:/dev/usb/lp1`（所检测到的第一个连接到 USB 端口的打印机）。

## 12.5.2 网络打印机

网络打印机可以支持多种协议，其中某些甚至是同时进行的。虽然大多数支持的协议是标准化的，但某些制造商因为测试尚未正确实施标准的系统或要提供标准中未提供的功能，所以对标准进行了扩展（修改）。于是制造商提供仅用于几个操作系统的驱动程序，解决使用这些系统遇到的困难。不过很少提供 Linux 驱动程序。当前的情况是您在执行操作时不能假定每个协议都可以在 Linux 中正常工作。因此，您可能需要试验不同的选项来实现工作正常的配置。

CUPS 支持 `socket`、`LPD`、`IPP` 和 `smb` 协议。下面是有关这些协议的详细信息：

**socket** 套接字是指未先执行数据握手就将数据发送到因特网套接字所使用的连接。一些常用的套接字端口号包括 9100 或 35。示例设备 URI 为 `socket://host-printer:9100/`。

**LPD**（行式打印机守护程序）RFC 1179 中对经过证明的 LPD 协议进行了介绍。在此协议下，在发送实际打印数据之前，将先发送一些与作业相关的数据，例如打印机队列的 ID。因此，在为数据传送配置 LPD 协议之前，必须指定打印机队列。不同打印机制造商的实施非常灵活，可以接受任何名称作为打印机队列。如果需要，打印机手册应该指出要使用的名称。通常使用 `LPT`、`LPT1`、`LP1` 或类似的名称。当然，可以在不同 Linux 或 Unix 主机的 CUPS 系统中配置 LPD 队列。LPD 服务的端口号是 515。示例设备 URI 为 `lpd://host-printer/LPT1`。

**IPP (因特网打印协议)** IPP 是一个基于 HTTP 协议的相对较新的(1999) 协议。使用 IPP, 所传送的与作业有关的数据比其他协议要多一些。CUPS 使用 IPP 进行内部数据传送。这是在两个 CUPS 服务器之间转发队列的首选协议。要正确配置 IPP, 必须提供打印队列的名称。IPP 的端口号是 631。示例设备 URI 是 `ipp://host-printer/ps` 和 `ipp://host-cupsserver/printers/ps`。

**SMB (Windows 共享)** CUPS 还支持在连接到 Windows 共享的打印机上进行打印。用于此目的的协议是 SMB。SMB 使用端口号 137、138 和 139。示例设备 URI 是 `smb://user:password@workgroup/server/printer`、`smb://user:password` 和 `smb://server/printer`。

必须在配置之前确定打印机支持的协议。如果制造商未提供所需的信息, 则可以使用命令 `nmap` (附带 `nmap` 包) 来猜测协议。`nmap` 检查主机是否有打开的端口。例如:

```
nmap -p 35,137-139,515,631,9100-10000 printerIP
```

### 12.5.3 配置任务

可以通过使用 YaST 或命令行工具执行配置任务。

#### 使用 YaST 在网络中配置 CUPS

应该使用 YaST 配置网络打印机。YaST 简化了配置工作, 并进行了良好的配备以处理 CUPS 中的安全限制 (请参见第 12.7.2 节 “通过 CUPS 万维网前端进行管理”)。

有关在网络中安装 CUPS 的指导, 请参见位于 <http://portal.suse.com> 的支持数据库中的文章 *CUPS in a Nutshell*。

#### 使用命令行工具进行配置

也可以使用 `lpadmin` 和 `lpoptions` 等命令行工具配置 CUPS。如果完成了准备工作 (如果您知道 PPD 文件和设备的名称), 则需要执行以下步骤:

```
lpadmin -p queue -v device-URI \ -P PPD-file -E
```

不要使用 `-E` 作为第一个选项。对于所有 CUPS 命令, 将 `-E` 用作第一个参数设置使用加密连接。要启用打印机, 必须使用 `-E`, 如下面的示例所示:



```
lpadmin -p ps -v parallel:/dev/lp0 -P \ /usr/share/cups/model/Postscript-ppd.gz -E
```

以下示例配置了网络打印机：

```
lpadmin -p ps -v socket://192.168.1.0:9100/ -P \ /usr/share/cups/model/Postscript-le
```

### 修改选项

在系统安装期间，某些选项被设置为默认值。可以为每个打印作业修改这些选项（根据所使用的打印工具）。也可以使用 YaST 来更改这些默认选项。使用命令行工具设置默认选项，如下所示：

1. 首先，列出所有选项：

```
lpoptions -p queue -l
```

示例：

```
Resolution/Output Resolution:150dpi *300dpi 600dpi
```

前面带有星号(\*)表明这是激活的默认选项。

2. 使用 lpadmin 更改选项：

```
lpadmin -p queue -o Resolution=600dpi
```

3. 检查新设置：

```
lpoptions -p queue -l
```

```
Resolution/Output Resolution:150dpi 300dpi *600dpi
```

## 12.6 为应用程序进行配置

应用程序与任何命令行工具一样依赖于现有的打印机队列。通常无需为某个特定应用程序重配置打印机，因为使用可用的队列就能够从应用程序进行打印。

## 12.6.1 从命令行打印

要从命令行打印，请输入命令 `lp -d <queuename> <filename>`，使用相应的名称替换 `<queuename>` 和 `<filename>`。

## 12.6.2 使用命令行工具从应用程序进行打印

有些应用程序依赖于 `lp` 命令来进行打印。在这种情况下，请在应用程序的打印对话框中输入正确的命令（通常无需指定 `<filename>`），例如 `lp -d <queuename>`。要使其可用于 KDE 程序，请启用‘通过外部程序打印’。否则就不能输入打印命令。

## 12.6.3 使用 CUPS 打印系统

像 `xpp` 这样的工具和 KDE 程序 `kprinter` 都提供了一个图形界面，用于选择队列以及设置 CUPS 标准选项和通过 PPD 文件可用的打印机特定的选项。您可以使用 `kprinter` 作为非 KDE 应用程序的标准打印接口，方法是指定 `kprinter` 或 `kprinter --stdin` 作为这些应用程序打印对话框中的打印命令。应用程序本身的行为决定要从这两个命令中选择哪一个。如果正确设置，应用程序无论何时发布打印作业均应调用 `kprinter` 对话框，这样您就可以使用此对话框来选择队列并设置其他打印选项了。这要求应用程序本身的打印设置不能与 `kprinter` 的打印设置冲突，而且在启用 `kprinter` 后，只能通过它来更改打印选项。

# 12.7 SUSE LINUX 中的特殊功能

已对 CUPS 的许多功能进行了调整以用于 SUSE LINUX。这里将介绍一些最重要的更改。

## 12.7.1 CUPS 服务器和防火墙

有多种方法可以将 CUPS 配置为网络服务器的客户机。

- 对于网络服务器上的每个队列，您都可以配置一个本地队列，通过该本地队列将所有作业转发到对应的网络服务器。通常，不建议使用此方法，原因是当网络服务器的配置更改时，必须重配置所有客户机。

- 也可以将打印作业直接转发到一个网络服务器。对于这种类型的配置，不要运行本地 CUPS 守护程序。lp 或其他程序的相应库调用可以将作业直接发送到网络服务器。但是，如果还要在本地打印机上打印，则此配置不起作用。
- CUPS 守护程序可以监听其他网络服务器发送以通知可用队列的 IPP 广播包。这是通过远程 CUPS 服务器进行打印的最佳 CUPS 配置。但是，这种方法存在的风险在于，如果攻击者向守护程序 IPP 广播发送队列，则本地守护程序将访问假冒的队列。如果它随后使用与本地服务器上的另一个队列相同的名称来显示该队列并且较早收到了 IPP 包，则作业的拥有者可能认为作业已被发送到本地服务器，而实际上作业被发送到攻击者的服务器上。要使用此方法，必须为进入的包打开端口 631/UDP。

通过扫描所有网络主机以查看其是否提供 CUPS 服务并监听 IPP 广播，YaST 可以找到 CUPS 服务器。第二种方法用于在系统安装期间为建议查找 CUPS 服务器。它要求对进来的包打开端口 631/UDP。

显示在建议对话框的防火墙默认设置拒绝在任何接口上执行 IPP 广播。因此，用来检测远程队列的第二种方法和用来访问远程队列的第三种方法不能工作。因此，必须通过将接口之一（该接口默认打开端口）标为 `internal` 或通过显式打开 `external` 接口的端口来修改防火墙配置。出于安全原因，这些端口都不会默认打开。使用第二种方法打开端口来配置对远程队列的访问将引起安全问题，因为用户可能会接受攻击者广播的服务器。

必须修改建议的防火墙配置以在安装过程中启用 CUPS 检测远程队列，并在正常操作过程中从本地系统访问远程服务器。或者，用户可以通过主动扫描本地网络主机检测 CUPS 服务器或手工配置所有队列。但是，由于上述原因，不建议使用此方法。

## 12.7.2 通过 CUPS 万维网前端进行管理

要通过万维网前端 (CUPS) 或打印机管理工具 (KDE) 进行管理，必须将 `root` 用户设置为 CUPS 管理员，并同时设置 CUPS 管理组 `sys` 和 CUPS 口令。使用以下命令以 `root` 用户身份执行此操作：

```
lppasswd -g sys -a root
```

如果未执行此操作，则不能使用万维网接口或管理工具进行管理，因为如果未配置 CUPS 管理员，鉴定将失败。除了 `root` 用户之外，也可以将任何其他用户指定为 CUPS 管理员（请参见第 12.7.3 节“CUPS 打印服务 (cupsd) 中的更改”）。

### 12.7.3 CUPS 打印服务 (cupsd) 中的更改

这些更改最初应用于 SUSE LINUX 9.1。

#### **cupsd** 以用户 **lp** 的身份运行

启动时，**cupsd** 从用户 **root** 更改为用户 **lp**。这提供了更高的安全性级别，因为 CUPS 打印服务不会以不受限制的权限运行，只以打印服务所需的权限运行。

但是，不能通过 `/etc/shadow` 执行鉴定（口令检查），因为 **lp** 没有对 `/etc/shadow` 的访问权限。而必须使用通过 `/etc/cups/passwd.md5` 执行的 CUPS 特定的鉴定。为此，必须在 `/etc/cups/passwd.md5` 中输入 CUPS 管理员以及 CUPS 管理组 **sys** 和 CUPS 口令。要执行此操作，必须以 **root** 用户身份输入以下内容：

```
lppasswd -g sys -a CUPS-admin-name
```

当 **cupsd** 以 **lp** 运行时，不能生成 `/etc/printcap`，因为不允许 **lp** 在 `/etc/` 中创建文件。因此，**cupsd** 生成 `/etc/cups/printcap`。为了确保只能从 `/etc/printcap` 读取队列名的应用程序继续正常工作，`/etc/printcap` 是指向 `/etc/cups/printcap` 的符号链接。

当 **cupsd** 以 **lp** 运行时，不能打开端口 631。因此，不能使用 `rccups reload` 重载 **cupsd**。请改用 `rccups restart`。

#### **BrowseAllow** 和 **BrowseDeny** 的一般功能

为 **BrowseAllow** 和 **BrowseDeny** 设置的访问权限适用于发送到 **cupsd** 的所有类型的包。`/etc/cups/cupsd.conf` 中的默认设置如下所示：

```
BrowseAllow @LOCAL BrowseDeny All
```

and

```
<Location /> Order Deny,Allow Deny From All Allow From 127.0.0.1 Allow From 1
```

这样，只有 **LOCAL** 主机可以访问 CUPS 服务器上的 **cupsd**。**LOCAL** 主机是其 IP 地址属于非 PPP 接口（未设置其 `IFF_POINTOPOINT` 标志的接口）并且其 IP 地址与 CUPS 服务器属于同一个网络的主机。将立即拒绝来自所有其他主机的包。

## 默认激活 cupsd

在标准安装中，将自动激活 cupsd，从而可以方便地访问 CUPS 网络服务器的队列，而无需执行任何其他手工操作。前两项（请参见第12.7.3节“cupsd 以用户 lp 的身份运行”和第12.7.3节“BrowseAllow 和 BrowseDeny 的一般功能”）是此功能的重要前提，因为如果不是这样，安全性将不足以自动激活 cupsd。

## 12.7.4 多种包中的 PPD 文件

### 仅使用 PPD 文件进行打印机配置

YaST 打印机配置仅使用系统上 `/usr/share/cups/model/` 中安装的 PPD 文件为 CUPS 设置队列。为确定用于某个打印机型号的合适的 PPD 文件，YaST 将在硬件检测过程中确定的供应商和型号与存在于系统上 `/usr/share/cups/model/` 中的所有 PPD 文件中的供应商和型号进行比较。为此，YaST 打印机配置根据从 PPD 文件抽取的供应商和型号信息生成一个数据库。当您从供应商和型号列表中选择打印机时，将收到符合该供应商和型号的 PPD 文件。

仅使用 PPD 文件而不使用其他信息源的配置的优点在于可以随意修改 `/usr/share/cups/model/` 中的 PPD 文件。YaST 打印机配置可以识别更改并重新生成供应商和型号数据库。例如，如果您具有 PostScript 打印机，通常您不需要 `cups-drivers` 包中的 Foomatic PPD 文件或 `cups-drivers-stp` 包中的 Gimp-Print PPD 文件。而可以将您的 PostScript 打印机的 PPD 文件直接复制到 `/usr/share/cups/model/`（如果它们尚不存在于 `manufacturer-PPDs` 包中）以实现打印机的最佳配置。

### cups 包中的 CUPS PPD 文件

为 PostScript 级别 1 和级别 2 打印机调整的 Foomatic PPD 文件对 cups 包中的通用 PPD 文件进行了补充：

- `/usr/share/cups/model/Postscript-level1.ppd.gz`
- `/usr/share/cups/model/Postscript-level2.ppd.gz`

### cups-drivers 包中的 PPD 文件

通常，Foomatic 打印机过滤器 `foomatic-rip` 与非 PostScript 打印机的 Ghostscript 一起使用。适用的 Foomatic PPD 文件应具有项 `*NickName: ... Foomatic/Ghostscript driver` 和 `*cupsFilter: ... foomatic-rip`。这些 PPD 文件位于 `cups-drivers` 包中。

如果具有项 `*NickName: ... Foomatic ... (recommended)` 的 Foomatic PPD 文件符合打印机型号并且 `manufacturer-PPDs` 包不包含更合适的 PPD 文件（请参见下文），则 YaST 倾向于使用 Foomatic PPD 文件。

### **cups-drivers-stp 包中的 Gimp-Print PPD 文件**

Gimp-Print 中的 CUPS 过滤器 `rastertoprinter`（而不是 `foomatic-rip`）可用于许多非 PostScript 打印机。`cups-drivers-stp` 包中提供此过滤器和合适的 Gimp-Print PPD 文件。Gimp-Print PPD 文件位于 `/usr/share/cups/model/stp/` 中并具有项 `*NickName: ... CUPS+Gimp-Print` 和 `*cupsFilter: ... rastertoprinter`。

### **manufacturer-PPDs 包中来自打印机制造商的 PPD 文件**

`manufacturer-PPDs` 包中包含来自打印机制造商的 PPD 文件，这些文件是在充分自由的许可证下发布的。应该用打印机制造商的合适 PPD 文件配置 PostScript 打印机，因为此文件支持使用 PostScript 打印机的所有功能。如果满足以下条件，YaST 倾向于使用 `manufacturer-PPDs` 包中的 PPD 文件：

- 硬件检测过程中确定的供应商和型号符合 `manufacturer-PPDs` 包的 PPD 文件中的供应商和型号。
- `manufacturer-PPDs` 包中的 PPD 文件是唯一适合该打印机型号的 PPD 文件，或者有一个具有 `*NickName: ... Foomatic/Postscript (recommended)` 项的 Foomatic PPD 文件，该项也符合该打印机型号。

因此，在以下情况下，YaST 不使用 `manufacturer-PPDs` 包中的任何 PPD 文件：

- `manufacturer-PPDs` 包中的 PPD 文件不符合供应商和型号。如果 `manufacturer-PPDs` 包只包含用于类似型号的 PPD 文件（例如，如果某个型号系列中的各个型号没有单独的 PPD 文件，而是在 PPD 文件中以类似于 `Funprinter 1000 series` 的形式指定型号名），则可能发生这种情况。
- 不建议使用 Foomatic PostScript PPD 文件。这可能是由于该打印机型号在 PostScript 方式中不能充分地操作。例如，因为打印机内存太少而导致它在这种方式中不可靠，或者因为处理器太弱而导致打印机速度太慢。此外，因为 PostScript 支持只作为可选模块提供，所以打印机可能不默认支持 PostScript。

如果 `manufacturer-PPDs` 包中的 PPD 文件适合 PostScript 打印机，但 YaST 由于上述原因不能对其进行配置，则在 YaST 中手工选择相应的打印机型号。

## 12.8 查错

下面几节介绍一些最常遇到的打印机硬件和软件问题以及解决或避免这些问题的方法。

### 12.8.1 打印机没有标准打印机语言支持

不支持任何常用打印机语言并且只能使用特殊控制序列处理的打印机被称为 *GDI 打印机*。这些打印机只能用于制造商提供了驱动程序的操作系统的版本。*GDI* 是 Microsoft 为图形设备开发的编程接口。实际问题不是编程接口，而是 *GDI* 打印机只能通过相应打印机型号的专用打印机语言进行处理。

某些打印机可进行切换以 *GDI* 方式或一种标准打印机语言进行操作。某些制造商为他们的 *GDI* 打印机提供专有驱动程序。专有打印机驱动程序的缺点在于不能保证这些驱动程序可用于已安装的打印系统，也不能保证它们适合各种硬件平台。相反，支持标准打印机语言的打印机不依赖于特殊的打印系统版本或特殊的硬件平台。

与其花时间使专有 Linux 驱动程序工作，不如购买一台支持的打印机，这样更经济一些。这可以一次性全部解决驱动程序问题，从而无需安装并配置特殊驱动程序软件，也无需获取由于打印系统中开发的新功能而必须安装的驱动程序更新。

### 12.8.2 没有合适的 PPD 文件可用于 PostScript 打印机

如果 `manufacturer-PPDs` 包不包含任何用于 PostScript 打印机的合适 PPD 文件，则可以使用打印机制造商提供的驱动程序 CD 上的 PPD 文件或从打印机制造商万维网网页下载合适的 PPD 文件。

如果以 zip 存档 (.zip) 或自解压缩 zip 存档 (.exe) 的形式提供 PPD 文件，则用 `unzip` 命令将其解包。首先，查看 PPD 文件的许可协议条款。然后，使用 `cupstestppd` 实用程序检查该 PPD 文件是否符合“Adobe PostScript 打印机描述文件格式规范（版本 4.3）”。如果该实用程序返回“FAIL”，则 PPD 文件中的错误非常严重，很可能导致严重问题。应该解决 `cupstestppd` 报告的问题点。如果需要，询问打印机制造商是否提供合适的 PPD 文件。

### 12.8.3 并行端口

最安全的方法是将打印机直接连接到第一个并行端口并在 BIOS 中选择以下并行端口设置：

- I/O 地址：378（十六进制）
- 中断：无关
- 方式：Normal、SPP 或 Output Only
- DMA：已禁用

如果即便进行了这些设置仍无法对并行端口上的打印机进行寻址，则按照 BIOS 中的设置在 `/etc/modprobe.conf` 中以 `0x378` 形式显式输入 I/O 地址。如果有两个并行端口，分别被设置为 I/O 地址 378 和 278（十六进制），则以 `0x378,0x278` 形式输入这两个端口。

如果中断 7 可用，则可以用例 12.1 “`/etc/modprobe.conf`：第一个并行端口的中断方式”中显示的项将其激活。在激活中断方式之前，检查文件 `/proc/interrupts` 看看哪些中断仍在使用中。只显示当前正在使用的中断。根据哪些硬件部件处于活动状态，这可能会有所变化。用于并行端口的中断一定不能以任何其他设备使用。如果您不确定，则使用巡回检测方式，设置 `irq=none`。

#### 12.1: `/etc/modprobe.conf`：第一个并行端口的中断方式

```
alias parport_lowlevel parport_pc options parport_pc io=0x378 irq=7
```

### 12.8.4 网络打印机连接

**确定网络问题** 将打印机直接连接到计算机。出于测试目的，将该打印机配置为本地打印机。如果打印机可以工作，则问题与网络有关。

**检查 TCP/IP 网络** TCP/IP 网络和名称解析必须可以正常工作。

**检查远程 lpd** 使用以下命令测试是否可以与 `<host>` 上的 lpd（端口 515）建立 TCP 连接：

```
netcat -z host 515 && echo ok || echo failed
```

如果不能建立与 lpd 的连接，则 lpd 可能不处于活动状态或可能存在基本网络问题。

以 root 用户身份使用以下命令查询远程 `<host>` 上 `<queue>` 的状态报告（可能非常长），前提是相应的 lpd 处于活动状态并且主机接受查询：



```
echo -e "\004queue" \ | netcat -w 2 -p 722 host 515
```

如果 `lpd` 不响应，则它可能不处于活动状态或可能存在基本网络问题。如果 `lpd` 响应，响应应该显示为什么在 `host` 的 `queue` 上不能进行打印。如果您接收到类似例12.2 “来自 `lpd` 的错误讯息” 中的响应，则问题是由远程 `lpd` 引起的。

### 12.2: 来自 `lpd` 的错误讯息

```
lpd: your host does not have line printer access lpd: queue does not exist print
```

检查远程 `cupsd` 默认情况下，CUPS 网络服务器应该每隔 30 秒在 UDP 端口 631 上广播其队列。因此，以下命令可用于测试网络中是否有 CUPS 网络服务器。

```
netcat -u -l -p 631 & PID=$!; sleep 40 ; kill $PID
```

如果广播 CUPS 网络服务器存在，则输出如例12.3 “来自 CUPS 网络服务器的广播” 所示。

### 12.3: 来自 CUPS 网络服务器的广播

```
ipp://host.domain:631/printers/queue
```

以下命令可用于测试是否可以与 `<host>` 上的 `cupsd` (端口 631) 建立 TCP 连接：

```
netcat -z host 631 && echo ok || echo failed
```

如果不能建立与 `cupsd` 的连接，则 `cupsd` 可能不处于活动状态或可能存在基本网络问题。`lpstat -h host -l -t` 返回 `<host>` 上所有队列的状态报告 (可能非常长)，前提是相应的 `cupsd` 处于活动状态并且主机接受查询。

下一个命令用于测试 `<host>` 上的 `<queue>` 是否接受由单个回车字符组成的打印作业。不应打印任何内容。可能会弹出一页空白纸。

```
echo -en "\r" \ | lp -d queue -h host
```

## 对网络打印机或打印服务器计算机进行查错

当在打印服务器计算机中运行的假脱机程序要处理大量打印作业时，有时会导致出现问题。因为这是由打印服务器计算机中的假脱机程序引起的，所以没什么办法。作为替代解决方法，可以通过直接通过 TCP 套接字对连接到打印服务器计算机的打印机进行寻址来绕过打印服务器计算机中的假脱机程序。请参见第 12.5.2 节“网络打印机”。

这样，打印服务器计算机仅用作数据传送（TCP/IP 网络和本地打印机连接）各种不同形式之间的转换器。要使用此方法，您需要知道打印服务器计算机上的 TCP 端口。如果打印机连接在打印服务器计算机上并且打开了电源，则通常可以在打开打印服务器计算机的电源后使用 nmap 包中的 nmap 实用程序确定此 TCP 端口。例如，nmap *<IP-address>* 可能会传递打印服务器计算机的以下输出：

```
Port      State      Service 23/tcp      open      telnet 80/tcp
```

此输出指出可以在端口 9100 上通过 TCP 套接字对连接到打印服务器计算机的打印机进行寻址。默认情况下，nmap 只检查在 /usr/share/nmap/nmap-services 中列出的一些常见的端口。要检查所有可能的端口，请使用命令 nmap -p *<from\_port>*-*<to\_port>* *<IP-address>*。这可能要花一些时间。有关详细信息，请参考 nmap 手册页。

输入如下命令

```
echo -en "\rHello\r\n" | netcat -w 1 IP-address port cat file | netcat -
```

将字符串或文件直接发送到相应的端口以测试是否可以在该端口上对打印机进行寻址。

## 12.8.5 打印件有问题但没有错误讯息

对于打印系统，打印作业完成的标志是 CUPS 后端完成到接收方（打印机）的数据传送。如果在接收方的进一步处理失败（例如，如果打印机无法打印打印机特定的数据），则打印系统不会对此进行通知。如果打印机无法打印打印机特定的数据，则选择另一个更适合该打印机的 PPD 文件。

## 12.8.6 禁用的队列

如果向接收方传送数据在多次尝试后都失败，则 CUPS 后端（例如 usb 或 socket）向打印系统（向 cupsd）报告一个错误。后端决定在报告数据传送无法完成之前是否继续尝试以及进行多少次尝试。由于继续尝试可能也是徒劳，cupsd 将禁用相应队列的打印。在消除了问题的起因后，系统管理员必须使用 /usr/bin/enable 命令重启用打印。

## 12.8.7 CUPS 浏览：删除打印作业

如果 CUPS 网络服务器通过浏览向客户机主机广播其队列并且客户机主机上合适的本地 `cupsd` 处于活动状态，则客户机 `cupsd` 接受来自应用程序的打印作业并将它们转发到服务器上的 `cupsd`。当 `cupsd` 接受打印作业后，会为该作业指派一个新的作业号。因此，客户机主机上的作业号与服务器上的作业号不同。因为通常都将打印作业立即转发，所以不能用客户机主机上的作业号将其删除，原因是一旦将打印作业转发到服务器 `cupsd`，客户机 `cupsd` 就会将打印作业视为已完成。

要在服务器上删除打印作业，使用命令（例如 `lpstat -h print-server -o`）确定服务器上的作业号，前提是服务器尚未完成该打印作业（即尚未将其发送到打印机）。使用此作业号，可以删除服务器上的打印作业：

```
cancel -h print-server queue-jobnumber
```

## 12.8.8 有问题的打印作业和数据传送错误

如果打印进程中将打印机关闭或关闭计算机，则打印作业保留在队列中，当打开打印机或重引导计算机后，打印继续。必须使用 `cancel` 从队列中去除有问题的打印作业。

如果打印作业有问题或主机和打印机之间的通讯出现错误，则打印机会打印出很多张带有乱码的纸张，这是因为它不能正确处理数据。要解决此问题，请执行以下步骤：

1. 要停止打印，请将所有纸张从喷墨打印机中取出或打开激光打印机的纸盒。高质量的打印机具有一个用于取消当前打印件的按钮。
2. 打印作业可能仍在队列中，因为只有将作业完全发送到打印机后才会将它们去除。使用 `lpstat -o` 或 `lpstat -h <print-server> -o` 检查哪个队列当前正在打印。使用 `cancel <queue>-<jobnumber>` 或 `cancel -h <print-server> <queue>-<jobnumber>` 删除打印作业。
3. 即使已将打印作业从队列中删除，某些数据仍会被传送到打印机。检查 CUPS 后端进程是否仍在为相应的队列运行并将其终止。例如，对于连接到并行端口的打印机，可以使用命令 `fuser -k /dev/lp0` 终止仍在访问打印机（更准确地说是并行端口）的所有进程。
4. 通过关闭打印机一段时间完全重设置打印机。然后插入纸张并打开打印机。

## 12.8.9 对 CUPS 打印系统进行调试

使用以下过程确定 CUPS 打印系统中的问题：

1. 在 `/etc/cups/cupsd.conf` 中设置 `LogLevel debug`。
2. 停止 `cupsd`。
3. 去除 `/var/log/cups/error_log*` 从而无需搜索非常长的日志文件。
4. 启动 `cupsd`。
5. 重复导致问题的操作。
6. 检查 `/var/log/cups/error_log*` 中的讯息以确定问题的原因。

## 12.8.10 参考信息

支持数据库中提供了对许多特定问题的解决方案。如果打印机出现问题，请参考支持数据库文章 [安装打印机](#) 和 [从 SUSE LINUX 9.2 进行打印机配置](#)，通过搜索关键字“打印机”可以找到这些文章。

# Linux 中的移动计算

本章对使用 Linux 进行移动计算的各个方面进行了概述。其中简要介绍了各个应用领域，并说明了所用硬件的关键功能。此外还涉及了针对特殊要求的软件解决方案和用于优化性能的选件，同时提出了最大限度降低能耗的种种可能。本章最后概述了关于这一主题的最为重要的信息源。

13.1	便携式计算机 . . . . .	228
13.2	移动硬件 . . . . .	233
13.3	手提电话和 PDA . . . . .	234
13.4	参考信息 . . . . .	234

多数人在谈及移动计算时都会联想到便携式计算机、PDA 和手提电话以及它们之间的数据交换。本章将重点扩展到可连接到便携式计算机或台式机的移动硬件部件，如外部硬盘、闪存盘或数码相机。

## 13.1 便携式计算机

### 13.1.1 便携式计算机硬件的特殊属性

便携式计算机的硬件不同于普通台式机的硬件。这是因为交换能力、占用空间和能耗能之类的标准都是要考虑的属性。移动硬件制造商已经开发了 PCMCIA（个人计算机存储卡国际协会）标准。该标准涉及存储卡、网络接口卡、ISDN 卡和调制解调器卡以及外部硬盘等多个硬件。在 Linux 中如何实现对上述硬件的支持、配置期间需要考虑哪些事项、哪些软件可用于控制 PCMCIA，以及如何对可能的问题查错，这些内容均在第14章 PCMCIA 中进行了说明。

### 13.1.2 省电

由于在制造便携式计算机时加入了能量优化系统部件，这使得不必连接电源线即可使用便携式计算机。便携式计算机在省电方面的贡献并不亚于操作系统在这方面的贡献。SUSE LINUX 支持多种方法，可以影响便携式计算机的能耗并对使用电池时的运行时间有各种影响。下面的列表按照在省电方面的贡献从大到小排列：

- 节制 CPU 流速
- 在暂停期间关闭显示器
- 手工调整显示器照明
- 断开不使用的支持热插拔的附件（USB CD-ROM、外部鼠标、不使用的 PCMCIA 卡等）
- 在硬盘闲置时降低其转速

有关 SUSE LINUX 中的电源管理以及运行 YaST 电源管理模块的详细背景信息，请参见第16章 电源管理。

### 13.1.3 在变化的操作环境中集成

在用于移动计算时，您的系统需要适应变化的操作环境。很多服务都依赖环境，因此必须重新配置底层的客户机，而这一切都可以由 SUSE LINUX 代劳。

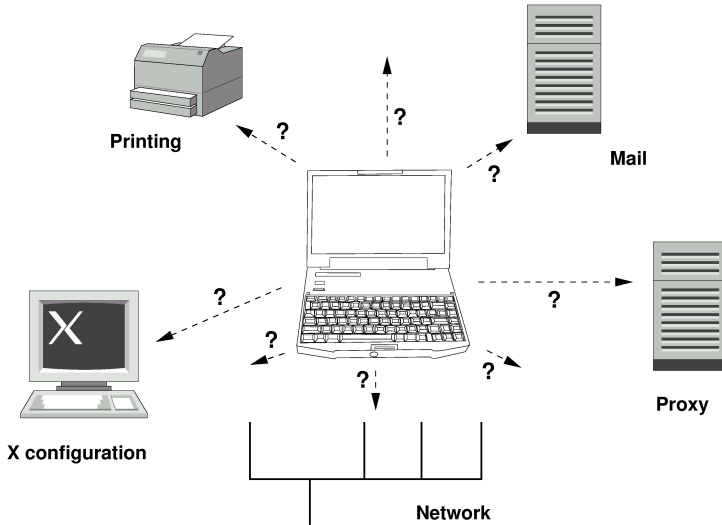


Figure 13.1: 在网络中集成便携式计算机

对于在小型家庭网络和办公网络之间往来通讯的便携式计算机，受影响的服务包括：

**网络配置** 涉及 IP 地址分配、名称解析、因特网连接和与其他网络的连接。

**打印** 必须存在可用打印机的当前数据库和可用的打印服务器（具体取决于网络）。

**电子邮件和代理** 就像在打印中那样，当前必须存在一组相应的服务器。

**配置 X** 如果您的便携式计算机暂时连接到投影机或外部监视器，则需要有不同的显示配置。

SUSE LINUX 提供两种方式，可以将便携式计算机集成到现有的可合并的操作环境中。

**SCPM** SCPM (系统配置配置文件管理) 允许将系统的任意配置状态储存为一种称为配置文件的“快照”。可以为不同的情况创建配置文件。在变化的环境(家庭网络、办公网络)中操作系统时,这些配置文件十分有用。可以随时在配置文件间切换。有关 SCPM 的信息,请参见第15章 系统配置 配置文件管理。使用 KDE 中的 kicker 小程序 Profile Chooser 可以在配置文件间切换。该应用程序要求在切换之前提供根口令。

**SLP** 服务位置协议(SLP) 简化了便携式计算机与现有网络的连接。没有 SLP, 便携式计算机的管理员通常需要详细了解网络中可用的服务。使用 SLP 则可以向本地网络中的所有客户机广播某种服务是否可用。支持 SLP 的应用程序可以处理 SLP 发送的信息,并进行自动配置。SLP 甚至还可用于安装系统,而不必再费事地搜索适合的安装源。有关 SLP 的详细信息,请参见第23章 网络中的 SLP 服务。

SCPM 强调的是启用和维护可重新生成的系统状况。而 SLP 可用于自动执行联网计算机的大部分配置工作,从而极大地简化了配置工作。

### 13.1.4 软件选择

移动应用中有多项特殊任务是通过专用软件来完成的:如系统监视(特别是电池电量监视)、数据同步,以及与外设和因特网之间的无线通讯。以下各节说明了 SUSE LINUX 为各项任务提供的最为重要的应用程序。

#### 系统监视

SUSE LINUX 提供了两种 KDE 系统监视工具。kicker 中的小程序 KPowersave 可以处理便携式计算机充电电池的单纯状态显示。复杂的系统监视则通过 KSysguard 来执行。如果使用的是 GNOME,上述功能则分别由 GNOME ACPI (类似面板小程序) 和 System Monitor 来完成。

**KPowersave** KPowersave 是在控制面板中显示充电电池状态的小程序。该图标将随电源类型调整显示。如果使用交流电,则显示一个小的插头图标。如果使用电池,则改为显示电池图标。提供根口令之后,相应的菜单会打开用于电源管理的 YaST 模块。此时即可设置使用不同电源时的系统行为。有关电源管理和相应的 YaST 模块的信息,请参见第16章 电源管理。

**KSysguard** KSysguard 是一个独立的应用程序,它可以将所有可测量系统参数收集到一个监视环境中。KSysguard 提供用于 ACPI (电池状态)、CPU 负载、网络、分区和内存使用等方面的监视程序。它还可以进一步观察和显示所有系统进程。可以自定义收集到的数据的表示和过滤方式。可以监



视不同数据页中的不同系统参数，也可以跨网络并行收集不同计算机上的数据。KSysguard 还可以在不具备 KDE 环境的计算机上作为守护程序运行。有关此程序的详细信息，请参见此程序中集成的帮助功能或 SUSE 帮助页。

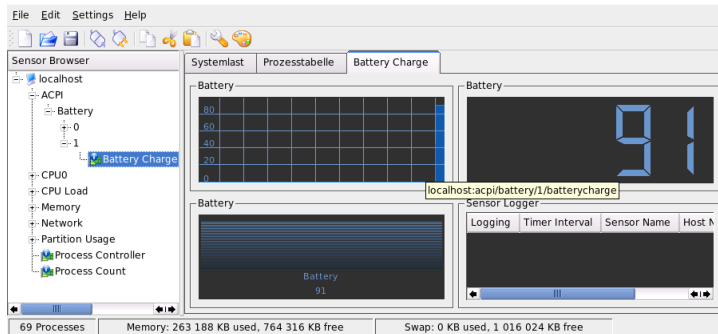


Figure 13.2: 使用 KSysguard 监视电池状态

## 同步数据

如果要在以下两种工作方式（在与网络断开的移动计算机上工作和在办公室中的联网工作站上工作）之间切换，则需要所有实例间都保持处理数据的同步。要同步的可能包括电子邮件文件夹、目录和单个文件，这些数据需要保持最新，以便在途中和办公室中处理。适用于这两种情况的解决方案如下：

**电子邮件的同步** 用一个 IMAP 帐户来储存办公网络中的电子邮件。随后可以从工作站使用任意断开连接的支持 IMAP 的电子邮件客户程序（如 *用户指南* 中所述的 Mozilla Thunderbird Mail、Evolution 或 KMail）来访问这些电子邮件。需要配置电子邮件客户程序，以便始终从同一文件夹访问 Sent messages（已发送信件）。这样能确保在完成同步过程之后可以提供所有信件及其状态信息。使用邮件客户程序中实施的 SMTP 服务来发送信件，取代系统范围内使用的 MTA postfix 或 sendmail 来接收有关未发送邮件的可靠反馈。

**同步文件和目录** 有若干实用程序适合在便携式计算机和工作站之间同步数据。有关详细信息，请参考第31章 *文件同步*。

## 无线通讯

便携式计算机不仅可以通过缆线连接家庭或办公网络，而且可以无线连接到其他计算机、外设、手提电话或 PDA。Linux 支持三种类型的无线通讯：

**WLAN** WLAN 在这三种无线技术中覆盖范围最广，是唯一一种适用于大型（有时甚至是在空间上分离的）网络的操作的技术。单独的计算机可以通过互连形成独立的无线网络或访问因特网。称为访问点的设备充当支持 WLAN 的设备的基站，并作为访问因特网的中介。移动用户可以在多个访问点之间切换，具体取决于所在位置及哪个访问点正在提供最佳连接。类似移动电话的情况，WLAN 用户可以访问一个大型网络，而不必被集中到某个位置来访问这个网络。有关 WLAN 的详细信息，请参见第17.1节“无线 LAN”。

**蓝牙** 蓝牙技术是所有无线技术中应用范围最广的技术。正如 IrDA 那样，蓝牙技术可用于计算机（便携式计算机）和 PDA 或手提电话之间的通讯。它还可用于连接视线范围内的多台计算机。蓝牙技术还可用于连接键盘或鼠标之类的无线系统部件。但这种技术的覆盖范围还不够大，无法将远程系统连接到网络中。WLAN 是穿越墙壁之类的有形障碍物进行通讯的首选技术。有关蓝牙技术、该技术的应用和配置的详细信息，请参见第17.2节“蓝牙”。

**IrDA** IrDA 是覆盖范围最小的无线技术。通讯双方必须在彼此的视线范围之内。无法穿越墙壁这样的障碍物。将文件从便携式计算机传送到手提电话就是 IrDA 的一种应用方式。使用 IrDA 即可覆盖由便携式计算机到手提电话之间的较短路径。要在较大范围内将文件传输给接收方，则需要通过移动网络来处理。IrDA 的另一种应用方式就是在办公室中无线传送打印作业。有关 IrDA 的详细信息，请参见第17.3节“红外线数据传送”。

### 13.1.5 数据安全性

要防止他人未经授权访问您的便携式计算机上的数据，您最好同时采用多种方式。可以在以下方面采取各种可能的安全措施：

**防止被盗** 始终尽可能地利用实物来防止您的系统被盗。零售店中就出售各种防盗工具，如锁链。

**保护系统中的数据** 重要数据不仅要在传送过程中加密，而且要在硬盘上加密。这样即使被盗也能保证数据不外泄。第34.3节“对分区和文件进行加密”中对如何使用 SUSE LINUX 创建加密分区进行了说明。

**网络安全** 无论采用哪种形式传送数据，任何形式的传送都应受到保护。有关 Linux 和网络的一般安全事项，请参见第34.4节“安全性和机密性”。有关无线联网的安全措施，请参见第17章 无线通讯。

## 13.2 移动硬件

SUSE LINUX 支持通过防火墙 (IEEE 1394) 或 USB 自动检测移动储存设备。移动储存设备一词可以指任意类型的防火墙或 USB 硬盘、USB 闪存盘或数码相机。这些设备一通过相应的接口连接到系统，即可通过热插拔自动检测和配置。subfs 和 submount 可以确保将这些设备装入文件系统中的相应位置。用户完全不必再像在以往版本的 SUSE LINUX 中那样手工装入和卸装设备。只要没有程序访问设备即可断开与系统的连接。

**外部硬盘 (USB 和防火墙)** 一旦系统正确识别出外部硬盘，相应的图标即显示在‘我的电脑’(KDE)或‘计算机’(GNOME)中的已装入驱动器列表中。左键单击该图标将显示该驱动器的内容。可以在此创建文件夹和文件，并执行编辑或删除操作。要将系统指定的硬盘名称重命名，请右击该图标，从打开的菜单中选择相应的菜单项。只有在文件管理器中才能显示这种名称更改。装入该设备时所用的描述符（在 /media/usb-xxx 或 /media/ieee1394-xxx 中）不会受到更改的影响。

**USB 闪存盘** 系统会按照处理外部硬盘的方式来处理这些设备。同样也可以重命名文件系统中的项。

**数码相机 (USB 和防火墙)** 系统识别出的数码相机也作为外部驱动器显示在文件管理器的概览中。KDE 允许读取和访问 URL camera:/ 中的图片。随后可以使用 digikam 或 The GIMP 对图像进行处理。使用 GNOME 时，Nautilus 将在自有文件夹内显示图片。GThumb 是一种简单的图像处理和管理实用程序。The GIMP 则可以完成更高级的照片处理。除 GThumb 之外，上述程序均在 *用户指南* 中进行了说明。另有还有一章专门说明数码相机。

### Important

#### 保护移动数据驱动器

移动硬盘或闪存盘像便携式计算机一样容易被盗。建议在这些设备上创建一个加密分区（如第34.3节“对分区和文件进行加密”中所述），以防止被第三方滥用。

Important

## 13.3 手提电话和 PDA

通过蓝牙或 IrDA 连接可以在台式机（或便携式计算机）和手提电话之间通讯。有些手提电话型号两种协议都支持，另一些则只支持其中的一种。这两种协议的使用范围以及相应的展开文档都已在第13.1.4节“无线通讯”中有所涉及。手提电话自带的手册中对如何在手提电话上配置这些协议进行了说明。第17.2节“蓝牙”和第17.3节“红外线数据传送”中则对 Linux 一端的配置进行了说明。

Evolution 和 Kontact 中已经内嵌了与 Palm, Inc. 制造的手持设备进行同步的支持功能。最初与手持设备连接时，无论使用哪种应用程序都可以借助向导轻松连接。一旦配置了针对 Palm Pilots 的支持，则需要确定应该同步哪种数据（地址、约会等）。上述两种群件应用程序都在 *用户指南* 中进行了说明。

Kontact 中集成的 KPilot 程序也可以作为单独的实用程序来运行。*用户指南* 中对该程序进行了说明。也可以使用程序 KitchenSync 来同步地址数据。

有关 Evolution 和 Kontact 的详细信息，请参见 *用户指南*。

## 13.4 参考信息

<http://tuxmobil.org/> 是与移动设备和 Linux 有关的所有问题的集中参考来源。该万维网站点的各个章节论述了便携式计算机、PDA、手提电话和其他移动硬件的软硬件问题。

<http://www.linux-on-laptops.com/> 中也提供了与 <http://tuxmobil.org/> 类似的参考资源。可以在此站点中找到有关便携式计算机和手持设备的信息。

SUSE 维护着一个德文邮件列表，专门讨论便携式计算机这一主题。请参见 <http://lists.suse.com/archive/suse-laptop/>。在这个列表中，用户和开发人员讨论的是有关 SUSE LINUX 中的移动计算的各个方面的问题。用英文投递的信件都有答复，但存档信息中大部分都只有德文信息。

如果便携式计算机出现 SUSE LINUX 电源管理问题，建议阅读 `/usr/share/doc/packages/powersave` 中的 README 文件。此目录经常包含来自测试人员和开发人员的最新反馈，所以为解决这类问题提供了有价值的提示。

# PCMCIA

本节介绍便携式计算机中 PCMCIA 硬件和软件的一些特性。PCMCIA 代表个人计算机存储卡国际协会，是用于所有相关硬件和软件的集合术语。

14.1	硬件	236
14.2	软件	236
14.3	配置	237
14.4	实用程序	239
14.5	查错	239
14.6	参考信息	241

## 14.1 硬件

最重要的部件是 PCMCIA 卡。PCMCIA 卡有两种类型：

**PC 卡** PCMCIA 刚出现时，PC 卡已无处不在。PC 卡使用 16 位总线进行数据传输，通常很便宜。某些目前的 PCMCIA 桥很难检测到这些 PC 卡。但是，只要检测到 PC 卡，通常它们能顺畅运行，不会出现任何问题。

**CardBus 卡** 这是一个较新的标准。这种卡使用 32 位总线，速度更快，但更贵。CardBus 卡就像 PCI 卡一样集成在系统中，也可以顺畅运行。

如果 PCMCIA 服务是活动的，则命令 `cardctl ident` 可以显示所插入卡的类型。目录 `/usr/share/doc/packages/pcmcia` 下的文件 `SUPPORTED.CARDS` 中提供了所支持的卡的列表。此目录中还有 PCMCIA HOWTO 文档的最新版本。

另一个重要的部件是 PCMCIA 控制器（PC 卡桥或 CardBus 桥），它在卡和 PCI 总线之间建立连接。支持所有常见型号。可以使用命令 `pcic_probe` 确定控制器的类型。如果是 PCI 设备，则命令 `lspci -vt` 可以提供详细信息。

## 14.2 软件

### 14.2.1 基础模块

必需的内核模块位于内核包中。另外，还需要 `pcmcia` 和 `hotplug` 包。在启动 PCMCIA 时，将装载模块 `pcmcia_core`、`yenta_socket` 和 `ds`。在非常特殊的情况下，需要模块 `tcic` 而不是 `yenta_socket`。这些模块初始化现有的 PCMCIA 控制器并提供基本功能。

### 14.2.2 卡管理器

因为可以在系统运行时更换 PCMCIA 卡，所以必须监控插槽中的活动。此任务是由在基础模块中实施的卡服务进行处理的。卡管理器（用于 PC 卡）或内核热插拔系统（用于 CardBus 卡）处理插入卡的初始化。在装载基础模块后，PCMCIA 启动脚本将启动卡管理器。热插拔是被自动激活的。

如果插入卡，则卡管理器或热插拔确定其类型和功能，然后装载相应的模块。成功装载模块后，卡管理器或热插拔根据卡的功能启动特定的初始化脚本。初始化

脚本建立网络连接、装入外部 SCSI 硬盘的分区或执行其他硬件特定的操作。卡管理器的脚本位于目录 `/etc/pcmcia` 中。热插拔的脚本位于 `/etc/hotplug` 中。拆卸卡后，卡管理器或热插拔用相同的脚本终止所有卡活动。随后，将卸载不再需要的模块。

这些操作被称为热插拔事件。每当添加硬盘或分区（块事件），热插拔脚本就会使用 `subfs` 使新媒体可以立即在 `/media` 中使用。要通过以前的 PCMCIA 脚本装入媒体，应在热插拔中禁用 `subfs`。

系统日志文件 (`/var/log/messages`) 中记录了 PCMCIA 的启动和卡事件。此日志文件中还记录了装载的模块和执行的脚本。

理论上，无需执行任何附加操作就可以拆卸 PCMCIA 卡。只要不再有活动的网络连接，就可以放心地拆卸网卡、调制解调器和 ISDN 卡。但对于外部硬盘的已装入分区或 NFS 目录，则不能这样做。必须正确地同步并卸装此类装置。当然，取出卡后就无法进行同步和卸装了。如果您不能确定，则可以使用命令 `cardctl eject` 取消激活仍插在便携式计算机中的所有卡。要只取消激活其中一个卡，请指定插槽编号，例如 `cardctl eject 0`。

## 14.3 配置

使用 YaST 运行级别编辑器，确定引导系统时是否应启动 PCMCIA。可以使用‘系统’→‘运行级别编辑器’启动此模块。

文件 `/etc/sysconfig/pcmcia` 中定义了以下三个变量：

**PCMCIA\_PCIC** 包含控制 PCMCIA 控制器的模块的名称。通常，启动脚本应自动确定模块。如果自动确定失败，请在这里输入模块。否则，此变量应留空。

**PCMCIA\_CORE\_OPTS** 此变量是为 `pcmcia_core` 模块的参数设计的。但很少使用这些参数。`pcmcia_core(4)` 手册页中有这些选项的介绍。因为此手册页引用了 David Hinds 的 `pcmcia-cs` 包中的同名模块，所以它列出的参数要多于内核中模块实际支持的参数，即以 `cb_` 和 `pc_debug` 开头的的所有参数。

**PCMCIA\_BEEP** 启用和禁用卡管理器的声音信号。

文件 `/etc/pcmcia/config` 和 `/etc/pcmcia/*.conf` 为 PC 卡指派驱动程序。系统首先读取 `config`，然后按字母顺序读取 `/*.conf`。将使用为卡找到的最后一项。`pcmcia(5)` 手册页提供了有关这些文件语法的详细信息。

被指定为 `/etc/sysconfig/hardware/hwcfg-<configurationname>` 的文件为 CardBus 卡指派驱动程序。这些文件是在配置卡时由 YaST 创建的。`/usr/share/doc/packages/sysconfig/README` 和 `getcfg(8)` 手册页提供了有关配置名称的详细信息。

### 14.3.1 网卡

可以像配置普通网卡一样使用 YaST 配置 Ethernet 网卡、无线 LAN 网卡和 TokenRing 网卡。如果未检测到您的卡，则在硬件设置中选择卡类型 PCMCIA。第 22.4 节“使用 YaST 配置网络连接”中提供了有关网络配置的所有其他详细信息。

### 14.3.2 ISDN

与其他 ISDN 卡类似，在很多情况下 ISDN PC 卡也可以使用 YaST 进行配置。这与选择了哪个列出的 PCMCIA ISDN 卡无关，只要这个卡是 PCMCIA 卡即可。当配置硬件并选择提供商时，运行方式必须始终是 `hotplug`，而不是 `onboot`。ISDN 调制解调器也以 PCMCIA 卡的形式存在。这些卡是附带 ISDN 连接工具包的调制解调器卡或多功能卡。可以像处理调制解调器一样处理这些卡。

### 14.3.3 调制解调器

通常，对于调制解调器 PC 卡，没有 PCMCIA 特定的设置。一插入调制解调器，它就在 `/dev/modem` 下可用。某些 PCMCIA 调制解调器卡是不受 Linux 支持的软调制解调器。如果这些卡有驱动程序，则必须在系统中安装这些驱动程序。

### 14.3.4 SCSI 和 IDE

卡管理器或热插拔装载合适的驱动程序模块。一插入 SCSI 或 IDE 卡，就可以使用所连接的设备。设备名是动态确定的。`/proc/scsi` 和 `/proc/ide` 下提供了有关可用 SCSI 和 IDE 设备的信息。

在将 PCMCIA 插入插槽前，必须打开外部硬盘、CD-ROM 驱动器和类似的设备。对 SCSI 设备使用活动终止。



## Warning

### 拆卸 SCSI 或 IDE 卡

在拆卸 SCSI 或 IDE 卡前，必须使用命令 `umount` 卸装已连接设备的所有分区。如果您忘记执行此操作，则只有在重引导系统后才能访问这些设备。

## Warning

## 14.4 实用程序

上述 `cardctl` 实用程序是获取 PCMCIA 信息或执行某些操作的主要工具。有关详细信息，请参考 `cardctl(8)` 手册页。输入 `cardctl` 可以获得有效选项的列表。可以使用图形前端 `cardinfo` 来控制 `cardctl` 的主要功能。要使用图形前端，请安装 `pcmcia-cardinfo` 包。

`ifport`、`ifuser`、`probe` 和 `rcpcmcia` 是 `pcmcia` 包中的一些其他实用程序。但是，不经常使用这些实用程序。要了解 `pcmcia` 包中包含哪些文件，请输入命令 `rpm -ql pcmcia`。

## 14.5 查错

便携式计算机和某些卡上大多数与 PCMCIA 有关的问题都可以通过系统地接近问题使问题得以确定和解决。首先，查明问题是由卡还是 PCMCIA 基础系统引起的。为查明这一情况，请在不插入任何卡的情况下引导计算机。如果基础系统工作正常，则插入卡。`/var/log/messages` 中将记录所有讯息。在搜索问题的原因时，请用 `tail -f /var/log/messages` 监视此文件。这样，可以将问题起因的范围缩小到以下两种情况之一。

### 14.5.1 PCMCIA 基础系统不工作

如果在引导系统时系统挂起并出现讯息“PCMCIA: Starting services”（PCMCIA：正在启动服务）或异常现象，则在引导提示符处输入 `NOPCMCIA=yes` 重引导系统并禁用 PCMCIA。要隔离错误，请一个一个地手工装载 PCMCIA 系统的三个基础模块。

要手工装载 PCMCIA 模块，请以 `root` 用户身份执行命令 `modprobe pcmcia_core`、`modprobe yenta_socket` 和 `modprobe ds`。在非常特殊

的情况下，使用 `tcic`、`i82365` 或 `i82092` 而不是 `yenta_socket`。首先装载的两个模块是关键模块。

如果在装载 `pcmcia_core` 时出现错误，请参考 `pcmcia_core(4)` 手册页。可以使用命令 `modprobe` 首先测试本手册页中介绍的选项。例如，测试可用 I/O 范围。有时，如果此测试影响了其他硬件部件，则会引发问题。可以使用选项 `probe_io=0` 来避免此问题：

```
modprobe pcmcia_core probe_io=0
```

如果所选选项成功执行，则将文件 `/etc/sysconfig/pcmcia` 中的变量 `PCMCIA_CORE_OPTS` 设置为值 `probe_io=0`。使用空格分隔多个选项：

```
PCMCIA_CORE_OPTS="probe_io=0 setup_delay=10"
```

装载 `yenta_socket` 模块时出现错误表明问题是更基本的问题，例如 ACPI 进行的资源分配。

文件 `/etc/pcmcia/config` 和 `/etc/pcmcia/config.opts` 由卡管理器进行解释。这些文件包含的设置中一部分用于启动 `cardmgr`，一部分用于装载 PC 卡的驱动程序模块。文件 `/etc/pcmcia/config.opts` 中可以包含或排除 IRQ、I/O 端口和内存地址范围。在特殊情况下，访问不正确的 I/O 范围将导致系统崩溃。如果是这样，请尝试限制这些范围。

## 14.5.2 PCMCIA 卡工作不正常

基本上，有三种类型的错误：未检测到卡、无法装载驱动程序或驱动程序提供的接口配置错误。一定要了解卡是由卡管理器控制的还是由热插拔控制的。卡管理器控制 PC 卡，而热插拔控制 CardBus 卡。

**插入卡时无反应** 如果在插入卡时系统没有反应，并且手工执行命令 `cardctlinsert` 也不起作用，则 PCI 设备的中断分配可能不正确。如果是这种情况，则其他 PCI 设备（例如网卡）也可能有问题。在这种情况下，可以使用引导参数 `pci=noacpi` 或其他 PCI 或 ACPI 参数。

**未检测到卡** 如果未检测到卡，则 `/var/log/messages` 中将出现讯息“unsupported Card in Slot x”（不支持插槽 x 中的卡）。此讯息仅指出卡管理器不能为卡指派驱动程序。需要使用文件 `/etc/pcmcia/config` 或 `/etc/pcmcia/*.conf` 进行这一指派。通过将现有项用作模板可以轻松地扩展此驱动程序数据库。输入命令 `cardctl ident` 查找

有关卡的详细信息。PCMCIA HOWTO 文档（第 6 部分）和 `pcmcia(5)` 手册页提供了有关卡的详细信息。编辑 `/etc/pcmcia/config` 或 `/etc/pcmcia/*.conf` 后，使用命令 `rcpcmciareload` 重装载所指定的驱动程序。

**未装载驱动程序** 出现此问题的一个原因可能是驱动程序数据库包含不正确的指派。例如，如果制造商在外观不变的卡型号上使用了不同的芯片，则可能发生这种情况。某些型号只有在使用预先选择的驱动程序之外的驱动程序时才能工作或更好地工作。在这种情况下，您需要有关卡的详细信息。如有必要，请将问题发送到邮件列表或请求高级支持。

对于 CardBus 卡，必须在文件 `/etc/sysconfig/hotplug` 中插入项 `HOTPLUG_DEBUG=yes`。随后，检查系统日志中是否有指出是否（成功）装载驱动程序的讯息。

另一个可能的原因是资源冲突。对于大多数 PCMCIA 卡，使用哪个 IRQ、I/O 端口或内存范围进行操作没有关系，但也有例外。如果是这种情况，则一次只测试一个卡，并临时禁用其他系统部件，例如声卡、IrDA、调制解调器和打印机。以 `root` 用户身份使用命令 `lsdev` 查看系统的资源分配。多个 PCI 设备使用同一个 IRQ 完全没有问题。

一个可能的解决方案是为卡驱动程序模块查找合适的选项。输入 `modinfo (driver)` 列出这些选项。大多数模块都提供有手册页。`rpm -ql pcmcia | grep man` 列出 `pcmcia` 包中包含的所有手册页。要测试这些选项，可以手工卸载卡驱动程序。

找到解决方案后，通常可以在 `/etc/pcmcia/config.opts` 中允许或禁止使用某些资源。还可以在此文件中输入卡驱动程序的选项。例如，要使 `pcnet_cs` 模块以独占方式使用 IRQ 5，请添加下面一项：

```
module pcnet_cs opts irq_list=5
```

**配置错误的接口** 在这种情况下，使用 `getcfg` 仔细检查接口的配置和配置的名称以消除配置错误。应该将文件 `/etc/sysconfig/network/config` 中的变量 `DEBUG` 和文件 `/etc/sysconfig/hotplug` 中的变量 `HOTPLUG_DEBUG` 设置为 `yes`。对于其他卡或如果此操作不起作用，则可以在卡管理器或热插拔执行的脚本中添加行 `set-vx`（请参见 `/var/log/messages`）。这样，系统日志将记录脚本的每一个命令。如果在脚本中找到关键部分，则在终端中输入并测试相应的命令。

## 14.6 参考信息

位于 <http://linux-laptop.net> 的 Linux 便携式计算机主页提供了特定便携

式计算机型号的实用信息。位于 <http://tuxmobil.org/> 的 TuxMobil 主页提供了另一个有用的信息来源。这些页提供了便携式计算机 Howto 文档、IrDA Howto 文档和许多其他相关信息。此外，位于 <http://portal.suse.com> 的支持数据库包含多篇有关在移动设备中使用 Linux 的文章。要查找这些文章，请在搜索对话框中输入关键字 *notebook* 或 *laptop*。

# 系统配置配置文件管理

本章介绍 SCPM（系统配置配置文件管理）。借助于 SCPM，可以调整计算机的配置以适应不同的操作环境或硬件配置。SCPM 管理一组用于不同方案的系统配置文件。使用 SCPM 可以在两个系统配置文件之间方便地切换，从而无需手工重配置系统。

15.1	术语	244
15.2	使用命令行配置 SCPM	244
15.3	YaST 配置文件管理器	247
15.4	查错	251
15.5	引导系统时选择配置文件	251
15.6	参考信息	252

某些情况需要经过修改的系统配置。这对在不同位置运行的移动计算机而言是很常见的情况。如果要使用普通硬件部件之外的其他硬件部件临时运行台式机系统，则使用 SCPM 会很方便。恢复最初的系统配置也会很容易，同时可以重现对系统配置的修改。利用 SCPM，可以在自定义的配置文件中保存系统配置的任何部分。

SCPM 的主要应用领域是便携式计算机上的网络配置。不同的网络配置通常需要对其他服务（例如，电子邮件或代理）进行不同的设置。SCPM 还可以用于其他方面，例如在家里和办公室使用不同的打印机、为开会时使用多媒体投影仪而设置的自定义 X 服务器配置、旅行时采用的特殊省电设置以及海外子公司所在的不同时区。

## 15.1 术语

以下是 SCPM 文档和 YaST 模块中使用的一些术语。

- **系统配置**这一术语是指计算机的完整配置。它包括所有基础设置，例如硬盘分区的使用、网络设置、时区选择和键盘映射。
- **配置文件**也称为**配置配置文件**，是一个已被保留并且可以随时恢复的状态。
- **活动配置文件**是指最后选择的配置文件。这并不意味着当前系统配置完全与此配置文件对应，因为配置可以随时进行修改。
- SCPM 上下文中的**资源**是指用于系统配置的元素。资源可以是文件或包含元数据（例如，用户、权限或访问时间）的软链接。资源也可以是在此配置文件中运行、但在另一个配置文件中取消的系统服务。
- 每个资源都属于一个特定的**资源组**。这些组包含逻辑上在一起的所有资源——大多数组既包含服务又包含其配置文件。组装由 SCPM 管理的资源非常容易，原因是此操作不需要了解所需服务的配置文件。SCPM 附带有足以满足大多数方案的预配置资源组的集合。

## 15.2 使用命令行配置 SCPM

本节介绍 SCPM 的命令行配置。了解如何启动、配置 SCPML 并使用配置文件。

## 15.2.1 启动 SCPM 并定义资源组

在使用前，必须激活 SCPM。使用 `scpmenable` 激活 SCPM。如果是第一次运行，则将初始化 SCPM，这需要花几秒钟。可以随时用 `scpm disable` 取消 SCPM，以防止意外切换配置文件。随后的重新激活只恢复初始化。

默认情况下，SCPM 处理网络和打印机设置以及 X.Org 配置。要管理特殊的服务或配置文件，请激活相应的资源组。要列出预定义的资源组，请使用 `scpmlist_groups`。要只查看已经激活的资源组，请使用 `scpmlist_groups -a`。请在命令行以 `root` 用户身份发出这些命令。

```
scpm list_groups -a
```

```
nis          Network Information Service client
mail         Mail subsystem
ntpd         Network Time Protocol daemon
xf86         X Server settings
autofs       Automounter service
network     Basic network settings
printer      Printer settings
```

使用 `scpmactivate_group NAME` 或 `scpmdeactivate_group NAME` 可以激活或取消某个组。用相关组名替换 `NAME`。

## 15.2.2 创建和管理配置文件

激活 SCPM 后，名为 `default` 的配置文件已经存在。可以使用 `scpm list` 获得所有可用配置文件的列表。这一现有的配置文件也是活动配置文件，可以使用 `scpm active` 进行校验。配置文件 `default` 是一个基本配置，其他配置文件都是从该配置派生的。为此，应该首先创建应在所有配置文件中相同的所有设置。然后，用 `scpm reload` 将这些修改储存在活动配置文件中。可以以 `default` 配置文件为基础，进行复制和重命名以生成新配置文件。

可以使用两种方法来添加新配置文件。如果新配置文件（这里名为 `work`）以配置文件 `default` 为基础，则用 `scpmcopy default work` 创建此配置文件。命令 `scpmswitch work` 可以切换为新配置文件，然后可以对其进行修改。您可能要为特殊目的修改系统配置并将更改保存到新配置文件中。命令 `scpmadd work` 将创建一个新配置文件，方法是将当前系统配置保存在配置文件 `work` 中并将其标记为活动配置文件。随后运行 `scpm reload` 将更改保存到配置文件 `work` 中。

可以使用 `scpmrename x y` 和 `scpmdelete z` 重命名或删除配置文件。例如，要将 `work` 重命名为 `project`，请输入 `scpmrename work project`。要删除 `project`，请输入 `scpm delete project`。不能删除活动配置文件。

### 15.2.3 切换配置配置文件

可以使用命令 `scpm switch work` 切换到另一个配置文件（在本例中是配置文件 `work`）。切换到活动配置文件以将系统配置的已修改设置包括在配置文件中。此操作对应于命令 `scpmreload`。

切换配置文件时，SCPM 首先检查已修改了活动配置文件的哪些资源。然后，SCPM 查询是将每个资源的修改添加到活动配置文件中，还是将它们删除。如果需要这些资源的单独列表（SCPM 以前的版本采用这种形式），则使用带有 `-r` 参数的切换命令：`scpm switch -r work`。

```
scpm switch -r work
```

```
Checking for modified resources
Checking for Resources to be started/shut down
Checking for dependencies
Restoring profile default
```

随后，SCPM 将当前系统配置与要切换到的配置文件进行比较。在这个阶段，SCPM 评估由于相互依赖性或为反映配置中的更改，需要停止或重新启动哪些系统服务。这类似于部分系统重引导，系统的一小部分进行重引导，而其余部分继续运行，无任何影响。也就是在此时系统服务停止，写入所有已修改的资源（例如配置文件），然后重新启动系统服务。

### 15.2.4 高级配置文件设置

您可以为通过 `scpm list` 显示的每个配置文件输入说明。对于活动配置文件，使用 `scpm set description "text"` 进行设置。对于不活动的配置文件，请提供配置文件名，例如 `scpm set description "text" work`。有时在切换配置文件时可能需要执行一些不是由 SCPM 提供的其他操作。每个配置文件最多可以附加 4 个可执行文件。将在切换进程的不同阶段调用这些文件。这些阶段被称为：

**prestop** 离开配置文件时，在停止服务之前

**poststop** 离开配置文件时，在停止服务之后



**prestart** 激活配置文件时，在启动服务之前

**poststart** 激活配置文件时，在启动服务之后

使用命令 `set` 插入这些操作，方法是输入 `scpm`  
`set prestop filename`、`scpm set poststop filename`、`scpm`  
`set prestart filename` 或 `scpm set poststart filename`。这些脚本  
必须可执行并且引用了正确的解释器。

### Warning

集成自定义脚本

对超级用户 (root) 来说，将由 SCPM 执行的其他脚本必须是可读的和可执行的。必须阻止所有其他用户对这些文件的访问。输入命令 `chmod 700 filename` 和 `chown root:root filename` 向 root 用户授予对这些文件的独占权限。

### Warning

用 `set` 和 `get` 查询输入的所有其他设置。例如，命令 `scpm get poststart` 返回 `poststart` 调用的名称，如果未附加任何对象，则不返回任何内容。使用 "" 覆盖可以重置这些设置。使用命令 `scpm set prestop ""` 可以去掉附加的 `prestop` 程序。

可以使用与添加注释相同的方法将所有 `set` 和 `get` 命令应用到任何一个配置文件中。例如，`scpm get prestop filename work` 或 `scpm get prestop work`。

## 15.3 YaST 配置文件管理器

通过 YaST 控制中心启动 YaST 配置文件管理器（‘系统’→‘配置文件管理器’）。第一次启动时，通过选择如图 15.1 “YaST SCPM 选项” 所示的 ‘SCPM 选项’ 对话框中的 ‘启用’ 来显式启用 SCPM。在 ‘设置’ 中，确定是否应自动关闭进度弹出窗口，是否显示有关 SCPM 配置进度的冗长讯息。对于 ‘切换方式’，确定在切换配置文件时，是应保存还是应丢弃活动配置文件的已修改资源。如果将 ‘切换方式’ 设置为 ‘常规’，则在切换时保存活动配置文件的所有更改。要定义引导时 SCPM 的行为，请将 ‘引导方式’ 设置为 ‘保存更改’（默认设置）或设置为 ‘丢弃更改’。

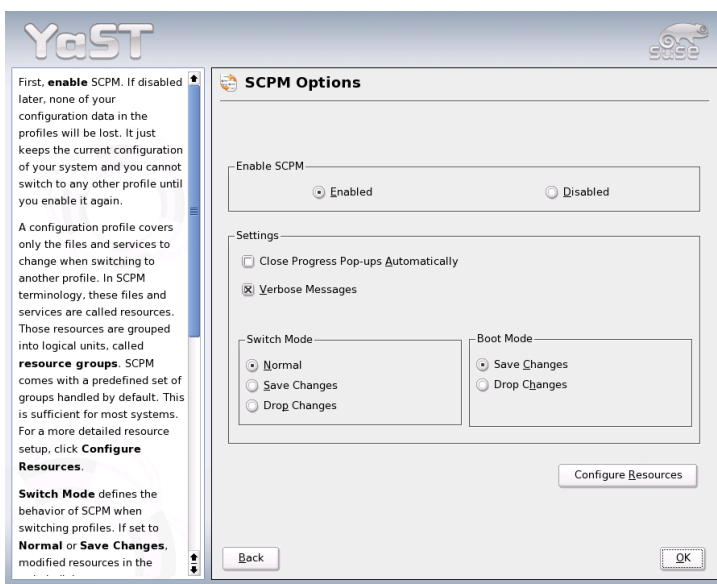


Figure 15.1: YaST SCPM 选项

### 15.3.1 配置资源组

要对当前资源配置进行更改，请在‘SCPM 选项’对话框中选择‘配置资源’。下一个对话框‘资源组配置’（如图15.2“资源组配置”中所示）列出系统上可用的所有资源组。要添加或编辑资源组，请指定或修改‘资源组’和‘说明’。例如，对于LDAP 服务，输入 ldap 作为‘资源组’，输入 LDAP 客户机服务作为‘说明’。然后，输入相应的资源（服务、配置文件或两者），或修改现有的资源。删除未使用的这些资源。要重置所选资源的状态（丢弃对资源所进行的任何更改并返回到初始配置值），请选择‘重设置组’。您的更改将被保存到活动配置文件中。

### 15.3.2 新建配置文件

要创建新的配置文件，请在开始对话框（‘系统配置配置文件管理’）中单击‘添加’。在打开的窗口中，选择应将新配置文件基于当前系统配置（SCPM 自动检索当前配置并将其写入配置文件中），还是基于现有的配置文件。如果将当前

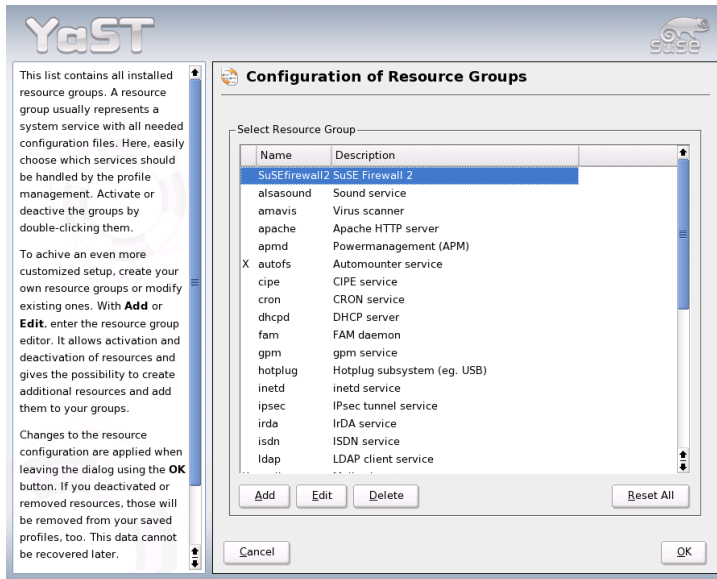


Figure 15.2: 资源组配置

系统配置用作新配置文件的基础，则可将新配置文件标记为新的活动配置文件。此操作不会对旧配置文件进行任何更改，也不会启动或停止任何服务。

在随后的对话框中为新配置文件提供名称和简短说明。为使 SCPM 在切换配置文件时执行特殊脚本，请输入每个可执行文件的路径（请参见图 15.3 “特殊的配置文件设置”）。有关详细信息，请参考第 15.2.4 节“高级配置文件设置”。SCPM 对新配置文件的资源执行检查。在此测试成功完成后，就可使用新配置文件了。

### 15.3.3 修改现有的配置文件

要修改现有的配置文件，请在开始对话框（‘系统配置配置文件管理’）中单击‘编辑’。然后，按需要修改名称、说明、脚本和资源。

### 15.3.4 切换配置文件

要切换配置文件，请打开配置文件管理器。活动配置文件标记有一个箭头。选

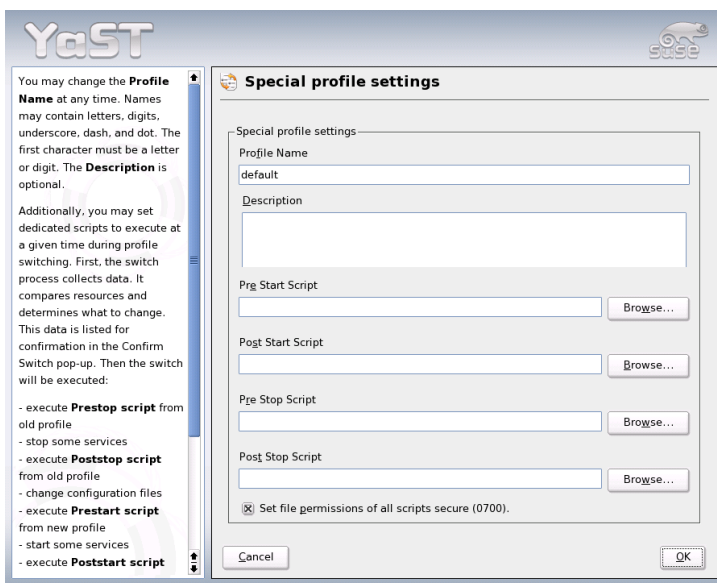


Figure 15.3: 特殊的配置文件设置

择要切换到的配置文件，然后单击‘切换到’。如有必要，SCPM 将检查新资源或已修改的资源，然后添加这些资源。

如果已修改了资源，则 YaST 打开‘确认切换’对话框。‘活动配置文件的已修改资源组’列出已修改但尚未保存到活动配置文件中的所有资源组。当前所选资源组的‘保存或忽略’确定是应将对此资源组所进行的更改保存到活动配置文件中，还是应丢弃这些更改。或者，选择每个资源并单击‘细节’来详细分析这些更改。这将显示属于此已修改资源组的所有配置文件或可执行文件的列表。要获得旧版本和新版本行对行的比较，请单击‘显示更改’。在分析了这些更改后，请在‘操作’中确定要对这些更改执行的操作：

**保存资源** 将此资源保存到活动配置文件中，但保留所有其他配置文件不动。

**忽略资源** 保留活动资源不动。此更改被丢弃。

**保存到所有配置文件中** 将此资源的整个配置复制到所有其他配置文件中。

**修补所有配置文件** 只将最新的更改应用到所有配置文件。

‘保存或忽略全部’只保存或丢弃此对话框中显示的所有资源的更改。

在确认对活动配置文件的更改后，通过单击‘确定’退出‘确认切换’对话框。然后，SCPM 切换到新配置文件。在切换时，SCPM 执行旧配置文件的 `prestop` 和 `poststop` 脚本及新配置文件的 `prestart` 和 `poststart` 脚本。

## 15.4 查错

本节介绍使用 SCPM 时经常遇到的问题。了解这些问题是如何产生的和如何解决这些问题。

### 15.4.1 在切换进程中终止

有时，SCPM 会在切换过程中停止工作。这可能是由某个外部原因引起的，例如用户中止、电源故障或甚至是 SCPM 本身的错误。如果出现这种情况，则在您下次启动 SCPM 时将显示一条错误讯息，指出 SCPM 已被锁定。这是出于系统安全的考虑，因为储存在其数据库中的数据可能与系统的状态不同。要解决此问题，请运行 `scpm recover`。SCPM 执行上次运行时缺少的所有操作。您还可以运行 `scpm recover -b`，它尝试撤消上次运行时所有已执行的操作。如果使用 YaST 配置文件管理器，则在启动时将获得提供执行上述命令的恢复对话框。

### 15.4.2 更改资源组配置

要在初始化 SCPM 后修改资源组的配置，请在添加或去除组后输入 `scpmrebuild`。这样，新资源就会被添加到所有配置文件中，而去除的资源也会被永久删除。如果删除的资源在各个配置文件中有不同的配置，则此配置数据将丢失（系统中当前版本除外，因为 SCPM 不对其进行处理）。如果使用 YaST 修改配置，则不需要输入重建命令，因为 YaST 对此进行了处理。

## 15.5 引导系统时选择配置文件

要在引导系统时选择配置文件，请在出现引导屏幕时按 **(F4)** 键访问可用配置文件列表。使用箭头键选择配置文件，然后按 **(Enter)** 键进行确认。所选的配置文件随后将被用作引导选项。

## 15.6 参考信息

SCPM 信息页提供了最新的文档，您可以使用 Konqueror 或 Emacs (konqueror info:scpm) 等工具进行查看。在控制台中，输入 info 或 pinfo。 /usr/share/doc/packages/scpm 提供了开发人员所需的信息。

# 电 源 管 理

本章概述了 Linux 中的各种电源管理技术，并详细介绍了所有可用的 APM（高级电源管理）、ACPI（高级配置和电源接口）和 CPU 频率调节设置。

16.1	省电功能	254
16.2	APM	255
16.3	ACPI	256
16.4	硬盘的休眠	261
16.5	powersave 包	262
16.6	YaST 电源管理模块	269

与 APM（以前只在便携式计算机上用于电源管理）不同，硬件信息和配置工具 ACPI 可用于所有目前的计算机（便携式计算机、台式机和服务器）。在目前许多类型的硬件上，可以根据情况调节 CPU 频率，这特别有助于延长移动设备上宝贵的电池时间（CPU 频率调节）。

所有电源管理技术都需要合适的硬件和 BIOS 例程。大多数便携式计算机、许多目前的台式机和服务器都符合这些要求。APM 用在许多以前的计算机上。因为 APM 主要由在 BIOS 中实施的功能集组成，所以 APM 支持的级别因硬件的不同而有所不同。而这对 ACPI 而言就更是如此，ACPI 更加复杂。因此，实际上很难决定是向您推荐 APM 还是 ACPI。在您的硬件上测试各种过程，然后选择支持情况最好的技术即可。

### Important

用于 AMD64 处理器的电源管理  
具有 64 位内核的 AMD64 处理器只支持 ACPI。

Important

## 16.1 省电功能

省电功能不仅对便携式计算机的移动使用很重要，而且对台式机系统也很重要。下面几段简要介绍电源管理系统 APM 和 ACPI 中的主要功能及其用法：

**待机** 此运行方式将关闭屏幕显示。在某些计算机上，处理器性能会受到限制。但并不是所有 APM 实施中都可以使用此功能。此功能对应于 ACPI 状态 S1 或 S2。

**暂停（到内存）** 此方式将整个系统状态写入 RAM。随后，除 RAM 外，整个系统都进入休眠状态。在此状态下，计算机消耗的电量非常少。此状态的优点是无需引导和重新启动应用程序就可以在数秒内在同一点恢复工作。通常在使用 APM 时，可以通过合上机盖暂停便携式计算机，然后打开机盖可将其激活。此功能对应于 ACPI 状态 S3。对此状态的支持仍在开发中，因此目前主要依靠硬件来实现支持。

**休眠（暂停到磁盘）** 在此运行方式下，将整个系统状态写入硬盘并关闭系统电源。从此状态重新激活大概需要 30 到 90 秒。将恢复到暂停之前的状态。某些制造商提供这种方式有用的混合变体（例如 IBM Thinkpads 中的 RediSafe）。对应的 ACPI 状态是 S4。在 Linux 中，由独立于 APM 和 ACPI 的内核例程执行暂停到磁盘。



**电池监视** ACPI 和 APM 检查电池电量状态并提供有关电量状态的信息。另外，当达到临界电量状态时，两个系统都将协调要执行的操作。

**自动关闭电源** 关闭后，将关闭计算机的电源。当在电池电量用完前立即执行自动关闭时，此功能特别重要。

**系统部件的关闭** 关闭硬盘是整个系统中省电潜力最大的一个方面。根据整个系统的可靠性，硬盘可以休眠一段时间。但是，休眠期间丢失数据的风险也会增加。可以通过 ACPI 对其他部件取消激活（至少在理论上如此），或在 BIOS 设置中对这些部件永久取消激活。

**处理器速度控制** 在与 CPU 有关的方面，可以采用三种不同的省电方法：频率和电压调节（也称为 PowerNow! 或 Speedstep）、节流和让处理器进入休眠状态（C 状态）。根据计算机的运行方式，还可以将这三种方法结合起来使用。

## 16.2 APM

APM BIOS 本身会执行一些省电功能。在许多便携式计算机上，可以使用组合键或通过合上机盖来激活待机状态和暂停状态，无需任何特殊的操作系统功能。但是，要通过命令激活这些方式，则必须在暂停系统前触发某些特定的操作。要查看电池电量水平，需要特殊的程序包和合适的内核。

SUSE LINUX 内核具有内置的 APM 支持。但是，只有在 BIOS 中未实施 ACPI 且检测到 APM BIOS 的情况下才能激活 APM。要激活 APM 支持，必须在引导提示符下使用 `acpi=off` 禁用 ACPI。输入 `cat /proc/apm` 检查 APM 是否处于活动状态。由多个数字组成的输出表示一切正常。现在应该能使用命令 `shutdown-h` 关闭计算机。

在引导提示符下以 `apm=<parameter>` 形式输入所有参数：

不完全符合标准的 BIOS 实施可能使 APM 出现问题。可以使用特殊的引导参数来避免某些问题。在引导提示符下以 `apm=parameter` 形式输入所有参数：

**on or off** 启用或禁用 APM 支持。

**(no-)allow-ints** 在执行 BIOS 功能时允许中断。

**(no-)broken-psr** BIOS 的“GetPowerStatus”功能工作不正常。

**(no-)realmode-power-off** 关闭前将处理器重设置为实际方式。

**(no-)debug** 在系统日志中记录 APM 事件。

**(no-)power-off** 在关闭后关闭系统电源。

**bounce-interval=*n*** 暂停事件后的一段时间（以百分之一秒为单位），在这段时间中将忽略其他暂停事件。

**idle-threshold=*n*** 系统不活动百分比，从这个百分比开始执行 BIOS 功能 `idle`（0 表示始终执行，100 表示从不执行）。

**idle-period=*n*** 开始测量系统活动前所经过的时间（以百分之一秒为单位）。

不再使用 APM 守护程序 (`apmd`)。其功能现在由新的 `powersaved` 处理，后者还支持 ACPI 和 CPU 频率调节。

## 16.3 ACPI

ACPI（高级配置和电源接口）支持操作系统设置和控制各个硬件部件。ACPI 可以取代 PnP 和 APM。它提供有关电池、AC 适配器、温度、风扇和系统事件（例如“合上机盖”或“电池电量低”）的信息。

BIOS 提供包含有关各个部件和硬件访问方法信息的表。操作系统使用这些信息执行指派中断或激活和取消激活部件等任务。因为操作系统执行 BIOS 中储存的命令，所以功能取决于 BIOS 实施。`/var/log/boot.msg` 中报告了 ACPI 能够检测并装载的表。有关对 ACPI 问题进行查错的详细信息，请参见第 16.3.4 节“查错”。

### 16.3.1 使用 ACPI

如果内核在引导系统时检测到 ACPI BIOS，则会自动激活 ACPI 并取消激活 APM。某些较旧的计算机可能需要引导参数 `acpi=on`。计算机必须支持 ACPI 2.0 或更高版本。检查 `/var/log/boot.msg` 中的内核引导讯息，了解是否已激活了 ACPI。

随后，必须装载多个模块。这是由 ACPI 守护程序的启动脚本完成的。如果这些模块中的任何一个模块引起问题，则可以在 `/etc/sysconfig/powersave/common` 中排除相应模块的装载或卸载。系统日志 (`/var/log/messages`) 包含模块的讯息，使您了解检测到了哪些部件。

`/proc/acpi` 目前包含多个文件，这些文件提供有关系统状态的信息，也可用于更改某些状态。某些功能仍在开发中，所以尚不能使用，而对某些功能的支持主要取决于制造商的实施。

通过 `cat` 可以读取所有文件（`dsdt` 和 `fadt` 除外）。在某些文件中，可以使用 `echo` 修改设置，例如 `echo X > file` 可以为 `X` 指定合适的值。请始终使用命令 `powersave` 访问此信息和控制选项。下面介绍一些最重要的文件：

`/proc/acpi/info` 有关 ACPI 的一般信息。

`/proc/acpi/alarm` 这里指定应将系统从休眠状态唤醒的时间。当前不完全支持此功能。

`/proc/acpi/sleep` 提供有关可能的休眠状态的信息。

`/proc/acpi/event` 在这里报告所有事件并由 Powersave 守护程序 (`powersaved`) 对这些事件进行处理。如果没有任何守护程序访问此文件，则可以通过 `cat/proc/acpi/event` 读取事件（例如，短暂单击 `(Power)` 按钮或合上机盖）（用 `(Ctrl)-C` 终止）。

`/proc/acpi/dsdt` 和 `/proc/acpi/fadt`  
这些文件包含 ACPI 表 DSDT（区分系统说明表和 FADT（固定 ACPI 说明表））。可以使用 `acpidmp`、`acpidisasm` 和 `dmdecode` 读取这些文件。包 `pmtools` 中提供了这些程序及其文档。例如，`acpidmpDSDT` | `acpidisasm`。

`/proc/acpi/ac_adapter/AC/state`  
显示是否连接了 AC 适配器。

`/proc/acpi/battery/BAT*/{alarm,info,state}`  
有关电池状态的详细信息。通过将 `info` 中的 `last full capacity` 与 `state` 中的 `remaining capacity` 进行比较来读取电量水平。一个更方便的方法是使用第 16.3.3 节“ACPI 工具”中引入的特殊程序之一。可以在 `alarm` 中指定电量水平，达到该电量水平将触发电池事件。

`/proc/acpi/button` 此目录包含有关各种开关的信息。

`/proc/acpi/fan/FAN/state` 显示风扇当前是否处于活动状态。将 0（开）或 3（关）写入此文件可以手工激活或取消激活风扇。但是，当设备变得过热时，内核中的 ACPI 代码和硬件（或 BIOS）将覆盖此设置。

`/proc/acpi/processor/CPU*/info`  
有关处理器省电选项的信息。

`/proc/acpi/processor/CPU*/power`  
有关当前处理器状态的信息。C2 旁边的星号表示处理器处于空闲状态。这是最常见的状态，可以从 `usage` 值中观察到。

`/proc/acpi/processor/CPU*/throttling`

可用于设置处理器时钟的节流。通常，可以将节流分为 8 个级别。节流独立于 CPU 的频率控制。

`/proc/acpi/processor/CPU*/limit`

如果守护程序自动控制性能（已过时）和节流，则可以在这里指定最大限制。某些限制是由系统确定的。某些限制可由用户进行调整。

`/proc/acpi/thermal_zone/` 每个热区有单独的子目录。热区是具有类似热属性的区域，其编号和名称由硬件制造商指定。但是，很少实施 ACPI 提供的许多功能。而温度控制通常是由 BIOS 处理的。因为关系到硬件的使用寿命，所以操作系统很少有机会进行干预。因此，下面的某些说明只具有理论价值。

`/proc/acpi/thermal_zone/*/temperature`

热区的当前温度。

`/proc/acpi/thermal_zone/*/state`

此状态指出一切是否 ok 或 ACPI 是采用 active 散热还是 passive 散热。对于独立于 ACPI 的风扇控制，此状态始终是 ok。

`/proc/acpi/thermal_zone/*/cooling_mode`

选择由 ACPI 控制的散热方法。选择被动散热方式（性能较低，但很经济）还是主动散热方式（全部性能，但有风扇噪音）。

`/proc/acpi/thermal_zone/*/trip_points`

允许您确定温度限制，达到这些温度限制将触发特定操作（例如，被动散热或主动散热、暂停(hot)或关闭(critical)）。DSDT 中定义了可能的操作（取决于设备）。ACPI 规范中确定的临界点是 critical、hot、passive、active1 和 active2。即使不是实施所有临界点，也必须始终在此文件中以此顺序输入它们。例如，项 `echo 90:0:70:0:0 > trip_points` 将 critical 的温度设置为 90，将 passive 的温度设置为 70（所有温度以摄氏度为单位）。

`/proc/acpi/thermal_zone/*/polling_frequency`

如果在温度改变时没有自动更新 temperature 中的值，请在这里切换巡回检测方式。使用命令 `echo X > /proc/acpi/thermal_zone/*/polling_frequency` 将每 X 秒查询一次温度。设置 X=0 禁用巡回检测。

不需要手工编辑这些设置、信息和事件。这可以通过 Powersave 守护程序 (powersaved) 和多个应用程序（例如 powersave、kpowersave 和 wmpowersave）来完成。请参见第 16.3.3 节“ACPI 工具”。因为 powersaved 包含了以前 acpid 的功能，所以不再需要 acpid。

### 16.3.2 控制 CPU 性能

CPU 可以采用三种省电方法。根据计算机的运行方式，还可以将这三种方法结合起来使用。省电还意味着系统温度不会升得过高并且激活风扇的频率会降低。

**频率和电压调节** PowerNow! 和 Speedstep 是 AMD 和 Intel 为这一技术指定的名称。但是，其他制造商的处理器中也应用了这一技术。CPU 的时钟频率及其核心电压同时降低，因而采用这一技术所节省的电量远远超过了线性省电量。这意味着，如果频率减半（一半的性能），所节省的电量远不止一半。此技术独立于 APM 或 ACPI，并需要为性能调节频率和电流需求的守护程序。可以在目录 `/sys/devices/system/cpu/cpu*/cpufreq/` 下进行这些设置。

**节流时钟频率** 此技术将忽略一定百分比的 CPU 时钟信号脉冲。如果节流 25%，则将忽略四分之一的脉冲，如果节流 87.5%，则只有八分之一的脉冲到达处理器。但是，采用这种方法所节省的电量稍微低于线性省电量。通常，只有在频率调节不可用或要最大程度节省电量时才使用节流。此技术也必须由特殊的进程控制。系统接口是 `/proc/acpi/processor/*/throttling`。

**使处理器进入休眠状态** 操作系统在处理器不执行任何任务时使处理器进入休眠状态。在这种情况下，操作系统向 CPU 发送一个 `halt` 命令。休眠状态有三种类型：C1、C2 和 C3。最经济的状态是 C3，在这种状态下，连处理器超速缓存与主存储器之间的同步都将被暂停。因此，只有在没有任何其他设备通过总线主控芯片活动修改主存储器的内容时才能应用此状态。某些驱动程序禁止使用 C3。当前状态显示在 `/proc/acpi/processor/*/power` 中。

只有当处理器忙时，才需要进行频率调节和节流，这是因为当处理器处于空闲状态时总是会应用最经济的 C 状态。如果 CPU 忙，则建议采用的省电方法是频率调节。处理器经常只在部分负载的状态下工作。在这种情况下，可以以较低的频率运行。通常，守护程序（例如 `powersaved`）控制的动态频率调节是最佳方法。如果使用电池工作或如果您想让计算机冷却或安静，则静态设置为低频率会非常有用。

节流应作为最后没有办法时采用的方法，例如，虽然系统负载很高，但为延长电池工作时间而采用节流。但是，如果节流程度过高，某些系统将不会正常运行。此外，如果 CPU 处理的任务量很少，则 CPU 节流就没什么作用。

在 SUSE LINUX 中，这些技术是由 Powersave 守护程序控制的。第 16.5 节“powersave 包”对此配置进行了说明。

### 16.3.3 ACPI 工具

一系列相对全面的 ACPI 实用程序包含这样一些工具：只显示信息（例如，电池电量水平和温度）的工具（`acpi`、`klaptopdaemon` 和 `wmacpimon` 等）、简化对 `/proc/acpi` 中的结构进行访问的工具或协助监视更改的工具（`akpi`、`acpiw` 和 `gtkacpiw`）以及用于编辑 BIOS 中 ACPI 表的工具（包 `pmtools`）。

### 16.3.4 查错

问题有两种不同的类型。一方面，内核的 ACPI 代码可能包含未及时检测出的错误。在这种情况下，可以通过下载获得解决方案。但问题经常是由 BIOS 引起的。有时，会故意将一些不符合 ACPI 规范的配置集成在 BIOS 中，用于避免其他常用操作系统中 ACPI 实施中的错误。在 ACPI 实施中有严重错误的硬件部件会被记录在一个黑名单中，防止 Linux 内核对这些部件使用 ACPI。

在遇到问题时，首先要做的是更新 BIOS。如果计算机根本未引导，则使用以下引导参数之一可能会解决问题：

`pci=noacpi` 不使用 ACPI 配置 PCI 设备。

`acpi=oldboot` 只执行简单的资源配置。不要将 ACPI 用于其他目的。

`acpi=off` 禁用 ACPI。

#### Warning

不使用 ACPI 引导会出现问题

某些较新的计算机（特别是 SMP 系统和 AMD64 系统）需要 ACPI 以正确配置硬件。在这些计算机上，禁用 ACPI 可能会产生问题。

#### Warning

引导后，用命令 `dmesg | grep -2i acpi` 来监视系统的引导讯息（或所有讯息，因为问题可能不是由 ACPI 引起的）。如果在分析 ACPI 表时发生错误，则可以用改进的版本替换最重要的表 DSDT。在这种情况下，将忽略 BIOS 中有问题的 DSDT。第 16.5.4 节“查错”中对这一过程进行了介绍。

在内核配置中，可以使用开关来激活 ACPI 调试讯息。如果已编译并安装了具有 ACPI 调试功能的内核，则支持对详细信息执行错误专家搜索。

如果遇到 BIOS 或硬件问题，则最好与制造商联系。特别是如果制造商不常对 Linux 提供支持，他们就应该面对这些问题。只有在制造商意识到有很多客户在使用 Linux 时，他们才会重视这一问题。

## 参考信息

有关 ACPI 的其他文档和帮助：

- <http://www.cpqlinux.com/acpi-howto.html> (详细的 ACPI HOWTO 文档, 包含 DSDT 增补程序)
- <http://www.intel.com/technology/iapc/acpi/faq.htm> (ACPI FAQ @Intel)
- <http://acpi.sourceforge.net/> (Sourceforge 中的 ACPI4Linux 项目)
- <http://www.poupinou.org/acpi/> (Bruno Ducrot 开发的 DSDT 增补程序)

## 16.4 硬盘的休眠

在 Linux 中, 如果不使用硬盘, 则可以使硬盘完全进入休眠状态, 或者在更经济或更安静的方式下运行。在目前的便携式计算机上, 您无需手工关闭硬盘, 因为硬盘会在不运行时自动进入经济的运行方式。但是, 如果要最大限度地省电, 请尝试使用以下一些方法。通过 powersaved 可以控制大多数功能。

hdparm 应用程序可用于修改多种硬盘设置。选项 `-y` 将硬盘立即切换到待机方式。`-Y` (小心使用) 使硬盘进入休眠状态。`hdparm -S x` 会使硬盘在一段时间 (这段时间内未活动) 后运行速度减慢。按以下规则替换  $\langle x \rangle$ : 0 表示禁用此机制, 让硬盘继续运行。值 1 到 240 表示的时间为所选的值乘以 5 秒。值 241 到 251 对应的时间分别是 30 分钟的 1 到 11 倍。

使用选项 `-B` 可以控制硬盘的内部省电选项。在 0 到 255 之间选择一个值, 0 表示最大省电方式, 255 表示最大吞吐量方式。结果取决于所使用的硬盘, 难以估算。要让硬盘安静一些, 请使用选项 `-M`。在 128 到 254 之间选择一个值, 128 表示最安静, 254 表示速度最快。

通常, 让硬盘进入休眠状态并不容易。在 Linux 中, 大量的进程对硬盘执行写操作, 因而会经常将其唤醒。因此, 一定要了解 Linux 如何处理需要写入硬盘的数据。首先, 在 RAM 中对所有数据进行缓冲。此缓冲区由内核更新守护程序 (kupdated) 进行监视。当数据达到一定的有效期限或缓冲区已被填充到一定程度时, 就会清理缓冲区, 将其中的内容写入硬盘。缓冲区大小是动态的, 取决于内存的大小和系统负载。默认情况下, 将 kupdated 设置为较短的时间间隔可以获得最好的数据完整性。它每 5 秒检查一次缓冲区, 当数据超过 30 秒或缓冲区

填充程度达到 30% 时，它会向 `bdflush` 守护程序发出通知。随后，`bdflush` 守护程序将数据写入硬盘。此守护程序还独立于 `kupdated` 写入数据，例如，当缓冲区已满时。

## Warning

对数据完整性的损害

更改内核更新守护程序设置将损害数据完整性。

## Warning

除了这些进程之外，日记文件系统（例如 ReiserFS 和 Ext3）独立于 `bdflush` 写入它们的元数据，这也会妨碍硬盘减慢运行速度。为了避免这种情况，已为移动设备开发了特殊的内核扩展。有关详细信息，请参见 `/usr/src/linux/Documentation/laptop-mode.txt`。

另一个重要因素是活动程序的行为方式。例如，好的编辑器会定期将当前已修改文件的隐藏备份写入硬盘，而这会唤醒磁盘。可以禁用此类功能，但这会影响数据的完整性。

在此连接中，邮件守护程序 `postfix` 使用变量 `POSTFIX_LAPTOP`。如果将此变量设为 `yes`，则 `postfix` 访问硬盘的频率将显著降低。但是，如果增加 `kupdated` 的时间间隔，则这样做没有什么作用。

## 16.5 powersave 包

在使用电池工作时，`powersave` 包负责便携式计算机的省电功能。它的某些功能还可用于普通工作站和服务器，例如暂停、待机、ACPI 按钮功能以及使 IDE 硬盘进入休眠状态。

此包包含计算机的所有电源管理功能。它支持使用 ACPI、APM、IDE 硬盘以及 PowerNow! 或 SpeedStep 技术的硬件。包 `apmd`、`acpid`、`ospm` 和 `cpufreqd`（现在是 `cpuspeed`）中的功能已被合并到 `powersave` 包中。因此不应将这些包中的守护程序与 `powersave` 守护程序同时运行。

即使您的系统未包含上面列出的所有硬件元素，也应使用 `powersave` 守护程序控制省电功能。因为 ACPI 和 APM 是互斥的，所以您的计算机上只能使用其中一个系统。此守护程序将自动检测硬件配置中的任何更改。



**Important**

有关 **powersave** 的信息

`/usr/share/doc/packages/powersave` 中还提供了有关 **powersave** 包的信息。

**Important**

## 16.5.1 配置 powersave 包

通常，**powersave** 的配置分布在多个文件中：

### `/etc/sysconfig/powersave/common`

此文件包含 **powersave** 守护程序的一般设置。例如，通过增大变量 `POWERSAVE_DEBUG` 的值可以增加（`/var/log/messages` 中）调试讯息的数量。

### `/etc/sysconfig/powersave/events`

**powersave** 守护程序需要此文件来处理系统事件。可以将事件指派给外部操作，也可以指派给守护程序本身执行的操作。对于外部操作，守护程序尝试运行 `/usr/lib/powersave/scripts/` 中的可执行文件。预定义的内部操作有：

- `ignore`
- `throttle`
- `dethrottle`
- `suspend_to_disk`
- `suspend_to_ram`
- `standby`
- `do_suspend_to_disk`
- `do_suspend_to_ram`
- `do_standby`

`throttle` 按照 `POWERSAVE_MAX_THROTTLING` 中定义的值减慢处理器的运行速度。此值取决于当前方案。`dethrottle` 设置处理器发挥全部性能运行。`suspend_to_disk`、`suspend_to_ram` 和 `standby` 触发休眠方式的系统事件。这三个操作通常负责触发休眠方式，但应始终将它们与特定的系统事件关联起来。

目录 `/usr/lib/powersave/scripts` 包含处理事件的脚本：

**notify** 通过控制台、X 窗口或声音信号发出的有关事件的通知。  
**screen\_saver** 激活屏幕保护程序。  
**switch\_vt** 如果暂停或待机后屏幕移位，则可以使用此脚本。  
**wm\_logout** 保存设置并从 GNOME、KDE 或其他窗口管理器中注销。  
**wm\_shutdown** 保存 GNOME 或 KDE 设置并关闭系统。

例如，如果设置了变量 `POWERSAVE_EVENT_GLOBAL_-SUSPEND2DISK="prepare_suspend_to_disk do_suspend_to_disk"`，则用户一将采用休眠方式 `suspend to disk` 的命令提供给 `powersaved`，就将按指定的顺序处理这两个脚本或操作。守护程序首先运行外部脚本 `/usr/lib/powersave/scripts/prepare_suspend_to_disk`。在成功处理此脚本后，守护程序运行内部操作 `do_suspend_to_disk`，然后在此脚本卸载了关键模块并停止了服务后，将计算机设置为休眠方式。

可以修改 `(sleep)` 按钮事件的操作，如 `POWERSAVE_EVENT_BUTTON_-SLEEP="notify suspend_to_disk"` 中所示。在这种情况下，外部脚本 `notify` 会将有关暂停的信息通知给用户。随后，生成事件 `POWERSAVE_EVENT_GLOBAL_-SUSPEND2DISK`，这导致执行上述操作并进入安全的系统暂停方式。使用 `/etc/sysconfig/powersave/common` 中的变量 `POWERSAVE_NOTIFY_METHOD` 可以自定义脚本 `notify`。

**/etc/sysconfig/powersave/cpufreq**  
包含用于优化动态 CPU 频率设置的变量。

**/etc/sysconfig/powersave/battery**  
包含电池电量限制和其他电池特定的设置。

**/etc/sysconfig/powersave/sleep**  
在此文件中，激活休眠方式并确定在暂停事件或待机事件之前应卸载哪些关键模块以及停止哪些服务。当系统恢复时，将重装载这些模块并重启动这些服务。您甚至可以延迟已触发的休眠方式（为了能够保存文件）。默认设置主要涉及 USB 和 PCMCIA 模块。暂停或待机失败通常是由某些模块引起的。有关确定错误的详细信息，请参见第 16.5.4 节“查错”。

**/etc/sysconfig/powersave/thermal**  
激活散热和热量控制。文件 `/usr/share/doc/packages/powersave/README.thermal` 中提供了有关此主题的详细信息。

**/etc/sysconfig/powersave/scheme\_\***  
提供多种耗电量与特定部署方案相适应的方案。许多方案都是预配置的，使用时无需进行更改。可以在这里保存自定义方案。

## 16.5.2 配置 APM 和 ACPI

### 暂停和待机

默认情况下，休眠方式处于非活动状态，原因是这些休眠方式在某些计算机上仍不工作。有三种基本的 ACPI 休眠方式和两种 APM 休眠方式：

**暂停到磁盘（ACPI S4、APM 暂停）** 将整个内存内容保存到硬盘。计算机完全关闭且不消耗任何电源。

### 暂停到 RAM（ACPI S3、APM 暂停）

将所有设备的状态保存到主存储器。只有主存储器继续消耗电源。

**待机（ACPI S1、APM 待机）** 关闭某些设备（取决于制造商）。

为了正确处理暂停、待机和恢复，确保在文件 `/etc/sysconfig/powersave/events` 中设置了以下默认选项（SUSE LINUX 安装之后的默认设置）：

```
POWERSAVE_EVENT_GLOBAL_SUSPEND2DISK=
    "prepare_suspend_to_disk do_suspend_to_disk"
POWERSAVE_EVENT_GLOBAL_SUSPEND2RAM=
    "prepare_suspend_to_ram do_suspend_to_ram"
POWERSAVE_EVENT_GLOBAL_STANDBY=
    "prepare_standby do_standby"
POWERSAVE_EVENT_GLOBAL_RESUME_SUSPEND2DISK=
    "restore_after_suspend_to_disk"
POWERSAVE_EVENT_GLOBAL_RESUME_SUSPEND2RAM=
    "restore_after_suspend_to_ram"
POWERSAVE_EVENT_GLOBAL_RESUME_STANDBY=
    "restore_after_standby"
```

### 自定义电池状态

文件 `/etc/sysconfig/powersave/battery` 中定义了三个电池电量水平（用百分比表示），在达到这些电池电量水平时，将触发系统警报或特定的操作。

```
POWERSAVED_BATTERY_WARNING=20
POWERSAVED_BATTERY_LOW=10
POWERSAVED_BATTERY_CRITICAL=5
```

配置文件 `/etc/sysconfig/powersave/events` 中定义了电池电量水平降到指定限制之下时要执行的操作或脚本。可以按第16.5.1节“配置 powersave 包”中所述修改按钮的标准操作。

```
POWERSAVE_EVENT_BATTERY_NORMAL="ignore"  
POWERSAVE_EVENT_BATTERY_WARNING="notify"  
POWERSAVE_EVENT_BATTERY_LOW="notify"  
POWERSAVE_EVENT_BATTERY_CRITICAL="wm_shutdown"
```

### 调整耗电量以适应各种情况

可以根据电源类型调整系统行为。当系统从 AC 电源断开并使用电池运行时，应降低系统的耗电量。同样，系统一连接到 AC 电源，性能就应自动提高。CPU 频率、IDE 省电功能和许多其他参数都可以进行修改。

`/etc/sysconfig/powersave/events` 中定义了计算机从 AC 电源断开或连接到 AC 电源时要执行的操作。在 `/etc/sysconfig/powersave/common` 中选择要使用的方案：

```
POWERSAVE_AC_SCHEME="performance"  
POWERSAVE_BATTERY_SCHEME="powersave"
```

这些方案被储存在 `/etc/sysconfig/powersave` 下的文件中。文件名采用格式 `scheme_nameofthescheme`。本示例涉及两个方案：`scheme_performance` 和 `scheme_powersave`。`performance`、`powersave`、`presentation` 和 `acoustic` 都是预配置的。借助于 YaST 电源管理模块（请参见第 16.6 节“YaST 电源管理模块”），可以编辑、创建、删除现有的方案，也可以将其与不同的电源状态关联起来。

### 16.5.3 其他 ACPI 功能

如果您使用 ACPI，则可以控制系统对 *ACPI 按钮*（电源、休眠、机盖打开和机盖合上）的响应。在 `/etc/sysconfig/powersave/events` 中配置操作的执行。有关各个选项的解释，请参考此配置文件。

```
POWERSAVE_EVENT_BUTTON_POWER="wm_shutdown"
```

当按下电源按钮时，系统关闭相应的窗口管理器（KDE、GNOME、fvwm 等）。

```
POWERSAVE_EVENT_BUTTON_SLEEP="suspend_to_disk"
```

当按下休眠按钮时，系统会被设置为暂停到磁盘方式。

```
POWERSAVE_EVENT_BUTTON_LID_OPEN="ignore"
```

当机盖打开时，不执行任何操作。

`POWERSAVE_EVENT_BUTTON_LID_CLOSED="screen_saver"`

当机盖合上时，激活屏幕保护程序。

如果在指定时间内 CPU 负载未超过指定的限制，则可以进一步限制 CPU 性能。在 `POWERSAVED_CPU_LOW_LIMIT` 中指定负载限制，在 `POWERSAVED_CPU_IDLE_TIMEOUT` 中指定超时。

## 16.5.4 查错

文件 `/var/log/messages` 中记录了所有错误讯息和警报。如果您未能找到所需的信息，请使用文件 `/etc/sysconfig/powersave/common` 中的 `DEBUG` 增加 `powersave` 讯息的详细程度。请将变量的值增加到 7，或甚至增加到 15，然后重新启动守护程序。`/var/log/messages` 中更详细的错误讯息应有助于您找到错误。以下几节介绍 `powersave` 最常见的问题。

硬件支持已激活 **ACPI**，但功能不工作

如果使用 **ACPI** 时遇到问题，请使用以下命令在 `dmesg` 的输出中搜索 **ACPI** 特定的讯息：`dmesg|grep-i acpi`。可能需要更新 BIOS 来解决问题。请转到便携式计算机制造商的主页，查找已更新的 BIOS 版本，然后安装它。要求制造商遵循最新的 **ACPI** 规范。如果在更新 BIOS 后错误仍然存在，则按以下步骤用已更新的 `DSDT` 替换 BIOS 中有问题的 `DSDT` 表。

1. 从 <http://acpi.sourceforge.net/dsdt/tables> 为您的系统下载 `DSDT`。检查是否已解压缩并编译了此文件，如果文件扩展名是 `.aml` (**ACPI** 机器语言)，则表明已完成这些操作。如果是这种情况，请继续执行第 3 步。
2. 如果已下载表的文件扩展名是 `.asl` (**ACPI** 源语言)，则必须使用 `iasl` (包 `pmttools`) 对其进行编译。为此，请输入命令 `iasl -sa file.asl`。<http://developer.intel.com/technology/iapc/acpi/downloads.htm> 提供了最新版本的 `iasl` (**Intel ACPI** 编译器)
3. 将文件 `DSDT.aml` 复制到任何位置 (建议的位置为 `/etc/DSDT.aml`)。编辑 `/etc/sysconfig/kernel` 并相应地调整指向 `DSDT` 文件的路径。启动 `mkinitrd` (包 `mkinitrd`)。您卸载内核并使用 `mkinitrd` 创建 `initrd` 后，引导系统时就会集成并装载已修改的 `DSDT`。

## CPU 频率不工作

请参考内核源代码 (kernel-source) 查看是否支持您的处理器。您可能需要特殊内核模块或模块选项来激活 CPU 频率控制。/usr/src/linux/Documentation/cpu-freq/\* 中提供了此信息。如果需要特殊模块或模块选项，则通过变量 CPUFREQD\_MODULE 和 CPUFREQD\_MODULE\_OPTS 在文件 /etc/sysconfig/powersave/cpufreq 中进行配置。

## 暂停和待机不工作

有多个与内核有关的问题会妨碍在 ACPI 系统上使用暂停和待机：

- 目前，具有超过 1 GB RAM 的系统不支持暂停。
- 目前，多处理器系统和使用 P4 处理器（有超线程功能）的系统不支持暂停。

错误还可能是有问题的 DSDT 实施 (BIOS) 引起的。如果是这种情况，则安装新的 DSDT。

在 ACPI 和 APM 系统上：当系统尝试卸载有问题的模块时，系统被阻止执行操作或暂停事件未被触发。如果您未卸载模块或停止阻止成功暂停的服务，也会发生相同的情况。在这两种情况下，尝试确定阻止采用休眠方式的有问题的模块。/var/log/sleepmode 中的 powersave 守护程序生成的日志文件对确定有问题的模块很有用。如果计算机未进入休眠方式，则原因在最后卸载的模块上。请配置 /etc/sysconfig/powersave/sleep 中的下列设置以在暂停或待机前卸载有问题的模块。

```
POWERSAVE_UNLOAD_MODULES_BEFORE_SUSPEND2DISK=""
POWERSAVE_UNLOAD_MODULES_BEFORE_SUSPEND2RAM=""
POWERSAVE_UNLOAD_MODULES_BEFORE_STANDBY=""
POWERSAVE_SUSPEND2DISK_RESTART_SERVICES=""
POWERSAVE_SUSPEND2RAM_RESTART_SERVICES=""
POWERSAVE_STANDBY_RESTART_SERVICES=""
```

如果您在不断变化的网络环境中使用暂停或待机，或将暂停或待机用于远程装入的文件系统（例如 Samba 和 NIS），请使用 automounter 装入它们，或在上述变量中添加相应的服务，例如 smbfs 或 nfs。在远程装入的文件系统进入暂停或待机前，如果某个应用程序访问此文件系统，则无法正确停止服务且无法正确卸载该文件系统。在恢复系统后，文件系统可能被损坏，因此必须重装入文件系统。

使用 ACPI，但 Powersave 不通知电池电量限制

使用 ACPI，操作系统可以在电池电量水平低于某个特定限制时请求 BIOS 发送讯息。此方法的优点是，不需要经常巡回检测电池状态，巡回检测会降低计算机的性能。但是，当电量水平低于指定的限制时，有可能不进行此通知，即使我们认为 BIOS 支持此功能。如果您的系统上发生这种情况，请将文件 `/etc/sysconfig/powersave/battery` 中的变量 `POWERSAVED_FORCE_BATTERY_POLLING` 设置为 `yes` 以强制执行电池巡回检测。

## 16.6 YaST 电源管理模块

YaST 电源管理模块可以配置上述所有电源管理设置。当通过‘系统’→‘电源管理’从 YaST 控制中心启动此模块时，会打开此模块的第一个对话框。图16.1 “方案选择”中显示了这一工具。

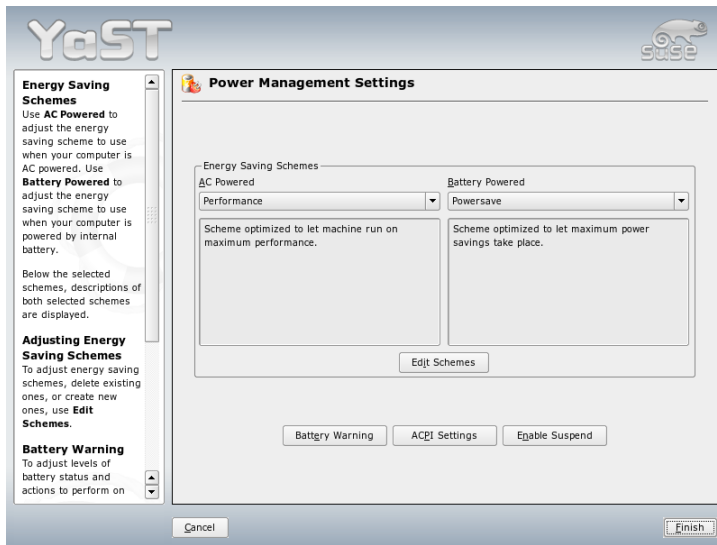


Figure 16.1: 方案选择

在此对话框中，选择使用电池运行和使用 AC 运行时要使用的方案。要添加或修改方案，请单击‘编辑方案’，这将打开现有方案的概述，如图16.2 “现有方案的概述”所示。

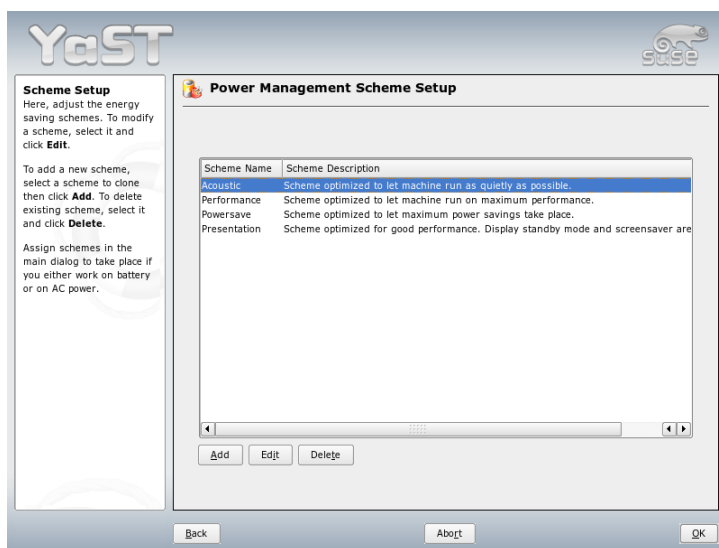


Figure 16.2: 现有方案的概述

在方案概述中，选择要修改的方案，然后单击‘编辑’。要创建新方案，请单击‘添加’。这两种情况打开的是同一个对话框，如图16.3“添加方案”所示。

首先，为新方案或已编辑的方案输入合适的名称和说明。确定此方案是否应控制CPU性能及如何控制CPU性能。确定是否应使用频率调节和节流以及调节和节流的程度。随后的对话框是针对硬盘的，它定义了‘待机策略’是为实现最佳性能还是为实现最大省电。‘声音策略’控制硬盘的噪音级别（少数几种硬盘支持这一功能）。‘散热策略’确定要使用的散热方法。遗憾的是BIOS很少支持这种类型的热量控制。请阅读 `/usr/share/doc/packages/powersave/README.thermal` 以了解如何使用风扇和被动散热方法。单击‘下一步’进入配置已连接显示器省电方式的对话框。选中复选框‘启用屏幕保护程序’以在不使用计算机时降低显示器的耗电量。‘启用显示器电源管理’用于控制显示器进入待机、暂停或关闭电源方式前所经过的时间。完成此方案的所有设置，然后单击‘确定’返回到起始对话框。在起始对话框中，将自定义方案指派给两个运行方式中的一个。要激活您的设置，请单击‘确定’退出此对话框。

可以在最初的对话框中使用‘电池警告’、‘ACPI设置’或‘启用暂停’进行全局电源管理设置。单击‘电池警告’可以访问电池电量水平对话框，如图16.4“电池电量水平”所示。



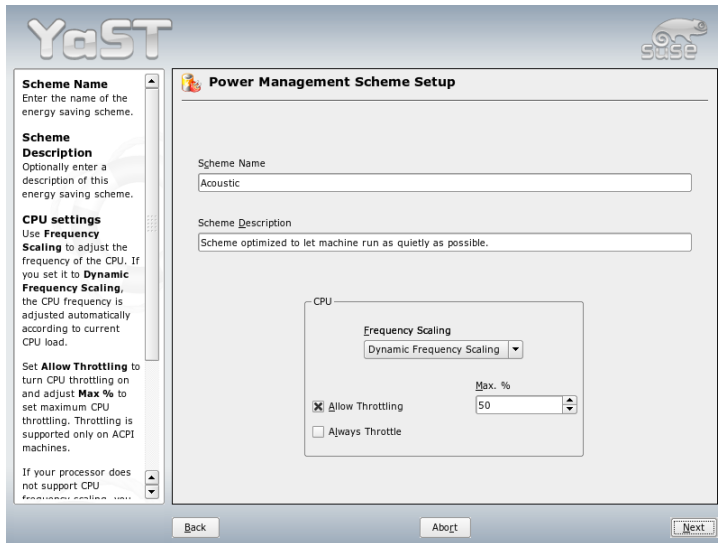


Figure 16.3: 添加方案

每当电量水平降至特定的可配置限制之下时，系统的 BIOS 就会通知操作系统。在此对话框中，定义三个限制：‘警告电量’、‘低电量’和‘关键电量’。当电量水平降至这些限制之下时将触发特定的操作。通常，前两个状态只触发发送给用户的通知。第三个关键电量水平触发关闭操作，原因是剩余的电量不足以支持系统继续运行。选择合适的电量水平和所需的操作，然后单击‘确定’返回到起始对话框。

通过‘ACPI 设置’访问用于配置 ACPI 按钮的对话框。图16.5 “ACPI 设置”中显示了这一工具。对 ACPI 按钮进行的设置确定系统应如何对特定的开关进行响应。配置对按下电源按钮、按下休眠按钮和合上便携式计算机盖的系统响应。单击‘确定’完成配置并返回到起始对话框。

单击‘启用暂停’进入一个对话框，可以在这个对话框中确定此系统的用户是否及如何使用暂停或待机功能。单击‘确定’返回到主对话框。再次单击‘确定’退出此模块并确认您的电源管理设置。

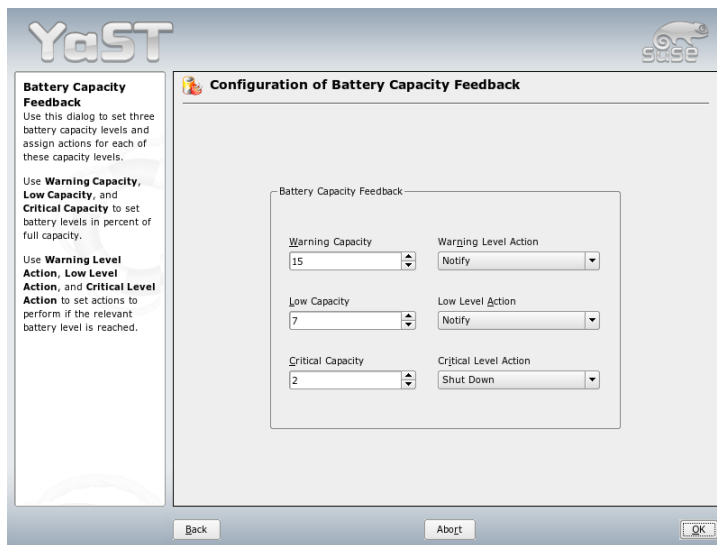


Figure 16.4: 电池电量水平

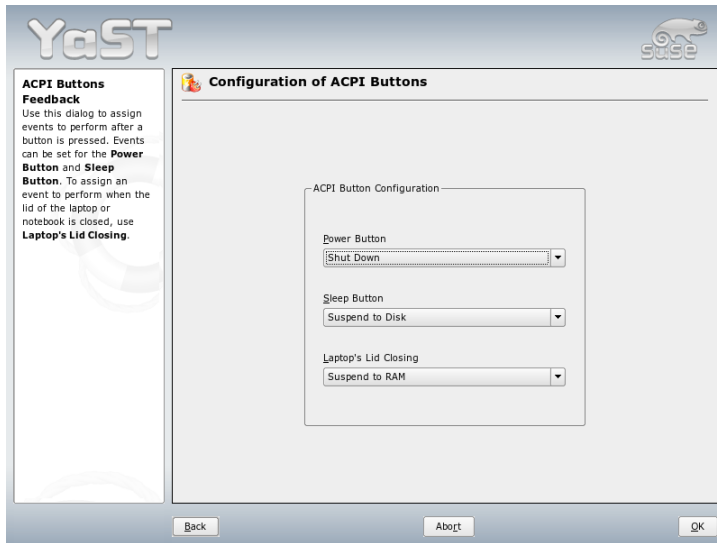


Figure 16.5: ACPI 设置



# 无线通讯

可以通过多种方法使用您的 Linux 系统与其他计算机、手提电话或外围设备进行通讯。WLAN（无线 LAN）可用于将便携式计算机联网。蓝牙可用于将单独的系统部件（鼠标、键盘）、外围设备、手提电话、PDA 和单独的计算机互相连接。IrDA 通常用于与 PDA 或手提电话的通讯。本章介绍所有这些技术和它们的配置。

17.1	无线 LAN . . . . .	276
17.2	蓝牙 . . . . .	283
17.3	红外线数据传送 . . . . .	291

## 17.1 无线 LAN

无线 LAN 已成为移动计算不可或缺的一个方面。当今，大多数笔记本电脑都配有内置 WLAN 卡。用于 WLAN 卡无线通讯的 802.11 标准是由 IEEE 组织制订的。最初，此标准实现的最大传送速率是 2 MBit/s。此后，此标准进行了多次补充以提高数据传送速率。这些补充定义了调制、传送输出和传送速率等详细内容：

*Table 17.1: 各种 WLAN 标准的概述*

名称	频 带 (GHz)	最大传送速率 (MBit/s)	注释
802.11	2.4	2	已过时；目前市场上不销售采用此标准的最终设备
802.11b	2.4	11	广泛采用
802.11a	5	54	较少使用
802.11g	2.4	54	向后兼容 11b

此外还有一些专有标准，如 Texas Instruments 对 802.11b 进行调整后形成的标准（有时也称为 802.11b+），其最大传送速率为 22 MBit/s。但采用这种标准的卡的普及程度有限。

### 17.1.1 硬件

SUSE LINUX 不支持 802.11 卡。它支持采用 802.11a、802.11b 和 802.11g 的大多数卡。现在新推出的卡通常符合 802.11g 标准，但市面上仍可买到采用 802.11b 的卡。通常，支持具有以下芯片的卡：

- Lucent/Agere Hermes
- Intel PRO/Wireless 2100、2200BG、2915ABG
- Intersil Prism2/2.5/3
- Intersil PrismGT
- Atheros 5210、5211、5212

- Atmel at76c502、at76c503、at76c504、at76c506
- Texas Instruments ACX100、ACX111

还支持许多很少用到且市面上不再有售的较早的卡。*AbsoluteValue Systems* 的万维网站点中提供了一个列表，详尽列出了各种 WLAN 卡及其使用的芯片，网址是：[http://www.linux-wlan.org/docs/wlan\\_adapters.html.gz](http://www.linux-wlan.org/docs/wlan_adapters.html.gz)。<http://wiki.uni-konstanz.de/wiki/bin/view/Wireless/ListeChipsatz> 提供了各种 WLAN 芯片的概述。

有些卡需要固件映像，在初始化驱动程序时必须将固件映像装载到卡中。Intersil PrismGT、Atmel ACX100、ACX111 就是这种情况。使用 YaST 联机更新可以方便地安装固件。Intel PRO-Wireless 卡的固件附带了 SUSE LINUX，在检测到此类型的卡时，YaST 将立即自动安装该卡。已安装系统的 `/usr/share/doc/packages/wireless-tools/README.firmware` 中提供了有关此主题的详细信息。

没有本机 Linux 支持的卡可以通过运行 `ndiswrapper` 应用程序来使用。`ndiswrapper` 使用随大多数 WLAN 卡附带的 Windows 驱动程序。有关 `ndiswrapper` 的说明位于 `/usr/share/doc/packages/ndiswrapper/README.SUSE` 之下（前提是已安装 `ndiswrapper` 软件包）。有关 `ndiswrapper` 的深入信息，请参考该项目的万维网站点 <http://ndiswrapper.sourceforge.net/support.html>。

## 17.1.2 功能

本节介绍无线网络的基本信息。您可了解不同的操作方式、鉴定和加密类型。

### 操作方式

无线网络基本上可分为受控网络和特殊网络。受控网络具有一个管理元素：访问点。在这种方式（也称为基础结构方式）下，网络中 WLAN 工作站的所有连接均在此访问点上运行，此访问点可能还充当到 Ethernet 的连接。特殊网络没有访问点。各个工作站直接互相通讯。在特殊网络中，传送范围和参与工作站的数目都受到很大限制。因此，采用访问点通常更加有效。甚至可以将 WLAN 卡用作访问点。大多数卡支持此功能。

与使用缆线连接的网络相比，无线网络中的数据更容易被截获，无线网络更容易受到攻击，所以各标准都包括了鉴定和加密方法。IEEE 802.11 标准最初的版本在术语 WEP 下对这些方法进行了说明。但是，WEP 被证明是不安全的（请参见第 17.1.5 节“安全性”），因此 WLAN 行业（组织名为 *Wi-Fi 联盟*）制订了一个名为 WPA 的新扩展，用以弥补 WEP 的缺陷。后来的 IEEE 802.11i 标准（也称为 WPA2，因为 WPA 基于 802.11i 的草案版本）包括 WPA 和其他一些鉴定和加密方法。

## 鉴定

为了确保只有经过授权的工作站才能连接，受控网络中使用了多种鉴定机制：

**打开** 开放系统是不要求鉴定的系统。任何工作站都可以加入网络。不过，可以使用 WEP 加密（请参见第 17.1.2 节“加密”）。

**共享密钥（按照 IEEE 802.11）** 在此过程中，使用 WEP 密钥来进行鉴定。但不建议采用此过程，因为它使 WEP 密钥更容易受到攻击。只要攻击者对工作站和访问点之间的通讯监听足够长的时间，他们就有条件进行攻击。在鉴定过程中，双方将交换相同的信息，一次使用的是加密形式，一次使用的是未加密形式。因此，使用适当的工具就可以重新构建出密钥。因为这种方法使用 WEP 密钥进行鉴定和加密，所以它并没有提高网络的安全性。具有正确 WEP 密钥的工作站可以执行鉴定、加密和解密。不具有密钥的工作站无法解密接收到的包。因此，无论它是否必须对本身进行鉴定都不能进行通讯。

**WPA-PSK（按照 IEEE 802.1x）** WPA-PSK（PSK 代表“预共享密钥”）的工作方式与共享密钥过程类似。所有参与工作站和访问点需要相同的密钥。密钥长度为 256 位，通常以通行密码的形式输入。此系统不需要像 WPA-EAP 那样的复杂密钥管理，更适合个人使用。因此，有时将 WPA-PSK 称为“家庭”WPA。

**WPA-EAP（按照 IEEE 802.1x）** 实际上，WPA-EAP 不是一个鉴定系统，而是一个传输鉴定信息的协议。WPA-EAP 用于保护企业中的无线网络。在个人网络中，很少使用 WPA-EAP。因此，有时将 WPA-EAP 称为“企业”WPA。

## 加密

有多种加密方法可以确保未经授权的用户不能读取无线网络中交换的数据包，也不能访问此网络：

**WEP（在 IEEE 802.11 中定义）** 此标准使用 RC4 加密算法，最初密钥长度为 40 位，后来也使用 104 位的密钥。通常，将此长度声明为 64 位或 128 位，这取决于是否包括初始化矢量的 24 位。但是，此标准有一些缺陷。攻击者能够成功攻击此系统生成的密钥。不过，使用 WEP 总比根本不对网络进行加密要好。

**TKIP（在 WPA/IEEE 802.11i 中定义）**

WPA 标准中定义的这一密钥管理协议使用与 WEP 相同的加密算法，但弥补了其缺陷。它为每个数据包生成一个新密钥，从而有效阻止了对这些密钥的攻击。TKIP 与 WPA-PSK 一起使用。



CCMP（在 IEEE 802.11i 中定义）CCMP 对密钥管理进行了说明。通常，它与 WPA-EAP 一起使用，但它也可以与 WPA-PSK 一起使用。加密依照 AES 进行，该加密比 WEP 标准的 RC4 加密更强大。

### 17.1.3 用 YaST 进行配置

要配置您的无线网卡，请启动 YaST ‘网卡’模块。在‘网络地址设置’中，选择设备类型‘无线’，然后单击‘下一步’。

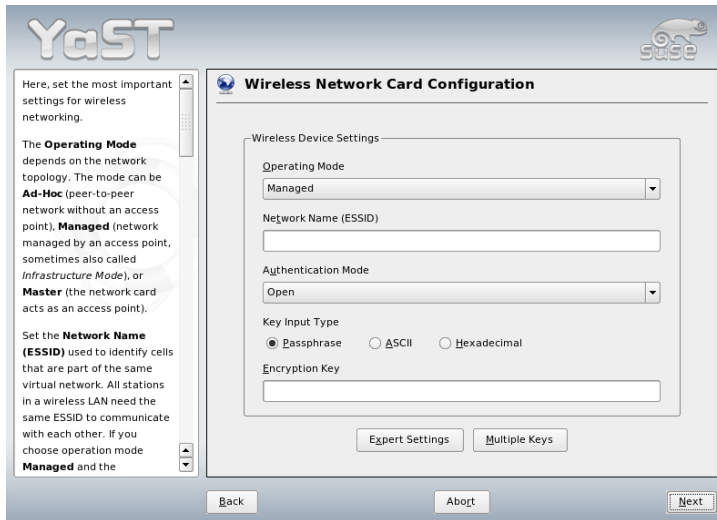


Figure 17.1: YaST: 配置无线网卡

在‘无线网卡配置’（如图 17.1 “YaST: 配置无线网卡” 所示）进行 WLAN 操作的基本设置：

**操作方式** 我们可以将工作站以三种不同的方式集成到 WLAN 中。哪种方式合适取决于要在其中进行通讯的网络：‘特殊’（对等网络，无访问点）、‘受控’（网络由访问点管理）或‘主’（您的网卡将充当访问点）。

**网络名称 (ESSID)** 为实现相互通讯，无线网络中的所有工作站都需要相同的 ESSID。如果未指定任何内容，则网卡会自动选择一个访问点，但它可能不是您所希望使用的。

**鉴定方式** 为您的网络选择合适的鉴定方法：‘开放’、‘共享密钥’或‘WPA-PSK’。如果选择了‘WPA-PSK’，则必须设置网络名称。

**专家设置** 单击此按钮将打开一个对话框，用于对 WLAN 连接进行详细配置。下面的内容提供了此对话框的详细说明。

完成基本设置后，即可将您的工作站部署在 WLAN 中。

## Important

### 无线网络中的安全性

确保使用所支持的鉴定和加密方法之一来保护您的网络通讯。如果未加密 WLAN 连接，则第三方便可以截获所有网络数据。即使进行弱加密 (WEP) 也比根本不加密要好。相关信息请参考第 17.1.2 节“加密”和第 17.1.5 节“安全性”。

## Important

根据所选的鉴定方法，YaST 会提示您在另一个对话框中微调这些设置。对于‘开放’，无需进行任何配置，因为此设置实施的是无需鉴定的未加密操作。

**WEP 密钥** 设置密钥输入类型。选择‘通行密码’、‘ASCII’或‘十六进制’。您最多可以保留 4 个不同的密钥来加密所传送的数据。单击‘多个密钥’进入密钥配置对话框。选择密钥的长度：‘128 位’或‘64 位’。默认设置是‘128 位’。在对话框底部的列表区域中，最多可以指定 4 个不同的密钥，您的工作站将使用这些密钥进行加密。按‘设置默认密钥’可将其中一个密钥定义为默认密钥。除非更改默认设置，否则 YaST 会将第一个输入的密钥用作默认密钥。如果删除了标准密钥，则必须将其他密钥中的一个手工标记为默认密钥。单击‘编辑’可以修改现有列表项或创建新密钥。此时将出现一个弹出窗口，提示您选择输入类型（‘通行密码’、‘ASCII’或‘十六进制’）。如果选择的是‘通行密码’，则输入一个单词或字符串，将从该单词或字符串按照先前指定的长度生成密钥。‘ASCII’要求为 64 位密钥输入 5 个字符，为 128 位密钥输入 13 个字符。如果选择的是‘十六进制’，则按照十六进制表示法为 64 位密钥输入 10 个字符，或为 128 位密钥输入 26 个字符。

**WPA-PSK** 要输入用于 WPA-PSK 的密钥，请选择输入方法‘通行密码’或‘十六进制’。在‘通行密码’方式下，输入必须为 8 到 63 个字符。在‘十六进制’方式下，请输入 64 个字符。

单击‘专家设置’可退出 WLAN 连接的基本配置对话框并进入专家配置对话框。此对话框中有以下选项可用：

**通道** 只有在‘特殊’和‘主’方式下才需要指定 WLAN 工作站要在其上工作的通道。在‘受控’方式下，网卡将自动搜索访问点的可用通道。在‘特殊’方式下，可从所提供的 12 个通道中选择一个，用于在您的工作站和其他工作站之间进行通讯。在‘主’方式下，确定您的网卡应该在哪个通道上提供访问点功能。此选项的默认设置是‘自动’。

**位速率** 根据网络的性能，您可能要为从一点到另一点之间的传送设置特定速率。在默认设置‘自动’中，系统会尽可能地使用最高数据传送速率。一些 WLAN 卡不支持位速率设置。

**访问点** 在具有多个访问点的环境中，通过指定 MAC 地址可以预先选择多个访问点中的一个。

**使用电源管理** 当您在旅途中时，使用省电技术可以最大限度地延长电池的工作时间。有关电源管理的详细信息，请参考第16章 电源管理。

## 17.1.4 实用程序

hostap (hostap 软件包) 用于将 WLAN 卡作为访问点运行。有关此软件包的详细信息，请访问项目主页 (<http://hostap.epitest.fi/>)。

kismet (kismet 软件包) 是一个网络诊断工具，使用它可以监听 WLAN 数据包通讯。通过这种方式，您可以检测到网络中的任何侵入企图。有关详细信息，请参见手册页和 <http://www.kismetwireless.net/>。

## 17.1.5 建立 WLAN 的提示和技巧

了解如何精确调整 WLAN 的速度、稳定性和安全性。

### 稳定性和速度

无线网络的性能和可靠性主要取决于参与工作站是否能够接收到来自其他工作站的清楚信号。障碍物（例如，墙壁）极大地削弱了信号。信号强度越低，传送速率就越慢。在网络运行过程中，可以在命令行（Link Quality 字段）中使用 iwconfig 实用程序或在 KDE 中使用 kwifimanager 来检查信号强度。如果信号质量存在问题，可尝试将设备放在其他位置，或调整访问点天线的位置。很多 PCMCIA WLAN 卡都配有辅助天线，可充分提高接收效果。制造商指定的速率（例如 54 MBit/s）是一个额定值，仅表示理论最大值。实际上，最大数据吞吐量不到该值的一半。

## 安全性

如果要建立一个无线网络，则一定要记住，如果不实施任何安全措施，则传送范围内的任何人都可以方便地访问此网络。因此，一定要激活某种加密方法。所有 WLAN 卡和访问点都支持 WEP 加密。虽然这并非完全安全，但还是对潜在攻击者设置了一道屏障。WEP 通常可满足个人使用。WPA-PSK 的安全性更好，但不能在较早的访问点或具有 WLAN 功能的路由器中实施。在某些设备上，可以通过固件更新来实施 WPA。此外，Linux 并非在所有硬件部件上都支持 WPA。在准备此文档时，WPA 只适用于使用 Atheros 或 Prism2/2.5/3 芯片的卡。On the latter, WPA only works if the hostap driver is used (see 第17.1.6节“有关 Prism2 卡的问题”)。If WPA is not available, WEP is better than no encryption. 在具有高级安全要求的企业中，无线网络工作时必须采用 WPA。

### 17.1.6 查错

如果您的 WLAN 卡不能响应，则应检查是否下载了所需固件。请参考第17.1.1节“硬件”。以下几段介绍了一些常见问题。

#### 多个网络设备

Modern laptops usually have a network card and a WLAN card. If you configured both devices with DHCP (automatic address assignment), you may encounter problems with the name resolution and the default gateway. 可以 Ping 路由器但不能浏览因特网就是这方面问题的典型示例。位于 <http://portal.suse.com> 的支持数据库提供了一篇有关这一主题的文章。要查找此文章，可以在搜索对话框中输入“DHCP”。

#### 有关 Prism2 卡的问题

采用 Prism2 芯片的设备有多个驱动程序可用。不同的卡与不同的驱动程序之间的适用性是不一样的。使用这些卡时，只有在使用 hostap 驱动程序时，才能实施 WPA。如果这样的卡不能正常工作或根本不工作，或者您要使用 WPA，请参见 </usr/share/doc/packages/wireless-tools/README.prism2>。

#### WPA

SUSE LINUX 中首次实施了 WPA 支持。在 Linux 中，WPA 支持仍处于开发阶段。因此，YaST 只允许配置 WPA-PSK。许多卡都不支持 WPA。要支持 WPA，需要对这些卡中的一些进行固件更新。如果要使用 WPA，请参见 </usr/share/doc/packages/wireless-tools/README.wpa>。

### 17.1.7 参考信息

Jean Tourrilhes 开发了用于 Linux 的无线工具，他的因特网网页上有很多关于无线网络的有用信息。请参见 [http://www.hpl.hp.com/personal/Jean\\_Tourrilhes/Linux/Wireless.html](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html)。

## 17.2 蓝牙

蓝牙是用于连接各种设备（例如，移动电话、PDA、外围设备、便携式计算机，或者键盘或鼠标等系统部件）的无线技术。蓝牙 (Bluetooth) 取自丹麦国王 Harold Bluetooth 的名字，正是他结束了斯堪的纳维亚半岛分裂混战的状态。蓝牙徽标是以 Rune（古代北欧文字）的“H”（像一颗星星）和“B”的组合为基础设计的。

蓝牙在多个重要方面区别于 IrDA。首先，各个设备不需要直接“看到”对方；其次，可以在网络中连接多个设备。但最大数据传送速率是 720 Kbps（在当前版本 1.2 中）。理论上，蓝牙甚至可以穿墙进行通讯。但实际上，这取决于墙和设备类的属性。有三种设备类的传送范围在 10 到 100 米之间。

### 17.2.1 基础

以下几节介绍了蓝牙的基本工作原理。了解需要符合哪些软件要求、蓝牙如何与系统交互以及蓝牙配置文件的工作原理。

#### 软件

为了能使用蓝牙，您需要蓝牙适配器（内置适配器或外部设备）、驱动程序和蓝牙协议堆栈。Linux 内核已包含使用蓝牙所需的基本驱动程序。Bluez 系统用作协议堆栈。要确保应用程序能使用蓝牙，必须安装基础包 `bluez-libs` 和 `bluez-utils`。这些包提供了多个所需的服务和实用程序。此外，某些适配器（Broadcom 和 AVM BlueFritz!）要求安装 `bluez-firmware` 包。`bluez-cups` 包支持通过蓝牙连接进行打印。

#### 常规交互

蓝牙系统由 4 个提供所需功能的互相关联的层组成：

**硬件** 实现 Linux 内核支持所需的适配器和合适的驱动程序。

**配置文件** 用于控制蓝牙系统。

**守护程序** 由配置文件控制并提供功能的服务。

**应用程序** 应用程序允许用户使用和控制守护程序提供的功能。

当插入蓝牙适配器时，热插拔系统将装载相应的驱动程序。在装载驱动程序后，系统检查配置文件以查看是否要启动蓝牙。如果要启动蓝牙，则它确定要启动的服务。根据此信息，启动相应的守护程序。安装时检测蓝牙适配器。如果找到一个或多个蓝牙适配器，则启用蓝牙。否则取消激活蓝牙系统。此后添加的任何蓝牙设备都必须手工启用。

## 配置文件

在蓝牙中，服务是通过配置文件（例如，文件传送配置文件、基本打印配置文件和个域网配置文件）进行定义的。为使某台设备能使用另一台设备的服务，这两台设备都必须了解同一个配置文件—设备包和手册中通常缺少的一些信息。遗憾的是，某些制造商没有严格遵守各配置文件的定义。不过，设备间的通讯通常能顺畅进行。

在下文中，本地设备是在物理上与计算机连接的那些设备。而只能通过无线连接进行访问的所有其他设备都被称为远程设备。

## 17.2.2 配置

本节介绍蓝牙配置。了解涉及哪些配置文件、需要哪些工具以及如何使用 YaST 或手工配置蓝牙。

### 使用 YaST 配置蓝牙

使用 YaST 蓝牙模块（如图 17.2 “YaST 蓝牙配置” 中所示）在您的系统中配置蓝牙支持。热插拔在检测到系统中的蓝牙适配器（例如在引导过程中，或如果插入了一个适配器）后，蓝牙立即使用蓝牙模块中配置的设置自动启动。

配置的第一步是确定在您的系统中是否要启动蓝牙服务。如果已启用了蓝牙服务，则可以配置两项内容。首先配置‘设备名’。这是您的计算机被发现时其他设备显示的名称。可以有两个占位符—%h 代表系统的主机名（很有用，例如，如果它是 DHCP 动态指派的），而 %d 插入接口号（只有在计算机中安装了多个蓝牙适配器才有用）。例如，如果在此字段中输入 Laptop %h 且 DHCP 将名称 unit123 指派给计算机，则其他远程设备将您的计算机识别为 Laptop unit123。

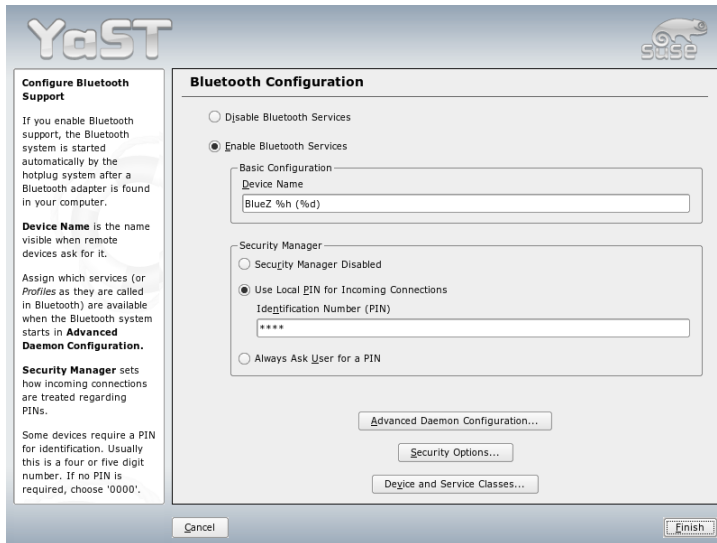


Figure 17.2: YaST 蓝牙配置

第二个参数‘安全性管理器’与远程设备尝试连接时本地系统的行为相关。不同之处是在对 PIN 号码的处理上。这种处理要么允许任何设备不使用 PIN 进行连接，要么确定在需要 PIN 时如何选择正确的 PIN。可在相应的输入字段中输入 PIN（储存在配置文件中）。如果设备尝试连接，则它首先使用这个 PIN。如果连接失败，则它切换为不使用 PIN 进行连接。为获得最大的安全性，最好选择第三个选项“始终要求用户提供 PIN”。此选项允许您对不同的（远程）设备使用不同的 PIN。

单击‘高级守护程序配置’进入用于选择和配置可用服务（在蓝牙中称为配置文件）的对话框。所有可用服务都显示在列表中，可以通过单击‘激活’或‘取消激活’启用或禁用这些服务。单击‘编辑’打开一个对话框，可以在这个对话框中为所选服务（守护程序）指定其他参数。除非您熟悉服务，否则请不要进行任何更改。在完成守护程序的配置后，单击‘确定’退出此对话框。

回到主对话框中，单击‘安全性选项’进入安全性对话框并指定加密、鉴定和扫描设置。然后，退出安全性对话框返回主对话框。在单击‘完成’关闭主对话框后，就可以使用您的蓝牙系统了。

您还可以从主对话框进入‘设备和服务类’对话框。蓝牙设备被分为不同的“设备类”。在这个对话框中，为您的计算机选择正确的设备类，例如“台式机”或

“便携式计算机”。与“服务类”不同，设备类不太重要，但也在这里进行设置。有时，远程蓝牙设备（如移动电话）如果可以检测到在系统中设置的正确服务类，则只允许特定的功能。对于要在允许与计算机之间传送文件之前获得名为“对象传送”类的移动电话而言，往往是这种情况。您可以选择多个类。但无需为了“以防万一”而选择所有的类。在大多数情况，默认选择应足够了。

要使用蓝牙设置网络，则激活‘高级守护程序配置’对话框中的‘PAND’，然后单击‘编辑’设置守护程序的方式。对于一个有效的蓝牙网络连接，一个 pand 必须在‘监听’方式下运行，而对应的同级必须在‘搜索’方式下运行。默认情况下，预设为‘监听’方式。调整本地 pand 的行为。此外，在 YaST ‘网卡’模块中配置 bnepX 接口（x 代表系统中的设备编号）。

### 手工配置蓝牙

Bluez 系统各个部件的配置文件都位于目录 `/etc/bluetooth` 中。唯一的例外是用于启动部件的文件 `/etc/sysconfig/bluetooth`，该文件由 YaST 模块进行修改。

只有用户 `root` 可以修改下面介绍的配置文件。当前，没有用于更改所有设置的图形用户界面。可以使用 YaST 蓝牙模块设置最重要的设置，如第 17.2.2 节“使用 YaST 配置蓝牙”中所述。所有其他设置都只能在出现特殊情况时由有经验的用户进行设置。但通常默认设置可以满足要求。

使用 PIN 号码可以基本防止不需要的连接。移动电话在建立第一个联系时（或在电话上建立设备联系时）通常查询 PIN。为使两台设备能进行通讯，这两台设备必须用相同的 PIN 对自身进行标识。在计算机上，PIN 位于文件 `/etc/bluetooth/pin` 中。

### Important

#### 蓝牙连接的安全性

尽管有 PIN，但两台设备之间的数据传送也不是完全安全的。默认情况下，蓝牙连接的鉴定和加密处于取消激活状态。对某些蓝牙设备而言，激活鉴定和加密可能产生通信问题。

### Important

可以在配置文件 `/etc/bluetooth/hcid.conf` 中更改各种设置，例如设备名和安全性方式。通常，默认设置可以满足要求。此文件包含介绍各个设置选项的注释。

所包含文件中有两部分分别被指定为 `options` 和 `device`。前者包含 `hcid` 用于启动的一般信息。后者包含各个本地蓝牙设备的设置。



options 部分中最重要的设置之一是 `security auto;`。如果设置为 `auto`，则 `hcid` 尝试对进来的连接使用本地 PIN。如果失败，则它切换到 `none` 并建立连接。为提高安全性，应将默认设置设置为 `user`，以确保每次建立连接时都要求用户输入 PIN。

在 `device` 部分设置在另一个设备中显示该计算机所使用的设备名。本部分定义设备类，例如 `Desktop`、`Laptop` 或 `Server`。还在这里启用或禁用鉴定和加密。

### 17.2.3 系统部件和实用程序

蓝牙的可操作性取决于各种服务的交互。至少需要两个后台守护程序：`hcid`（主机控制器接口）和 `sdpd`（服务发现协议），前者充当蓝牙设备的接口并控制蓝牙设备，而设备可以通过后者找到主机提供的服务。如果在启动系统时未自动激活 `hcid` 和 `sdpd`，则可以使用命令 `rcbluetooth start` 激活这两个守护程序。必须以 `root` 用户身份执行此命令。

下面几段简要介绍了可用于使用蓝牙的最重要的 Shell 工具。尽管现在可以使用各种图形组件来控制蓝牙，但这些程序也值得您去关注。

某些命令只能以 `root` 用户身份执行。这包含用于测试远程设备连接的命令 `l2ping(device_address)`。

#### **hcitool**

`hcitool` 可用于确定是否检测到本地和远程设备。命令 `hcitool dev` 列出本地设备。输出为每个检测到的本地设备生成一行，形式为 `<interface_name>` `<device_address>`。

可以使用命令 `hcitool inq` 搜索远程设备。对于每个检测到的远程设备，返回 3 个值：设备地址、时钟偏移和设备类。设备地址很重要，因为其他命令用它来标识目标设备。时钟偏移主要用于技术目的。而设备类以十六进制值的形式指定设备类型和服务类型。

命令 `hcitoolname <device-address>` 可用于确定远程设备的设备名。对于远程计算机，设备类和设备名对应于其 `/etc/bluetooth/hcid.conf` 中的信息。使用本地设备地址将生成错误输出。

#### **hciconfig**

命令 `/usr/sbin/hciconfig` 提供有关本地设备的详细信息。如果不带任何参数执行 `hciconfig`，则输出将显示设备名 (`hciX`) 等设备信息、物理设备地址（12 位数字，形式为 `00:12:34:56:78`）和有关已传送数据量的信息。

`hciconfig hci0 name` 显示当您的计算机接收来自远程设备的请求时它返回的名称。除查询本地设备的设置外，`hciconfig` 还可用于修改这些设置。例如，`hciconfig hci0 name TEST` 将名称设置为 `TEST`。

## sdptool

程序 `sdptool` 可用于检查特定设备提供哪些服务。命令 `sdptoolbrowse <device_address>` 返回某个设备的所有服务。使用命令 `sdptool search <service_code>` 搜索特定的服务。此命令扫描所有可访问的设备来搜索请求的服务。如果其中一台设备提供服务，则此程序将输出此设备返回的完整服务名及简短说明。输入不带参数的 `sdptool` 可以查看所有可能的服务代码列表。

## 17.2.4 图形应用程序

在 *Konqueror* 中，输入 URL `bluetooth:/` 列出本地和远程蓝牙设备。双击某个设备可以获得此设备提供的服务的概述。如果将鼠标指针移过其中一个指定的服务，则浏览器的状态栏将显示将哪个配置文件用于此服务。如果您单击某个服务，则将打开一个对话框，询问您要执行的操作：保存、使用服务（必须启动应用程序来执行此操作）还是取消操作。如果您不希望此对话框再次显示，而是要始终执行所选的操作，则选中特定复选框。对于某些服务，尚不提供支持。而对于其他服务，可能需要安装附加包。

## 17.2.5 示例

本节介绍两个典型的可能蓝牙方案的示例。第一个示例说明如何通过蓝牙在两台主机之间建立网络连接。第二个示例介绍计算机和移动电话之间的连接。

### 两台主机间的网络连接

在第一个示例中，在主机 *H1* 和 *H2* 之间建立网络连接。这两台主机包含蓝牙设备地址 `baddr1` 和 `baddr2`（通过上面介绍的 `hcitooldev` 命令在这两台主机上确定）。应用 IP 地址 `192.168.1.3` (*H1*) 和 `192.168.1.4` (*H2*) 标识这两台主机。

借助于 `pand`（个域网）建立蓝牙连接。必须由用户 `root` 执行以下命令。这里主要介绍蓝牙特定的操作，不提供网络命令 `ip` 的详细说明。

输入命令 `pand -s` 在主机 *H1* 上启动 `pand`。随后，可以通过命令 `pand-c <baddr1>` 在主机 *H2* 上建立连接。如果您在其中一台主机上输入 `ip link show` 以列出可用的网络接口，则输出应包含如下所示的项：

```
bnep0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
link/ether 00:12:34:56:89:90 brd ff:ff:ff:ff:ff:ff
```

输出应包含本地设备地址 *baddr1* 或 *baddr2*，而不是 00:12:34:56:89:90。现在，必须为此接口指派一个 IP 地址并将此接口激活。在 *H1* 上，可以通过以下两个命令完成此操作：

```
ip addr add 192.168.1.3/24 dev bnep0
ip link set bnep0 up
```

在 *H2* 上：

```
ip addr add 192.168.1.4/24 dev bnep0
ip link set bnep0 up
```

现在，可以从 *H2* 使用 IP 192.168.1.3 访问 *H1*。使用命令 `ssh 192.168.1.4` 从 *H1* 访问 *H2*（假定 *H2* 运行 `sshd`，SUSE LINUX 中默认激活此程序）。也可以以普通用户身份运行命令 `ssh 192.168.1.4`。

### 将数据从移动电话传送到计算机

第二个示例说明如何将使用带有内置数码相机的移动电话拍摄的照片传送到计算机（传送多媒体讯息不会产生额外的费用）。尽管各种移动电话上的菜单结构有所不同，但过程通常十分类似。如有必要，请参考您的电话的手册。本示例介绍将 Sony Ericsson 移动电话中的照片传送到便携式计算机的过程。计算机上必须提供服务 Obex-Push，并且计算机必须向移动电话授予访问权限。第一步是在便携式计算机上使服务可用。可以通过包 `bluez-utils` 中的 `opd` 守护程序完成此步骤。用以下命令启动该守护程序：

```
opd --mode OBEX --channel 10 --daemonize --path /tmp --sdp
```

此命令使用了两个重要参数：`--sdp` 向 `sdpd` 注册服务 `--path /tmp` 指示程序保存接收数据的位置（在本例中保存到 `/tmp`）。您还可以指定具有写访问权限的任何其他目录。

现在，移动电话必须联系上计算机。为此，打开电话中的‘连接’菜单，然后选择‘蓝牙’。如有必要，请在选择‘我的设备’前单击‘打开’。选择‘新设备’并让电话搜索便携式计算机。如果检测到设备，其名称显示在屏幕上。选择与便携式计算机关联的设备。如果出现 PIN 查询，则输入 `/etc/bluetooth/pin` 中指定的 PIN。现在，您的电话就能识别便携式计算机了，并能与便携式计算机交换数据。退出当前菜单并转到图像菜单。选择要传送的图像并按‘更多’。在下一个菜单中，按‘发送’选择传送方式。选择‘通过蓝牙’。便携式计算机将被作为目标设备列出。选择便携式计算机开始传送。图像随即被保存在使用 `opd` 命令指定的目录中。可以采用相同的方法将音频曲目传送到便携式计算机。

## 17.2.6 查错

如果在建立连接时遇到困难，则根据以下列表继续操作。请记住，错误可能发生在连接的任何一端，或甚至两端都有错误。如果可能，请使用另一个蓝牙设备重现所发生的问题，以校验目前的设备是否有问题。

本地设备是否列在 `hcitooldev` 的输出中？

如果此输出中未列出本地设备，则可能是未启动 `hcid` 或设备未被识别为蓝牙设备。产生这种情况有多种原因。设备可能有问题或缺少正确的驱动程序。内置蓝牙的便携式计算机通常有一个用于无线设备（例如，WLAN 和蓝牙）的开关。请查看便携式计算机的手册以了解您的设备是否有这样一个开关。用命令 `rcbluetoothrestart` 重新启动蓝牙系统，并检查 `/var/log/messages` 中是否报告了任何错误。

您的蓝牙适配器是否需要固件文件？如果需要，请安装 `bluez-bluefw` 并用 `rcbluetoothrestart` 重新启动蓝牙系统。

`hcitoolinq` 的输出是否返回其他设备？

请多次测试此命令。连接可能受到干扰，因为其他设备也在使用蓝牙的频段。

PIN 是否匹配？检查计算机的 PIN 号码（在 `/etc/bluetooth/pin` 中）是否与目标设备的 PIN 号码相匹配。

远程设备是否能“看到”您的计算机？

尝试从远程设备建立连接。检查此设备是否能看到您的计算机。

是否可以建立网络连接（请参见示例 1）？

第一个示例（网络连接）可能无法工作，原因是多方面的。例如，这两台计算机中的一台可能不支持 `ssh` 协议。尝试 `ping192.168.1.3` 或 `ping192.168.1.4`。如果有效，则检查 `sshd` 是否是活动的。另一个问题可能是，在这两台设备中，有一台设备的网络设置与示例中的地址 `192.168.1.x` 冲突。如果是这种情况下，请尝试使用不同的地址，例如 `10.123.1.2` 和 `10.123.1.3`。

便携式计算机是否作为目标设备显示出来（示例 2）？移动设备是否识别便携式计算机上的 `Obex-Push` 服务？

在‘我的设备’中，选择相应的设备并查看‘服务’的列表。如果未显示 `Obex-Push`（甚至在更新列表后也未显示），则问题是由便携式计算机上的 `opd` 引起的。`opd` 是否是活动的？您是否对指定的目录具有写访问权限？

第二个示例是否能实现反方向的传送？

如果安装了 `obexftp` 包，则可在某些设备上使用命令 `obexftp -b <device_address> -B 10 -p <image>`。已对多部 Siemens 和 Sony Ericsson 移动电话进行了测试，可以正常工作。请参考 `/usr/share/doc/packages/obexftp` 中包的文档。

### 17.2.7 参考信息

对于蓝牙的使用和配置，<http://www.holtmann.org/linux/bluetooth/> 提供了各种说明的广泛概述。其他有用的信息和说明：

- 集成在内核中的蓝牙协议堆栈的正式 Howto 文档：<http://bluez.sourceforge.net/howto/index.html>
- 连接到 PalmOS PDA：<http://www.cs.ucl.ac.uk/staff/s.zachariadis/btpalmlinux.html>

## 17.3 红外线数据传送

IrDA（红外线数据标准协会）是用红外线进行无线通讯的行业标准。目前市场上销售的许多便携式计算机都配有与 IrDA 兼容的收发器，支持与其他设备（例如打印机、调制解调器、LAN 或其他便携式计算机）进行通讯。数据传送速度从 2400 bps 到 4 Mbps。

有两种 IrDA 工作方式。标准方式 SIR，通过串行接口访问红外线端口。这种方式可以在几乎所有系统上使用，足以满足大多数要求。更快的方式 FIR，要求 IrDA 芯片有特殊的驱动程序。因为缺少适当的驱动程序，所以并非所有芯片类型都可以以 FIR 方式工作。在计算机的 BIOS 中设置所需的 IrDA 方式。BIOS 还显示了 SIR 方式使用了哪个串行接口。

Werner Heuser 编写的 IrDA How-To 文档（位于 <http://tuxmobil.org/Infrared-HOWTO/Infrared-HOWTO.html>）提供了有关 IrDA 的信息。另外，请参考位于 <http://irda.sourceforge.net/> 的 Linux IrDA Project 万维网站点。

### 17.3.1 软件

内核包中包含所需的内核模块。包 `irda` 提供了支持红外线接口所需的帮助应用程序。安装此包后，可以在 `/usr/share/doc/packages/irda/README` 中找到文档。

## 17.3.2 配置

在引导系统时，不会自动启动 IrDA 系统服务。因此使用 YaST IrDA 模块激活此服务。此模块中只能修改一个设置：红外线设备的串行接口。测试窗口显示两个输出。其中一个 `irdadump` 的输出，它记录了所有发送的和接收的 IrDA 包。此输出应包含计算机的名称以及传送范围内所有红外线设备的名称。第 17.3.4 节“查错”显示了这些讯息的示例。窗口的下部列出了存在 IrDA 连接的所有设备。

IrDA 消耗相当多的电池电量，这是因为每隔几秒就会发送一个发现包以检测其他外围设备。因此，如果使用电池电源，则只在必要时才应启动 IrDA。输入命令 `rcirdastart` 激活 IrDA，或输入 `rcirdastop` 取消激活 IrDA。当激活接口时，系统将自动装载所有需要的内核模块。

可以在文件 `/etc/sysconfig/irda` 中执行手工配置。此文件只包含一个变量 `IRDA_PORT`，它确定在 SIR 方式下要使用的接口。

## 17.3.3 用法

可以将数据发送到设备文件 `/dev/irlpt0` 进行打印。设备文件 `/dev/irlpt0` 就像普通 `/dev/lp0` 电缆接口一样，只是它通过红外线以无线方式发送打印数据。要进行打印，确保打印机位于计算机红外线接口的可视范围内并且启动了红外线支持。

可以使用 YaST 打印机模块配置通过红外线接口运行的打印机。因为不能自动检测打印机，所以请通过单击“其他（未检测到）”手工配置打印机。在随后的对话框中，选择“IrDA 打印机”。通常，`irlpt0` 是正确的连接。第 12 章 打印机操作提供了在 Linux 中操作打印机的详细信息。

通过设备文件 `/dev/ircomm0` 可以与其他主机以及移动电话或其他类似设备进行通讯。例如，Siemens S25 和 Nokia 6210 移动电话可以通过使用红外线接口的 `wvdial` 应用程序拨号并连接到因特网。还可以与 Palm Pilot 同步数据，但前提是已将相应的应用程序的设备设置设置为 `/dev/ircomm0`。

如果愿意，您可以只对支持打印机或 IrCOMM 协议的设备进行寻址。通过特殊应用程序（例如 `irobexpalm` 和 `irobexreceive`）可以对支持 IROBEX 协议的设备（例如 3Com Palm Pilot）进行寻址。有关信息，请参考 *IR-HOWTO* (<http://tldp.org/HOWTO/Infrared-HOWTO/>)。 `irdadump` 输出中设备名后的方括号中列出了此设备支持的协议。IrLAN 协议支持仍在开发中。

## 17.3.4 查错

如果连接到红外线端口的设备不响应，请使用命令 `irdadump`（以 `root` 用户身

份) 检查计算机是否识别其他设备。当 Canon BJC-80 打印机在计算机的可视范围内时, 会经常出现类似例 17.1 “irdadump 的输出” 的内容:

### 17.1: irdadump 的输出

```
21:41:38.435239 xid:cmd 5b62bed5 > ffffffff S=6 s=0 (14)
21:41:38.525167 xid:cmd 5b62bed5 > ffffffff S=6 s=1 (14)
21:41:38.615159 xid:cmd 5b62bed5 > ffffffff S=6 s=2 (14)
21:41:38.705178 xid:cmd 5b62bed5 > ffffffff S=6 s=3 (14)
21:41:38.795198 xid:cmd 5b62bed5 > ffffffff S=6 s=4 (14)
21:41:38.885163 xid:cmd 5b62bed5 > ffffffff S=6 s=5 (14)
21:41:38.965133 xid:rsp 5b62bed5 < 6cac38dc S=6 s=5 BJC-80
                    hint=8804 [Printer IrCOMM ] (23)
21:41:38.975176 xid:cmd 5b62bed5 > ffffffff S=6 s=* earth
                    hint=0500 [ PnP Computer ] (21)
```

如果没有输出或其他设备不回应, 则检查接口的配置。校验是否使用了正确的接口。有时红外线接口位于 `/dev/ttyS2` 或 `/dev/ttyS3`, 并且有时使用了 IRQ 3 之外的中断。可以在几乎每台便携式计算机的 BIOS 设置菜单中检查和修改这些设置。

还可以使用普通数码摄像机来帮助确定红外线 LED 是否发光。大多数数码摄像机都可以看到红外线, 但人眼却看不到。





# 热插拔系统

热插拔系统控制计算机中大多数设备的初始化。热插拔系统不但用于可以在操作期间插入和拆卸的设备，而且用于在系统引导时检测到的所有设备。它工作时与 `sysfs` 文件系统和 `udev` 紧密相关，后者在第19章使用 `udev` 创建动态设备节点中进行介绍。

18.1	设备和接口	296
18.2	热插拔事件	297
18.3	热插拔代理	297
18.4	自动模块装载	299
18.5	对 PCI 设备进行热插拔	300
18.6	引导脚本冷插拔	300
18.7	错误分析	300

在引导内核之前，只有绝对需要的设备（如总线系统、引导磁盘和键盘）才能得到初始化。内核触发所有检测到的设备的热插拔事件。udev守护程序监听这些事件并调用相应的热插拔脚本以初始化这些设备。对于无法自动检测到的设备或在早期引导时丢失其事件的设备，可以使用冷插拔。冷插拔对未初始化的设备重放记录的事件或扫描系统，并对以前的设备（如 ISA）使用静态配置。

除以前个别特殊设备之外，大多数设备将在能够访问时立即进行初始化（在系统引导期间或设备热插拔时）。在初始化期间，系统将接口注册到内核。此注册将触发导致相应接口进行自动配置的更多的热插拔事件。

在 SUSE LINUX 以前的版本中，使用一组静态配置数据作为初始化设备的基础。而现在，系统查看每个可用的设备并搜索合适的配置数据或生成配置数据。

最重要的热插拔功能是在两个文件中配置的。第一个文件 `/etc/sysconfig/hotplug` 包含影响 `hotplug` 和 `coldplug` 行为的变量。对所有变量都进行了注释。第二个文件 `/proc/sys/kernel/hotplug` 包含由内核调用的可执行程序的名称。设备配置位于 `/etc/sysconfig/hardware` 中。从 SUSE LINUX 9.3 开始，此文件通常为空白，因为 `udev` 通过 `netlink` 套接字接收热插拔消息。

## 18.1 设备和接口

热插拔系统不仅管理设备，而且还管理接口。设备链接到总线或接口上。总线可以被视作多个接口。接口将设备互相链接起来，或将设备链接到应用程序。此外，还有虚拟设备，例如网络隧道。通常，设备需要内核模块形式的驱动程序。大多数情况下，接口由 `udev` 创建的设备节点表示。了解设备和接口之间的区别对了解整体概念非常重要。

在 `sysfs` 文件系统中输入的设备可在 `/sys/devices` 下找到。接口位于 `/sys/class` 或 `/sys/block` 下。`sysfs` 中的所有接口都应有一个到其设备的链接。但是，仍有一些驱动程序将不自动添加此链接。没有此链接，就不了解此接口所属的设备，也无法找到合适的配置。

设备通过设备说明进行寻址。以下可能是 `sysfs` 中的设备路径 (`/sys/devices/pci0000:00/0000:00:1e.0/0000:02:00.0`)、连接点的说明 (`bus-pci-0000:02:00.0`)、个人 ID (`id-32311AE03FB82538`) 或某些类似的内容。在过去，接口通过其名称进行寻址。这些名称只是现有设备的简单编号，可能随设备的添加或拆卸而改变。

还可以通过关联设备的说明对接口进行寻址。通常，上下文会指出此说明是指向设备本身，还是指向其接口。设备、接口和说明的典型示例有：

**PCI 网卡** 连接到 PCI 总线 (`/sys/devices/pci0000:00/0000:00:1e.0/0000:02:00.0` 或 `bus-pci-0000:02:00.0`) 并具有网络接口 (`eth0`、`id-00:0d:60:7f:0b:22` 或 `bus-pci-0000:02:00.0`) 的设备。网络接口由网络服务使用,或被连接到虚拟网络设备(如隧道或 VLAN)上,而虚拟网络设备也有一个接口。

**PCI SCSI 控制器** 采用总线(`/sys/class/scsi_host/host1`)形式使多个物理接口可用的设备(`/sys/devices/pci0000:20/0000:20:01.1/host1/1:0:0:0` 或 `bus-scsi-1:0:0:0`)。

**SCSI 硬盘** 有几个接口(`/sys/block/sda*`)的设备(`/sys/devices/pci0000:20/0000:20:01.1/host1/1:0:0:0` 或 `bus-scsi-1:0:0:0`)。

## 18.2 热插拔事件

每台设备和每个接口都有关联的热插拔事件,由 `udev` 和相应的热插拔代理处理此事件。当建立或去除与设备的链接或当驱动程序注册或检测到接口时,内核将触发热插拔事件。在 SUSE LINUX 9.3 之后, `udev` 接收并分配热插拔事件。`udev` 要么直接监听来自内核的 `netlink` 讯息,要么必须在 `/proc/sys/kernel/hotplug` 中指定 `/sbin/udevsend`。在 `udev` 完成其工作(请参见第 19 章使用 `udev` 创建动态设备节点)后,它将在 `/etc/hotplug.d/` 中搜索与事件类型相符的热插拔代理。

## 18.3 热插拔代理

热插拔代理是一个对事件执行适当操作的可执行程序。设备事件的代理位于 `/etc/hotplug.d/⟨event name⟩` 和 `/etc/hotplug.d/default` 中。这些目录中后缀为 `.hotplug` 的所有程序都按字母顺序执行。

要确保忽略特定类型的事件,请从相应热插拔代理中去除可执行位。或者,将后缀 `.hotplug` 改为其他后缀。

通常,设备代理装载内核模块,但有时它们还调用其他命令。在 SUSE LINUX 中,这由 `/sbin/hwup` 或 `/sbin/hwdown` 处理。这些程序在目录 `/etc/sysconfig/hardware` 中搜索适合设备的配置并应用配置。要防止某台设备被初始化,请创建合适的配置文件,将启动方式设置为 `manual` 或 `off`。如果 `/sbin/hwup` 未找到任何配置,则代理将自动装载模块。在这种情况下,

某些代理将自动为 `hwup` 生成配置文件。这将使代理在下次运行时速度更快。有关详细信息，请参见第18.4节“自动模块装载”。有关 `/sbin/hwup` 的详细信息，可以在文件 `/usr/share/doc/packages/sysconfig/README` 和 `man hwup` 的手册页中获得。

在调用接口代理前，`udev` 通常生成系统可以访问的设备节点。`udev` 支持为接口指派持久性名称。有关详细信息，请参见第19章使用 `udev` 创建动态设备节点。随后，各个代理将设置接口。下面说明了某些接口的过程。

### 18.3.1 激活网络接口

网络接口通过 `/sbin/ifup` 进行初始化，并通过 `/sbin/ifdown` 取消激活。文件 `/usr/share/doc/packages/sysconfig/README` 和 `man ifup` 的手册页提供了详细信息。

如果某台计算机有多台网络设备，而这些设备有不同的驱动程序，则在引导系统时，如果另一个驱动程序装载的速度较快，则可能会更改接口的指定。出于这个原因，SUSE LINUX 通过队列管理 PCI 网络设备的事件。通过文件 `/etc/sysconfig/hotplug` 中的变量 `HOTPLUG_PCI_QUEUE_NIC_EVENTS=no` 可以禁用此功能。

但是，最佳解决方案是使用持久性接口指定。您可以在配置文件中指定各个接口的名称。有关此方法的详细信息，可以在文件 `/usr/share/doc/packages/sysconfig/README` 中获得。从 SUSE LINUX 9.3 之后，`udev` 还处理网络接口，尽管这些接口不是设备节点。这样就能以更标准化的方式使用持久性接口名称。

### 18.3.2 激活储存设备

必须装入储存设备的接口才能进行访问。此操作可以完全自动完成，也可以预先配置。配置在 `/etc/sysconfig/hotplug` 的变量 `HOTPLUG_DO_MOUNT`、`HOTPLUG_MOUNT_TYPE` 和 `HOTPLUG_MOUNT_SYNC` 以及文件 `/etc/fstab` 中进行。通过设置变量 `HOTPLUG_DO_MOUNT=yes` 可以激活完全自动化的操作。通过将变量设置为 `no` 将其取消激活。

使用变量 `HOTPLUG_MOUNT_TYPE` 可以在两种方式间切换：`subfs` 或 `fstab`。在 `HOTPLUG_MOUNT_TYPE=subfs` 方式下，将在目录 `/media` 下创建一个子目录。此子目录的名称是从设备属性派生而来的。在访问媒体时，`submountd` 将自动装入和卸装媒体。在此方式下，如果不再可访问某设备，则可以方便地将其去除。在 `HOTPLUG_MOUNT_TYPE=fstab` 方式下，将用传统方式装入储存设备，即在文件 `/etc/fstab` 中输入合适的内容。

可以设置变量 `HOTPLUG_MOUNT_SYNC` 来采用同步或异步方式访问。在异步方式下，写访问较快，原因是结果进行了缓冲。但是，如果不小心拆卸了数据媒体，则可能导致数据写入不完整。在同步方式下，所有数据都立即写入，但访问会花费较长的时间。必须用 `umount` 手工卸载设备。

建议使用永久设备名，因为传统的设备名可能因初始化顺序的不同而发生变化。有关永久设备名的详细信息可以在第19章使用 `udev` 创建动态设备节点获得。

## 18.4 自动模块装载

如果无法通过 `/sbin/hwup` 初始化设备，则代理将在模块映射中搜索合适的驱动程序。首先，代理搜索 `/etc/hotplug/*.handmap` 中包含的映射。如果在这里未找到驱动程序，则它还将在 `/lib/modules/<kernelversion>/modules.*map` 中搜索。要对内核使用标准驱动程序之外的驱动程序，请将此驱动程序输入 `/etc/hotplug/*.handmap` 中，因为这是第一个被读取的文件。

USB 代理还在文件 `/etc/hotplug/usb.usermap` 和 `/etc/hotplug/usb/*.usermap` 中搜索用户方式驱动程序。用户方式驱动程序是控制对设备而非内核模块进行访问的程序。这样，可以对特定设备调用可执行程序。

对于 PCI 设备，`pci.agent` 首先就有关驱动程序模块查询 `hwinfo`。只有在 `hwinfo` 不了解任何驱动程序的情况下，代理才查看 `pci.handmap` 和内核映射。因为 `hwinfo` 已查找过此处，所以查询必定失败。`hwinfo` 还有一个用于驱动程序映射的数据库。但是，它还将装载 `pci.handmap` 以确保应用了此文件中的任何个别映射。

可以限制代理 `pci.agent` 只访问 `/lib/modules/<kernelversion>/kernel/drivers` 的某个子目录中某种类型的设备或驱动程序模块。在第一种情况下，可以将文件 `/usr/share/pci.ids` 末尾找到的 PCI 设备类输入文件 `/etc/sysconfig/hotplug` 的变量 `HOTPLUG_PCI_CLASSES_WHITELIST` 和 `HOTPLUG_PCI_CLASSES_BLACKLIST` 中。对于第二种情况，在变量 `HOTPLUG_PCI_DRIVERTYPE_WHITELIST` 和 `HOTPLUG_PCI_DRIVERTYPE_BLACKLIST` 中指定一个或几个目录。不会装载被排除的目录中的模块。在这两种情况下，白名单为空表示允许使用除黑名单排除的设备或驱动程序模块之外的所有设备或驱动程序模块。您还可以将单个模块排除在装载之外。请在文件 `/etc/hotplug/blacklist` 中输入代理一定不能装载的模块。每个模块名称列在单独的行上。

如果在映射文件中找到多个合适的模块，则只装载第一个模块。要装载所有模块，请设置变量 `HOTPLUG_LOAD_MULTIPLE_MODULES=yes`。但是，最好为此设备创建单独的设备配置 `/etc/sysconfig/hardware/hwcfg-*`。

通过 `hwup` 装载的模块不受影响。自动模块装载只在特殊情况下进行，并将在 SUSE LINUX 以后的版本中被进一步限制。但是，如果找到合适的模块，则代理将创建一个 `hwup` 配置文件，将在下次使用这个文件。这将提高设备初始化的速度。

## 18.5 对 PCI 设备进行热插拔

某些计算机还允许对 PCI 设备进行热插拔。要充分利用此功能，必须装载一些特殊的内核模块。但是，这些模块可能会导致非 PCI 热插拔计算机出现问题。不过，无法自动检测热插拔 PCI 插槽。要手工进行配置，请在文件 `/etc/sysconfig/hotplug` 中将变量 `HOTPLUG_DO_REAL_PCI_HOTPLUG` 设置为 `yes`。

## 18.6 引导脚本冷插拔

`boot.coldplug` 负责处理不能自动检测到的和没有为其生成任何热插拔事件的所有设备。此命令只为指定为 `/etc/sysconfig/hardware/hwcfg-static-*` 的每个静态设备配置调用 `hwup`。还可以使用此命令按照与热插拔时不同的顺序来初始化内置设备，因为 `coldplug` 在 `hotplug` 之前执行。

## 18.7 错误分析

### 18.7.1 日志文件

除非另行指定，否则 `hotplug` 只向 `syslog` 发送一些重要讯息。要获得更多信息，请将文件 `/etc/sysconfig/hotplug` 中的变量 `HOTPLUG_DEBUG` 设置为 `yes`。如果将此变量设置为值 `max`，则将为所有热插拔脚本记录每个 `shell` 命令。这意味着 `syslog` 储存所有讯息的 `/var/log/messages` 将变得很大。但是，因为 `syslog` 在引导进程期间在 `hotplug` 和 `coldplug` 后启动，所以第一条讯息可能未被记录。如果这些讯息对您很重要，则通过变量 `HOTPLUG_SYSLOG` 指定不同的日志文件。有关此主题的信息可以在 `/etc/sysconfig/hotplug` 中获得。

## 18.7.2 引导问题

如果计算机在引导进程中挂起，则禁用 hotplug 或 coldplug，方法是在引导提示符下输入 `NOHOTPLUG=yes` 或 `NOCOLDPLUG=yes`。由于热插拔处于取消激活状态，所以内核将不发出任何热插拔事件。在正在运行的系统中，可以输入命令 `/etc/init.d/boot.hotplugstart` 激活热插拔。将发出并处理在此之前生成的所有事件。要拒绝已排队的事件，首先在 `/proc/sys/kernel/hotplug` 中输入 `/bin/true`，过一段时间后，将输入重设置为 `/sbin/hotplug`。因为冷插拔处于取消激活状态，所以将不应用静态配置。要应用静态配置，稍后输入 `/etc/init.d/boot.coldplug start`。

要查明是否是 hotplug 装载的特定模块导致此问题的产生，请在引导提示符下输入 `HOTPLUG.TRACE=<N>`。在实际装载模块之前，将在屏幕上列出所有要装载的模块的名称，显示时间为 `<N>` 秒。在此过程中，您不能进行干预。

## 18.7.3 事件记录器

`/sbin/hotplug` 对每个事件执行脚本 `/sbin/hotplugeventrecorder`。如果目录 `/events` 存在，则所有热插拔事件都以各自文件的形式储存在此目录中。因此，可以重新生成事件以进行测试。如果此目录不存在，则不记录任何内容。





# 使用 udev 创建动态设备节点

Linux 内核 2.6 为具有永久设备指定的动态设备目录 `/dev` 引入了一个新的用户空间解决方案：`udev`。它只为实际存在的设备提供文件。它创建或去除的设备节点文件通常位于 `/dev` 目录中，并且它会重命名网络接口。以前使用 `devfs` 的 `/dev` 实施不再有效，它已被 `udev` 取代。

19.1	创建规则	304
19.2	使用 NAME 和 SYMLINK 实现自动操作	305
19.3	关键字中的正则表达式	305
19.4	关键字选择	305
19.5	用于大容量储存设备的永久名称	306

以前，设备节点都被储存在 Linux 系统的 `/dev` 目录中。无论设备是否实际存在于系统中，每个可能的设备类型都有一个节点。结果导致此目录占用了大量空间。命令 `devfs` 带来了重大改进，因为 `/dev` 中只向真正存在的设备提供设备节点。

`udev` 引入一种创建设备节点的新方法。它将 `sysfs` 提供的信息与用户以规则形式提供的数据进行比较。`sysfs` 是内核 2.6 中的新文件系统。它提供了有关连接到系统的设备的基本信息。`sysfs` 被装入在 `/sys` 下。

用户无需创建规则。如果连接了设备，就将创建适当的设备节点。虽然如此，使用规则使更改节点名称成为可能。这样可以方便地将含义模糊的设备名替换为容易记住的设备名，也可以在连接了相同类型的两个设备时使用永久设备名。

除非另行指定，否则系统为两台打印机指定的名称为 `/dev/lp0` 和 `/dev/lp1`。为哪个设备指定哪个设备节点取决于打开设备的顺序。另一个示例是外部大容量储存设备，例如 USB 硬盘。`udev` 命令允许在 `/etc/fstab` 中输入准确的设备路径。

## 19.1 创建规则

`udev` 在 `/dev` 下创建设备节点之前，会按字母顺序读取 `/etc/udev/rules.d` 中后缀为 `.rules` 的所有文件。它将使用适合此设备的第一条规则，即使其他规则也适合此设备。备注以井号 (`#`) 开头。规则采用以下形式：

```
key, [key,...] NAME [, SYMLINK]
```

必须指定至少一个关键字，因为需要根据这些关键字将规则指派给设备。还必须指定名称。在 `/dev` 中创建的设备节点将采用此名称。可选 `symlink` 参数允许在其他位置创建节点。用于打印机的规则可以采用以下形式：

```
BUS="usb", SYSFS{serial}="12345", NAME="lp_hp", SYMLINK="printers/hp"
```

本例中有两个关键字 `BUS` 和 `SYSFS{serial}`。`udev` 将此序列号与连接到 USB 总线的设备的序列号进行比较。要将名称 `lp_hp` 指派给 `/dev` 目录中的设备，所有关键字都必须相同。此外，还创建了指向设备节点的符号链接 `/dev/printers/hp`。同时还自动创建 `printers` 目录。随后可以将打印作业发送到 `/dev/printers/hp` 或 `/dev/lp_hp`。

## 19.2 使用 NAME 和 SYMLINK 实现自动操作

参数 NAME 和 SYMLINK 允许使用操作符进行自动指派。这些操作符指代对应设备上的内核数据。以下的简单示例可以说明此过程：

```
BUS="usb", SYSFS{vendor}="abc", SYSFS{model}="xyz", NAME="camera%n"
```

此名称中的操作符 %n 将被替换为相机设备的编号，例如 camera0 或 camera1。另一个有用的操作符是 %k，它将被替换为内核的标准设备名（例如 hda1）。您还可以在 udev 规则中调用外部程序，并使用 NAME 值和 SYMLINK 值中返回的字符串。udev 的手册页提供了所有操作符的列表。

## 19.3 关键字中的正则表达式

在 udev 规则的关键字中，您可以使用 shell 样式的模式匹配，即通配符。例如，字符 \* 可以用作任意个字符的占位符，而 ? 只能用作一个任意字符的占位符。

```
KERNEL="ts*", NAME="input/%k"
```

此规则将标准目录中的标准内核名称指派给名称以字母“ts”开头的设备。有关在 udev 规则中使用正则表达式的详细信息，请参见手册页 `man udev`。

## 19.4 关键字选择

对于每个有效的 udev 规则而言，适当的关键字至关重要。以下是一些标准关键字的示例：

**BUS** 设备总线类型

**KERNEL** 内核使用的设备名

**ID** 总线上的设备编号（例如，PCI 总线 ID）

**PLACE** 连接设备的物理位置（例如在 USB 上）

**SYSFS{...}** sysfs 设备特性，如标签、厂商、序列号等。

关键字 ID 和 PLACE 可能很有用，但通常使用关键字 BUS、KERNEL 和 SYSFS{...}。udev 配置还提供了调用外部脚本并对其结果求值的关键字。详细信息请参见 `man udev`。

文件系统 `sysfs` 将包含硬件信息的小型文件储存在目录树中。每个文件通常只包含一个信息项，如设备名、厂商或序列号。这些文件都可用作关键字的值。但是，要在一个规则中使用多个 SYSFS 关键字，您只能将同一个目录下的文件用作关键字值。工具 `udevinfo` 有助于查找有用的关键字值。

您必须找到指向相关设备并包含文件 `dev` 的 `/sys` 的一个子目录。这些目录都位于 `/sys/block` 或 `/sys/class` 下。如果设备已有设备节点，则 `udevinfo` 可为您找到正确的子目录。命令 `udevinfo -q path -n /dev/sda` 将输出 `/block/sda`。这表示所需目录为 `/sys/block/sda`。现在使用命令 `udevinfo -a -p /sys/block/sda` 调用 `udevinfo`。也可以将这两个命令结合起来使用，如 `udevinfo -a -p `udevinfo -q path -n /dev/sda`` 中所示。以下内容是从输出中抽取的一部分：

```
BUS="scsi"
ID="0:0:0:0"
SYSFS{detach_state}="0"
SYSFS{type}="0"
SYSFS{max_sectors}="240"
SYSFS{device_blocked}="0"
SYSFS{queue_depth}="1"
SYSFS{scsi_level}="3"
SYSFS{vendor}="          "
SYSFS{model}="USB 2.0M DSC      "
SYSFS{rev}="1.00"
SYSFS{online}="1"
```

从输出信息中，查找将不会更改的合适关键字。请记住，不能使用不同目录下的关键字。

## 19.5 用于大容量储存设备的永久名称

SUSE LINUX 附带一些脚本，这些脚本允许您始终为硬盘和其他储存设备（不论其初始化顺序如何）指派相同的名称。`/sbin/udev.get_persistent_device_name.sh` 是一个包装程序脚本。此脚本首先调用 `/sbin/udev.get_unique_hardware_path.sh`，后者为指定设备查找硬件路径。`/sbin/udev.get_unique_drive_id.sh` 可以检索序列号。然后，将两个输出都传递到

udev，它在 `/dev` 下创建指向设备节点的符号链接。可以在 udev 规则中直接使用此包装程序。以下是有关 SCSI 的一个示例，该示例也适用于 USB 或 IDE（输入时不要换行）：

```
BUS="scsi",  
PROGRAM="/sbin/udev.get_persistent_device_name.sh",  
NAME="%k", SYMLINK="%c{1+}"
```

在装载大容量储存设备的驱动程序后，该驱动程序会将所有可用硬盘注册到内核。每个硬盘均触发一个调用 udev 的热插拔块事件。然后 udev 会读取规则来确定是否需要创建符号链接。

如果通过 `initrd` 装载驱动程序，则热插拔事件丢失。但是，`sysfs` 中储存有全部信息。udevstart 实用程序查找 `/sys/block` 和 `/sys/class` 下的所有设备文件并启动 udev。

还有一个启动脚本 `boot.udev`，它在引导进程中重新创建所有设备节点。但是，必须通过 YaST 运行级别编辑器或命令 `insserv boot.udev` 激活启动脚本。

### Tip

只有在 `/dev/sda` 是 SCSI 硬盘且 `/dev/hda` 是 IDE 磁盘的情况下，才可以使用许多工具和程序。如果不是这种情况，则不能使用这些程序。YaST 依赖于这些工具，因此只在具有内核设备指定时才能发挥作用。

Tip



# Linux 中的文件系统

Linux 支持多种不同的文件系统。本章简要概述了最常见的 Linux 文件系统，详细介绍了其设计概念、优点以及应用领域。本章还提供了有关 Linux 中 LFS（大型文件支持）的一些信息。

20.1	术语	310
20.2	Linux 中的主要文件系统	310
20.3	其他一些支持的文件系统	315
20.4	Linux 中对大型文件的支持	316
20.5	参考信息	318

## 20.1 术语

**元数据** 文件系统 – 确保能正确组织和访问磁盘上所有数据的内部数据结构。从本质上讲，它是“有关数据的数据”。几乎每个文件系统都有自己的元数据结构，这也是文件系统为何表现出不同性能特性的部分原因。维护元数据的完整性非常重要，因为如果不这样，则可能无法访问文件系统中的所有数据。

**inode** Inode 包含关于文件的各种信息，包括大小，链接数，创建、修改和访问的日期和时间，以及指向实际储存文件内容的磁盘块的指针。

**日记** 在文件系统的上下文中，日记是包含某种日志的磁盘上结构，文件系统将对文件系统的元数据进行的更改储存在此日志中。日记可以显著缩短 Linux 系统的恢复时间，因为它取消了在系统启动时检查整个文件系统这一冗长的搜索过程。而只是重放日记。

## 20.2 Linux 中的主要文件系统

与两三年前不同，为 Linux 系统选择文件系统不再是花几秒钟就能完成的操作（选择 Ext2 或 ReiserFS）。从版本 2.4 开始，内核提供了多种供选择的文件系统。下面概述了这些文件系统的基本工作原理以及它们的优点。

您一定要记住一点，即没有任何一个文件系统能适合所有应用环境。每个文件系统都有各自的特定优点和缺点，必须将这些因素考虑在内。但是，即使是最复杂的文件系统也不能取代合理的备份策略。

本章中使用的术语**数据完整性和数据一致性**并不是指用户空间数据（您的应用程序写入其文件的数据）的一致性。此数据是否一致必须由应用程序本身控制。

### Important

设置文件系统

除非本章另行声明，否则可以使用 YaST 模块执行设置或更改分区和文件系统所需的所有步骤。

Important

### 20.2.1 ReiserFS

作为 2.4 内核版本的正式关键功能之一，ReiserFS 早在 SUSE LINUX 版本 6.4 后就作为 2.2.x SUSE 内核的内核增补程序提供。ReiserFS 是由 Hans Reiser 和



Namesys 开发小组设计的。ReiserFS 已证明它自己是以前的 Ext2 功能强大的替代系统。ReiserFS 的主要优点是更高的磁盘空间利用率、更高的磁盘访问性能以及更快的崩溃恢复速度。但它也有一个小缺点：ReiserFS 太注重元数据而非数据本身。ReiserFS 的后续版本将包括数据日记（将元数据和实际数据都写入日记）及顺序写入。

以下内容是对 ReiserFS 优点的详细说明：

**更高的磁盘空间利用率** 在 ReiserFS 中，采用一种名为 B\*-Tree（平衡树）的结构组织所有数据。这种树结构有助于提高磁盘空间的利用率，这是因为可以将小文件直接储存在平衡树的页节点而不是其他位置，并且只维护一个指向实际磁盘位置的指针。此外，不按照 1 kB 或 4 kB 的大块来分配储存区，而是根据所需的准确大小进行。另一个优点是 inode 的动态分配。这使得此文件系统比传统的文件系统（例如 Ext2）更灵活，而传统文件系统中的 inode 密度必须在创建文件系统时指定。

**更高的磁盘访问性能** 对于小文件，文件数据和“stat\_data”（inode）信息经常被储存在相邻的位置。这样通过一个磁盘 I/O 操作就可以访问它们，这意味着只需访问磁盘一次就可以检索到所有需要的信息。

**更快的崩溃恢复速度** 使用日记来跟踪最近的元数据更改使对文件系统的检查可以很快完成，即使对大型文件系统也是如此。

**通过数据日记确保可靠性** ReiserFS 还支持与 Ext3 一节第 20.2.3 节“Ext3”中介绍过的概念类似的数据日记和有序数据方式。默认方式是 data=ordered，它确保了数据和元数据的完整性，但只对元数据使用日记。

## 20.2.2 Ext2

Ext2 的起源可以追溯到 Linux 历史的早期。1992 年 4 月推出了 Ext2 的前身 Extended File System（扩展文件系统），并将其集成到 Linux 0.96c 中。扩展文件系统经过多次修改，并（像 Ext2 一样）成为多年来最流行的 Linux 文件系统。但随着日记文件系统的创建以及其恢复时间的大大缩短，Ext2 的重要性逐渐降低。

简要总结 Ext2 的优点有助于您了解为什么它以前是（在某些领域现在仍是）许多 Linux 用户最喜欢使用的 Linux 文件系统。

**可靠性** Ext2 确实是一个“老古董”，它经历了许多改进和频繁的测试。这可能是人们将其称之为坚如磐石的文件系统的原因。在系统中断后，如果

无法彻底卸装文件系统，则 e2fsck 将开始分析文件系统数据。系统使元数据恢复一致的状态，并将挂起的文件或数据块写入指定的目录（名为 `lost+found`）。与日记文件系统相比，e2fsck 会分析整个文件系统，而不仅仅是最近修改的元数据位。这种操作所花的时间要远远超过检查日记文件系统的日志数据所花的时间。根据文件系统的大小，此过程可能需要半小时或更长时间。因此，对于任何要求高可用性的服务器，不要选择 Ext2。但是，因为 Ext2 不维护日记且使用的内存也少得多，所以其速度常常超过其他文件系统。

可方便地升级 Ext2 的代码是 Ext3 成为广受欢迎的下一代文件系统的坚实基础。它的可靠性和稳定性与日记文件系统的优点完美地结合在一起。

## 20.2.3 Ext3

Ext3 是由 Stephen Tweedie 设计的。与所有其他下一代文件系统不同，Ext3 并没有采用全新的设计原则。它是在 Ext2 基础上设计的。这两个文件系统密切关联。可以方便地在 Ext2 文件系统中建立 Ext3 文件系统。Ext2 和 Ext3 最重要的区别是 Ext3 支持日记。总之，Ext3 有三个主要优点：

**方便并高度可靠地从 Ext2 升级** 因为 Ext3 以 Ext2 代码为基础并且共享 Ext2 的磁盘上格式和元数据格式，所以从 Ext2 升级到 Ext3 非常简单。与转换到其他日记文件系统（如 ReiserFS、JFS 或 XFS）不同，转换到 Ext3 只需要花几分钟，而转换到其他日记文件系统会相当繁琐（备份整个文件系统并从头开始重新创建文件系统）。升级到 Ext3 还很安全，因为从头重新创建整个文件系统可能会出现。考虑到等待升级到日记文件系统的现有 Ext2 系统的数量，就很容易明白为什么 Ext3 对许多系统管理员来说如此重要。从 Ext3 降级到 Ext2 与升级一样简单。只需彻底卸装 Ext3 文件系统并重新装入 Ext2 文件系统即可。

**可靠性和性能** 某些其他日记文件系统采用“仅元数据”的日记方法。这意味着元数据始终保持一致的状态，但无法自动保证文件系统数据本身一致。Ext3 的设计既可以照顾到元数据，又可以照顾到数据。“照顾”的程度可以自定义。在 `data=journal` 方式中启用 Ext3 可以提供最大的安全性（数据完整性），但因为要将元数据和数据都记入日记，所以可能降低系统的速度。一个相对较新的方法是采用 `data=ordered` 方式，这种方式确保了数据和元数据的完整性，但只对元数据使用日记。文件系统驱动程序收集与一次元数据更新对应的所有数据块。这些数据块在更新元数据之前被写入磁盘中。这样，在不牺牲性能的情况下，元数据和数据的一致性得以实现。第三个可以使用的选项是 `data=writeback`，它允许在将某些

数据的元数据提交给日记后，将这些数据写入主文件系统。在性能方面，此选项常被认为是最佳选项。但它在维护内部文件系统完整性的同时，允许以前的数据在系统崩溃并恢复后再次出现在文件中。除非指定了其他选项，否则运行 Ext3 时，`data=ordered` 为默认设置。

## 20.2.4 将 Ext2 文件系统转换为 Ext3

从 Ext2 转换为 Ext3 涉及两个单独的步骤：

**创建日记** 以 `root` 身份登录并运行 `tune2fs -j`。此命令将用默认参数创建 Ext3 日记。要自己确定日记的大小和所在的设备，请改为运行 `tune2fs -J`，同时带所需的日记选项 `size=` 和 `device=`。可以在 `tune2fs` 程序的手册页 (`tune2fs(8)`) 中获得关于此程序的更多信息。

**在 `/etc/fstab` 中指定文件系统类型** 要确保正确地识别 Ext3 文件系统，请编辑文件 `/etc/fstab`，将为对应的分区指定的文件系统类型从 `ext2` 更改为 `ext3`。此更改将在下次重引导后生效。

**对根目录使用 Ext3** 要引导设置为 Ext3 分区的根文件系统，请将模块 `ext3` 和 `jbd` 包含在 `initrd` 中。为此，请编辑文件 `/etc/sysconfig/kernel`，将这两个模块包含在 `INITRD_MODULES` 下，然后执行命令 `mkinitrd`。

## 20.2.5 Reiser4

在发布了内核 2.6 之后，日记文件系统家族又加入了一个成员：Reiser4 与以前的 ReiserFS（版本 3.6）截然不同。它引入了插件概念，从而将文件系统功能和更严格的安全性概念结合起来。

**严格的安全性概念** 在设计 Reiser4 时，其开发者将重点放在与安全性有关功能的实施上。因此，Reiser4 附带有一组专用的安全性插件。最重要的一点是引入了文件“项目”概念。当前，文件访问控制是对每个文件定义的。如果一个大型文件包含与多个用户、组或应用程序有关的信息，则访问权限就无法很精确地包含所涉及的所有各方。在 Reiser4 中，您可以将这些文件分成较小的部分（“项目”）。然后，可以分别为每个项目和每个用户设置访问权限，从而进行更精确的文件安全性管理。一个很好的示例是 `/etc/passwd`。到此为止，只有 `root` 可读取和编辑此文件，而非 `root` 用户对这个文件只有读访问权限。采用 Reiser4 的项目概念，就可以将这

个文件分成一组项目（每个用户一个项目），且允许用户或应用程序修改它们自己的数据，但不访问其他用户的数据。这一概念提高了安全性和灵活性。

**通过插件增强扩展性** 在 Reiser4 中，文件系统通常使用的许多文件系统功能和外部功能可以以插件的形式实施。这些插件可方便地添加到基础系统中。您不再需要重新编译内核或重新格式化硬盘就可将新功能添加到文件系统中。

**通过延迟分配产生较好的文件系统布局**

与 XFS 类似，Reiser4 也支持延迟分配。请参见第 20.2.7 节“XFS”。甚至可以对元数据使用延迟分配以产生较好的整体布局。

## 20.2.6 JFS

JFS（*日记文件系统*）是由 IBM 开发的。JFS Linux 的第一个 Beta 版在 2000 年夏季进入 Linux Community。2001 年发布了版本 1.0.0。JFS 的设计使其非常适合高吞吐量服务器环境，在这种环境中，性能是最重要的。作为完整的 64 位文件系统，JFS 支持大型文件和分区，这也是在服务器环境中使用它的另一个原因。

对 JFS 的进一步了解将使您明白为何这种文件系统被证明是适合 Linux 服务器的选择：

**有效的日记** JFS 采用“仅元数据”方法。这种方法只检查最近文件系统活动生成的元数据更改，而不进行大量检查，这会在恢复时节省大量的时间。可以将需要多个并行日志输入的并行操作合并为一个组提交，这样可以通过多个写入操作大大降低文件系统的性能损失。

**有效的目录组织** JFS 采用两种不同的目录组织。对于小目录，JFS 允许将目录的内容直接储存在 inode 中。对于较大的目录，JFS 使用 B<sup>+</sup> 树，这大大地简化了目录管理。

**通过动态 inode 分配更好地利用空间** 对于 Ext2，您必须提前定义 inode 密度（管理信息占用的空间），这限制了文件系统中文件或目录的最大数目。而对于 JFS，您就无需考虑这些——它动态分配 inode 空间，并在不再需要时将其释放。

## 20.2.7 XFS

SGI 在 20 世纪 90 年代初开始开发 XFS，最初计划将 XFS 作为 IRIX OS 的文件系统。开发 XFS 的目的是创建一个高性能的 64 位日记文件系统来满足当今对计算能力的极高要求。XFS 适合操纵大型文件，在高端硬件上表现优异。但即使是 XFS 也有缺点。与 ReiserFS 类似，XFS 非常注重元数据的完整性，但不太注重数据的完整性。

快速回顾 XFS 的关键功能将解释为什么此文件系统被证明是在高端计算方面其他日记文件系统的强大竞争对手。

**通过使用分配组实现高伸缩性** 在创建 XFS 文件系统时，文件系统底层的块设备被分成 8 个或 8 个以上相同大小的线性区域。这些线性区域被称为分配组。每个分配组管理自己的 inode 和可用空间。实际上，可以将分配组看作文件系统中的一个文件系统。因为分配组相互独立，所以内核可同时对多个分配组进行寻址。此功能对 XFS 优异的可伸缩性非常关键。独立分配组的概念自然适合多处理器系统的需要。

**通过有效管理磁盘空间获得高性能** 可用空间和 inode 是由分配组内的 B<sup>+</sup> 树处理的。使用 B<sup>+</sup> 树将大大增强 XFS 的性能和可伸缩性。XFS 使用延迟分配。它通过将进程分成两部分来处理分配。将挂起事务储存在 RAM 中并保留适当数量的空间。XFS 仍不决定应储存数据的准确位置（指出文件系统块）。此决定将被延迟到最后的时刻。某些生存期很短的临时数据可能永远不会被储存到磁盘上，这是因为在 XFS 决定保存它们的实际位置时，这些数据可能已经过时了。这样，XFS 增强了写性能，并减少了文件系统碎片的数目。因为延迟分配引起写事件的频率比其他文件系统引起写事件的频率要低，所以如果写操作期间发生系统崩溃，则数据丢失可能会更加严重。

**进行预分配以避免文件系统碎片** 在将数据写入文件系统前，XFS 保留（预分配）文件所需的可用空间。这样会大大减少文件系统碎片的数目。因为文件的内容不会分散在整个文件系统中，所以性能得以提高。

## 20.3 其他一些支持的文件系统

表 20.1 “Linux 中的文件系统类型” 对 Linux 支持的其他一些文件系统进行了总结。支持这些文件系统主要是为了确保与不同类型的媒体或异操作系统实现兼容和数据交换。

Table 20.1: Linux 中的文件系统类型

---

cramfs	压缩的 ROM 文件系统：一种经压缩的只读 ROM 文件系统。
hpfs	高性能文件系统：IBM OS/2 标准文件系统 — 只在只读方式下支持此文件系统。
iso9660	CD-ROM 上的标准文件系统。
minix	此文件系统源自有关操作系统的学术项目，是在 Linux 中使用的第一个文件系统。目前，它被用作软盘的文件系统。
msdos	<i>fat</i> （最初由 DOS 使用的文件系统）现在已被多种操作系统采用。
ncpfs	通过网络装入 Novell 卷的文件系统。
nfs	网络文件系统：在此文件系统中，可以将数据储存在网络中的任何计算机上，并可以通过网络授予访问权限。
smbfs	Windows 等产品使用服务器讯息块来支持通过网络进行文件访问。
sysv	在 SCO UNIX、Xenix 和 Coherent（用于个人电脑的商用 UNIX 系统）上使用。
ufs	供 BSD、SunOS 和 NeXTstep 使用。只在只读方式下支持此文件系统。
umsdos	MSDOS 上的 UNIX：应用于常规 <i>fat</i> 文件系统上，通过创建特殊文件获得 UNIX 功能（权限、链接和长文件名）。
vfat	虚拟 FAT： <i>fat</i> 文件系统的扩展（支持长文件名）。
ntfs	Windows NT 文件系统，只读。

---

## 20.4 Linux 中对大型文件的支持

最初，Linux 支持的最大文件大小为 2 GB。在大量使用多媒体之前，只要用户不在 Linux 中操纵大型数据库，这个大小已足够了。但由于服务器计算变得越来越重要，所以当使用一组应用程序必须使用的新接口时，对内核和 C 库进行了修改以支持大小大于 2 GB 的文件。当今，几乎所有的主要文件系统都提供 LFS 支

持，从而允许您执行高端计算。表20.2 “文件系统的最大大小（磁盘上格式）”概述了 Linux 文件和文件系统的当前限制。

Table 20.2: 文件系统的最大大小（磁盘上格式）

文件系统	文件大小（字节）	文件系统大小（字节）
Ext2 或 Ext3（1 kB 块大小）	$2^{34}$ (16 GB)	$2^{41}$ (2 TB)
Ext2 或 Ext3（2 kB 块大小）	$2^{38}$ (256 GB)	$2^{43}$ (8 TB)
Ext2 或 Ext3（4 kB 块大小）	$2^{41}$ (2 TB)	$2^{44}$ (16 TB)
Ext2 或 Ext3（8 kB 块大小）（系统采用 8 kB 的页，与 Alpha 类似）	$2^{46}$ (64 TB)	$2^{45}$ (32 TB)
ReiserFS v3	$2^{46}$ (64 GB)	$2^{45}$ (32 TB)
XFS	$2^{63}$ (8 EB)	$2^{63}$ (8 EB)
JFS（512 字节块大小）	$2^{63}$ (8 EB)	$2^{49}$ (512 TB)
JFS（4 kB 块大小）	$2^{63}$ (8 EB)	$2^{52}$ (4 PB)
NFSv2（客户端）	$2^{31}$ (2 GB)	$2^{63}$ (8 EB)
NFSv3（客户端）	$2^{63}$ (8 EB)	$2^{63}$ (8 EB)

## Important

### Linux 内核限制

表20.2 “文件系统的最大大小（磁盘上格式）”介绍了有关磁盘上格式的限制。2.6 内核将自己的大小限制强加给它处理的文件和文件系统。下面介绍了这些限制：

**文件大小** 在 32 位系统上，文件不能超过 2 TB（ $2^{41}$  字节）。

**文件系统大小** 文件系统最大可以为  $2^{73}$  字节。但是，目前可用的硬件尚不会超出这一限制。

Important

## 20.5 参考信息

上面介绍的每个文件系统项目都有自己的主页，可以在其中找到邮件列表信息、更多文档和常见问题解答。

- <http://e2fsprogs.sourceforge.net/>
- <http://www.zipworld.com.au/~akpm/linux/ext3/>
- <http://www.namesys.com/>
- <http://oss.software.ibm.com/developerworks/opensource/jfs/>
- <http://oss.sgi.com/projects/xfst/>

可以在 *IBM developerWorks* 中找到关于 Linux 文件系统的多部分综合性教程，网址为：<http://www-106.ibm.com/developerworks/library/l-fs.html>。有关 Linux 中不同日记文件系统的比较，请阅读 *Linuxgazette* 中 Juan I. Santos Florido 的文章，网址为：<http://www.linuxgazette.com/issue55/florido.html>。对于那些对 Linux 中 LFS 的深入分析感兴趣的用戶，请访问 Andreas Jaeger 的 LFS 站点：[http://www.suse.de/~aj/linux\\_lfs.html](http://www.suse.de/~aj/linux_lfs.html)。



# 通过 PAM 进行鉴定

Linux 在鉴定进程中使用 PAM（可插拔鉴定模块）作为用户和应用程序之间的中间层。PAM 模块在整个系统范围内可用，因此任何应用程序都可以请求 PAM 模块。本章介绍模块化鉴定机制的工作原理和配置方法。

21.1	PAM 配置文件的结构 . . . . .	320
21.2	sshd 的 PAM 配置 . . . . .	321
21.3	PAM 模块的配置 . . . . .	323
21.4	参考信息 . . . . .	325

系统管理员和编程人员经常要将访问限制在系统的某些部分或限制对应用程序某些功能的使用。如果不使用 PAM，则每次引入新的鉴定机制（例如 LDAP 或 SAMBA）时都必须对应用程序进行调整。但是，此过程相当耗费时间并且容易出现错误。避免这些缺点的一种方法是将应用程序从鉴定机制中分开并将后者委托给集中管理的模块。当需要使用最近所需的鉴定方案时，只要调整或编写合适的 PAM 模块供相关程序使用即可。

依赖于 PAM 机制的每个程序在目录 `/etc/pam.d/<programname>` 中都有它自己的配置文件。这些文件定义用于鉴定的 PAM 模块。另外，`/etc/security` 下有用于大多数 PAM 模块的全局配置文件，全局配置文件准确地定义这些模块的行为（示例包括 `pam_env.conf`、`pam_pwcheck.conf`、`pam_unix2.conf` 和 `time.conf`）。使用 PAM 模块的每个应用程序实际上调用一组 PAM 函数，这些函数随后将处理不同配置文件中的信息并将结果返回到调用这些函数的应用程序。

## 21.1 PAM 配置文件的结构

PAM 配置文件中的每一行最多包含 4 列：

```
<Type of module> <Control flag> <Module path> <Options>
```

PAM 模块是成批处理的。不同类型的模块具有不同的用途。例如一个模块检查口令，另一个模块校验访问系统的位置，第三个模块读取用户特定的设置。PAM 可以识别四种不同类型的模块：

**auth** 这种类型的模块的用途是检查用户的真实性。传统上，这是通过查询口令完成的，但也可以借助芯片卡或通过生物测定学（指纹或虹膜扫描）实现。

**account** 这种类型的模块检查用户是否具有使用所请求服务的一般权限。例如，应执行这种检查以确保任何人都不能使用失效帐户的用户名进行登录。

**password** 这种类型的模块的用途是启用鉴定令牌的更改。在大多数情况下，这是口令。

**session** 这种类型的模块负责管理和配置用户会话。在鉴定前后启动这些模块以在系统日志中注册登录尝试并配置用户的特定环境（邮件帐户、主目录、系统限制等）。

第二列包含的控制标志影响所启动模块的行为：

**required** 在进行鉴定之前，必须先成功处理带有此标志的模块。在处理带有 **required** 标志的模块失败后，将继续处理带有相同标志的所有其他模块，之后用户才会收到有关鉴定尝试失败的讯息。

**requisite** 也必须成功处理带有此标志的模块，处理方式在很大程度上与带有 **required** 标志的模块类似。但是，如果某个带有此标志的模块失败，将立即向用户提供反馈并且不再继续处理其他模块。如果成功，则将处理随后的模块，就像带有 **required** 标志的任何模块一样。**requisite** 标志可用于基本过滤器，该过滤器检查进行正确鉴定所必需的某些条件是否存在。

**sufficient** 在成功处理带有此标志的模块后，发出调用的应用程序立即收到处理成功的讯息并且不再处理其他模块，但前提是前面没有带有 **required** 标志的模块失败。带有 **sufficient** 标志的模块失败没有任何直接后果，所有随后的模块都将按其各自的顺序进行处理。

**optional** 带有此标志的模块成功或失败不会产生任何直接后果。此标志可用于只用来显示讯息（例如，通知用户收到了邮件）而不采取任何进一步操作的模块。

**include** 如果给出此标志，则在此处插入指定为参数的文件。

只要模块位于默认目录 `/lib/security`（对于 SUSE LINUX 支持的所有 64 位平台，默认目录是 `/lib64/security`）中，就无需显式指定模块路径。第四列可能包含给定模块的选项，例如 `debug`（启用调试）或 `nullok`（允许使用空口令）。

## 21.2 sshd 的 PAM 配置

为了说明 PAM 背后的工作原理，让我们看一下 `sshd` 的 PAM 配置这一实际示例：

### 21.1: sshd 的 PAM 配置

```
#%PAM-1.0
auth    include      common-auth
auth    required     pam_nologin.so
account include     common-account
```

```

password include      common-password
session include      common-session
# Enable the following line to get resmgr support for
# ssh sessions (see /usr/share/doc/packages/resmgr/README.SuSE)
#session optional    pam_resmgr.so fake_ttyname

```

应用程序（在本例中是 sshd）的典型 PAM 配置包含 4 个 include 语句，它们指向 4 种模块类型的配置文件：common-auth、common-account、common-password 和 common-session。这 4 个文件包含每种模块类型的默认配置。通过将它们包含在内而不是单独为每个 PAM 应用程序调用各个模块，在管理员更改默认值时可自动更新 PAM 配置。以前，在 PAM 发生更改或安装新应用程序时，必须手工调整所有应用程序的所有配置文件。现在则通过默认配置文件的方式来继承 PAM 配置和对它的所有更改。

第一个 include 文件 (common-auth) 调用 auth 类型的两个模块：pam\_env 和 pam\_unix2。请参见例 21.2 “auth 部分的默认配置”。

### 21.2: auth 部分的默认配置

```

auth    required    pam_env.so
auth    required    pam_unix2.so

```

第一个模块 pam\_env 装载文件 /etc/security/pam\_env.conf 以按照此文件中指定的内容设置环境变量。这可以用于将 DISPLAY 变量设置为正确的值，原因是 pam\_env 模块知道进行登录的位置。第二个模块 pam\_unix2 根据 /etc/passwd 和 /etc/shadow 检查用户的登录名和口令。

在成功调用 common-auth 中指定的模块后，第三个模块 pam\_nologin 将检查文件 /etc/nologin 是否存在。如果存在，则只有 root 用户方可登录。在 sshd 得到登录是否成功的任何反馈之前，整批 auth 模块都将完成处理。假设这批模块中的所有模块都带有 required 控制标志，则必须先成功处理所有这些模块，在此之后 sshd 才能收到有关处理成功的讯息。如果其中的某个模块不成功，则仍将继续处理整批模块，在此之后 sshd 才能得到处理失败的通知。

成功处理了 auth 类型的所有模块后，将立即处理另一个 include 语句（在本例中即例 21.3 “account 部分的默认配置”中的语句）。common-account 只包含 pam\_unix2 一个模块。如果 pam\_unix2 返回的结果证明用户存在，则 sshd 会收到一条处理成功的消息，然后处理下一批模块 (password)，如例 21.4 “password 部分的默认配置”中所示。

### 21.3: account 部分的默认配置

```
account required      pam_unix2.so
```

#### 21.4: password 部分的默认配置

```
password required    pam_pwcheck.so  nullok
password required    pam_unix2.so    nullok use_first_pass use_authtok
#password required   pam_make.so     /var/yp
```

这里，sshd 的 PAM 配置再次只涉及一个指向 password 模块的默认配置的 include 语句，这些模块位于 common-password 中。当应用程序请求鉴定令牌的更改时，必须成功完成这些模块（控制标志 required）。更改口令或另一个鉴定令牌需要进行安全检查。使用 pam\_pwcheck 模块可完成此操作。随后使用的 pam\_unix2 模块存有来自 pam\_pwcheck 的任何旧口令和新口令，因此用户无需再次鉴定。该模块还确保不能绕过 pam\_pwcheck 所执行的检查。只要前面的 account 或 auth 类型的模块被配置为指出失效的口令，就应该使用 password 类型的模块。

#### 21.5: session 部分的默认配置

```
session required     pam_limits.so
session required     pam_unix2.so
```

最后，调用 session 类型的模块（捆绑在 common-session 文件中）以根据相关用户的设置来配置会话。虽然再次处理 pam\_unix2，但由于在该模块的相应配置文件 pam\_unix2.conf 中指定了 none 选项，所以没有实际后果。pam\_limits 模块装载文件 /etc/security/limits.conf，该文件定义对某些系统资源使用的限制。当用户注销时，将再次调用 session 模块。

## 21.3 PAM 模块的配置

某些 PAM 模块是可配置的。对应的配置文件位于 /etc/security 中。本节简要介绍与 sshd 示例相关的一些配置文件——pam\_unix2.conf、pam\_env.conf、pam\_pwcheck.conf 和 limits.conf。

### 21.3.1 pam\_unix2.conf

传统的基于口令的鉴定方法是由PAM模块 `pam_unix2` 控制的。它可以从 `/etc/passwd`、`/etc/shadow`、NIS 映射、NIS+ 表或 LDAP 数据库中读取必要的数据库。通过配置各个应用程序自己的PAM选项或通过编辑 `/etc/security/pam_unix2.conf` 进行全局配置可以影响此模块的行为。例21.6 “`pam_unix2.conf`” 中说明了该模块一个非常基本的配置文件。

#### 21.6: *pam\_unix2.conf*

```
auth:    nullok
account:
password:    nullok
session:    none
```

`auth` 和 `password` 类型的模块的 `nullok` 选项指定对应类型的帐户允许使用空口令。允许用户更改他们帐户的口令。`session` 类型的模块的 `none` 选项指定不为它记录任何讯息（这是默认设置）。通过文件本身中的注释和 `pam_unix2(8)` 的手册页可以了解其他配置选项。

### 21.3.2 pam\_env.conf

此文件可用于定义调用 `pam_env` 模块时为用户设置的标准化环境。它允许您使用以下语法预设环境变量：

```
VARIABLE [DEFAULT=[value]] [OVERRIDE=[value]]
```

**VARIABLE** 要设置的环境变量的名称。

**[DEFAULT=[value]]** 设置的管理员所需的默认值。

**[OVERRIDE=[value]]** 可能由 `pam_env` 查询并设置的值，覆盖默认值。

默认值应该被 `pam_env` 覆盖的一个非常常见的示例是 `DISPLAY` 变量，每次进行远程登录，该变量都会被更改。请参见例21.7 “`pam_env.conf`”。

#### 21.7: *pam\_env.conf*

```
REMOTEHOST    DEFAULT=localhost OVERRIDE=@{PAM_RHOST}
DISPLAY       DEFAULT=${REMOTEHOST}:0.0 OVERRIDE=${DISPLAY}
```

第一行将 `REMOTEHOST` 变量的值设置为 `localhost`，当 `pam_env` 不能确定任何其他值时就会使用该值。`DISPLAY` 变量又包含 `REMOTEHOST` 的值。文件 `/etc/security/pam_env.conf` 中的注释提供了详细信息。

### 21.3.3 pam\_pwcheck.conf

此配置文件用于 pam\_pwcheck 模块，该模块为所有 password 类型的模块读取此文件中的选项。储存在此文件中的设置优先于单个应用程序的 PAM 设置。如果尚未定义应用程序特定的设置，则应用程序使用全局设置。例 21.8 “pam\_pwcheck.conf” 指示 pam\_pwcheck 允许使用空口令和修改口令。文件 /etc/security/pam\_pwcheck.conf 中介绍了模块的更多选项。

21.8: pam\_pwcheck.conf

```
password: nullok
```

### 21.3.4 limits.conf

可以在文件 limits.conf 中以用户或组为基础设置的系统限制，该文件由 pam\_limits 模块读取。该文件允许您设置硬限制（根本不能超出的限制）和软限制（可以临时超出的限制）。要了解语法和可用选项，请阅读文件中包含的注释。

## 21.4 参考信息

在已安装系统的 /usr/share/doc/packages/pam 目录中，可以找到以下附加文档：

**README 文件** 在此目录的最高一级，存有一些常规 README 文件。子目录 modules 保存有关可用 PAM 模块的 README 文件。

**Linux-PAM 系统管理员指南** 此文档包括系统管理员应该了解的有关 PAM 的所有内容。它讨论了一系列主题，从配置文件的语法到 PAM 的安全特性。此文档以 PDF 文件、HTML 格式和纯文本格式提供。

**Linux-PAM 模块编写人员手册** 此文档从开发人员的角度对多个主题进行了总结，提供了有关如何编写符合标准的 PAM 模块的信息。此文档以 PDF 文件、HTML 格式和纯文本格式提供。

**Linux-PAM 应用程序开发人员指南** 此文档包括要使用 PAM 库的应用程序开发人员所需的所有内容。此文档以 PDF 文件、HTML 格式和纯文本格式提供。

Thorsten Kukuk 为 SUSE LINUX 开发了许多 PAM 模块并提供了有关这些模块的信息，网址为：<http://www.suse.de/~kukuk/pam/>。





# Part III

## 服 務



# 基本联网知识

Linux 实际上就是因特网的产物，它提供集成到各种网络结构中所需的所有联网工具和功能。TCP/IP 是 Linux 惯用的协议，具有多种服务和特殊功能，本章将对此进行介绍。使用网卡、调制解调器或其他设备的网络访问可以通过 YaST 来配置。也可以手工进行配置。不过本章的讨论仅限于基本机制和相关网络配置文件。

22.1	IP 地址和路由	332
22.2	IPv6 — 下一代的因特网	335
22.3	名称解析	341
22.4	使用 YaST 配置网络连接	342
22.5	手工配置网络连接	351
22.6	作为拨号助手的 smpppd	359

Linux 和其他 Unix 操作系统均使用 TCP/IP 协议。该协议不是单个网络协议，而是提供多种服务的一系列网络协议。表22.1 “TCP/IP 系列协议中的若干协议”中所列的协议专用于在两台计算机之间通过 TCP/IP 交换数据。由 TCP/IP 连接而成的网络构成了世界范围的网络，就整体而言也称作“因特网”。

RFC 代表请求注释。RFC 由一些文档组成，用来说明各种因特网协议和操作系统及其应用程序的实施过程。RFC 文档用来说明如何设置因特网协议。要进一步了解某个协议，请参考相应的 RFC 文档。可以通过 <http://www.ietf.org/rfc.html> 访问这些联机文档。

Table 22.1: TCP/IP 系列协议中的若干协议

协议	说明
TCP	传送控制协议：一种面向连接的安全协议。要传送的数据首先由应用程序作为数据流发送，然后由操作系统转换为相应的格式。数据到达目标主机上的相应应用程序时采用最初发送时的原始数据流格式。TCP 确定传送过程中是否丢失了数据，并确保格式没有混乱。只要涉及到数据序列就会实施 TCP。
UDP	用户数据报文协议：一种无连接的不安全的协议。要传送的数据以应用程序生成的数据包的形式发送。不能保证数据以正确的顺序到达接收方，还可能丢失数据。UDP 适用于面向记录的应用程序。它的等待时间比 TCP 稍短。
ICMP	因特网控制讯息协议：这实际上不是一个面向最终用户的协议，而是一个特殊的控制协议，用来发出错误报告，还可以控制参与 TCP/IP 数据传送的计算机的行为。此外，它还提供一种特殊的回应方式，可以通过 ping 程序查看该方式。
IGMP	因特网组管理协议：此协议控制实施 IP 多路广播时的计算机行为。

如图22.1 “TCP/IP 的简化层次模型”中所示，数据交换在不同的层中进行。实际的通讯层是通过 IP（因特网协议）的不安全数据传送。IP 的上面是 TCP（传送控制协议），它能够确保一定程度的数据传送安全性。IP 层又受底层硬件相关协议（例如 Ethernet）的支持。

该图为每一层都提供了一到两个示例。如您所见，层次按照抽象程度排序。最底层非常接近硬件。最上层则几乎就是硬件的完全抽象化。每一层都有自己的特殊功能。每一层的特殊功能多隐含在其说明中。位传送层和安全性层表示所用的物理网络（如 Ethernet）。

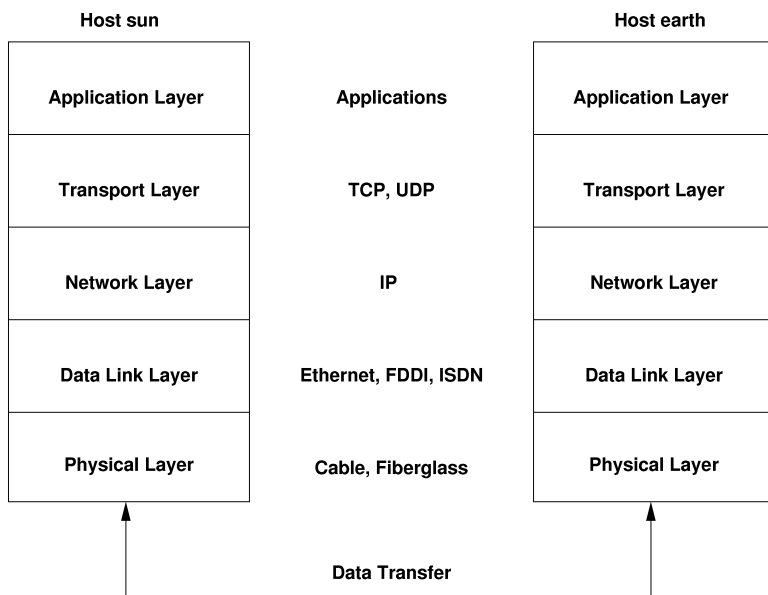


Figure 22.1: TCP/IP 的简化层次模型

几乎所有硬件协议都在面向数据包的基础上发挥作用。因为无法一次传送所有数据，所以要将这些数据封装在包中。TCP/IP 包最大约为 64 KB。通常的包还要小得多，因为可能还要受到网络硬件的限制。Ethernet 上的数据包最大约为 1500 字节。通过 Ethernet 发送数据时，TCP/IP 包不能超过这个限额。如果传送更多数据，操作系统需要发送更多的数据包。

为使层实现其指定功能，必须在数据包中保存与每层相关的附加信息。这些信息保存在数据包的报头中。每一层都在每个新包的开头附加一小块称为协议报头的数据。图 22.2 “TCP/IP Ethernet 包” 演示了一个通过 Ethernet 电缆传送的示例 TCP/IP 数据包。校验和位于包的末尾而不是开头，这样更便于网络硬件处理。

当应用程序通过网络发送数据时，数据会穿越每个层次，所有传递都在 Linux 内核中实施（只有第 1 层 - 网卡除外）。每一层都负责准备好数据，以便传递到下一层。最底层最后负责发送数据。接收数据时则逆向执行整个过程。正像剥洋葱皮那样，在每一层中都要从传输数据中去除协议报头。最后，第 4 层负责使数据可供目标上的应用程序使用。通过这种方式，每一层只与其上一层或下一层通讯。对于应用程序，无论数据是通过 100 MBit/s（兆位/秒）的 FDDI 网络传送

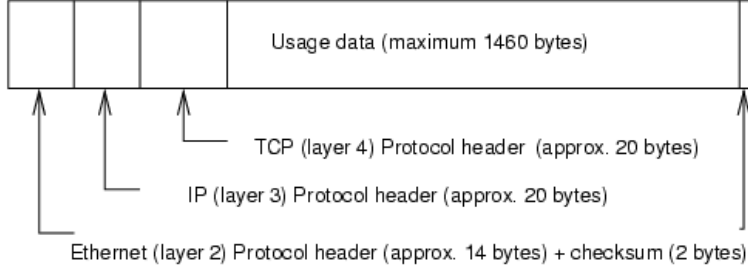


Figure 22.2: TCP/IP Ethernet 包

还是通过 56 kbit/s (千位/秒) 的调制解调器线路传送, 都毫不相关。同样, 只要数据包的格式正确, 传送哪种数据对数据线也无关系要。

## 22.1 IP 地址和路由

各节的论述仅限于 IPv4 网络。有关 IPv6 协议 (IPv4 的后续协议) 的信息, 请参考第 22.2 节 “IPv6 — 下一代的因特网”。

### 22.1.1 IP 地址

因特网上的每台计算机都有一个唯一的 32 位地址。这些 32 位 (或 4 字节) 地址通常按例 22.1 “如何书写 IP 地址” 的第二行所示的格式书写。

#### 22.1: 如何书写 IP 地址

```
IP Address (binary): 11000000 10101000 00000000 00010100
IP Address (decimal): 192. 168. 0. 20
```

在十进制格式中, 四字节以十进制数书写, 其间以句点分隔。IP 地址被指派给主机或网络接口。除此之外不能用在其他任何地方。这条规则当然也有例外, 但这些例外对下文的影响微乎其微。

Ethernet 网卡本身有自己的唯一地址 - MAC，即媒体访问控制地址。该地址长 48 位，在全球范围内唯一，并且由网卡厂商编程到硬件中。但是，厂商指派的地址存在一个缺陷 — MAC 地址并未构成分级系统，而差不多是随机分布的。因此，不能用这些地址对远程计算机寻址。不过，MAC 地址仍然在本地网络主机间的通讯中发挥着重要作用，并且还是第 2 层协议报头的主要组成部分。

IP 地址中的点表示分级系统。直到 20 世纪 90 年代，IP 地址仍然有严格的分类。不过，这个系统经证实太过死板，因此已经废止。现已改为使用无类别路由 - (CIDR，无类别域间路由)。

## 22.1.2 网络掩码和路由

一般认为网络掩码是用于向 IP 地址为 192.168.0.1 的主机通知 IP 地址为 192.168.0.20 的主机的位置。简单地说，带有 IP 地址的主机的网络掩码可以定义哪些主机是内部的，哪些是外部的。位于内部（“位于同一子网内”）的主机可以直接响应。位于外部（“不在同一子网内”）的主机只能通过网关或路由器来响应。由于每个网络接口能够接收自己的 IP 地址，情况可能变得十分复杂。

发送网络包之前，计算机上会运行以下操作：IP 地址通过逻辑 AND 链接到网络掩码，发送主机的地址要通过逻辑 AND 连接到网络掩码。如果同时有多个网络接口，通常会校验所有可能的发送方地址。随后比较 AND 链接的结果。如果未比较出差异，目标（或接收主机）则位于同一子网内。否则必须通过网关来访问。网络掩码中的“1”位越多，可直接访问的主机越少，通过网关访问的主机越多。例 22.2 “将 IP 地址链接到网络掩码”演示了几个示例。

### 22.2: 将 IP 地址链接到网络掩码

```
IP address (192.168.0.20):  11000000 10101000 00000000 00010100
Netmask   (255.255.255.0): 11111111 11111111 11111111 00000000
-----
Result of the link:      11000000 10101000 00000000 00000000
In the decimal system:   192.    168.    0.      0

IP address (213.95.15.200): 11010101 10111111 00001111 11001000
Netmask   (255.255.255.0): 11111111 11111111 11111111 00000000
-----
Result of the link:      11010101 10111111 00001111 00000000
In the decimal system:   213.    95.    15.    0
```

网络掩码和 IP 地址一样，也采用由句点分隔的十进制格式。由于网络掩码也是 32 位值，需要连续书写四组数字值。对于哪些主机充当网关以及通过哪些网络接口访问哪些地址域，都必须进行配置。

再举一个例子：通过同一 Ethernet 电缆相连的所有计算机通常都位于同一子网中，所以可直接访问。如果用交换机或网桥分隔该 Ethernet，这些主机仍然可以访问。

不过，较为经济的 Ethernet 不适用于覆盖较远的距离。您必须将 IP 包传送到其他硬件（如 FDDI 或 ISDN）。用于这类传送的设备称为路由器或网关。Linux 计算机可以执行这样的任务。相应的选项称为 `ip_forwarding`。

如果配置了网关，IP 包将被发送到相应的网关。网关随后会尝试以相同的方式转发该包——从主机到主机——直到到达目标主机或超过该包的 TTL（存活时间）。

Table 22.2: 特定地址

地址类型	说明
基本网络地址	这是网络掩码和该网络中的任意地址，如例 22.2 “将 IP 地址链接到网络掩码” 中的 Result（结果）所示。不能将此地址指派给任何主机。
广播地址	这大体表示“访问此子网内的所有主机”。要生成此地址，需要将网络掩码反转为二进制格式，并使用逻辑 OR 链接到基本网络地址。由此上例生成的结果为 192.168.0.255。不能将此地址指派给任何主机。
本地主机	地址 127.0.0.1 只能指派给每台主机的“回路设备”。可以使用此地址与您自己的计算机建立连接。

由于 IP 地址必须在全球范围内唯一，您不能随机选择地址。共有三个地址域可用于建立基于 IP 的专用网络。使用这些地址域不能与因特网上的其他主机建立任何连接（除非运用特定的技巧），因为这些地址不能通过因特网传送。这些地址域在 RFC 1597 中指定，并且列在表 22.3 “专用 IP 地址域” 中。

Table 22.3: 专用 IP 地址域

网络/网络掩码	域
10.0.0.0/255.0.0.0	10.x.x.x
172.16.0.0/255.240.0.0	172.16.x.x-172.31.x.x
192.168.0.0/255.255.0.0	192.168.x.x



## 22.2 IPv6 — 下一代的因特网

由于 WWW（万维网）的出现，过去十五年中越来越多的计算机开始通过 TCP/IP 通讯，这使因特网有了突飞猛进的发展。自从 1990 年在 CERN (<http://public.web.cern.ch>) 任职的 Tim Berners-Lee 开创了 WWW，因特网主机的数量已从几千台猛增至上亿台。

如上所述，IP 地址只有 32 位。而且还有不少 IP 地址丢失——因网络组织结构的原因而无法使用。子网中可用的地址数量是位数的平方减 2。举例来说，某个子网可以有 2 个、6 个或 14 个可用地址。如果要将 128 台主机连接到因特网，您的子网要提供 256 个 IP 地址，其中只有 254 个可用，因为有两个 IP 地址需要供该子网本身的结构使用：即作为广播地址和基础网络地址。

在当前的 IPv4 协议下，DHCP 或 NAT（网络地址转换）是用来避免出现地址短缺的典型机制。这些方法与用来分隔专用地址空间和公用地址空间的规定相结合，肯定能够缓解短缺状况；它们的问题在于不仅配置烦琐，而且也加重了维护的负担。要在 IPv4 网络内设置主机，您需要若干地址项，如主机本身的 IP 地址、子网掩码、网关地址，可能还要提供名称服务器地址。所有这些项都是必需的，而且无法从其他任何地方得到这些项。

利用 IPv6，地址的短缺和复杂的配置都将成为过去。以下各节进一步说明了 IPv6 带来的改进和优点，以及如何从旧协议过渡到新协议。

### 22.2.1 优点

新协议中最为重要同时也最为显著的改进在于对可用地址空间的极大扩容。IPv6 地址由 128 位值而不是传统的 32 位值组成，它提供的 IP 地址数目多达 10 的 15 次方的若干倍。

不过，IPv6 与以前的不同不仅限于长度，其内部结构也发生了变化，这种结构可以包含更多的有关系统和系统所属网络的具体信息。有关详细信息，请参见第 22.2.2 节“地址类型和结构”。

以下列出了新协议的其他一些优点：

**自动配置** IPv6 使网络可以支持“即插即用”，这意味着无需任何手工配置即可将新安装的系统集成到（本地）网络中。新主机可以使用其自动配置机制，依赖名为邻居发现 (ND) 的协议从邻近的路由器提供的信息中得到自己的地址。这种方法不要求管理员参与，并且无需维护用于分配地址的中央服务器——这是 IPv4 无法媲美的（IPv4 中需要使用 DHCP 服务器来自动分配地址）。

**移动能力** 利用 IPv6，为一个网络接口同时指派多个地址成为可能。用户能够轻松访问多个网络，就如同享受移动电话公司提供的国际漫游服务：将移动电话携至境外时，电话一进入国外服务区即自动登录相应的国外服务，这样，无论您身处何地都可以用同一个号码进行联系，而且您还能够像在本地服务区内那样拨打电话。

**安全通讯** 在 IPv4 中，网络安全是一项附加功能。IPv6 则将 IPSec 作为其核心功能之一，允许系统通过安全隧道通讯，避免被因特网上的外来者窃听。

**向后兼容** 实际上，要想将整个因特网一下子从 IPv4 转换为 IPv6 是不可能的。因此，这两个协议不仅要能在因特网上同时存在，还应能够同时存在于一个系统中，这一点至关重要。要实现这一点，一方面两种地址应兼容（IPv4 地址可以轻松转换为 IPv6 地址），另一方面还要使用一定数量的隧道。请参见第 22.2.3 节“IPv4 与 IPv6 并存”。此外，系统可以依赖双栈 IP 技术同时支持两种协议，这意味着系统中有两种完全分开的网络堆栈，从而避免这两种版本的协议相互影响。

**通过多路广播的自定义服务** 在 IPv4 中，有些服务（如 SMB）需要向本地网络中的所有主机广播其数据包。IPv6 则采用一种更为精确的方式，通过多路广播支持服务器对主机寻址——即对作为组中一部分的若干主机寻址（这不同于通过广播对所有主机寻址或通过单路广播对每台主机逐个寻址）。将哪些主机作为一组来寻址可能要取决于具体的应用程序。可使用一些预定义的组来寻址，例如对所有名称服务器寻址（*所有名称服务器多路广播组*），或对所有路由器寻址（*所有路由器多路广播组*）。

## 22.2.2 地址类型和结构

如上所述，当前的 IP 协议在以下两个重要方面存在缺陷：IP 地址越来越短缺，而且配置网络和维护路由表的任务越来越复杂和繁重。IPv6 通过将地址空间扩展到 128 位解决了第一个问题。通过引入分级地址结构，结合先进的网络地址分配技术和多宿主功能（将多个地址指派给同一个设备，从而支持对多个网络的访问），第二个问题也迎刃而解。

使用 IPv6 时，了解三种类型的地址十分有用：

**单路广播** 这类地址只与一个网络接口关联。采用这类地址的包只传递到一个目标。因此，使用单路广播地址可以将包传送到本地网络或因特网上的单个主机。

**多路广播** 这类地址与一组网络接口相关。采用这类地址的包将传递到属于该组的所有目标。多路广播地址主要供特定网络服务使用，用于以有序的方式与特定的主机组通讯。

**任意广播** 这类地址与一组接口相关。采用这类地址的包将根据基础路由协议的原则，传递给该组中与发送方最为接近的成员。任意广播地址便于主机在特定网络区域内找到提供特定服务的服务器。同一类型的所有服务器都具有相同的任意广播地址。在请求服务时，主机会收到路由协议决定的最接近它的服务器的回复。如果出于某种原因此服务器无法回复，协议会自动选择距离稍远一些的服务器，依此类推。

IPv6 地址分为八组，每组四位数字，代表十六位，采用十六进制表示法。各组之间用冒号(:)分隔。可以删除某组中的前置零字节，但不能删除组中或组末的零。另一个约定是：连续的零字节若超过四个，则可以省略为双冒号形式。不过，每个地址只允许有一个这样的::。例22.3“示例 IPv6 地址”中演示了这种简写表示法，其中的三行全部表示同一地址。

### 22.3: 示例 IPv6 地址

```
fe80 : 0000 : 0000 : 0000 : 0000 : 10 : 1000 : 1a4
fe80 :    0 :    0 :    0 :    0 : 10 : 1000 : 1a4
fe80 :                               : 10 : 1000 : 1a4
```

IPv6 地址的每个部分都有明确的功能。前面的字节构成前缀，用于指定地址类型。中间部分是地址的网络部分，但可以不用。地址的结尾构成主机部分。在 IPv6 中，网络掩码是通过在地址末尾的斜杠后指明前缀的长度来定义的。例22.4“指定前缀长度的 IPv6 地址”中的地址包含上述信息，即：前 64 位构成地址的网络部分，后 64 位构成地址的主机部分。换言之，64 表示网络掩码由左起的 64 个 1 位值构成。正如 IPv4，要用 AND 将 IP 地址与子网值结合起来，以确定主机位于同一子网中还是其他网络中。

### 22.4: 指定前缀长度的 IPv6 地址

```
fe80::10:1000:1a4/64
```

IPv6 可以识别几种预定义的前缀类型。其中有些列在表22.4“各种 IPv6 前缀”中。

Table 22.4: 各种 IPv6 前缀

前缀（十六进制）	定义
00	IPv4 地址和 IPv6 上的 IPv4 兼容地址。这些用于与 IPv4 保持兼容。要使用这些地址，仍然需要依赖路由器将 IPv6 包转换为 IPv4 包。有若干特殊地址（如用于回路设备的地址）也采用此前缀。
2 或 3 作为第一个数字	可聚合全局单路广播地址。类似 IPv4 的情况，可以指定某个接口作为特定子网的一部分。当前存在以下地址空间：2001::/16（生产质量地址空间）和 2002::/16（6to4 地址空间）。
fe80::/10	链路本地地址。不应路由带有这种前缀的地址，而只能从同一子网中访问。
fec0::/10	站点本地地址。可以路由这种地址，但只局限在它们所属的组织网络之内。实际上，这些是相当于当前的专用网络地址空间（如 10.x.x.x）的 IPv6 地址。
ff	这些是多路广播地址。

单路广播地址由三个基本部分组成：

**公共拓扑结构** 第一部分（也包含上述前缀之一）用于通过公共因特网路由数据包。其中包含提供因特网访问的公司或机构的相关信息。

**站点拓扑结构** 第二部分包含要将包传递到的子网的路由信息。

**接口 ID** 第三部分标识要将包传递到的接口。其中允许使用 MAC。由于 MAC 是硬件厂商编程到设备中的全球唯一的固定标识符，配置过程得到了极大简化。事实上，前 64 个地址位共同构成 EUI-64 令牌，后 48 位从 MAC 中提取，其余的 24 位包含有关令牌类型的特殊信息。这样还可以将 EUI-64 令牌指派给没有 MAC 的接口，如基于 PPP 或 ISDN 的接口。

在这个基础结构之上，IPv6 还区分五种不同的单路广播地址：

**::（未指定）** 在首次初始化接口时——即无法通过其他方法确定地址时，这类地址可用作主机的源地址。

**:::1（回路）** 回路设备的地址。

**IPv4 兼容地址** IPv6 地址由 IPv4 地址和 96 个零位组成的前缀构成。这类兼容地址用于隧道通讯进程（请参见第 22.2.3 节“IPv4 与 IPv6 并存”），以便 IPv4 和 IPv6 主机与在纯 IPv4 环境中操作的其他主机通讯。

**映射到 IPv6 的 IPv4 地址** 这类地址以 IPv6 表示法指定纯 IPv4 地址。

**本地地址** 有两类地址可供本地使用：

**链路本地** 这类地址只能在本地子网中使用。不能将源地址或目标地址采用此类地址的包路由到因特网或其他子网。这些地址包含特殊的前缀 (fe80::/10) 和网卡的接口 ID，中间部分为空字节。这类地址在自动配置过程中使用，用于与同一子网中的其他主机通讯。

**站点本地** 可以将采用这类地址的包路由到其他子网，但不能路由到更广阔的因特网——不能跨越组织自身的网络。这类地址用于内部网，相当于 IPv4 定义的专用地址空间。其中包含特殊的前缀 (fec0::/10)、接口 ID，及指定子网 ID 的十六位域。其余部分也是空字节。

作为 IPv6 引进的全新功能，每个网络接口通常可以获得多个 IP 地址，这个功能的优点即在于：可以通过同一接口访问多个网络。其中一个网络可以使用 MAC 和已知前缀进行完全的自动配置，这样一启用 IPv6（使用链路本地地址），即可访问本地网络中的所有主机。由于其中使用了 MAC，所用的任何 IP 地址都是全球唯一的。地址中只有指定**站点拓扑结构**和**公共拓扑结构**的部分才是可变部分，这取决于主机当前运行所在的实际网络。

要使主机在不同网络间切换，主机至少需要两个地址。其中之一 - **本地地址**，不仅包含接口 ID 而且包含该主机通常所属的本地网络的标识符（以及相应的前缀）。本地地址是静态地址，因此一般不变。所有要发送到移动主机的包仍可以传递到该主机，不管它是在本地网络还是其他任何网络中操作。这一点得益于 IPv6 引进的全新功能，如**无状态自动配置**和**邻居发现**。除本地地址之外，移动主机还获得一个或多个额外的地址，这些地址属于该主机漫游到的外地网络。这些地址称为**转交地址**。本地网络有一种功能，可以在主机漫游到外地时转发要发送给该主机的所有包。在 IPv6 环境中，这项任务由**本地代理**来完成，该代理可以接收要发送到本地地址的所有包，并通过隧道进行转发。另一方面，发送到转交地址的那些包可直接转发到移动主机，而不必进行任何特殊的迂回处理。

### 22.2.3 IPv4 与 IPv6 并存

将与因特网相连的所有主机从 IPv4 迁移到 IPv6 是一个逐步的过程。这两种协议将在未来一定时间内并存。通过双栈技术来实施这两种协议，可以在同一系统

上同时支持这两种协议。但这仍然没有解决支持 IPv6 的主机如何与 IPv4 主机通讯，以及应如何通过当前网络（主要基于 IPv4）传输 IPv6 包的问题。最好的解决方案就是提供隧道处理功能和兼容地址（请参见第 22.2.2 节“地址类型和结构”）。

在（全球）占主导地位的 IPv4 网络中相对孤立的 IPv6 主机可以通过隧道来通讯：IPv6 包被封装为 IPv4 包，在 IPv4 网络中传送。这种在两个 IPv4 主机间的连接被称为隧道。要实现这种功能，包必须包含 IPv6 目标地址（或相应的前缀），以及隧道接收端的远程主机的 IPv4 地址。根据主机管理员间的协议，可以手工配置基本的隧道。这也称作静态隧道。

但是，静态隧道的配置和维护往往过于烦琐，不能适应日常通讯需要。因此，IPv6 提供了三种不同的动态隧道方法：

**6over4** IPv6 包被自动封装为 IPv4 包，并通过支持多路广播的 IPv4 网络发送。这种方法诱导 IPv6 将整个网络（因特网）视为一个巨大的局域网 (LAN)。这样即可自动确定 IPv4 隧道的接收端。不过，这种方法不够灵活，并且还因为 IP 多路广播在因特网上尚未普及而不易推行。因此，它提供的解决方案仅适用于支持多路广播的小型公司网络或机构网络。RFC 2529 中对这种方法作出了规定。

**6to4** 利用这种方法，可以从 IPv6 地址自动生成 IPv4 地址，从而支持孤立的 IPv6 主机通过 IPv4 网络进行通讯。不过，用这种方法在孤立的 IPv6 主机和因特网之间通讯时存在一些问题。RFC 3056 中对这种方法进行了说明。

**IPv6 隧道中介程序** 这种方法依赖特殊的服务器为 IPv6 主机提供专用隧道。RFC 3053 中对此进行了说明。

---

## Important

### 6bone 倡议

在“以往”因特网的核心位置，通过隧道相连的遍布全球的 IPv6 子网已发展起来。这就是 *6bone* 网络 (<http://www.6bone.net>)，即可供编程人员和因特网服务提供者使用的 IPv6 测试环境。这些人可以利用这个网络开发和提供基于 IPv6 的服务，汲取实施新协议所需的经验。有关详细信息，请参见该项目的因特网站点。

---

Important

## 22.2.4 配置 IPv6

要配置 IPv6，通常无需在各个工作站上执行任何更改，但必须装载 IPv6 支持。为此，需要以 root 身份输入 `modprobe ipv6`。

由于 IPv6 使用自动配置，将给网卡指派链路-本地网络中的地址。一般不在工作stations上管理路由选择表。工作站可以使用路由广告协议查询网络路由器，了解应实施的前缀和网关。使用 radvd 程序可以设置 IPv6 路由器。此程序会通知工作站对 IPv6 地址使用哪个前缀和哪个路由器。或者，可以使用 zebra 自动配置两个地址和路由选择。

有关如何使用 `/etc/sysconfig/network` 文件设置各种隧道的信息，请参考 `ifup` 的手册页 (`man ifup`)。

### 22.2.5 参考信息

上文的概述中并未全面论述 IPv6 这一主题。有关这种新协议的深入讨论，请参考以下联机文档和书目：

<http://www.ngnet.it/e/cosa-ipv6.php>

包含一系列文章，对 IPv6 的基础知识进行了精彩介绍。是关于该主题的不错的入门指南。

<http://www.bieringer.de/linux/IPv6/>

在此可找到 Linux IPv6-HOWTO 以及许多与该主题有关的链接。

<http://www.6bone.net/> 如果想加入 IPv6 隧道网络，请访问此站点。

<http://www.ipv6.org/> 学习 IPv6 知识的起点。

RFC 2640 有关 IPv6 的基础 RFC。

**IPv6 Essentials** 该书详述了这一主题的所有重要方面。Silvia Hagen: *IPv6 Essentials*. O'Reilly & Associates, 2002 (ISBN 0-596-00125-8)。

## 22.3 名称解析

DNS 有助于将 IP 地址指派给一个或多个名称，并将名称指派给 IP 地址。在 Linux 中，这种转换通常由一种特殊的称为 `bind` 的软件来完成。负责这种转换的计算机称为名称服务器。这些名称构成了分级系统，各个名称组成部分之间用圆点分隔。不过，这个名称层次与上述 IP 地址层次无关。

考虑以 `hostname.domain` 格式书写的完整名称，如 `laurent.suse.de`。完整名称，即完全限定的域名 (FQDN)，由主机名和域名 (`suse.de`) 组成。后者还包含顶级域或 TLD (`de`)。

TLD 的指派由于历史原因已经变得十分混乱。传统的指派方法是美国所用的三字母域名，而世界其他地方采用的标准是双字母 ISO 国家/地区代码。此外，2000 年还引进了较长的 TLD，表示特定的活动领域（例如 .info、.name 和 .museum）。

在因特网发展的早期阶段（1990 年之前），文件 `/etc/hosts` 被用来储存因特网上表示的所有计算机的名称。后来事实证明随着接入因特网的计算机与日俱增，这种方法很快就行不通了。为此人们开发了一个分散式数据库，以十分分散的方式储存主机名。这个数据库类似名称服务器，它并不储存与因特网上的所有主机相关的数据，但可以向其他名称服务器发送请求。

位于层次顶级的是根名称服务器。这些根名称服务器管理顶级域，并由网络信息中心（即 NIC）运行。每个根名称服务器都了解负责特定顶级域的名称服务器。有关顶级域 NIC 的信息，请参见 <http://www.internic.net>。

DNS 不仅可以解析主机名，还能够为整个域识别出负责接收电子邮件的主机——邮件交换器 (MX)。

为解析 IP 地址，您的计算机必须了解至少一个名称服务器及其 IP 地址。借助 YaST 可以轻松指定这样的名称服务器。如果建立的是调制解调器拨号连接，则根本无需手工配置名称服务器。拨号协议可以在建立连接后提供名称服务器的地址。有关如何在 SUSE LINUX 中配置针对名称服务器的访问，请参见第 24 章 域名系统。

whois 协议与 DNS 密切相关。使用此程序可以快速找出负责特定域的服务器。

## 22.4 使用 YaST 配置网络连接

计算机必须装有受支持的网卡。通常，在安装过程中可以检测到网卡，随后会装载适合的驱动程序。要检查网卡是否与相应的驱动程序正确集成，请输入命令 `ipaddress list eth0`。命令输出中将列出有关网络设备 `eth0` 的所有信息，或显示一条错误讯息。

如果针对网卡的内核支持作为模块实施（SUSE 内核的默认实施方式），则必须在 `/etc/sysconfig/hardware/hwcfg-*` 中输入模块的名称。如果未在此文件中指定任何名称，`hotplug` 将自动选择一个驱动程序。不管网卡是什么类型（热插拔或内置），`hotplug` 都会指派一个驱动程序。

### 22.4.1 使用 YaST 配置网卡

启动 YaST 模块后，其中将显示一个常规网络配置对话框。上半部分显示所有待配置网卡的列表。在引导过程中正确自动检测到的所有网卡的名称都会列在此



处。未能检测到的设备显示为‘其他（未检测到）’。该对话框的下半部分显示目前已配置的设备列表，以及它们的网络类型和地址。现在可以配置新网卡或更改现有配置。

### 手工配置网卡

对于未能自动检测到的网卡（显示为‘其他’），需要配置以下项：

**网络配置** 设置接口的设备类型和配置名称。从所提供的选项中选择设备类型。按需指定配置名称。通常，默认设置很有用，可以接受该设置。有关配置名称的命名约定的信息，请参见 `getcfg` 的手册页。

**内核模块** ‘硬件配置名称’指定 `/etc/sysconfig/hardware/hwcfg-*` 文件的名称，该文件包含网卡的硬件设置，例如适合的内核模块的名称。通常，YaST 会为 PCMCIA 和 USB 硬件建议适用的名称。对于其他硬件，0 通常仅在使用 `hwcfg-static-0` 配置网卡时才有意义。

如果网卡为 PCMCIA 或 USB 设备，则激活相应的复选框，并选择‘下一步’退出此对话框。如果不是这些设备，请在‘从列表中选择’中选择您的网卡型号。YaST 将自动选择适合的内核模块。选择‘下一步’可退出此对话框。

### 设置网络地址

设置接口的设备类型和配置名称。从所提供的设备类型中选择一种。按需指定配置名称。通常，默认设置很有用，可以接受该设置。有关配置名称的命名约定的信息，请参见 `getcfg` 的手册页。

如果已选择‘无线’作为接口的设备类型，请在下一个对话框‘无线网卡配置’中配置操作方式、网络名称 (ESSID) 和加密方式。单击‘确定’可完成网卡的配置。有关配置 WLAN 网卡的详细说明，请参见第 17.1.3 节“用 YaST 进行配置”。对于其他所有接口类型，请继续进行网络地址的设置：

‘自动地址设置（通过 DHCP）’ 如果网络中有 DHCP 服务器，您可以依赖它来自动设置网络地址。如果您用的是 DSL 线路，但 ISP 没有给您指派静态 IP，此时也应该使用该选项。如果决定使用 DHCP，请在选择‘DHCP 客户机选项’后配置详细信息。指定 DHCP 服务器是否应始终允许广播请求并允许使用标识符。默认情况下，DHCP 服务器使用网卡的硬件地址来标识接口。如果您使用虚拟主机设置，其中不同的主机都通过同一接口通讯，则需要用标识符来区分它们。

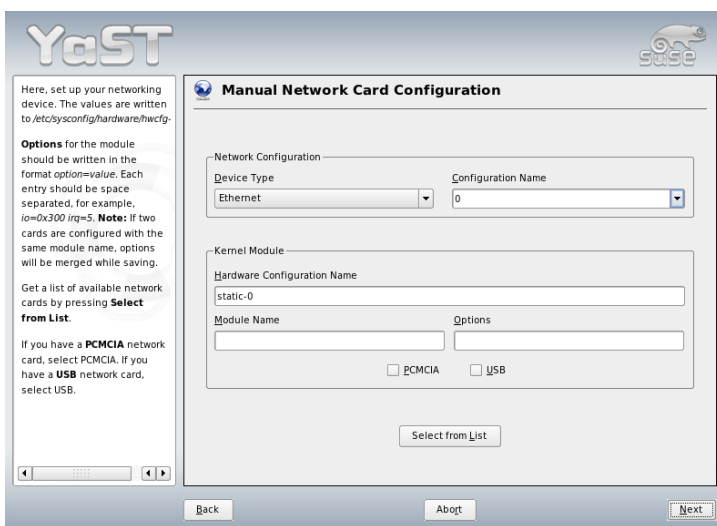


Figure 22.3: 配置网卡

‘静态地址设置’ 如果有静态地址，请启用相应的复选框。然后输入网络的地址和子网掩码。预设的子网掩码应该符合典型家庭网络的要求。

通过选择‘下一步’退出此对话框，或者继续配置主机名、名称服务器和路由选择详细信息（请参见 和 一节）。

使用‘高级’可以指定更复杂的设置。在‘详细信息’下，使用‘用户控制’将网卡的控制权从管理员（root）委托给普通用户。对于移动操作，这样便于用户激活或取消激活接口，也就能更灵活地适应变化的网络连接。还可以在此对话框中设置 MTU（最大传送单元）和‘设备激活’的类型。

## 22.4.2 调制解调器

在 YaST 控制中心中，可以在‘网络设备’下访问调制解调器配置。如果未自动检测到您的调制解调器，请打开用于手工配置的对话框。在随后打开的对话框中，请在‘调制解调器设备’下输入调制解调器连接到的接口。

如果处在专用交换机（PBX）之后，则可能需要输入拨号前缀。该前缀通常是零。请参考随 PBX 附带的说明了解相关信息。同时还要选择使用音频拨号还是脉冲

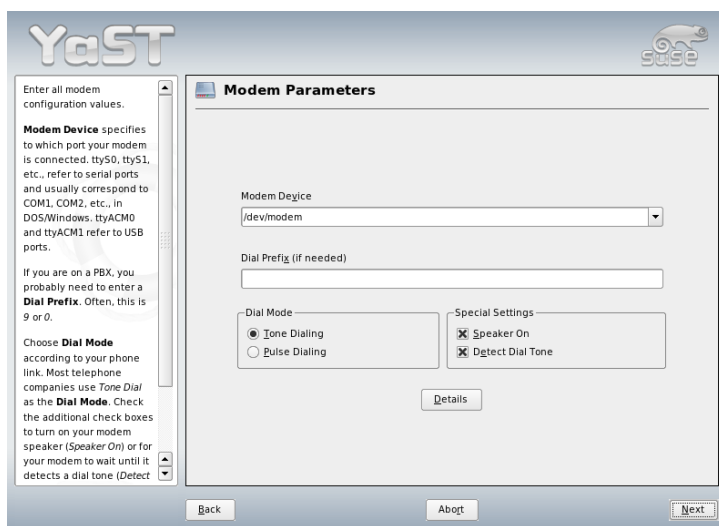


Figure 22.4: 调制解调器配置

拨号、是否打开扬声器，以及调制解调器是否应在检测到拨号音之前一直等待。如果调制解调器连接到交换机，则不应启用后一选项。

在‘详细信息’之下，设置波特率和调制解调器的初始化字符串。只有在调制解调器不是自动检测到的，或者需要特殊设置才能传送数据时，才应更改以上设置。这种情况主要发生在 ISDN 终端适配器上。单击‘确定’可退出此对话框。要将调制解调器的控制权委托给不具备根权限的普通用户，请激活‘用户控制’。这样，不具备管理员权限的用户即可激活或取消激活某个接口。在‘拨号前缀正则表达式’下，指定正则表达式。KInternet 中的‘拨号前缀’（可由普通用户修改）必须符合此正则表达式。如果将此字段留空，用户则无法在不具备管理员权限的情况下设置其他‘拨号前缀’。

在下一个对话框中，选择 ISP（因特网服务提供者）。要从您所在国家/地区的 ISP 的预定义列表中进行选择，请选择‘国家/地区’。也可以单击‘新建’打开一个对话框，从中为您的 ISP 提供数据。这些数据包括用于拨号连接的名称、ISP 的名称，以及 ISP 提供的登录名和口令。启用‘始终询问口令’，在您每次连接时都提示输入口令。

在最后一个对话框中，指定附加连接选项：

‘按需拨号’ 如果启用按需拨号，请指定至少一个名称服务器。

‘连接后修改 DNS’ 默认情况下启用此复选框，其作用是在每次连接因特网时都更新名称服务器地址。不过，如果启用‘按需拨号’，则应禁用此复选框，还应提供固定的名称服务器地址。

‘自动检索 DNS’ 如果提供者未在连接后传送其域名服务器，则禁用此选项并手工输入 DNS 数据。

‘愚蠢方式’ 默认情况下启用此选项。通过它可以忽略 ISP 服务器发送的输入提示，防止它们影响连接进程。

‘激活防火墙’ 选择此选项可以启用 SUSE 防火墙，在您与因特网连接时防止外部攻击。

‘空闲超时（秒）’ 使用此选项可以指定网络不活动的时间，一超过该时间调制解调器即自动断开连接。

‘IP 详细信息’ 使用此选项可打开地址配置对话框。如果您的 ISP 没有为您的主机指派动态 IP 地址，请禁用‘动态 IP 地址’，然后输入主机的本地 IP 地址及远程 IP 地址。请向您的 ISP 询问这些信息。保持‘默认路由’的启用状态，然后通过选择‘确定’关闭该对话框。

选择‘下一步’可返回初始对话框，其中显示调制解调器配置的概要。选择‘完成’可关闭此对话框。

### 22.4.3 ISDN

使用此模块可以为系统配置一个或多个 ISDN 网卡。如果 YaST 未能检测到您的 ISDN 网卡，则通过手工方式选择。可以使用多个接口，但您可以为一个接口配置多个 ISP。在随后的对话框中，设置该网卡正常工作所需的 ISDN 选项。

在下一个对话框中（如图 22.5 “ISDN 配置” 所示），选择要使用的协议。默认值是‘Euro-ISDN (EDSS1)’，但是对于旧式或大型交换机，请选择‘1TR6’。如果是在美国，请选择‘NI1’。在相关字段中选择您所在的国家/地区。相应的国家/地区代码将显示在该字段旁边的字段中。最后，提供您的‘区号’和拨号前缀（如果需要）。

‘启动方式’定义如何启动 ISDN 接口：使用‘OnBoot’可以在系统每次引导时初始化 ISDN 驱动程序。‘手工’要求您以 root 的身份使用命令 `rcisdn start` 装载 ISDN 驱动程序。‘热插拔’，用于 PCMCIA 或 USB 设备，用于在插入设备后加载驱动程序。在完成所有这些设置后，请选择‘确定’。

在下一个对话框中，为您的 ISDN 网卡指定接口类型，并将 ISP 添加到现有接口中。接口的类型可能是 SyncPPP 或 RawIP，但多数 ISP 以 SyncPPP 方式操作，如下文所述。



Figure 22.5: ISDN 配置

要为‘我的电话号码’输入的值取决于特定的设置：

ISDN 网卡直接连接到电话插座 标准的 ISDN 线路提供三个电话号码（称为多用户号码，或 MSN）。如果用户要求更多号码，最多可提供十个号码。必须在此处输入其中一个 MSN，但不要区号。如果输入的号码有误，您的电话运营商将自动退回到为您的 ISDN 线路指派的第一个 MSN。

ISDN 网卡连接到电话交换机 同样，配置可能随安装设备的不同而变化：

1. 适用于家庭的小型电话交换机大多使用 Euro-ISDN (EDSS1) 协议进行内部呼叫。这些交换机具有内部 S0 总线，并对与它们连接的设备使用内部号码。  
将其中一个内部号码用作您的 MSN。您应该至少能够使用支持直接向外拨号的交换机的 MSN 之一。如果无效，则尝试使用一个零。有关进一步信息，请参考随电话交换机附带的文档。
2. 为公司设计的大型电话交换机通常使用 1TR6 协议用于内部呼叫。它们的 MSN 称为 EAZ 并且通常对应直拨号码。要在 Linux 中配置，只需输入 EAZ 的最后一位即可。如果各种方法都行不通，可尝试 1 到 9 之间的各位数字。

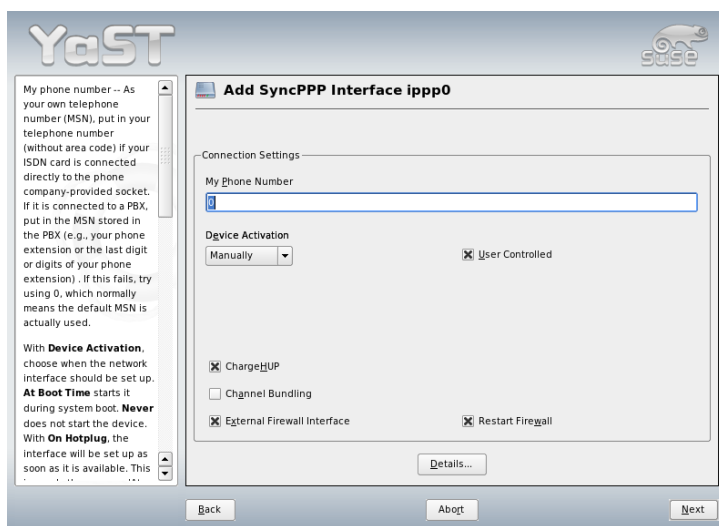


Figure 22.6: ISDN 接口配置

要在下一个收费单位开始之前及时终止连接，请启用‘ChargeHUP’。但要记住，该选项不是对每个 ISP 都奏效。您也可以通过选择相应的复选框启用信道绑定（多链接 PPP）。最后，您可以通过选择‘激活防火墙’为链接启用 SuSEfirewall2。要使不具备管理员权限的普通用户能够激活或取消激活接口，请选中‘用户控制’。

单击‘细节’打开一个对话框，在其中可实施更为复杂的连接方案，这与普通的家庭用户无关。选择‘下一步’可退出此对话框。

在下一个对话框中设置 IP 地址。如果您的提供者没有为您指定静态 IP，请选择‘动态 IP 地址’。否则，根据 ISP 指定的信息，使用提供的字段输入您主机的本地 IP 地址及远程 IP 地址。如果接口应该成为与因特网连接的默认路由，请选择‘默认路由’。每台主机都只能有一个接口配置为默认路由。选择‘下一步’可退出此对话框。

使用随后的对话框，您可以设置您所在的国家/地区并选择 ISP。列表中的 ISP 都只是 call-by-call（通过呼叫进行呼叫）提供者。如果列表中未列出您的 ISP，请选择‘新建’。随即打开‘提供者参数’对话框，可以在其中输入 ISP 的所有详细信息。输入电话号码时，切勿在数字之间加空格或逗号。最后，输入 ISP 为您提供的登录名和口令。输完之后，请选择‘下一步’。

要在独立工作站上使用‘按需拨号’，还需指定名称服务器（DNS 服务器）。多数 ISP 都支持动态 DNS，这意味着每次用户连接时，都由 ISP 发送名称服务器的 IP 地址。不过，对于单个工作站，您仍然需要提供 192.168.22.99 之类的占位符地址。如果您的 ISP 不支持动态 DNS，请指定 ISP 的名称服务器 IP 地址。如果需要，可以为连接指定超时值——即网络不活动的时间（以秒计），一超过该时间即自动终止连接。选择‘下一步’确认设置。YaST 将显示配置好的接口的概要。要使所有这些设置有效，请选择‘完成’。

#### 22.4.4 电缆调制解调器

在有些国家/地区（奥地利、美国），人们往往通过有线电视网访问因特网。有线电视用户通常将调制解调器一端接在有线电视插座上，另一端与计算机网卡相连（使用 10Base-TG 双绞线）。随后电缆调制解调器就能通过固定 IP 地址提供专用因特网连接。

在配置网卡时需要根据您的 ISP 提供的说明选择‘自动地址设置（通过 DHCP）’或‘静态地址设置’。目前多数提供商都使用 DHCP。通常只有特殊的公司帐户才使用静态 IP 地址。

#### 22.4.5 DSL

要配置 DSL 设备，请从 YaST ‘网络设备’部分选择 ‘DSL’ 模块。这个 YaST 模块包含若干对话框，可以在这些对话框中基于以下协议之一设置 DSL 链接参数。

- Ethernet 上的 PPP (PPPoE)
- ATM 上的 PPP (PPPoATM)
- 用于 ADSL 的 CAPI (Fritz 网卡)
- 点对点隧道协议 (PPTP) — 奥地利

基于 PPPoE 或 PPTP 配置 DSL 连接时，要求已经正确设置相应的网卡。如果尚未这样做，应首先通过选择‘配置网卡’来配置网卡（请参见第 22.4.1 节“使用 YaST 配置网卡”）。使用 DSL 链接时，可以自动指派地址但并不通过 DHCP，这就是不应启用‘自动地址设置（通过 DHCP）’选项的原因。相反，应该为接口输入静态虚设地址，如 192.168.22.1。在‘子网掩码’中，输入 255.255.255.0。如果配置的是独立工作站，则将‘默认网关’留空。

## Tip

'IP 地址'字段和'子网掩码'中的值只是占位符。它们只用于初始化网卡，而不会将 DSL 链接表示成这样。

## Tip

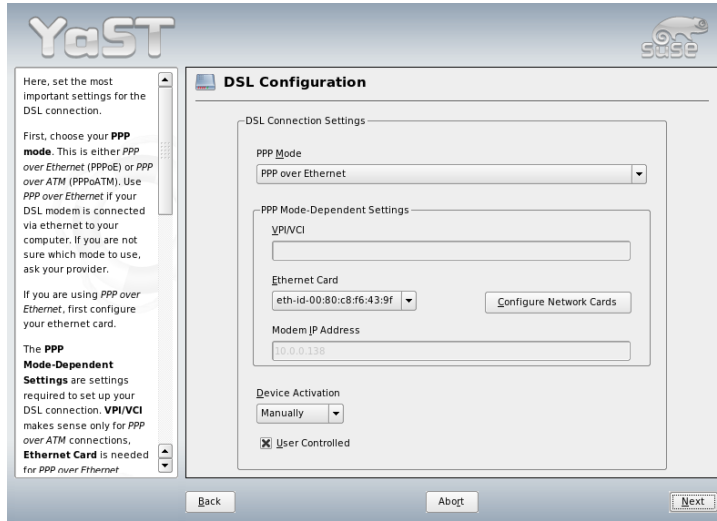


Figure 22.7: DSL 配置

要着手配置 DSL（请参见图 22.7 “DSL 配置”），首先应选择 PPP 方式及 DSL 调制解调器连接到的 Ethernet 网卡（多数情况下是 eth0）。然后使用‘设备激活’指定是否应在引导进程中建立 DSL 链接。单击‘用户控制’可授权不具备根权限的普通用户通过 KInternet 激活或取消激活接口。使用该对话框还可以选择您所在的国家/地区，并可以在该区域内的若干 ISP 中进行选择。随后的所有 DSL 配置对话框的详细信息都取决于目前已设置的选项，因此下面几段只对这些对话框进行了简要介绍。有关可用选项的详细信息，请阅读这些对话框中提供的详细帮助信息。

要在独立工作站上使用‘按需拨号’，还需指定名称服务器（DNS 服务器）。多数 ISP 都支持动态 DNS — 每次用户连接时，都由 ISP 发送名称服务器的 IP 地址。不过，对于单个工作站，应提供 192.168.22.99 之类的占位符地址。如果您的 ISP 不支持动态 DNS，请输入 ISP 提供的名称服务器 IP 地址。



‘空闲超时（秒）’定义网络不活动的时间，一超过该时间即自动终止连接。合理的超时值介于 60 到 300 秒之间。如果禁用了‘按需拨号’，则最好将超时值设置为零以防止自动挂断。

T-DSL 的配置与 DSL 的设置非常相似。只需将‘T-Online’选为您的提供者，YaST 将打开 T-DSL 配置对话框。在此对话框中，提供 T-DSL 所需的一些其他信息 — 线路 ID、T-Online 号码、用户代码和口令。所有这些信息都会包含在订阅到 T-DSL 后收到的信息中。

## 22.5 手工配置网络连接

应该始终将手工配置网络软件作为最后的选择。建议使用 YaST。但是，对网络配置背景信息的了解将对您使用 YaST 有所帮助。

可以通过热插拔检测并配置所有内置网卡及热插拔网卡（PCMCIA、USB 和某些 PCI 卡）。系统以两种不同的方式处理网卡，首先将其作为物理设备对待，其次将其作为接口对待。设备的插入或检测将触发一个热插拔事件。此热插拔事件通过脚本 `/sbin/hwup` 触发设备的初始化。当将网卡作为新网络接口进行初始化时，内核将生成另一个热插拔事件，该事件触发通过 `/sbin/ifup` 设置接口的操作。

内核按照接口注册的时间顺序对接口名进行编号。对于名称的指派，初始化顺序是决定性的。如果多个网卡中的一个失败，则所有随后初始化的网卡的编号将发生变化。对于真正的可热插拔网卡，连接设备的顺序非常重要。

为了实现灵活的配置，设备（硬件）的配置和接口的配置是分开进行的，配置到设备和接口的映射不再以接口名为基础进行管理。设备配置位于 `/etc/sysconfig/hardware/hwcfg-*` 中。接口配置位于 `/etc/sysconfig/network/ifcfg-*` 中。指派配置名的方式使这些名称可以描述所关联的设备和接口。因为以前从驱动程序到接口名的映射需要静态接口名，所以 `/etc/modprobe.conf` 中不再进行此映射。在新概念中，此文件中的别名项可能导致不希望出现的副作用。

配置名（`hwcfg-` 或 `ifcfg-` 后的任何内容）可以通过插槽、设备特定的 ID 或接口名对设备进行描述。例如，某个 PCI 卡的配置名可能是 `bus-pci-0000:02:01.0`（PCI 插槽）或 `vpid-0x8086-0x1014-0x0549`（厂商和产品 ID）。关联接口的名称可能是 `bus-pci-0000:02:01.0` 或 `wlan-id-00:05:4e:42:31:7a`（MAC 地址）。

若要将某个网络配置指派给某种类型的任何一个卡（一次只插入一个该类型的卡）而不是某个特定的卡，请选择不是非常特定的配置名。例如，可以

将 `bus-pcmcia` 用于所有 PCMCIA 卡。另一方面，可以通过在前面加上接口类型对名称进行限制。例如，可以为连接在 USB 端口上的 WLAN 卡指派 `wlan-bus-usb`。

系统始终使用对接口或提供接口的设备描述最清楚的配置。搜索最合适的配置是由 `/sbin/getcfg` 处理的。`getcfg` 的输出提供可用于描述设备的所有信息。`getcfg` 的手册页中介绍了有关指定配置名的详细信息。

使用上面介绍的方法，即使不总是以相同的顺序初始化网络设备，也可以用正确的配置对网络接口进行配置。但是，接口的名称仍取决于初始化顺序。有两种方法可以确保对特定网卡的接口进行可靠的访问：

- `/sbin/getcfg-interface<configuration name>` 返回关联的网络接口的名称。因此，在某些（但不是所有）配置文件中，可以输入配置名（例如，防火墙、`dhcpd`、路由、多种虚拟网络接口（隧道））而不是接口名（不是持久性的）。
- 可以将持久性接口名指派给其配置不具有接口名的所有接口。这可以通过在接口配置 (`ifcfg-*`) 中输入 `PERSISTENT_NAME=<pname>` 实现。但是，持久性名称 `<pname>` 不应该与内核可能自动指派的名称相同。因此，不允许使用 `eth*`、`tr*`、`wlan*`、`qeth*` 和 `iucv*` 等名称。请改用 `net*` 或者像 `external`、`internal` 或 `dmz` 等描述性名称。只能紧接在接口注册之后为接口指派持久性名称，这意味着必须重装网卡的驱动程序或者必须执行 `hwup <device description>`。命令 `rcnetworkrestart` 不足以实现此目的。

### Important

#### 使用持久性接口名

持久性接口名的使用尚未在所有领域中进行测试。因此，某些应用程序不能顺畅地处理所选的接口名。如果您遇到这种问题，请通过 <http://www.suse.de/feedback> 通知我们。

### Important

`ifup` 需要现有的接口，因为它不初始化硬件。硬件的初始化是由命令 `hwup` 处理的（由 `hotplug` 或 `coldplug` 执行）。当初始化设备时，将通过 `hotplug` 为新接口自动执行 `ifup`，如果启动方式是 `onboot`、`hotplug` 或 `auto` 并且启动了 `network` 服务，则将设置接口。以前，命令 `ifup <interfacename>` 触发硬件初始化。目前的过程与以前的过程相反。首先初始化硬件部件，随后执行所有其他操作。这样，使用现有的这组配置，总能以可能的最佳方式配置数目不固定的设备。

表22.5 “手工网络配置脚本”总结了网络配置中所涉及的最重要的脚本。只要有可能，硬件和接口将对脚本进行区分。

Table 22.5: 手工网络配置脚本

配置阶段	命令	功能
硬件	hw{up,down,status}	热插拔子系统执行此 hw* 脚本来初始化设备、撤消初始化或查询设备的状态。hwup 的手册页中提供了详细信息。
接口	getcfg	getcfg 可用于查询与配置名或硬件描述关联的接口名。getcfg 的手册页中提供了详细信息。
接口	if{up,down,status}	if* 脚本启动现有的网络接口或返回指定接口的状态。ifup 的手册页中提供了详细信息。

有关热插拔和持久性设备名的详细信息在第18章 热插拔系统 和第19章 使用 udev 创建动态设备节点 中有所介绍。

## 22.5.1 配置文件

本节对网络配置文件进行了概述并解释了它们的作用和所使用的格式。

### `/etc/sysconfig/hardware/hwcfg-*`

这些文件包含网卡和其他设备的硬件配置。它们包含所需的参数，例如内核模块、启动方式和脚本关联。有关详细信息，请参考 hwup 的手册页。不管现有硬件如何，都会在启动冷插拔时应用 hwcfg-static-\* 配置。

### `/etc/sysconfig/network/ifcfg-*`

这些文件包含网络接口的配置。它们包含启动方式和 IP 地址等信息。可能的参数在 ifup 的手册页中有所介绍。另外，如果应将常规设置只用于一个接口，则文件 dhcp、wireless 和 config 中的所有变量都可用于 ifcfg-\* 文件。

## **/etc/sysconfig/network/config, dhcp, wireless**

文件 `config` 包含 `ifup`、`ifdown` 和 `ifstatus` 行为的常规设置。`dhcp` 包含用于无线 LAN 卡的 DHCP 和 `wireless` 设置。对所有这三个配置文件中的变量都进行了注释，它们还可用于 `ifcfg-*` 文件中，该文件将以较高的优先级处理这三个配置文件。

## **/etc/sysconfig/network/routes,ifroute-\***

在这里确定 TCP/IP 包的静态路由。可以在 `/etc/sysconfig/network/routes` 文件中输入各项系统任务要求的所有静态路由：连接主机的路由、通过网关连接主机的路由和连接网络的路由。需要单独路由的各个接口要定义附加配置文件：`/etc/sysconfig/network/ifroute-*`。用接口名称替换 `*`。路由选择配置文件中的项如下所示：

```
DESTINATION          GATEWAY NETMASK   INTERFACE [ TYPE ] [ OPTIONS ]
DESTINATION          GATEWAY PREFIXLEN INTERFACE [ TYPE ] [ OPTIONS ]
DESTINATION/PREFIXLEN GATEWAY -        INTERFACE [ TYPE ] [ OPTIONS ]
```

要省略 GATEWAY、NETMASK、PREFIXLEN 或 INTERFACE，请用 `-` 代替。TYPE 项和 OPTIONS 项可以省略。

路由目标位于首列。此列可以包含网络或主机的 IP 地址，或者在有可访问名称服务器时，包含完全限定的网络或主机名。

第二列包含默认网关或通过其可访问主机或网络的网关。第三列包含网关后的网络或主机的子网掩码。例如，网关后主机的掩码为 `255.255.255.255`。

最后一列只与本地主机连接的网络有关，如回路、Ethernet、ISDN、PPP 和虚拟设备。必须在此输入设备名。

## **/etc/resolv.conf**

在此文件中指定主机所属的域（关键字 `search`）。同时列出的还有要访问的名称服务器地址的状态（关键字 `nameserver`）。可以指定多个域名。当解析不是完全限定的名称时，将尝试通过附加单独的 `search` 项生成一个完全限定的名称。通过输入多行可以输入多个名称服务器，每行都以 `nameserver` 开头。注释以 `#` 符号开头。YaST 在此文件中输入指定的名称服务器。例 22.5 “`/etc/resolv.conf`” 显示 `/etc/resolv.conf` 的内容。

*22.5: /etc/resolv.conf*

```
# Our domain
search example.com
#
# We use sun (192.168.0.20) as nameserver
nameserver 192.168.0.20
```

某些服务（例如 `pppd` (`wvdial`)、`ipppd` (`isdn`)、`dhcp` (`dhcpcd` 和 `dhclient`)、`pcmcia` 和 `hotplug` 通过脚本 `modify_resolvconf` 修改文件 `/etc/resolv.conf`。如果文件 `/etc/resolv.conf` 已被此脚本临时修改，则它将包含预定义的注释，给出有关修改它的服务的信息、备份原始文件的位置以及如何关闭自动修改机制。如果 `/etc/resolv.conf` 被多次修改，则该文件以嵌套的形式包括修改。即使还原的顺序不同于进行修改的顺序，也可以以一种彻底的方式进行还原。可能需要这种灵活性的服务包括 `isdn`、`pcmcia` 和 `hotplug`。

如果未以正常、彻底的方式终止服务，则可以使用 `modify_resolvconf` 恢复原始文件。另外，在系统引导时，将检查是否有由未彻底终止产生的、经修改的 `resolv.conf`（例如，系统崩溃后），如果有，则将恢复原始（未经修改的）`resolv.conf`。

YaST 使用命令 `modify_resolvconf check` 检查 `resolv.conf` 是否已被修改并将随后警告用户这些更改将在恢复文件后丢失。除此之外，YaST 将不使用 `modify_resolvconf`，这意味着通过 YaST 更改 `resolv.conf` 的效果与任何手工更改的效果相同。在两种情况下，更改都具有永久效果。上述服务请求的修改只是临时的。

## **/etc/hosts**

在此文件中，如例22.6 “`/etc/hosts`” 中所示，将为主机名指派 IP 地址。如果未实施名称服务器，则将其建立 IP 连接的所有主机必须列在此处。在此文件中为每个主机输入一行，包含 IP 地址、完全限定的主机名和主机名。IP 地址必须在每行的开头，各项用空格和制表符隔开。注释总是以 `#` 符号开头。

### *22.6: /etc/hosts*

```
127.0.0.1 localhost
192.168.0.20 sun.example.com sun
192.168.0.1 earth.example.com earth
```

## **/etc/networks**

在这里，网络名称被转换为网络地址。格式类似于 `hosts` 文件的格式，只是网络名称在地址的前面。请参见 例22.7 “`/etc/networks`”。

22.7: `/etc/networks`

```
loopback    127.0.0.0
localnet    192.168.0.0
```

## **/etc/host.conf**

此文件控制名称解析，即通过解析程序库转换主机名和网络名称。此文件只用于链接到 `libc4` 或 `libc5` 的程序。对于当前的 `glibc` 程序，请参考 `/etc/nsswitch.conf` 中的设置。参数必须始终单独在一行上。注释以 `#` 符号开头。表22.6 “`/etc/host.conf` 的参数”显示了可用的参数。例22.8 “`/etc/host.conf`”中显示了 `/etc/host.conf` 的示例。

Table 22.6: `/etc/host.conf` 的参数

---

<code>order hosts, bind</code>	指定访问服务以进行名称解析的顺序。可用参数有（使用空格或逗号隔开）：  <code>hosts</code> : 搜索 <code>/etc/hosts</code> 文件  <code>bind</code> : 访问名称服务器  <code>nis</code> : 使用 NIS
<code>multi on/off</code>	定义 <code>/etc/hosts</code> 中输入的主机是否可以具有多个 IP 地址。
<code>nospoof on spoofalert on/off</code>	这些参数影响名称服务器 <i>spoofing</i> ，但除此之外，它们对网络配置没有任何影响。
<code>trim domainname</code>	在主机名解析后，指定的域名与主机名分开（只要主机名包括域名）。如果只有本地域的名称在 <code>/etc/hosts</code> 文件中，但仍应通过附带的域名进行识别，则可以使用此选项。

---

22.8: `/etc/host.conf`

```
# We have named running
order hosts bind
# Allow multiple addrs
```

### **/etc/nsswitch.conf**

GNU C Library 2.0 的引入与名称服务转换 (NSS) 的引入是同时进行的。有关详细信息，请参考 `man 5 nsswitch.conf` 和 *GNU C Library 参考手册*。

查询的顺序是在文件 `/etc/nsswitch.conf` 中定义的。例22.9

“`/etc/nsswitch.conf`” 中显示了 `nsswitch.conf` 的示例。注释以 `#` 符号开头。在本例中，`hosts` 数据库下的项意味着通过 DNS（请参见第24章 域名系统）将请求发送到 `/etc/hosts (files)`。

#### 22.9: `/etc/nsswitch.conf`

```
passwd:      compat
group:       compat

hosts:       files dns
networks:    files dns

services:    db files
protocols:   db files

netgroup:    files
automount:   files nis
```

表22.7 “通过 `/etc/nsswitch.conf` 可用的数据库” 中列出了 NSS 上可用的“数据库”。另外，近期将推出 `automount`、`bootparams`、`netmasks` 和 `publickey`。表22.8 “NSS “数据库” 的配置选项” 中列出了 NSS 数据库的配置选项。

*Table 22.7: 通过 `/etc/nsswitch.conf` 可用的数据库*

<code>aliases</code>	sendmail 实施的邮件别名；请参见 <code>man 5 aliases</code> 。
<code>ethers</code>	Ethernet 地址。
<code>group</code>	<code>getgrent</code> 使用的用户组。另请参见 <code>group</code> 的手册页。

hosts	gethostbyname 和类似函数使用的主机名和 IP 地址。
netgroup	网络中用于控制访问权限的主机和用户列表，请参见 man 5 netgroup。
networks	getnetent 使用的网络名称和地址。
passwd	getpwent 使用的用户口令；请参见 man 5 passwd。
protocols	getprotoent 使用的网络协议；请参见 man 5 protocols。
rpc	getrpcbyname 和类似函数使用的远程过程调用名称和地址。
services	getservent 使用的网络服务。
shadow	getspnam 使用的用户的影子口令；请参见 man 5 shadow。

Table 22.8: NSS “数据库”的配置选项

files	直接访问文件，例如 /etc/aliases
db	通过数据库访问
nis、nisplus	NIS，另请参见第25章使用 NIS
dns	只能用作 hosts 和 networks 的扩展
compat	只能用作 passwd、shadow 和 group 的扩展

### /etc/nscd.conf

此文件用于配置 nscd（名称服务超速缓存守护程序）。请参见 man 8 nscd 和 man 5 nscd.conf。默认情况下，passwd 和 groups 的系统项由 nscd 进行缓存。这对于目录服务（例如 NIS 和 LDAP）的性能很重要，因为如果不是这样，每次访问名称或组都需要网络连接。默认情况下，不对 hosts 进行缓存，因为 nscd 中缓存主机的机制将导致本地系统无法信任正向和反向查找检查。请设置缓存 DNS 服务器，而不是让 nscd 缓存名称。

如果激活 passwd 的缓存，则通常需要 15 秒才能识别新添加的本地用户。通过使用命令 `rcnscd restart` 重新启动 nscd 可缩短此等待时间。



## /etc/HOSTNAME

此文件提供不附带域名的主机名。当引导计算机时，此文件将被多个脚本读取。它可能只包含一行，该行中设置了主机名。

## 22.5.2 启动脚本

除了上面介绍的配置文件之外，还有多个脚本在引导计算机时装载网络程序。只要系统切换到某个多用户运行级别，就将启动这些脚本。表22.9“网络程序的一些启动脚本”中介绍了其中的一些脚本。

Table 22.9: 网络程序的一些启动脚本

<code>/etc/init.d/network</code>	此脚本处理网络接口的配置。硬件必须已被 <code>/etc/init.d/coldplug</code> （通过 <code>hotplug</code> ）初始化。如果未启动 <code>network</code> 服务，当通过热插拔插入网络接口时，不实施任何网络接口。
<code>/etc/init.d/inetd</code>	启动 <code>xinetd</code> 。 <code>xinetd</code> 可用于使服务器服务在系统上可用。例如，它可以在初始化 FTP 连接时启动 <code>vsftpd</code> 。
<code>/etc/init.d/portmap</code>	启动 RPC 服务器所需的端口映射器，例如 NFS 服务器。
<code>/etc/init.d/nfsserver</code>	启动 NFS 服务器。
<code>/etc/init.d/sendmail</code>	控制 <code>sendmail</code> 进程。
<code>/etc/init.d/ypserv</code>	启动 NIS 服务器。
<code>/etc/init.d/ypbind</code>	启动 NIS 客户程序。

## 22.6 作为拨号助手的 smpppd

多数家庭用户由于不具备连接到因特网的专线，所以使用拨号连接。根据所用的拨号方法（ISDN 或 DSL），连接受 `ippd` 或 `pppd` 的控制。基本上，只要正确启动这些程序就可以联网了。

如果采用包月付费方式（拨号连接不产生任何附加费用），则只需启动相应的守护程序。用 KDE 小程序或命令行界面来控制拨号连接。如果因特网网关不是您

所用的主机，最好通过网络主机来控制拨号连接。

这时就需要 smpppd 了。该程序为辅助程序提供统一的界面，并且可以双向执行。首先，它要对所需的 pppd 或 ipppd 编程，并控制其拨号属性。然后需要使各种提供程序对用户程序可用，并传送有关当前连接状态的信息。由于还可以通过网络来控制 smpppd，该程序适用于从专用子网中的工作站控制与因特网的拨号连接。

## 22.6.1 配置 smpppd

YaST 可以自动配置由 smpppd 提供的连接。同时还会预先配置实际的拨号程序 kinternet 和 cinternet。只有在配置 smpppd 的附加功能（如远程控制）时，才需要手工设置。

smpppd 的配置文件为 `/etc/smpppd.conf`。默认情况下并未启用远程控制。此配置文件最重要的选项包括：

**open-inet-socket = {yes—no}** 要通过网络控制 smpppd，必须将此选项设置为 `yes`。smpppd 的监听端口为 3185。如果此参数设置为 `yes`，则还需相应设置 `bind-address`、`host-range` 和 `password` 等参数。

**bind-address = {ip}** 如果主机有多个 IP 地址，使用此参数可以确定 smpppd 应在哪个 IP 地址上接受连接。

**host-range = {min ip ; max ip}** 参数 `host-range` 用于定义网络范围。IP 地址属于这一范围的主机将被授予访问 smpppd 的权限。此范围之外的所有主机均不具备访问权。

**password = {password}** 通过指派口令可使客户机仅限于授权主机。由于这是个纯文本口令，不应高估该口令提供的安全性。如果未指派任何口令，所有客户机都有权访问 smpppd。

**slp-register = {yes—no}** 使用此参数，可以通过 SLP 在网络中声明 smpppd 服务。

有关 smpppd 的详细信息，请参见 `man 8 smpppd` 和 `man 5 smpppd.conf`。

## 22.6.2 配置供远程使用的 kinternet、cinternet 和 qinternet

kiinternet、cinternet 和 qinternet 可用于控制本地或远程 smpppd。cinternet 是与图形化的 kinternet 对应的命令程序。qinternet 基本上与 kinternet 相

同，但不使用 KDE 库，因此可在无 KDE 的情况下使用它，但必须单独进行安装。要使这些实用程序可用于远程 smpppd，请手工编辑配置文件 `/etc/smpppd-c.conf` 或使用 `kinternet`。此文件仅使用三个选项：

**sites = ;list of sites;** 此选项可以向前端通知 smpppd 的搜索位置。前端将按照在此指定的选项顺序来测试这些选项。选项 `local` 指示与本地 smpppd 建立连接。`gateway` 指向网关上的 smpppd。将按照 `config-file` 的 `server` 中的指定建立连接。`slp` 指示前端连接通过 SLP 发现的 smpppd。

**server = ;server;** 在此指定 smpppd 运行所在的主机。

**password = ;password;** 插入为 smpppd 选择的口令。

如果 smpppd 处于活动状态，现在即可访问它，例如通过 `cinternet --verbose --interface-list` 来访问。如果此时遇到困难，请参考 `man 5 smpppd-c.conf` 和 `man 8 cinternet`。



# 网络中的 SLP 服务

制定*服务位置协议* (SLP) 是为了简化本地网络中联网客户机的配置。要配置网络客户机（包括所有必需服务），管理员通常需要对网络中提供的服务器有详细了解。SLP 可以向本地网络中的所有客户机声明特定服务是否可用。支持 SLP 的应用程序则可以利用这一发布信息并进行自动配置。

23.1	注册您自己的服务 . . . . .	364
23.2	SUSE LINUX 中的 SLP 前端 . . . . .	364
23.3	激活 SLP . . . . .	365
23.4	参考信息 . . . . .	365

SUSE LINUX 支持使用通过 SLP 提供的安装源进行安装，并且包含许多集成了 SLP 支持的系统服务。YaST 和 Konqueror 都有适用于 SLP 的前端。您可以使用 SLP 为联网客户机提供核心功能，如 SUSE LINUX 上的安装服务器、YOU 服务器、文件服务器或打印服务器。

## 23.1 注册您自己的服务

SUSE LINUX 中的许多应用程序都已使用 `libslp` 库集成了 SLP 支持。如果服务未集成 SLP 支持，请使用以下方法之一使其可通过 SLP 声明。

通过 `/etc/slp.reg.d` 进行的静态注册

为每个新服务创建单独的注册文件。下面显示了注册扫描仪服务的文件的示例：

```
## Register a saned service on this system ## en means english language ## 65535
```

此文件中最重要的一行是以 `service:` 开头的服务 URL。其中包含服务类型 (`scanner.sane`) 以及该服务在服务器上的地址。 `<$HOSTNAME>` 自动用完整主机名替换。随后是可以找到相关服务的 TCP 端口的名称，端口与主机名之间用冒号分隔。然后输入服务的显示语言及以秒计的注册持续时间。应该用逗号分隔服务 URL 之后的各项内容。将注册持续时间设置为 0 到 65535 之间的值。0 表示禁止注册。65535 表示取消所有限制。

配置文件还包含 `watch-tcp-port` 和 `description` 这两个变量。第一个变量将 SLP 服务声明与相关服务是否处于活动状态联系起来（`slpd` 会检查服务的状态）。第二个变量为显示在适合的浏览器中的服务提供了更为准确的说明。

通过 `/etc/slp.reg` 进行的静态注册

与上述过程的唯一差别即在于：这种注册方式要将所有服务都集中到一个核心文件中。

使用 `slptool` 进行的动态注册 如果应该从专有脚本为某项服务注册 SLP 支持，请使用 `slptool` 命令行前端。

## 23.2 SUSE LINUX 中的 SLP 前端

SUSE LINUX 包含若干前端，支持检查 SLP 信息并通过网络使用这些信息。

**slptool** slptool 是一个简单的命令程序，可用于在网络中声明 SLP 查询或声明专有服务。slptool --help 列出了所有可用选项和功能。也可以从处理 SLP 信息的脚本调用 slptool。

**YaST SLP 浏览器** YaST 包含一个单独的 SLP 浏览器，可以在树状图的‘网络服务’→‘SLP 浏览器’下面列出本地网络中通过 SLP 声明的所有服务。

**Konqueror** 用作网络浏览器时，Konqueror 可以在 slp:/ 中显示本地网络中的所有可用 SLP 服务。单击主窗口中的图标可获得有关相关服务的详细信息。

如果在 Konqueror 中使用 service:/，在浏览器窗口中单击相关图标可与选定服务建立连接。

## 23.3 激活 SLP

如果要提供服务，必须在您的系统中运行 slpd。如果只需发出服务查询，则不必启动此守护程序。类似 SUSE LINUX 中的多数系统服务，slpd 守护程序通过单独的 ini 脚本来控制。默认情况下该守护程序处于非活动状态。要在会话期间激活该守护程序，请以 root 身份运行 rcsldap start 来启动它，运行 rcsldap stop 来停止它。使用 restart 或 status 可分别执行重新启动或状态检查。如果默认情况下 slpd 处于活动状态，请以 root 身份运行 insserv slpd 命令一次。这样即可在引导系统时将 slpd 自动加入要启动的一组服务。

## 23.4 参考信息

以下来源提供了有关 SLP 的详细信息：

**RFC 2608、2609、2610** RFC 2608 主要说明了 SLP 的定义。RFC 2609 更详细地说明了所用服务 URL 的语法；RFC 2610 则对通过 SLP 的 DHCP 进行了说明。

<http://www.openslp.com> OpenSLP 项目的主页。

`file:/usr/share/doc/packages/openslp/*`

此目录包含有关 SLP 的所有现有文档，其中包括 README.SuSE（包含 SUSE LINUX 详细信息）、上述 RFC 和两个介绍性的 HTML 文档。要使用 SLP 功能的编程人员应该安装 openslp-devel 包，以便参考随包附带的编程人员指南。





# 域名系统

需要使用 DNS（域名系统）将域和主机名解析成 IP 地址。这样，可以将 IP 地址指派给主机名，例如将 192.168.0.1 指派给 earth。在设置您自己的名称服务器前，请阅读第22.3节“名称解析”中有关 DNS 的一般信息。以下配置示例都涉及到 BIND。

24.1	用 YaST 进行配置	368
24.2	启动名称服务器 BIND	372
24.3	配置文件 /etc/named.conf	376
24.4	区域文件	380
24.5	区域数据的动态更新	382
24.6	安全事务	383
24.7	DNS 安全性	384
24.8	参考信息	384

## 24.1 用 YaST 进行配置

可以使用 YaST 的 DNS 模块来为您的本地网络配置 DNS 服务器。第一次启动此模块时，向导启动，提示您做出一些有关服务器管理的基本决定。完成此初始设置将生成一个非常基本的服务器配置，此配置可以使服务器在各基本方面正常工作。专家方式可用于处理更高级的配置任务，。

### 24.1.1 向导配置

向导由三个步骤或对话框组成。您可以在对话框的适当位置进入专家配置方式。

**转发器设置** 第一次启动模块时，您将看到图24.1 “DNS 服务器安装：转发器设置”中显示的对话框。此对话框允许您决定是让 PPP 守护程序在通过 DSL 或 ISDN 进行拨号时提供转发器列表（‘PPP 守护程序设置转发器’）还是提供您自己的列表（‘手工设置转发器’）。

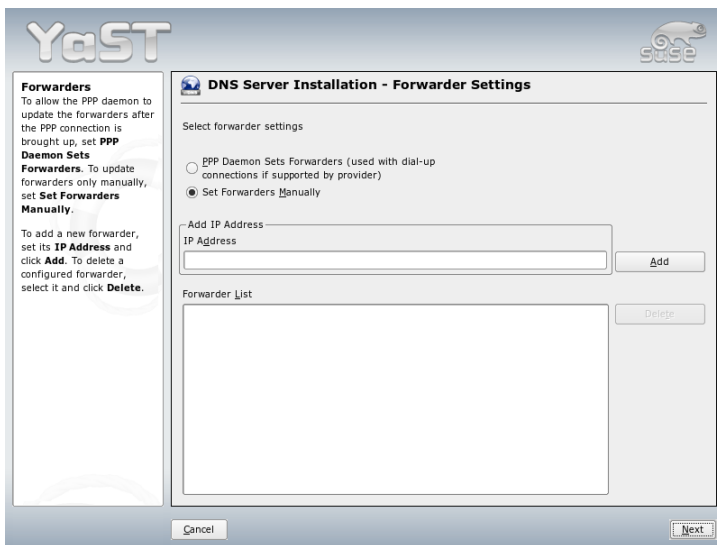


Figure 24.1: DNS 服务器安装：转发器设置

**DNS 区域** 此对话框由多个部分组成，负责管理区域文件（如第24.4节“区域文件”中所述）。对于新区域，请在‘区域名称’中为其提供一个名称。要添加反向区域，名称必须以 `.in-addr.arpa` 结尾。最后，选择‘区域类型’（主区域或从属区域）。请参见图24.2“DNS 服务器安装：DNS 区域”。单击‘编辑区域’来配置现有区域的其他设置。要去除区域，请单击‘删除区域’。

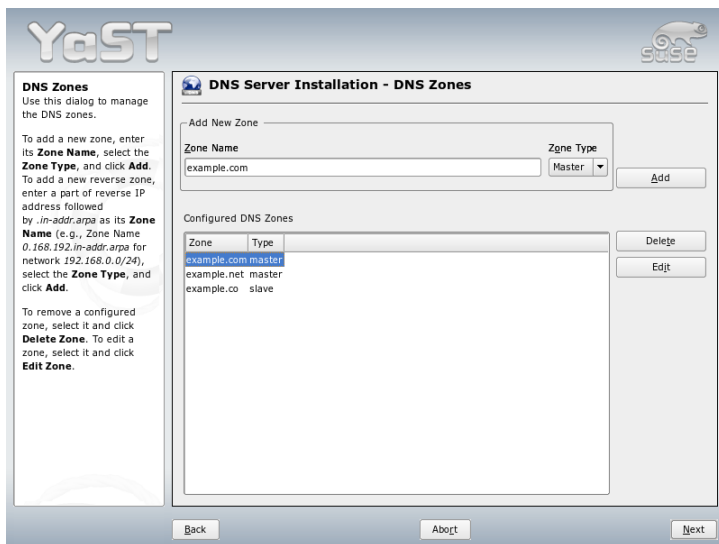


Figure 24.2: DNS 服务器安装：DNS 区域

**完成向导** 在最后的对话框中，您可以打开在安装期间激活的防火墙中的 DNS 服务端口，并确定是否启动 DNS。还可以从此对话框中访问专家配置。请参见图24.3“DNS 服务器安装：完成向导”。

## 24.1.2 专家配置

启动此模块后，YaST 将打开一个窗口，其中显示了多个配置选项。完成此窗口会生成具有基本功能的 DNS 服务器配置：

**启动** 在‘引导’下，定义 DNS 服务器在默认情况下应为‘On’还是‘Off’。要立即启动 DNS 服务器，请选择‘立即启动 DNS 服务器’。要停止 DNS 服务器，

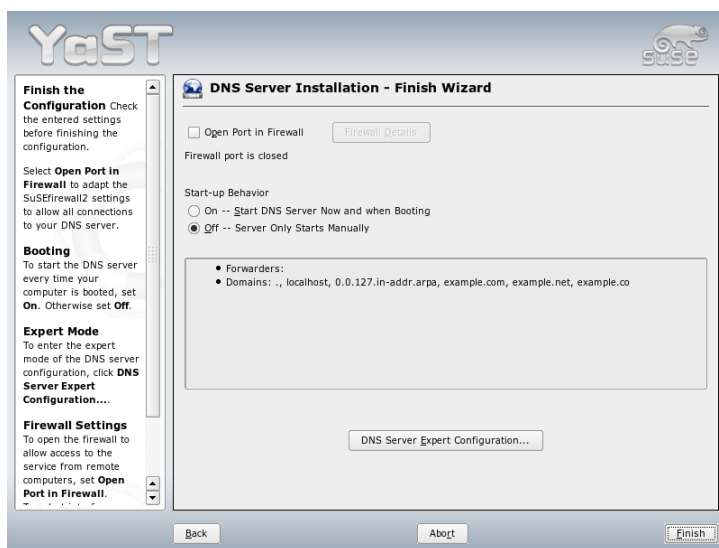


Figure 24.3: DNS 服务器安装：完成向导

请选择‘立即停止 DNS 服务器’。要保存当前设置，请选择‘立即保存设置并重新启动 DNS 服务器’。您可以用‘打开防火墙中的端口’打开防火墙中的 DNS 端口，并用‘防火墙细节’修改防火墙设置。

**转发器** 这个对话框与启动向导配置后打开的对话框是同一个对话框（请参见）。

**日志记录** 此部分允许您设置 DNS 服务器如何进行记录以及记录哪些内容。在‘日志类型’下，指定 DNS 服务器将日志数据写入的位置。选择‘记录到系统日志’使用系统级日志文件 `/var/log/messages`，或选择‘记录到文件’指定一个不同的文件。对于后者，还可以指定最大文件大小（以兆字节为单位）和要储存的日志文件的数目。

在‘附加日志记录’下可以使用其他一些选项：启用‘记录命名查询’将记录每个查询，在这种情况下，日志文件可能会变得非常大。出于这个原因，如果不是为了调试，则最好不要启用此选项。要记录区域更新期间 DHCP 和 DNS 服务器之间的通讯数据，请启用‘记录区域更新’。要记录将区域从主服务器传送到从属服务器期间的通讯数据，请启用‘记录区域传送’。请参见图 24.4 “DNS 服务器：日志记录”。

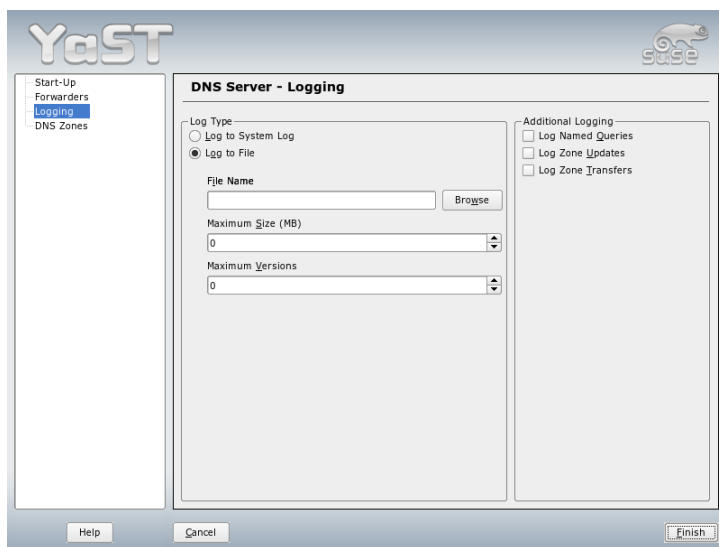


Figure 24.4: DNS 服务器：日志记录

**DNS 区域** 在介绍向导配置时已介绍了此对话框。请参见第24.1.1节“向导配置”。

**从属区域编辑器** 如果您在介绍的步骤中选择了区域类型‘从属’，则将打开此对话框。在‘主 DNS 服务器’下，指定从属服务器将从中获取数据的主服务器。要限制对此服务器的访问，可从列表选择一个 ACL。请参见图24.5“DNS 服务器：从属区域编辑器”。

**主区域编辑器** 如果您在介绍的步骤中选择了区域类型‘主’，则将打开此对话框。此对话框由多个页组成：‘基本’（首先打开）、‘NS 记录’、‘MX 记录’、‘SOA’ 和‘记录’。

**区域编辑器（NS 记录）** 此对话框用于为指定的区域定义替代名称服务器。确保已将自己的名称服务器包括在列表中。要添加记录，请在‘要添加的名称服务器’下输入其名称，然后用‘添加’按钮确认。请参见图24.7“DNS 服务器：区域编辑器（NS 记录）”。

**区域编辑器（MX 记录）** 要将当前区域的邮件服务器添加到现有的列表中，请输入相应的地址和优先级值。执行完此操作后，请选择‘添加’进行确认。请

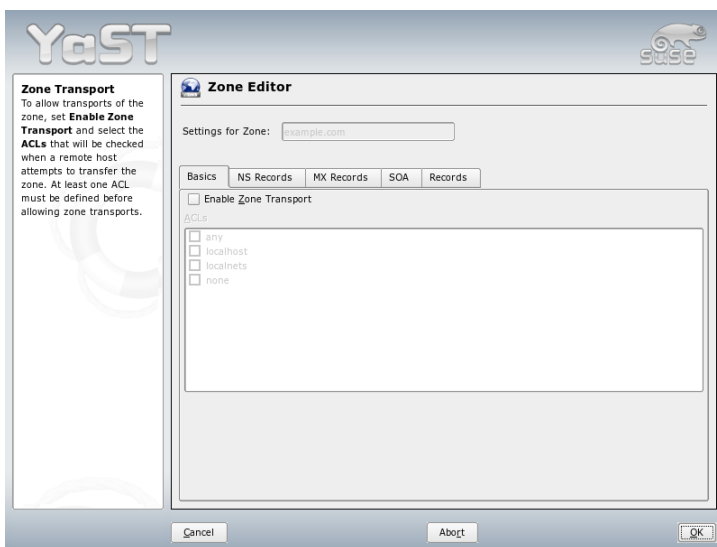


Figure 24.5: DNS 服务器：从属区域编辑器

参见图24.8 “DNS 服务器：区域编辑器（MX 记录）”。

**区域编辑器 (SOA)** 此页用于创建 SOA（起始授权机构）记录。有关各选项的说明，请参考例24.6 “文件 /var/lib/named/world.zone”。通过 LDAP 管理的动态区域不支持更改 SOA 记录。

**区域编辑器（记录）** 此对话框用于管理名称解析。在‘记录密钥’中，输入主机名并选择其类型。‘A 记录’表示主要项。此项的值应为一个 IP 地址。‘CNAME’是别名。对于要根据‘NS 记录’和‘MX 记录’选项卡中提供的信息而扩展的详细或部分记录，应使用类型‘NS’和‘MX’。这三种类型解析为现有的 A 记录。‘PTR’用于反向区域。它与 A 记录相反。

## 24.2 启动名称服务器 BIND

在 SUSE LINUX 系统中，已预先配置了名称服务器 BIND（*Berkeley Internet Name Domain*，*Berkeley 因特网名称域*），因此可以在安装后立即启动此名称服务器而不会出现任何问题。如果您已有一个有效的因特网连接并在

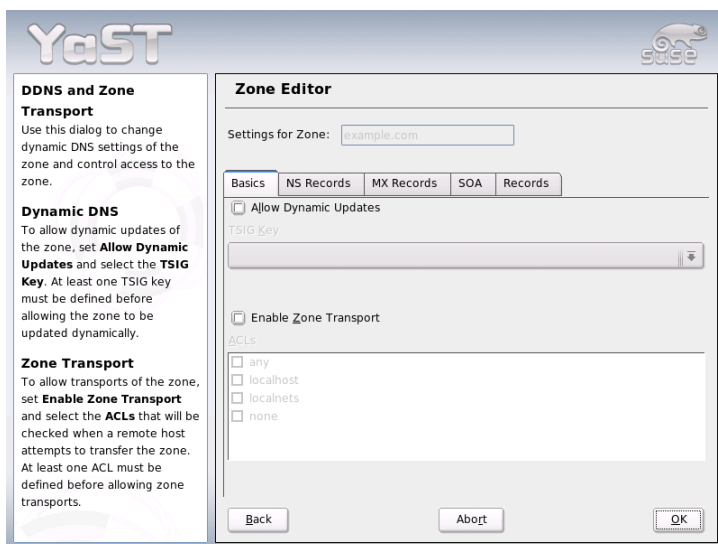


Figure 24.6: DNS 服务器：区域编辑器（基本）

`/etc/resolv.conf` 中输入了 `127.0.0.1` 作为 `localhost` 的名称服务器地址，那么您已经在使用名称解析功能了，而不需要知道提供商的 DNS。BIND 通过根名称服务器执行名称解析，这个过程非常慢。通常，应将提供商的 DNS 及其 IP 地址输入配置文件 `/etc/named.conf` 的 `forwarders` 下，以确保能进行有效而安全的名称解析。如果到目前为止是这种情况，则该名称服务器将作为仅用于超速缓存的纯名称服务器运行。只有在配置了该名称服务器自己的区域后，它才能成为一个正常的 DNS。`/usr/share/doc/packages/bind/sample-config` 的文档中包含了这种情况的一个简单示例。

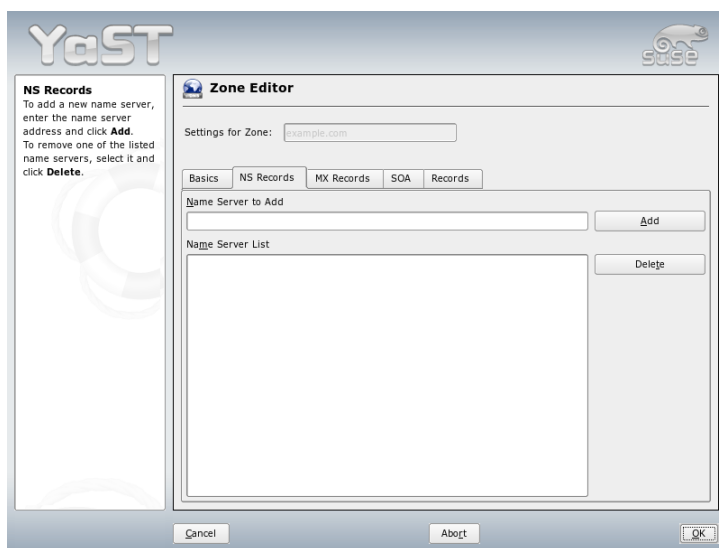


Figure 24.7: DNS 服务器：区域编辑器（NS 记录）

## Tip

### 名称服务器信息的自动适应

根据因特网连接或网络连接的类型，名称服务器信息可以自动适应当前的情况。为此，请将文件 `/etc/sysconfig/network/config` 中的变量 `MODIFY_NAMED_CONF_DYNAMICALY` 设置为 `yes`。

## Tip

但是，在相关机构指派域之前，请不要设置任何正式域。即使您有自己的域且提供商管理此域，也最好不要使用此域，因为如果使用此域，BIND 将不转发对此域的请求。例如，此域不能访问提供商的万维网服务器。

要启动名称服务器，请以 `root` 用户身份输入命令 `rcnamed start`。如果右侧出现绿色的“完成”，则说明已成功启动了名称服务器进程（该进程名为 `named`）。请用 `host` 或 `dig` 程序立即在本地系统上测试名称服务器，该测试应返回 `localhost` 作为默认服务器，地址为 `127.0.0.1`。如果未返回所需的结果，则 `/etc/resolv.conf` 可能包含不正确的名称服务器项或此文件根本不存在。如果是第一次测试，请输入 `host 127.0.0.1`，此命令应始



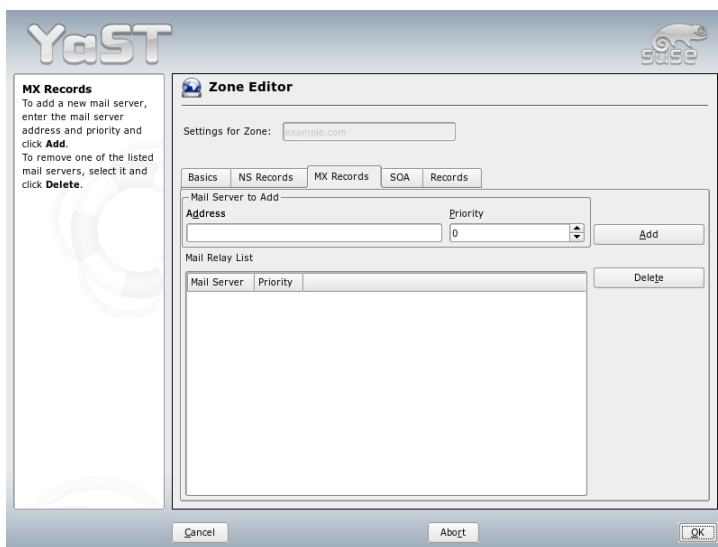


Figure 24.8: DNS 服务器：区域编辑器（MX 记录）

终有效。如果收到错误讯息，请使用 `rcnamed status` 查看服务器是否确实在运行。如果名称服务器未启动或出现意外的行为，则通常可以在日志文件 `/var/log/messages` 中找到原因。

要将提供商的名称服务器或网络上正在运行的名称服务器用作转发器，请在 `options` 部分的 `forwarders` 下输入相应的一个或多个 IP 地址。例24.1 “named.conf 中的转发选项”中包含的地址仅用作示例。请根据您的设置调整这些项。

#### 24.1: named.conf 中的转发选项

```
options {
    directory "/var/lib/named";
    forwarders { 10.11.12.13; 10.11.12.14; };
    listen-on { 127.0.0.1; 192.168.0.99; };
    allow-query { 127/8; 192.168.0/24; };
    notify no;
};
```

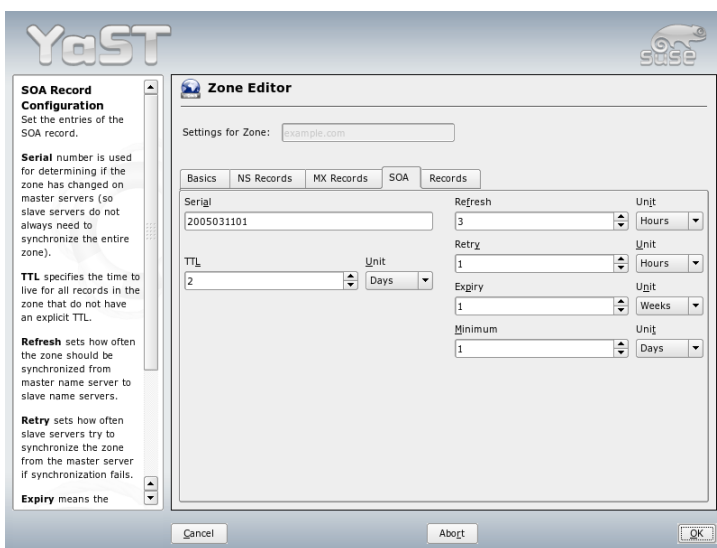


Figure 24.9: DNS 服务器：区域编辑器 (SOA)

options 项后面是区域的项 localhost 和 0.0.127.in-addr.arpa。“.”下的 type hint 项应始终存在。无需修改相应的文件，应照原样使用。还要确保每个项都以“;”结束，同时确保花括号位于正确的位置。在更改配置文件 /etc/named.conf 或区域文件后，通知 BIND 使用 `rndc reload` 重新读取这些文件。用 `rndc restart` 停止并重启动名称服务器也会获得相同的效果。输入 `rndc stop` 可以随时停止服务器。

## 24.3 配置文件 /etc/named.conf

BIND 名称服务器本身的所有设置都被储存在文件 /etc/named.conf 中。但是，将要处理的域的区域数据（由主机名、IP 地址等组成）储存在 /var/lib/named 目录下单独的文件中。有关此内容的详细信息在下面有所介绍。

/etc/named.conf 大致分为两部分。一部分是存放常规设置的 options 部分，另一部分由各个域的 zone 项组成。而 logging 部分和 acl（访问控制列表）项是可选的。注释行以 # 符号或 // 开头。例 24.2 “基本的 /etc/named.conf” 显示了一个最小的 /etc/named.conf。

## 24.2: 基本的 `/etc/named.conf`

```
options {
    directory "/var/lib/named";
    forwarders { 10.0.0.1; };
    notify no;
};

zone "localhost" in {
    type master;
    file "localhost.zone";
};

zone "0.0.127.in-addr.arpa" in {
    type master;
    file "127.0.0.zone";
};

zone "." in {
    type hint;
    file "root.hint";
};
```

### 24.3.1 重要的配置选项

**directory "*filename*";** 指定目录，BIND 可以在该目录中找到包含区域数据的文件。通常，此目录是 `/var/lib/named`。

**forwarders { *ip-address*; };** 指定在无法直接解析 DNS 请求的情况下应将其转发到的名称服务器（大多数情况下是提供商的名称服务器）。用 IP 地址（例如 `10.0.0.1`）替换 *ip-address*。

**forward first;** 在尝试通过根名称服务器解析 DNS 请求前，对 DNS 请求进行转发。可以写入 `forward only`（而不是 `forward first`）转发所有请求并且不将任何请求发送到根名称服务器。这可以用于防火墙配置。

**listen-on port 53 { 127.0.0.1; *ip-address*; };**

指示 BIND 通过哪个网络接口和哪个端口来接受客户机查询。不需要显式指定 `port 53`，因为 53 是默认端口。输入 `127.0.0.1` 允许接收来自 Localhost 的请求。如果完全省略此项，则在默认情况下使用所有接口。

**listen-on-v6 port 53 {any;};** 指示 BIND 应通过哪个端口监听 IPv6 客户机请求。唯一可以替代 any 的是 none。就 IPv6 而言，服务器只接受通配符地址。

**query-source address \* port 53;** 如果防火墙阻止外发的 DNS 请求，则需要此项。此项指示 BIND 从端口 53 向外部发送请求，而不使用端口号大于 1024 的任何端口。

**query-source-v6 address \* port 53;** 指示 BIND 将哪个端口用于 IPv6 查询。

**allow-query { 127.0.0.1; <net>;};** 定义客户机可以从中发送 DNS 请求的网络。用地址信息（例如 192.168.1/24）替换 <net>。末尾的 /24 是网络掩码的缩写表示，本例中，网络掩码是 255.255.255.0。

**allow-transfer !\*;;** 控制哪些主机可以请求区域传送。在本例中，用 !\* 可以完全拒绝此类请求。如果没有此项，则可以从任何位置请求区域传送，没有任何限制。

**statistics-interval 0;** 如果缺少此项，则 BIND 每小时在 /var/log/messages 中生成几行统计信息。将其设置为 0 可以完全禁止生成此类统计信息，也可以设置时间间隔（以分钟为单位）。

**cleaning-interval 720;** 此选项定义 BIND 间隔多长时间清除其超速缓存。每次出现此选项都会在 /var/log/messages 中触发一项。时间是以分钟为单位指定的。默认值是 60 分钟。

**interface-interval 0;** BIND 定期在网络接口中搜索新接口或不存在的接口。如果将该值设置为 0，则不执行搜索，BIND 只监听启动时检测到的接口。否则，采用分钟定义时间间隔。默认值是 60 分钟。

**notify no;** 指定 no 将阻止其他名称服务器在区域数据被更改或名称服务器被重新启动时得到通知。

## 24.3.2 日志记录

可以在 BIND 中详细配置日志记录的内容、方式和位置。通常，默认设置已足够了。例 24.3 “禁用日志记录的项”显示了此项最简单的形式，并完全禁止任何日志记录。

### 24.3: 禁用日志记录的项

```
logging {  
    category default { null; };  
};
```

### 24.3.3 区域项

#### 24.4: *my-domain.de* 的区域项

```
zone "my-domain.de" in { type master; file "my-domain.zone"; notify no; }
```

在 `zone` 后，指定要管理的域名 `my-domain.de`，后跟 `in` 和用花括号括起来的相关选项块，如例24.4 “*my-domain.de* 的区域项”所示。要定义从属区域，请将 `type` 切换为 `slave` 并将管理此区域的名称服务器指定为 `master`（它可能是另一个主区域的从属区域），如例24.5 “*other-domain.de* 的区域项”所示。

#### 24.5: *other-domain.de* 的区域项

```
zone "other-domain.de" in { type slave; file "slave/other-domain.zone"; masters { 10
```

区域选项：

**type master;** 通过指定 `master`，指示 BIND 由本地名称服务器对区域进行处理。这假定已用正确的格式创建了区域文件。

**type slave;** 从另一个名称服务器传送此区域。必须将它与 `masters` 一起使用。

**type hint;** `hint` 类型的区域。用于设置根名称服务器。此区域定义可以保留原样。

**file `my-domain.zone` 或 file “`slave/other-domain.zone`”;**

此项指定域的区域数据所在的文件。从属区域不需要此文件，因为此数据是从另一个名称服务器中获取的。要区分主文件和从属文件，请对从属文件使用目录 `slave`。

**masters { `<server-ip-address>`; };** 只有从属区域需要此项。它指定应从哪个名称服务器传送区域文件。

**allow-update { `!*`; };** 此选项控制外部写访问，这将允许客户端创建 DNS 项——出于安全原因，通常不希望出现这种情况。没有此项，就不允许进行区域更新。上述项可以实现相同的结果，因为 `!*` 有效地禁止了此类操作。

## 24.4 区域文件

所需的区域文件有两种类型。一种类型是将 IP 地址指派给主机名，另一种类型则正相反：为 IP 地址提供主机名。

### Tip

在区域文件中使用点

· 在区域文件中有重要的含义。如果给出主机名而末尾没有加 .，则会追加区域。通过完整域名指定的完整主机名必须以 . 结尾，避免再将域添加到主机名上。点丢失或放错位置可能是名称服务器配置出错最常见的原因。

### Tip

首先研究以下区域文件 `world.zone`，该区域文件负责域 `world.cosmos`，如例 24.6 “文件 `/var/lib/named/world.zone`” 所示。

24.6: 文件 `/var/lib/named/world.zone`

```
1 $TTL 2D world.cosmos. IN SOA      gateway root.world.cosmos. ( 2003072441 ;
2
3 IN NS      gateway IN MX      10 sun
4
5 gateway    IN A      192.168.0.1 IN A      192.168.1.1 sun      IN A
```

第 1 行： `$TTL` 定义默认存活时间，它适用于此文件中的所有项。在本例中，项在两天 (2 D) 内有效。

第 2 行： 这是 SOA (Start Of Authority, 起始授权机构) 控制记录开始的位置：

- 在第一个位置，要管理的域的名称是 `world.cosmos`。这个域名以 . 结尾，否则可能会再次追加区域。或者，可以在这里输入 @，在这种情况下，可以从 `/etc/named.conf` 中的相应项中抽取区域。
- IN SOA 之后是用作此区域的主服务器的名称服务器的名称。此名称从 `gateway` 扩展为 `gateway.world.cosmos`，因为它没有以 . 结尾。

- 随后是负责此名称服务器的用户的电子邮件地址。因为 @ 符号已经有特殊含义，所以在这里改为输入 `.`。对于 `root@world.cosmos`，此项必须是 `root.world.cosmos.`。必须在末尾包含 `.`，以防止添加区域。
- ( 和 ) 之间包含的所有行组成 SOA 记录。

第 3 行: `serial number` 可以是任一数字，每次更改此文件此数字都会增加。需要将这些更改通知给辅助名称服务器（从属服务器）。为此，日期和运行数字常采用十位数字格式，书写方式为 `YYYYMMDDNN`。

第 4 行: `refresh rate` 指定辅助名称服务器校验区域 `serial number` 的时间间隔。在本例中，此时间间隔为一天。

第 5 行: `retry rate` 指定辅助名称服务器在出现错误时尝试再次联系主服务器的时间间隔。这里的时间间隔是两小时。

第 6 行: `expiration time` 指定在辅助名称服务器无法重新联系上主服务器的情况下，辅助名称服务器在多长时间后丢弃超速缓存的数据。这里为一周。

第 7 行: SOA 记录中的最后一项指定 `negative caching TTL`— 超速缓存来自其他服务器的未解析 DNS 查询结果的时间。

第 9 行: `IN NS` 指定负责此域的名称服务器。`gateway` 被扩展为 `gateway.world.cosmos`，因为它没有以 `.` 结尾。可以有多个与此行类似的行——一行用于主名称服务器，其他每行分别用于每个辅助名称服务器。如果 `/etc/named.conf` 中未将 `notify` 设置为 `no`，则会将区域数据的更改通知给这里列出的所有名称服务器。

第 10 行: `MX` 记录指定接受、处理和转发域 `world.cosmos` 的电子邮件的邮件服务器。在本例中，邮件服务器是主机 `sun.world.cosmos`。主机名称前面的数字是优先顺序值。如果有多个 `MX` 项，则首先采用具有最小值的邮件服务器，如果向此服务器递送邮件失败，则尝试采用具有稍大一些值的邮件服务器。

第 12-17 行: 这些都是实际的地址记录，在这里将一个或多个 IP 地址指派到主机名。这里列出的名称不带 `.`，因为这些名称不包含它们的域，所以会在所有这些名称后面添加 `world.cosmos`。因为主机 `gateway` 有两个网卡，所以为其指派两个 IP 地址。只要主机地址是传统地址 (IPv4)，就将使用 `A` 标记该记录。如果地址是 IPv6 地址，则使用 `A6` 标记此项。IPv6 地址以前的标记是 `AAAA`，现在已经不使用该标记了。

第 18 行: 别名 `www` 可用于确定 `mond` (`CNAME` 是指规范名称) 的地址。

伪域 `in-addr.arpa` 用于 IP 地址到主机名的反向查找。它被追加到采用反向表示法的地址的网络部分。因此，将 `192.168.1` 解析成 `1.168.192.in-addr.arpa`。请参见例24.7“反向查找”。

## 24.7: 反向查找

```
1 $TTL 2D 1.168.192.in-addr.arpa. IN SOA gateway.world.cosmos. root.world.cosmos. ( 2003072441
2
3 IN NS          gateway.world.cosmos.
4
5 1              IN PTR          gateway.world.cosmos. 2              IN PTR
```

第 1 行： `$TTL` 定义应用于此处所有项的标准 TTL。

第 2 行： 此配置文件应激活网络 `192.168.1.0` 的反向查找。由于区域名为 `1.168.192.in-addr.arpa`，因此不应将此区域添加到主机名中。因此，采用完整形式输入所有主机名 — 带有它们的域并且在末尾有 `.`。其余的项对应于上一个 `world.cosmos` 示例介绍的那些内容。

第 3-7 行： 请参见上一个关于 `world.cosmos` 的示例。

第 9 行： 此行也是指定负责此区域的名称服务器。但这次采用完整形式输入名称，带有域且末尾带有 `.`。

第 11-13 行： 这些都是提示各自主机上 IP 地址的指针记录。只在行的开头输入 IP 地址的最后一部分，在末尾不加 `.`。将区域追加到这个地址（不带 `.in-addr.arpa`）将产生采用反向顺序的完整 IP 地址。

通常，可以在 BIND 的不同版本间传送区域，不会产生任何问题。

## 24.5 区域数据的动态更新

动态更新这一术语是指添加、更改或删除主服务器区域文件中项的操作。RFC 2136 对此机制进行了介绍。通过添加可选的 `allow-update` 或 `update-policy` 规则，可以为每个区域项单独配置动态更新。不应手工编辑要动态更新的区域。

用命令 `nsupdate` 将要更新的项传送到服务器。有关此命令的精确语法，请查看 `nsupdate` 的手册页 (`man 8 nsupdate`)。出于安全原因，应使用第24.6节“安全事务”中介绍的 TSIG 密钥执行此类更新。



## 24.6 安全事务

借助于基于共享密钥（也称为 TSIG 密钥）的事务签名 (TSIG) 可以实现安全事务。本节介绍如何生成和使用此类密钥。

不同服务器间的通讯和区域数据的动态更新需要安全事务。依靠密钥进行访问控制比只靠 IP 地址进行访问控制要安全得多。

使用下列命令生成 TSIG 密钥（有关详细信息，请参见 `man dnssec-keygen`）：

```
dnssec-keygen -a hmac-md5 -b 128 -n HOST host1-host2
```

此命令创建两个文件，名称与下面的名称类似：

```
Khost1-host2.+157+34265.private Khost1-host2.+157+34265.key
```

密钥本身（类似于 `ejIkuCyyGJwwuN3xAteKgg==` 的字符串）位于这两个文件中。要将密钥用于事务，必须将第二个文件 (`Khost1-host2.+157+34265.key`) 传送到远程主机，而且最好采用安全的方式（例如，使用 `scp`）传送。在远程服务器上，密钥必须包括在文件 `/etc/named.conf` 中以实现 `host1` 和 `host2` 之间的安全通讯：

```
key host1-host2. { algorithm hmac-md5; secret "ejIkuCyyGJwwuN3xAteKgg=="; }
```

### Warning

`/etc/named.conf` 的文件权限

确保正确限制了 `/etc/named.conf` 的权限。此文件的默认值是 0640，拥有者为 `root` 和组 `named`。或者，可以将密钥移到具有特殊限制权限的另一个文件中，然后将该文件包括在 `/etc/named.conf` 中。

### Warning

要使服务器 `host1` 能使用 `host2`（在本例中，其地址为 `192.168.2.3`）的密钥，服务器的 `/etc/named.conf` 必须包含下列规则：

```
server 192.168.2.3 { keys { host1-host2. ; }; };
```

必须将类似的项包括在 `host2` 的配置文件中。

向为 IP 地址和地址范围定义的任何 ACL（访问控制列表 — 请不要与文件系统的 ACL 混淆）添加 TSIG 密钥可以实现事务安全性。相应的项如下所示：

```
allow-update { key host1-host2. ;};
```

*BIND 管理员参考手册*的 `update-policy` 对此主题进行了详细介绍。

## 24.7 DNS 安全性

RFC 2535 中介绍了 DNSSEC (即 DNS 安全性)。BIND 手册讨论了可用于 DNSSEC 的工具。

被认为是安全的区域必须有一个或多个与之关联的区域密钥。这些密钥是通过 `dnssec-keygen` 生成的,就像主机密钥一样。当前使用 DSA 加密算法来生成这些密钥。应使用 `$INCLUDE` 规则将所生成的公共密钥包括在相应的区域文件中。

使用命令 `dnssec-makekeyset` 将生成的所有密钥打包成一个密钥集,然后必须采用安全方式将这个密钥集传送给父区域。在父区域,用 `dnssec-signkey` 对此密钥集进行签名。然后,通过 `dnssec-signzone` 使用此命令生成的文件对区域进行签名,这又会为 `/etc/named.conf` 中的每个区域生成要包含的文件。

## 24.8 参考信息

有关其他信息,请参考安装在 `/usr/share/doc/packages/bind/` 下的 *BIND 管理员参考手册*。另外,请考虑参考该手册中所引用的 RFC 和 BIND 附带的手册页。`/usr/share/doc/packages/bind/README.SuSE` 包含有关 SUSE LINUX 中 BIND 的最新信息。

# 使用 NIS

一旦网络内的多个 UNIX 系统都要访问公共资源，那么对于该网络内的所有计算机，所有用户和组身份是否相同就显得很重要了。网络对用户应该是透明的：无论用户使用什么计算机，其所处的环境应该是一成不变的。这可以通过 NIS 和 NFS 服务实现。NFS 通过网络分发文件系统，详细信息请参见第26章 通过 NFS 共享文件系统。

NIS（网络信息服务）可以说是一种数据库式服务，用于跨网络访问 `/etc/passwd`、`/etc/shadow` 和 `/etc/group` 的内容。NIS 也可用于其他目的（如提供 `/etc/hosts` 或 `/etc/services` 之类文件的内容），但这里不作介绍。人们常把 NIS 称作 YP，也就是网络中的“电话黄页”。

25.1	配置 NIS 服务器 . . . . .	386
25.2	配置 NIS 客户程序 . . . . .	388

## 25.1 配置 NIS 服务器

要配置服务器，请从 YaST 的‘网络服务’模块中选择‘NIS 服务器’。如果网络中还没有 NIS 服务器，请在下一屏幕中激活‘安装和设置主 NIS 服务器’。如果已有 NIS 服务器（主服务器），便可添加 NIS 从属服务器（例如，想要配置新的子网）。首先要说明的是主服务器的配置。

如果缺少某些必要的软件包，请按要求插入 CD 或 DVD 自动安装这些软件包。在配置对话框顶部输入域名，如图 25.1 “NIS 服务器配置工具” 所示。通过复选框定义该主机是否还应该是 NIS 客户程序，支持用户从 NIS 服务器登录并访问数据。

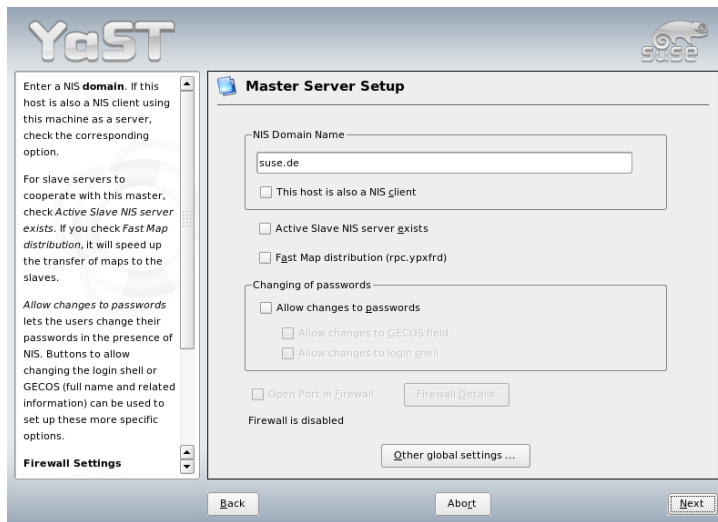


Figure 25.1: NIS 服务器配置工具

如果随后要配置网络中的其他 NIS 服务器（从属服务器），现在需要激活‘存在活动的从属 NIS 服务器’。选择‘快速映射分布’，设置将数据库项从主服务器快速传送到从属服务器。

要允许网络中的用户（本地用户和通过 NIS 服务器管理的用户）更改其在 NIS 服务器上的密码（使用命令 `yppasswd`），请激活相应的选项。这会相应激活‘允许更改 GECOS 字段’和‘允许更改登录 Shell’。“GECOS”意味着用户还可

以使用命令 `ypchfn` 更改其名称和地址设置。“SHELL”允许用户使用命令 `ypchsh` 更改其默认 shell，例如从 `bash` 改为 `sh`。

通过单击‘其他全局设置’，可访问图25.2“更改 NIS 服务器目录并同步文件”所示的屏幕，在该屏幕中可更改 NIS 服务器的源目录（默认为 `/etc`）。此外，也可以在此合并口令和组。应设置为‘是’，以便同步文件（`/etc/passwd`、`/etc/shadow` 和 `/etc/group`）。还可以确定最小用户 ID 和组 ID。按‘确定’确认设置并返回上一屏幕。然后单击‘下一步’。

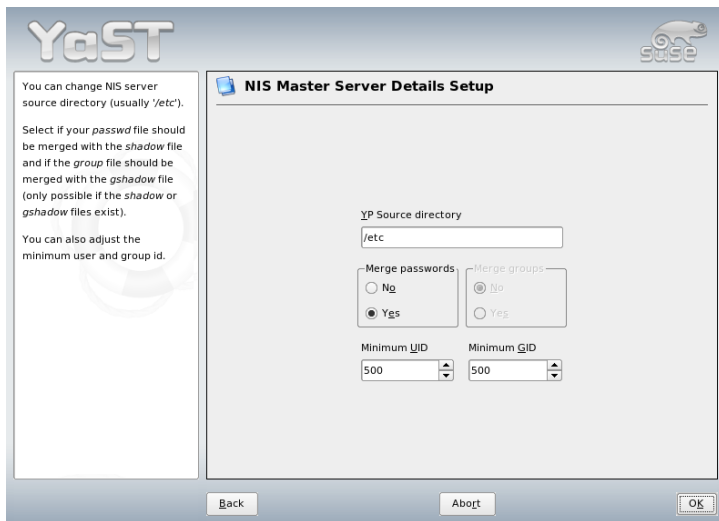


Figure 25.2: 更改 NIS 服务器目录并同步文件

如果先前启用了‘存在活动的从属 NIS 服务器’，则输入用作从属服务器的主机的名称，然后单击‘下一步’。如果未使用从属服务器，就会跳过从属服务器配置，直接转至数据库配置对话框。在此指定映射，可以将部分数据库从 NIS 服务器传送到客户机。默认设置通常就足够了。

选择‘下一步’可前进到最后一个对话框，如图25.3“为 NIS 服务器设置请求权限”所示。指定可以向 NIS 服务器发送来自哪个网络的请求。通常为内部网络。在当前情况下，应有以下两项：

```
255.0.0.0      127.0.0.0 0.0.0.0      0.0.0.0
```

第一项允许从您自己的主机（即 NIS 服务器）连接。第二项允许所有可访问同一网络的主机向服务器发送请求。

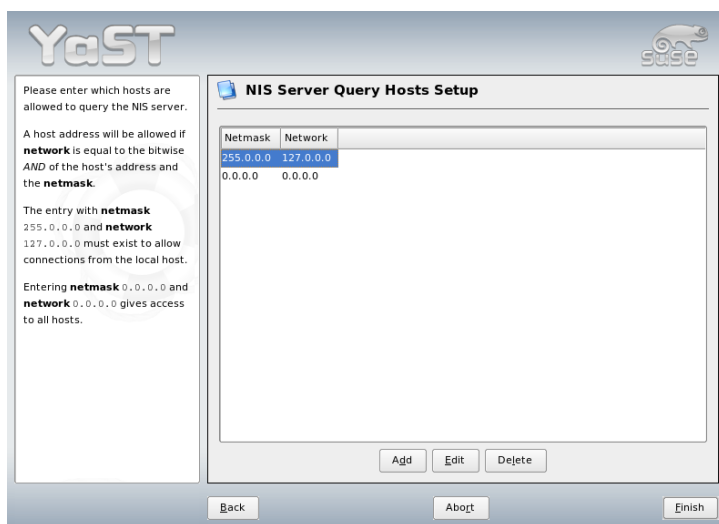


Figure 25.3: 为 NIS 服务器设置请求权限

## Important

### 自动配置防火墙

如果系统启用了防火墙 (SuSEfirewall2)，在选择‘打开防火墙中的端口’后，YaST 会通过启用 portmap 服务使防火墙的配置适应 NIS 服务器。

## Important

## 25.2 配置 NIS 客户程序

使用此模块配置 NIS 客户程序。选择使用 NIS 和 Automounter（视情况）后，将打开此对话框。选择主机使用固定 IP 地址还是接收 DHCP 分配的地址。DHCP 还提供 NIS 域和 NIS 服务器。有关 DHCP 的信息，请参见第 27 章 DHCP。如果使用静态 IP 地址，则手工指定 NIS 域和 NIS 服务器。请参见图 25.4 “设置 NIS 服务器的域和地址”。使用‘查找’可让 YaST 在网络中搜索活动的 NIS 服务器。

此外，可以指定多个域，其中一个为默认域。使用‘添加’可为各个域指定多个服务器（包括广播功能）。

在专家设置中，如果不希望其他主机能查询您的客户程序所用的服务器，请选中‘只应答本地主机’。通过选中‘断开的服务器’，客户程序将能够接收通过非特权端口通讯的服务器的答复。有关进一步信息，请参见 `man ypbind`。



Figure 25.4: 设置 NIS 服务器的域和地址





# 通过 NFS 共享文件系统

正如第25章使用NIS所述，NFS与NIS一起使网络在面向用户时是透明的。利用NFS可以在网络上分发文件系统。用户登录哪个终端并不重要。他们会发现自己总处在同一个环境中。

与NIS的情况相同，NFS也是一种非对称服务，有NFS服务器和NFS客户机之分。一台计算机可以充当这两种角色——它可以通过网络提供文件系统（导出），也可以从其他主机装入文件系统（导入）。这些计算机一般来说都是具有大容量硬盘的服务器，其他客户机要从该服务器装入文件系统。

26.1	使用 YaST 导入文件系统 . . . . .	392
26.2	手工导入文件系统 . . . . .	392
26.3	使用 YaST 导出文件系统 . . . . .	393
26.4	手工导出文件系统 . . . . .	394

## Important

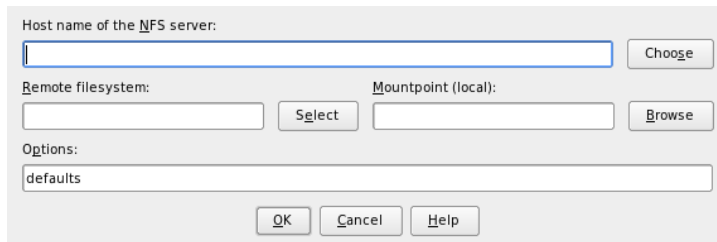
### 需要 DNS 的原因

从理论上讲，所有导出都可以仅使用 IP 地址来完成。但为避免超时，您应使用一个有效的 DNS 系统。至少为了日志记录目的也应使用此系统，因为 mountd 守护程序执行反向查找。

Important

## 26.1 使用 YaST 导入文件系统

经过授权的用户都可以将 NFS 服务器中的 NFS 目录装入自己的文件目录树。使用 YaST 的‘NFS 客户机’模块可以轻而易举地完成上述操作。只需输入 NFS 服务器的主机名、要导入的目录以及要在本地的哪个装入点装入此目录。单击第一个对话框中的‘添加’即可完成上述所有操作。请参见图 26.1 “使用 YaST 配置 NFS 客户机”。



The image shows a dialog box for configuring an NFS client. It has the following fields and buttons:

- Host name of the NFS server:** A text input field with a "Choose" button to its right.
- Remote filesystem:** A text input field with a "Select" button to its right.
- Mountpoint (local):** A text input field with a "Browse" button to its right.
- Options:** A text input field containing the text "defaults".
- At the bottom, there are three buttons: "OK", "Cancel", and "Help".

Figure 26.1: 使用 YaST 配置 NFS 客户机

## 26.2 手工导入文件系统

很容易从 NFS 服务器手工导入文件系统。唯一的前提条件是要运行 RPC 端口映射器，以 root 身份输入命令 `rpcportmap start` 即可启动该映射器。一旦满足此前提条件，就可以使用命令 `mount` 将导出到相应计算机上的远程文件系统装入文件系统（就像本地硬盘）。语法如下：

```
mount host:remote-path local-path
```

如果应该导入某台计算机（如 sun）上的用户目录，请使用以下命令：

```
mount sun:/home /home
```

## 26.3 使用 YaST 导出文件系统

使用 YaST 将网络中的某台主机转换为 NFS 服务器——即将目录和文件导出到所有有权访问它的主机的服务器。这样做可以为同一工作组中的所有同事提供应用程序，而不必在每台主机的本地都安装应用程序。要安装这样的服务器，请启动 YaST 并选择‘网络服务’→‘NFS 服务器’。打开一个如图 26.2 “NFS 服务器配置工具” 中所示的对话框。

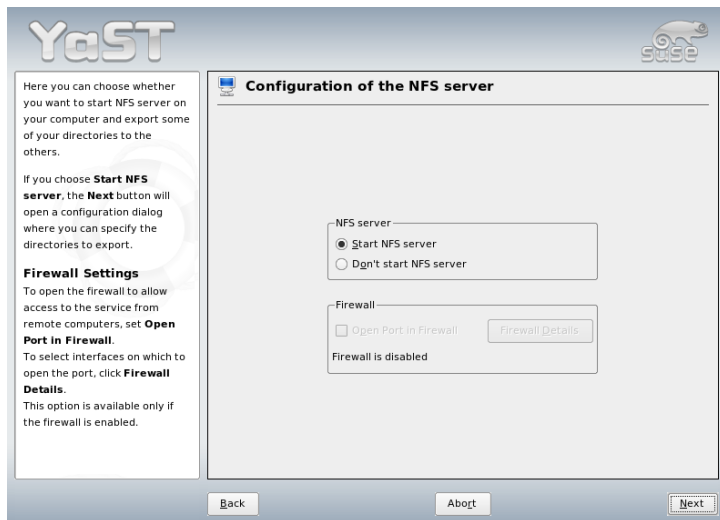


Figure 26.2: NFS 服务器配置工具

接下来，激活‘启动 NFS 服务器’，然后单击‘下一步’。在上面的文本字段中，输入要导出的目录。在下面输入应能访问这些目录的主机。图 26.3 “使用 YaST 配置 NFS 服务器” 中显示了此对话框。可以为每台主机设置四个选项：single host、netgroups、wildcards 和 IP networks。通过 man exports 可以提供这些选项的更为详尽的解释。选择‘退出’即完成配置。

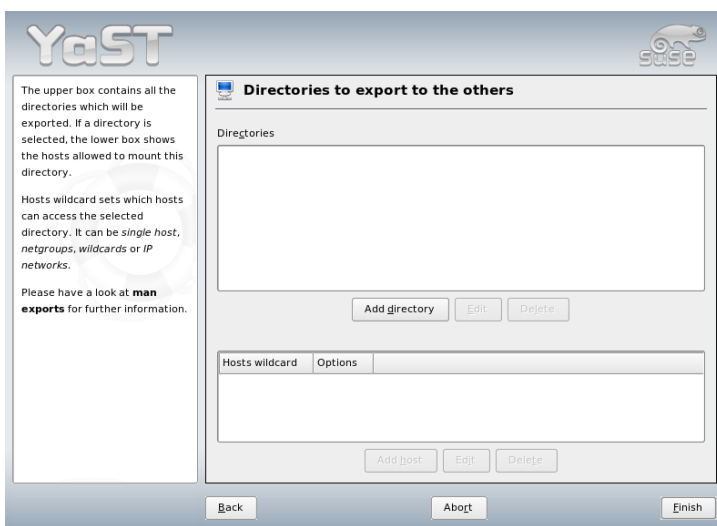


Figure 26.3: 使用 YaST 配置 NFS 服务器

### Important

#### 自动配置防火墙

如果系统启用了防火墙 (SuSEfirewall2)，在选择‘打开防火墙中的端口’后，YaST 会通过启用 `nfs` 服务使防火墙的配置适应 NFS 服务器。

Important

## 26.4 手工导出文件系统

如果不想使用 YaST，应确保以下系统运行在 NFS 服务器上：

- RPC 端口映射器 (`portmap`)
- RPC 装入守护程序 (`rpc.mountd`)
- RPC NFS 守护程序 (`rpc.nfsd`)

要在系统引导时使用脚本 `/etc/init.d/portmap` 和 `/etc/init.d/nfsserver` 启动上述服务，请输入命令 `insserv /etc/init.d/nfsserver` 和 `insserv /etc/init.d/portmap`。还需要在配置文件 `/etc/exports` 中定义应将哪些文件系统导出到哪台主机。

对于每个要导出的目录，需要有一行来设置哪些计算机可以使用什么权限来访问该目录。此目录的所有子目录也将自动导出。授权计算机通常以全名（包括域名）来指定，但也可以使用通配符，如 `*` 或 `?`（其扩展方式与 Bash shell 中的扩展方式相同）。如果在此未指定任何计算机，具有指定权限的所有计算机都可以导入此文件系统。

在计算机名称之后的括号内设置要导出文件系统的权限。表 26.1 “导出文件系统的权限” 中显示了最重要的选项。

Table 26.1: 导出文件系统的权限

选项	含义
<code>ro</code>	文件系统以只读权限导出（默认设置）。
<code>rw</code>	文件系统以读写权限导出。
<code>root_squash</code>	这样可以确保导入计算机的用户 <code>root</code> 对此文件系统不具备 <code>root</code> 权限。通过向用户 ID 为 0 ( <code>root</code> ) 的用户指派用户 ID 65534，可以实现这一点。应将此用户 ID 设置为 <code>nobody</code> （即默认设置）。
<code>no_root_squash</code>	不向用户 ID 65534 指派用户 ID 0，保持 <code>root</code> 权限有效。
<code>link_relative</code>	将绝对链接（以 <code>/</code> 开头）转换为一系列的 <code>../</code> 。此选项仅在装入计算机的整个文件系统时（默认设置）才有用。
<code>link_absolute</code>	保持符号链接不变。
<code>map_identity</code>	客户机和服务器上的用户 ID 完全相同（默认设置）。
<code>map_daemon</code>	客户机和服务器没有匹配的用户 ID。这会促使 <code>nfsd</code> 创建用户 ID 转换表。为此需要启动 <code>ugidd</code> 守护程序。

您的 `exports` 文件可能如例 26.1 “`/etc/exports`” 所示。`mountd` 和 `nfsd` 会读取 `/etc/exports`。如果对此文件作了任何改动，只有重新启动 `mountd` 和 `nfsd`

后，更改才能生效。使用 `rcnfsserver restart` 即可轻松地重启动。

### 26.1: */etc/exports*

```
# # /etc/exports # /home          sun(rw)   venus(rw) /usr/X11          sun(
```

# DHCP

*动态主机配置协议* (DHCP) 用于从服务器集中指派网络设置，而不必在每个工作stations本地逐一指派。被配置为使用 DHCP 的主机不能控制它自己的静态地址。DHCP 使它能够根据服务器的指示完全且自动地对自身进行配置。

27.1	使用 YaST 配置 DHCP 服务器	398
27.2	DHCP 软件包	399
27.3	DHCP 服务器 dhcpd	400
27.4	参考信息	404

使用 DHCP 的方法之一是使用网卡的硬件地址（它在大多数情况下是固定的）来标识每个客户机，然后在客户机每次连接到服务器时为其提供相同的设置。另一种方法是对 DHCP 进行配置，使服务器能够从为此目的而设置的地址池来为每个相关客户机动态指派地址。在后一种情况下，DHCP 服务器每次在收到客户机的请求时都会尝试向其指派相同的地址，即使相隔较长的时间也是如此。当然，只有当网络包含的客户机数目不大于地址数目时，DHCP 服务器才能这样做。

通过这些功能，DHCP 在两个方面简化了系统管理员的日常工作。与地址和网络配置相关的任何更改（甚至是较大的更改）一般都可以通过编辑服务器的配置文件来集中完成。这比重配置众多工作站要方便得多。此外，还可以更方便地将计算机（尤其是新计算机）集成到网络中，因为现在可以从池中为它们指派 IP 地址。如果经常在不同的网络中使用便携式计算机，则从 DHCP 服务器检索适当的网络设置会特别有用。

DHCP 服务器不仅提供了 IP 地址和网络掩码，还提供了客户机要使用的主机名、域名、网关和名称服务器地址。此外，DHCP 还允许您集中配置许多其他参数，例如客户机可能从中巡回检测当前时间的 NTP 时间服务器，甚至是打印服务器。

## 27.1 使用 YaST 配置 DHCP 服务器

首次启动此模块后，YaST 会启动一个由 4 部分组成的配置向导。您可以通过完成此向导来设置基本 DHCP 服务器。

**选择网络接口** 在第 1 步中，YaST 查找您的系统上可用的网络接口，然后将它们显示在列表中。从这个列表选择 DHCP 服务器要监听的接口，然后选择“打开所选接口的防火墙”来打开此接口的防火墙。请参见图 27.1 “DHCP 服务器：选择网络接口”。

**全局设置** 在输入字段中，提供 DHCP 服务器应管理的所有客户机的网络细节。这些细节包括域名、NTP 服务器的地址、主名称服务器和辅助名称服务器的地址、打印和 WINS 服务器的地址（在使用同时包含 Windows 和 Linux 客户机的混合网络的情况下）、网关地址和租用时间。请参见图 27.2 “DHCP 服务器：全局设置”。

**动态 DHCP** 在此步骤中，我们将配置如何为客户机指派动态 IP 地址。为此，应首先指定服务器为 DHCP 客户机指派地址时使用的 IP 范围。所有这些地址必须由同一个网络掩码来覆盖。还要指定租用时间，在此期间客户机可以保留它的 IP 地址，而无需请求续期。也可以选择指定最长租用时间——这是服务器为特定客户机保留某个 IP 地址的时间。请参见图 27.3 “DHCP 服务器：动态 DHCP”。



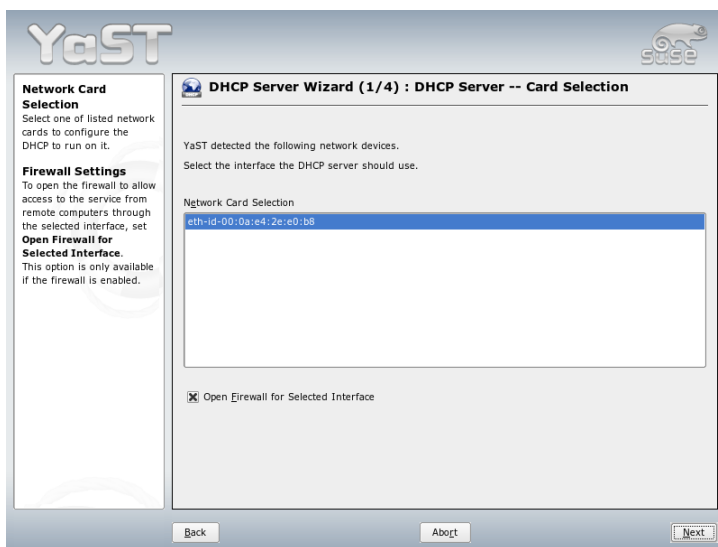


Figure 27.1: DHCP 服务器：选择网络接口

完成配置并设置启动方式 在完成配置向导的第 3 部分后，会出现最后一个对话框，用于定义应如何启动 DHCP 服务器。在这里确定是在引导系统时自动启动 DHCP 服务器，还是在需要时手工启动 DHCP 服务器（例如为了测试目的）。单击‘完成’完成对服务器的配置。请参见图 27.4 “DHCP 服务器：启动”。

## 27.2 DHCP 软件包

DHCP 服务器和 DHCP 客户机都可用于 SUSE LINUX。可用的 DHCP 服务器是 `dhcpcd`（由因特网软件联盟发布）。在客户端，在两个不同的 DHCP 客户程序中选择一个：`dhclient`（同样来自 ISC）和 `dhcpcd` 包中的 DHCP 客户机守护程序。

SUSE LINUX 在默认情况下会安装 `dhcpcd`。此程序非常易于处理，且在每次系统引导时会自动启动以监视 DHCP 服务器。它不需要配置文件即可工作，而且可以直接用在大多数标准设置中。对于更复杂的情况请使用 ISC `dhclient`，它是通过配置文件 `/etc/dhclient.conf` 来进行控制的。

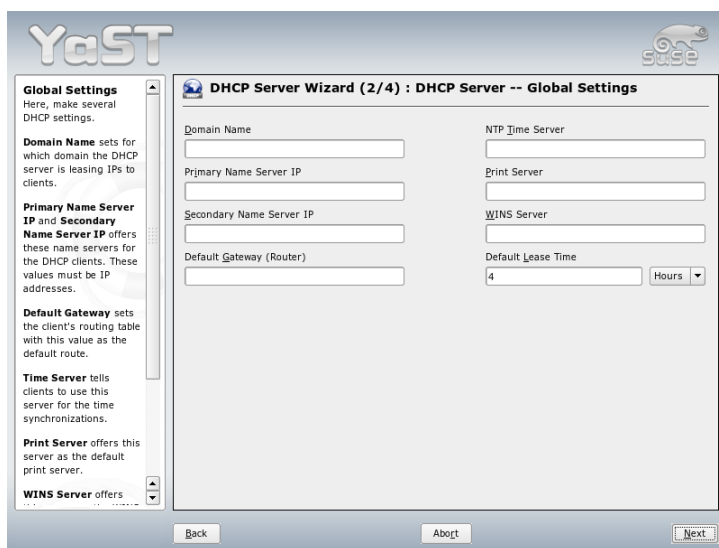


Figure 27.2: DHCP 服务器：全局设置

## 27.3 DHCP 服务器 dhcpd

任何 DHCP 系统的核心都是动态主机配置协议守护程序。根据配置文件 `/etc/dhcpd.conf` 中定义的设置，此服务器租出地址并监视它们的使用。通过更改此文件中的参数和值，系统管理员可以在许多方面影响程序的行为。让我们看一下例27.1“配置文件 `/etc/dhcpd.conf`”中的基本示例 `/etc/dhcpd.conf` 文件。

27.1: 配置文件 `/etc/dhcpd.conf`

```
default-lease-time 600;           # 10 minutes
max-lease-time 7200;             # 2 hours

option domain-name "cosmos.all";
option domain-name-servers 192.168.1.1, 192.168.1.2;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option subnet-mask 255.255.255.0;
```

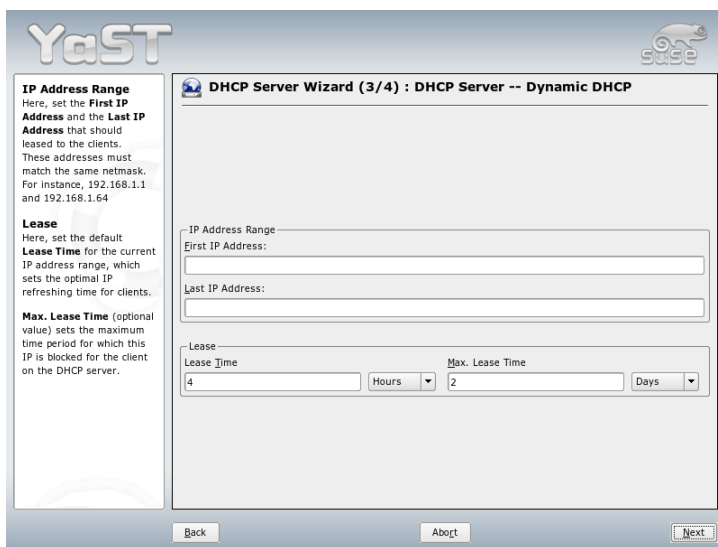


Figure 27.3: DHCP 服务器：动态 DHCP

```
subnet 192.168.1.0 netmask 255.255.255.0
{
    range 192.168.1.10 192.168.1.20;
    range 192.168.1.100 192.168.1.200;
}
```

这个简单的配置文件足以使 DHCP 服务器在网络中指派 IP 地址。确保在每行末尾插入一个分号，否则将不能启动 dhcpd。

上面的示例文件可以分为三部分。第一部分定义了将 IP 地址租出给请求它的客户机的默认秒数 (default-lease-time)，超过这个时间就应申请续期。此部分还包含一个最大期限语句，在此期限内计算机可以保留 DHCP 服务器指派的 IP 地址而无需申请续期 (max-lease-time)。

第二部分在全局级别上定义了一些基本网络参数：

- option domain-name 行用于定义网络的默认域。

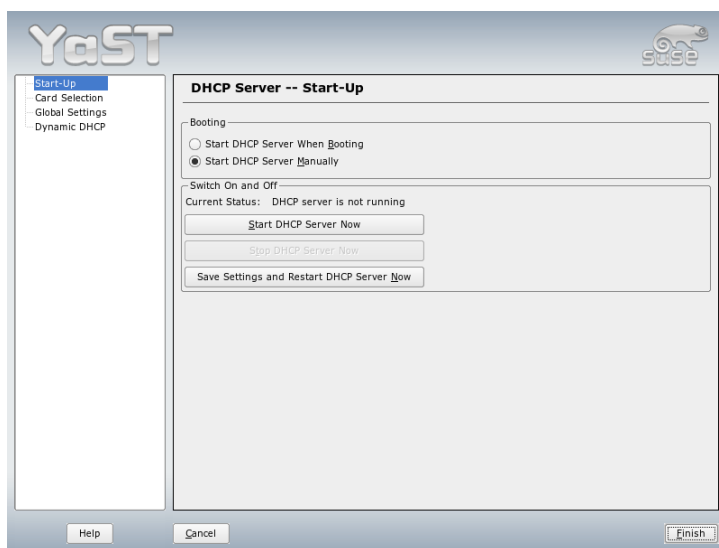


Figure 27.4: DHCP 服务器：启动

- `option domain-name-servers` 项用于指定 DNS 服务器将 IP 地址解析为主机名（反之亦然）时使用的值（最多 3 个）。理想情况下，应在设置 DHCP 之前在您的计算机上或网络中的其他位置配置一个名称服务器。这个名称服务器应为每个动态地址定义一个主机名（反之亦然）。要了解如何配置您自己的名称服务器，请参考第 24 章 域名系统。
- 行 `option broadcast-address` 定义了发出请求的客户机所使用的广播地址。
- `option routers` 用于指示服务器在无法将数据包发送到本地网络上的主机时应将其发送到的位置（根据所提供的源和目标主机地址以及子网掩码）。在大多数情况下，尤其是在较小的网络中，此路由器与因特网网关完全相同。
- `option subnet-mask` 用于指定为客户机指派的网络掩码。

文件的最后一部分用于定义网络，其中包含子网掩码。最后指定一个地址范围，DHCP 守护程序将使用此范围来向相关的客户机指派 IP 地址。在

此示例中，可以为客户机指派 192.168.1.10 和 192.168.1.20 之间以及 192.168.1.100 和 192.168.1.200 之间的任何地址。

在编辑这些行后，便可以使用命令 `rcdhcpd start` 来激活 DHCP 守护程序。随后将可以立即使用它。使用命令 `rcdhcpd check-syntax` 来执行简要的语法检查。如果配置出现意外问题 — 服务器由于错误而中止或在启动时不返回 `done` — 通过在主系统日志 `/var/log/messages` 或控制台 10 (`Ctrl-Alt-F10`) 上查找相关信息，就能够发现是什么出现了问题。

在默认 SUSE LINUX 系统上，由于安全原因，将在 `chroot` 环境中启动 DHCP 守护程序。必须将配置文件复制到 `chroot` 环境，以便守护程序能够找到它们。通常情况下无需担心这一点，因为命令 `rcdhcpd start` 会自动复制这些文件。

### 27.3.1 具有固定 IP 地址的客户机

如上所述，DHCP 也可以为每个请求而向特定客户机指派预定义的静态地址。显式指派的地址始终优先于来自地址池的动态地址。此外，静态地址永远不会像动态地址那样过期。例如，对于动态地址而言，如果没有足够的地址可用，服务器需要在客户机之间重新分发这些地址。

要标识配置有静态地址的客户机，`dhcpd` 将使用硬件地址，这是一个全局唯一的固定数字代码，其中包含 6 个八进制数对，用于标识所有网络设备（例如 `00:00:45:12:EE:F4`）。如果将相应的各行（如例 27.2 “配置文件的添加项” 中的行）添加到例 27.1 “配置文件 `/etc/dhcpd.conf`” 的配置文件，DHCP 守护程序在任何情况下都会将相同的一组数据指派到相应的客户机。

#### 27.2: 配置文件的添加项

```
host earth {
hardware ethernet 00:00:45:12:EE:F4;
fixed-address 192.168.1.21;
}
```

在第 1 行输入相应客户机的名称 (`host (hostname)`，在这里是 `earth`)，在第 2 行输入 MAC 地址。在 Linux 主机上，这个地址可通过使用命令 `ifstatus` 后跟网络设备（例如 `eth0`）来确定。如有必要，应首先使用 `ifup eth0` 激活网卡。输出应包含如下内容：

```
link/ether 00:00:45:12:EE:F4
```

在上面的示例中，网卡的 MAC 地址为 `00:00:45:12:EE:F4`，具有此网卡的客户机被自动指派了 IP 地址 `192.168.1.21` 和主机名 `earth`。虽然也支持在 IBM 系统上常见的 `token-ring`，但在几乎所有情况下，要输入的硬件类型都是 `ethernet`，

## 27.3.2 SUSE LINUX 版本

为了提高安全性，ISC 的 DHCP 服务器的 SUSE 版本附带有 Ari Edelkind 编写的非根/chroot 增补程序。这使得 dhcpd 能够使用用户 ID nobody 来运行，并可以在 chroot 环境 (/var/lib/dhcp) 中运行。要实现这一点，必须使配置文件 dhcpd.conf 位于 /var/lib/dhcp/etc 中。init 脚本在启动时会自动将文件复制到此目录。

通过文件 /etc/sysconfig/dhcpd 中的项来控制与此特性相关的服务器的行为。如果不希望在 chroot 环境中运行 dhcpd，请将 /etc/sysconfig/dhcpd 中的变量 DHCPD\_RUN\_CHROOTED 设置为“no”。

为了使 dhcpd 甚至能够解析来自 chroot 环境的主机名，还必须复制其他一些配置文件：

- /etc/localtime
- /etc/host.conf
- /etc/hosts
- /etc/resolv.conf

在启动 init 脚本时，将把这些文件复制到 /var/lib/dhcp/etc/。如果通过 /etc/ppp/ip-up 这样的脚本动态修改这些文件，则无论这些文件需要任何更改，都必须同时考虑这些副本。但是，如果配置文件仅指定 IP 地址（而不是主机名），就不需要担心这一点。

如果您的配置中包含应复制到 chroot 环境中的其他文件，请在文件 etc/sysconfig/dhcpd 中的变量 DHCPD\_CONF\_INCLUDE\_FILES 下指定它们。为了确保 DHCP 日志记录功能即使在重新启动 syslog 守护程序后也能继续工作，需要在 /etc/sysconfig/syslog 文件的 SYSLOGD\_PARAMS 下添加选项 "-a /var/lib/dhcp/dev/log"。

## 27.4 参考信息

有关 DHCP 的更多信息，请访问因特网软件联盟网站 (<http://www.isc.org/products/DHCP/>)。也可在 dhcpd、dhcpd.conf、dhcpd.leases 和 dhcp-options 的手册页中获得有关信息。

# 使用 xntp 同步时间

NTP（网络时间协议）是一种用于同步网络上的系统时间的协议。首先，计算机从作为可靠时间源的服务器获得时间。然后将此计算机用作网络中其他计算机的时间源。这样做有双重目的：既可维护绝对时间，又可保持网络中所有计算机系统时间的同步。

28.1	在网络中配置 xntp	406
28.2	设置本地参考时钟	406
28.3	使用 YaST 配置 NTP 客户机	407

维护确切的系统时间在许多情况下都非常重要。内置硬件 (BIOS) 时钟往往不能满足数据库这样的应用程序的要求。手工更正系统时间可能会导致许多严重问题, 例如向后调整时间将使关键应用程序出现故障。在一个网络内, 通常需要同步所有计算机的系统时间, 但手工调节时间是不可取的。xntp 提供了一种机制可解决这些问题。该机制随时借助网络中的可靠时间服务器调整系统时间。它还支持对本地参考时钟 (如无线电控制的时钟) 进行管理。

## 28.1 在网络中配置 xntp

xntp 已预先设置为以本地计算机时钟为时间参考。但是, 只有在没有更精确的时间源的情况下才使用 (BIOS) 时钟最为替代。要使用网络中的时间服务器, 最简便的方式就是设置服务器参数。例如, 如果网络中可以使用名为 `ntp.example.com` 的时间服务器, 请通过添加 `server ntp.example.com` 一行将其名称添加到文件 `/etc/ntp.conf` 中。要添加更多时间服务器, 请使用关键字 `server` 插入更多行。使用命令 `rcxntpd start` 初始化 `xntpd` 后, 等待时间稳定并且创建用于更正本地计算机时钟的偏移文件需要大约一个小时的时间。利用偏移文件, 只要计算机一启动, 就可以计算出硬件时钟的系统误差。可以立即使用更正功能, 使系统时间保持较高的稳定性。

有两种方法可将 NTP 机制用作客户机: 第一种方法是客户机可以定期从已知服务器查询时间。在存在许多客户机的情况下, 这种方法会给服务器带来很高的负荷。第二种方法是客户机可以等待网络中的广播时间服务器发送 NTP 广播。这种方法的缺点在于服务器的可靠性是未知的, 而且如果服务器发出错误信息将导致严重问题。

如果通过广播获取时间, 则不需要服务器名称。此时只需在配置文件 `/etc/ntp.conf` 中输入 `broadcastclient` 一行。要以独占方式使用一个或多个已知时间服务器, 请在以 `servers` 开头的行中输入它们的名称。

## 28.2 设置本地参考时钟

软件包 `xntp` 包含用于连接本地参考时钟的驱动程序。`xntp-doc` 软件包的文件 `/usr/share/doc/packages/xntp-doc/html/refclock.htm` 中提供了受支持时钟的列表。每个驱动程序都有一个关联数字。在 `xntp` 中, 实际配置是通过伪 IP 来完成的。时钟被输入 `/etc/ntp.conf` 文件, 就像已经在网络中存在一样。为此专门给它们指派了 `127.127.t.u` 格式的特殊 IP 地址。其中 `t` 代表时钟的类型并确定要使用的驱动程序, `u` 代表设备并确定要使用的接口。



通常，各个驱动程序都有特殊的参数来描述配置详细信息。文件 `/usr/share/doc/packages/xntp-doc/html/driverNN.htm`（其中 NN 是驱动程序的编号）提供了有关特定类型时钟的信息。例如，“8 型”时钟（通过串行接口的无线电时钟）需要额外的方式更精确地指定时钟。以 Conrad DCF77 接收模块为例，该模块需要使用 mode 5。要使用此时钟作为首选参考，应指定关键字 `prefer`。由此构成的 Conrad DCF77 接收模块的完整 `server` 行如下：

```
server 127.127.8.0 mode 5 prefer
```

其他时钟也采用相同的模式。安装 `xntp-doc` 包之后，可以在目录 `/usr/share/doc/packages/xntp-doc/html` 中找到 `xntp` 文档。文件 `/usr/share/doc/packages/xntp-doc/html/refclock.htm` 提供指向描述驱动程序参数的驱动程序页的链接。

## 28.3 使用 YaST 配置 NTP 客户机

除手工配置 `xntp` 之外，SUSE LINUX 还支持通过 YaST 便捷地配置 NTP 客户机。可使用快速配置或复杂配置。二者的说明见下文。

### 28.3.1 快速的 NTP 客户机配置

简单的 NTP 客户机配置包括两个对话框。在第一个对话框中设置 `xntpd` 的启动方式和要查询的服务器。要在引导系统时自动启动 `xntpd`，请单击‘引导系统时’。然后单击‘选择’，访问第二个对话框，从中为您的网络选择适合的时间服务器。

在详细的服务器选择对话框中，确定是使用本地网络中的时间服务器实施时间同步，还是使用考虑到所在时区的基于因特网的时间服务器（‘公共 NTP 服务器’）实施时间同步。要使用本地时间服务器，请单击‘查找’启动 SLP 查询，查找网络中的可用时间服务器。从搜索结果列表中选择最适合的时间服务器，然后单击‘确定’退出该对话框。要使用公共时间服务器，请选择您所在的国家/地区（时区），并从‘公共 NTP 服务器’列表中选择适合的服务器，然后单击‘确定’退出该对话框。在主对话框中，可单击‘测试’测试选定服务器的可用性，单击‘完成’退出该对话框。

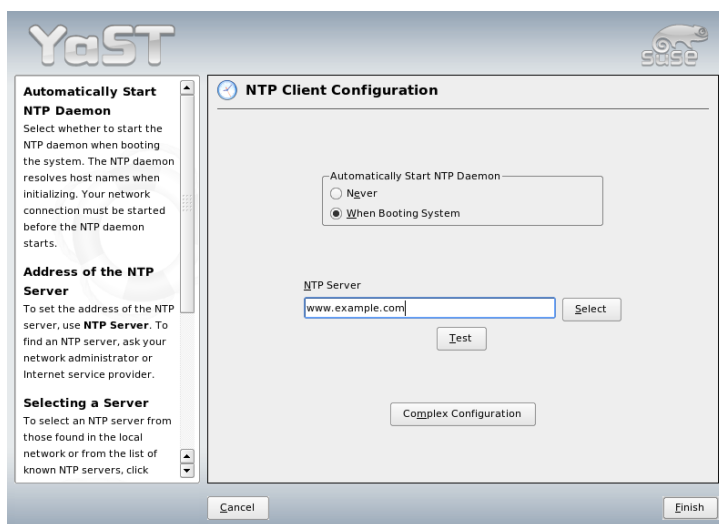


Figure 28.1: YaST: 配置 NTP 客户机

### 28.3.2 复杂的 NTP 客户机配置

选择快速配置中所述的启动方式之后，可以通过选择‘NTP 客户机’模块的主对话框中的‘复杂配置’对 NTP 客户机进行复杂配置（如图 28.1 “YaST: 配置 NTP 客户机”所示）。

在‘复杂的 NTP 客户机配置’中，确定是否应在 chroot jail 中启动 xntpd。在 chroot jail 中启动可以在遭受通过 xntpd 发起的攻击时提高安全性，因为这种方式可以防止攻击者危害整个系统。选择‘通过 DHCP 配置 NTP 守护程序’，可以设置 NTP 客户机通过 DHCP 获取网络中可用的 NTP 服务器列表。

供客户机查询的服务器和其他时间资源列在下半部分。使用‘添加’、‘编辑’和‘删除’可按需修改此列表。使用‘高级’可以查看客户机的日志文件，或调节 NTP 客户机配置的防火墙。

单击‘添加’可添加新的时间信息源。在随后的对话框中，选择要与其进行时间同步的源类型。有以下选项可用：

**服务器** 可供您选择 NTP 服务器的另一个对话框（如第 28.3.1 节“快速的 NTP 客户机配置”所述）。激活‘用于初始同步’，可以在引导系统时触发服务

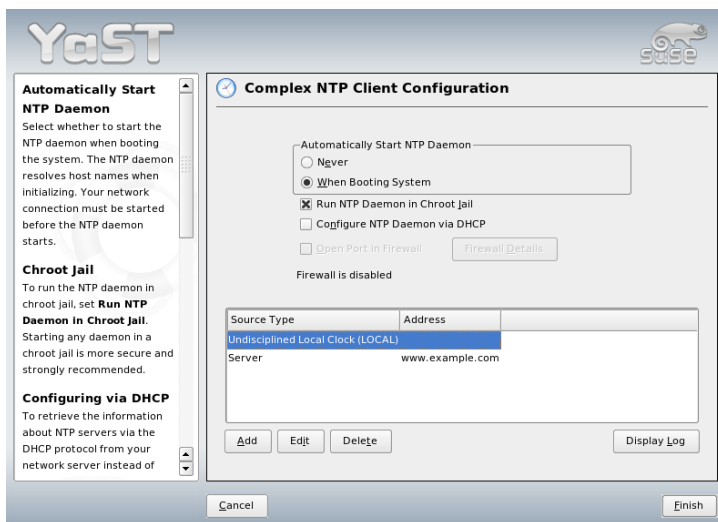


Figure 28.2: YaST: 复杂的NTP客户机配置

器和客户机之间的时间信息同步。有一个输入字段可用于为 xntpd 指定更多选项。有关详细信息，请参考 `/usr/share/doc/packages/xntp-doc`。

**同级** 同级是一台要与其建立对称关系的计算机：它将同时用作时间服务器和客户机。要在同一网络中用同级代替某个服务器，请输入相应系统的地址。该对话框的其他部分与‘服务器’对话框相同。

**无线电时钟** 要在系统中使用无线电时钟来同步时间，请在此对话框中输入时钟类型、单元号码、设备名和其他选项。单击‘驱动程序校准’可对该驱动程序进行微调。有关本地无线电时钟如何操作的详细信息，请参见 `/usr/share/doc/packages/xntp-doc/html/refclock.htm`。

**广播** 也可以通过在网络内广播的方式来传送时间信息和查询。在此对话框中，输入应将这类广播信息发送到的地址。除非使用了像无线电控制的时钟这样的可靠时间源，否则不要激活广播。

**接受广播包** 如果希望客户机通过广播接收信息，请在此字段中输入应接受来自哪个地址的相应数据包。



# LDAP — 目录服务

轻量级目录访问协议 (LDAP) 是一组设计用来访问和维护信息目录的协议。LDAP 可用于多种目的，如用户和组管理、系统配置管理或地址管理。本章简要介绍 LDAP 的工作原理以及如何使用 YaST 管理 LDAP 数据。

29.1	对比 LDAP 和 NIS . . . . .	413
29.2	LDAP 目录树的结构 . . . . .	413
29.3	使用 slapd.conf 配置服务器 . . . . .	416
29.4	LDAP 目录中的数据处理 . . . . .	420
29.5	YaST LDAP 客户程序 . . . . .	424
29.6	参考信息 . . . . .	429

在联网环境中保持重要信息组织有序并且访问便捷是非常重要的。这可以通过目录服务实现。目录服务就像常见的电话黄页，可以将信息组织得井然有序，便于快速搜索。

理想情况下，应该有一个中央服务器将数据组织到目录中，并使用特定协议将其分发给所有客户机。数据以特定的方式组织，以支持众多应用程序进行访问。这样，各种日历工具和电子邮件客户程序就不必保持自己的数据库——只需访问中央储存库即可。这种方式极大地减轻了管理这些信息的工作。利用 LDAP 之类的开放且标准化的协议，可以保证让尽量多的客户应用程序都能访问这些信息。这里所说的目录实际上是指一种为快速有效的读取和搜索而优化的数据库。

- 为支持尽量多的（并行）读取访问，需要限制写访问，只允许管理员执行次数较少的更新。要对常规数据库进行优化，使其能够在短时间内接受尽量多的数据。
- 由于只能在受限模式下执行写访问，所以可以采用目录服务来管理几乎不变的静态信息。常规数据库中的数据通常频繁变化（动态数据），而公司名录中的电话号码并不像会计数字（举例来说）那样经常变化。
- 管理静态数据时，极少更新现有数据集。而处理动态数据时，特别是在涉及像银行帐户或会计帐户这样的数据集时，数据的一致性举足轻重。如果要将从某处减去的数目加到另一个位置，必须在一个事务中同时执行这两个运算，以确保数据存量保持平衡。数据库支持处理这类事务，而目录却不行。短期内数据之间的不一致在目录中是完全可以接受的。

LDAP 这类目录服务并不是为支持复杂的更新或查询机制而设计的。访问此服务的所有应用程序都应能够便捷地获取访问权。

许多目录服务以前就存在，而且现在仍然存在于 Unix 和其他操作系统中。Novell NDS、Microsoft ADS、Banyan 的 Street Talk 及 OSI 标准 X.500 只是其中的几个例子。LDAP 最初是作为轻量级的 DAP（目录访问协议）设计的，后者是为访问 X.500 而开发的。X.500 标准用来管理目录项的层次组织。

LDAP 是 DAP 的精简版。在保留 X.500 项层次的同时，LDAP 新增了跨平台功能并能够节省资源。利用 TCP/IP，在对接应用程序和 LDAP 服务之间建立接口的工作得到大大简化。

与此同时，LDAP 还不断发展，已经越来越多地作为不带 X.500 支持的独立解决方案应用。LDAP 在 LDAPv3（软件包 `openldap2` 中的协议版本）中支持参照，从而能够实现分布式数据库。同时还新增了 SASL（简单鉴定和安全层）。LDAP 不仅能按最初设计的那样查询 X.500 服务器中的数据，它还提供开源代码服务器 `slapd`，该服务器可以将对象信息储存在本地数据库中。另外还提供名为 `slurpd` 的扩展程序，该程序负责复制多个 LDAP 服务器。

`openldap2` 软件包包含：

**slapd** 独立 LDAPv3 服务器，用于管理基于 BerkeleyDB 的数据库中的对象信息。

**slurpd** 使用此程序，可以将针对本地 LDAP 服务器数据的修改复制到网络上安装的其他 LDAP 服务器中。

用于系统维护的其他工具 `slapcat`、`slapadd`、`slapindex`

## 29.1 对比 LDAP 和 NIS

Unix 系统管理员以往使用 NIS 服务在网络内进行名称解析和数据分发。`/etc` 中的文件所包含的配置数据以及目录 `group`、`hosts`、`mail`、`netgroup`、`networks`、`passwd`、`printcap`、`protocols`、`rpc` 和 `services` 都通过客户程序在网络中分发。这些文件很容易维护，因为它们都是简单的文本文件。但随着数据量的不断增大，处理起来就会因为缺乏组织结构而愈发困难。NIS 专为 Unix 平台设计，因此无法在异构网络中让其充当中央数据管理器。

有别于 NIS，LDAP 服务不仅仅适用于单纯的 Unix 网络。Windows 2000 之后的服务器都支持 LDAP 作为目录服务。Novell 也提供 LDAP 服务。上述应用程序任务在非 Unix 系统中同样受支持。

LDAP 原理适用于所有需要集中管理的数据结构。以下是一些应用示例：

- 用于替代 NIS 服务
- 邮件路由选择 (`postfix`、`sendmail`)
- 邮件客户程序 (如 `Mozilla`、`Evolution` 和 `Outlook`) 的通讯录
- 为 BIND9 名称服务器管理区域说明

可以扩展此列表，因为 LDAP 是可扩展的，这是 NIS 所不能及的。由于更便于搜索数据，明确定义的数据层次结构简化了对大量数据的管理。

## 29.2 LDAP 目录树的结构

LDAP 目录具有树形结构。目录中的所有项 (称为对象) 在此层次结构中都有确定的位置。此层次结构称为 *目录信息树*，或简称 *DIT*。指向所需项的完整路径 (可以明确标识该项) 被称为 *判别名* 或 *DN*。沿路径指向此项的各级节点称为 *相对判别名* 或 *RDN*。通常可以向以下两种可能类型之一指派对象：

**树枝** 这些对象本身可以包含其他对象。这些对象类包括 `root`（目录树的根元素，实际并不存在）、`c`（国家/地区）、`ou`（组织单元）和 `dc`（域组件）。此模型类似文件系统中的目录（文件夹）。

**叶** 这些对象位于分支的末梢，没有任何从属对象。`person`、`InetOrgPerson` 或 `groupofNames` 都属于此类对象。

目录层次的顶端有一个根元素 `root`。其中可包含 `c`（国家/地区）、`dc`（域组件）或 `o`（组织）作为从属元素。LDAP 目录树中的关系在下例中尤为明显，如图 29.1 “LDAP 目录的结构” 所示。

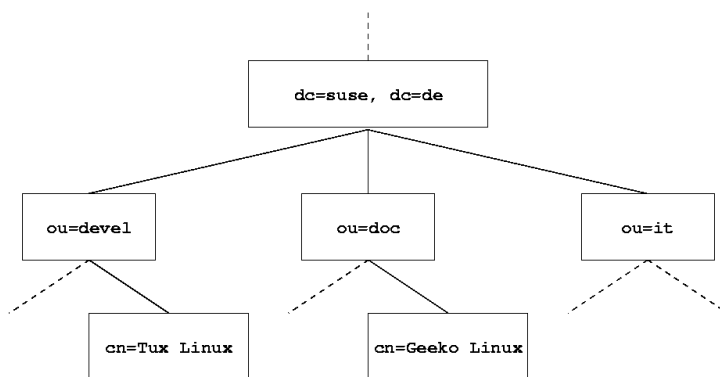


Figure 29.1: LDAP 目录的结构

完整的图中包含虚构的目录信息树。其中描述了三个层次上的项。每一项都对应图中的一个框。在本例中，虚构的 SUSE 雇员 `Geeko Linux` 的完整有效判别名为 `cn=Geeko Linux,ou=doc,dc=suse,dc=de`。该名称是通过将 RDN `cn=Geeko Linux` 添加到前一项的 DN `ou=doc,dc=suse,dc=de` 而构成的。

要从总体上决定应将哪类对象储存在 DIT 中，需要遵照方案来执行。对象类型由对象类决定。对象类决定必须或可以给相关对象指派哪些特性。因此，方案中必须包含所需应用方案中使用的所有对象类和特性的定义。已存在一些常用方案（请参见 RFC 2252 和 2256）。不过，可以创建自定义方案或使用多个互补的方案（如果 LDAP 服务器的运行环境要求这样做）。

表 29.1 “常用对象类和特性” 提供了示例所用的 `core.schema` 和 `inetorgperson.schema` 中的对象类的简要概览（包括必需特性和有效特性值）。



Table 29.1: 常用对象类和特性

对象类	含义	示例项	必需特性
dcObject	<i>domainComponent</i> (域的名称组件)	use	dc
organizationalUnit	<i>organizationalUnit</i> (组织单元)	oc	ou
inetOrgPerson	<i>inetOrgPerson</i> (内部网或因特网中与个人有关的数据)	Geeko Linux	sn 和 cn

例29.1 “schema.core 节选 (为便于解释对行进行了编号)” 节选自一个方案指令，并附有解释。

#### 29.1: schema.core 节选 (为便于解释对行进行了编号)

```
#1 attributetype (2.5.4.11 NAME ( 'ou' 'organizationalUnitName' )
#2     DESC 'RFC2256: organizational unit this object belongs to'
#3     SUP name )
...
#4 objectclass ( 2.5.6.5 NAME 'organizationalUnit'
#5     DESC 'RFC2256: an organizational unit'
#6     SUP top STRUCTURAL
#7     MUST ou
#8 MAY (userPassword $ searchGuide $ seeAlso $ businessCategory
    $ x121Address $ registeredAddress $ destinationIndicator
    $ preferredDeliveryMethod $ telexNumber
    $ teletexTerminalIdentifier $ telephoneNumber
    $ internationalISDNNumber $ facsimileTelephoneNumber
    $ street $ postOfficeBox $ postalCode $ postalAddress
    $ physicalDeliveryOfficeName
    $ st $ l $ description) )
...
```

特性类型 *organizationalUnitName* 和相应的对象类 *organizationalUnit* 在此仅作为示例。第 1 行说明特性的名称、其唯一 OID (对象标识符) (数字)，及特性缩写方式。

第 2 行通过 DESC 对该特性进行了简要说明。在此还提到了定义所基于的相应 RFC。第 3 行中的 SUP 表明此特性所属的上级特性类型。

从第 4 行开始是对象类 *organizationalUnit* 的定义；与特性定义类似，该定义也包含对象类的 OID 和名称。第 5 行对该对象类进行了简要说明。第 6 行通

过项 SUP top 表明此对象类不从属于其他对象类。第 7 行以 MUST 开头，列出必须与类型为 organizationalUnit 的对象一同使用的所有特性类型。第 8 行以 MAY 开头，列出可以与此对象类一同使用的所有特性类型。

有关方案用法的详尽介绍，请参见文档 OpenLDAP。安装 OpenLDAP 之后，可以在 /usr/share/doc/packages/openldap2/admin-guide/index.html 中找到该文档。

## 29.3 使用 slapd.conf 配置服务器

已安装系统的 /etc/openldap/slapd.conf 中包含 LDAP 服务器的完整配置文件。在此简述了其中的各个项并说明了必要的调整。以符号 (#) 为前缀的项处于非活动状态。必须取消这个注释字符才能激活这些项。

### 29.3.1 slapd.conf 中的全局指令

#### 29.2: slapd.conf: 用于方案的 Include 指令

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/inetorgperson.schema
```

这是 slapd.conf 中的第一个指令（如例 29.2 “slapd.conf: 用于方案的 Include 指令” 所示），用于指定组织 LDAP 目录所依据的方案。core.schema 是必需项。其他所需方案都追加到此指令中（在本例中添加了 inetorgperson.schema）。目录 /etc/openldap/schema 中提供了更多可用方案。要用类似的 LDAP 服务替换 NIS，应包含 rfc2307.schema 和 cosine.schema 这两个方案。相关信息可参见随附的 OpenLDAP 文档。

#### 29.3: slapd.conf: pidfile 和 argsfile

```
pidfile /var/run/slapd/slapd.pid
argsfile /var/run/slapd/slapd.args
```

这两个文件包含启动 slapd 进程所用的 PID（进程 ID）和一些参数。在此没必要进行修改。

## 29.4: *slapd.conf*: 访问控制

```
# Sample Access Control
#     Allow read access of root DSE
# Allow self write access
#     Allow authenticated users read access
#     Allow anonymous users to authenticate
# access to dn="" by * read
#     access to * by self write
#         by users read
#         by anonymous auth
#
# if no access controls are present, the default is:
#     Allow read by all
#
# rootdn can always write!
```

例29.4 “*slapd.conf*: 访问控制”节选自 *slapd.conf*，涉及如何管理服务器上的 LDAP 目录的访问权限。只要在数据库特定部分没有声明任何自定义访问规则，*slapd.conf* 的全局部分中的设置将始终有效。这些自定义声明会覆盖全局声明。如本例所示，所有用户都可以读取目录，但只有管理员 (*rootdn*) 才能写入此目录。LDAP 中的访问控制管理是一个非常复杂的过程。以下提示会有所帮助：

- 每条访问规则都具有如下结构：

```
access to <what> by <who> <access>
```

- *<what>* 是一个占位符，表示授权访问的对象或特性。可以使用单独的规则来明确保护各个目录分支。还可以使用正则表达式通过一条规则处理目录树的各个区域。*slapd* 按照各条规则列在配置文件中的先后顺序依次评估它们。较抽象的规则应列在较具体的规则之后 — 在评估 *slapd* 认为有效的第一条规则之后，随后的所有项都将被忽略。
- *<who>* 确定应该授权谁来访问 *<what>* 确定的区域。可以使用正则表达式。*slapd* 同样会在评估第一条有效规则之后中止对随后的 *who* 的评估，所以应将较具体的规则列在较抽象的规则之前。可以使用表29.2 “用户组及其访问授权”所示的项。

Table 29.2: 用户组及其访问授权

标签	作用域
*	所有用户，无一例外
anonymous	未鉴定（“匿名”）用户
users	已鉴定用户
self	与目标对象连接的用户
dn.regex=<regex>	与正则表达式匹配的所有用户

- `<access>` 指定访问类型。请使用表29.3“访问类型”所列的选项。

Table 29.3: 访问类型

标签	访问范围
none	无访问权
auth	用于联系服务器
compare	授予要进行比较访问的对象
search	用于应用搜索过滤器
read	读权限
write	写权限

slapd 会将客户机请求的访问权限与 `slapd.conf` 中授予的权限进行对比。如果规则允许的权限比请求的权限高或是相同，就可以授予客户机权限。如果客户机请求的权限高于规则中声明的权限，便会拒绝授予权限。

例29.5 “slapd.conf: 访问控制示例”显示了一个简单示例，使用正则表达式可以随意指定这样的简单访问控制。

### 29.5: slapd.conf: 访问控制示例

```
access to dn.regex="ou=([^\,]+),dc=suse,dc=de"  
by dn.regex="cn=administrator,ou=$1,dc=suse,dc=de" write  
by user read  
by * none
```

此规则声明只有各个 ou 项的管理员才有权写入他/她所管理的项。其他所有通过鉴定的用户只有读权限，其余人没有任何权限。

### Tip

#### 建立访问规则

如果没有 `access to` 规则或匹配的 `by` 指令，则拒绝访问。只有经过显式声明才能授予访问权限。如果根本没有声明任何规则，默认规则是管理员具有写权限，其他所有用户都具有读权限。

### Tip

有关 LDAP 访问权限的详细信息和示例配置，请参见所安装的 `openldap2` 软件包中的联机文档。

除了可以使用中央服务器配置文件 (`slapd.conf`) 管理访问权限之外，还可以使用 ACI - 访问控制信息。ACI 允许储存 LDAP 树中各个对象的访问信息。这种访问控制尚未普及，开发人员认为它目前仍处于试用阶段。相关信息请参考 <http://www.openldap.org/faq/data/cache/758.html>。

## 29.3.2 slapd.conf 中的数据库特定指令

### 29.6: slapd.conf: 数据库特定指令

```
database ldbm
suffix "dc=suse,dc=de"
rootdn "cn=admin,dc=suse,dc=de"
# Cleartext passwords, especially for the rootdn, should
# be avoided. See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw secret
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd/tools. Mode 700 recommended.
directory /var/lib/ldap
# Indices to maintain
index objectClass eq
```

数据库的类型（本例中为 LDBM）是在上述代码段的第一行中确定的（请参见例 29.6 “`slapd.conf`: 数据库特定指令”）。第二行通过 `suffix` 确定此服务器应该负责 LDAP 目录树的哪个部分。接下来的 `rootdn` 确定谁对此服务器拥有管理员权限。在此声明的用户不必拥有 LDAP 项，也不必是普通用户。管理员口令用 `rootpw` 来设置。在此不必使用 `secret`，可以输入 `slappasswd` 创建的

管理员口令的哈希值。`directory` 指令表明数据库目录储存在服务器的哪个目录中（在文件系统中）。最后一个指令 `index objectClass eq` 指示对所有对象类的索引进行维护。可以在此根据经验添加用户最常搜索的特性。在此为数据库自定义的 `Access` 规则将取代全局 `Access` 规则。

### 29.3.3 启动和停止服务器

在根据第29.4节“LDAP 目录中的数据处理”所述的模式对 LDAP 服务器进行全面配置并且设置全部所需项之后，可通过输入 `rcldap start` 以 `root` 身份启动 LDAP 服务器。要手工停止服务器，请输入命令 `rcldap stop`。使用 `rcldap status` 可请求运行中的 LDAP 服务器的状态。

可以使用第7.6节“Systems Services (RunLevel)”所述的 YaST 运行级别编辑器，在引导和暂停系统时相应地自动启动和停止服务器。还可以如第7.5.1节“添加 `init` 脚本”所述，在命令提示中使用 `insserv` 命令创建指向启动脚本和停止脚本的相应链接。

## 29.4 LDAP 目录中的数据处理

OpenLDAP 提供一系列工具，可用于在 LDAP 目录中管理数据。下文简要说明了四种最重要的工具，分别对储存的数据执行添加、删除、搜索和修改等操作。

### 29.4.1 将数据插入 LDAP 目录

在 `/etc/openldap/slapd.conf` 中正确配置 LDAP 服务器并且准备就绪后（即已配置好 `suffix`、`directory`、`rootdn`、`rootpw` 和 `index` 这些项之后），接下来可以输入记录。OpenLDAP 为此任务提供了 `ldapadd` 命令。如果可行，最好以捆绑方式向数据库添加对象，这是较为实用的做法。LDAP 能够为此处理 LDIF 格式（LDAP 数据交换格式）。LDIF 文件是一个简单的文本文件，可以包含任意数量的特性和值对。请参考 `slapd.conf` 中声明的方案文件，了解可用的对象类和特性。为图29.1“LDAP 目录的结构”中的示例创建大致框架的 LDIF 文件将如例29.7“LDIF 文件示例”所示：

#### 29.7: LDIF 文件示例

```
# The SUSE Organization
dn: dc=suse,dc=de
```

```
objectClass: dcObject
objectClass: organization
o: SUSE AG dc: suse

# The organizational unit development (devel)
dn: ou=devel,dc=suse,dc=de
objectClass: organizationalUnit
ou: devel

# The organizational unit documentation (doc)
dn: ou=doc,dc=suse,dc=de
objectClass: organizationalUnit
ou: doc

# The organizational unit internal IT (it)
dn: ou=it,dc=suse,dc=de
objectClass: organizationalUnit
ou: it
```

## Important

### LDIF 文件的编码

LDAP 采用 UTF-8 (Unicode) 编码。必须对元音变音符正确编码。请使用支持 UTF-8 的编辑器（如 Kate 或 Emacs 的最近版本）。要不然就避免使用元音变音符和其他特殊字符，或使用 `recode` 将输入内容重新编码为 UTF-8。

## Important

以 `.ldif` 为后缀保存该文件，然后使用以下命令将其传递给服务器：

```
ldapadd -x -D <dn of the administrator> -W -f <file>.ldif
```

`-x` 在本例中用于关闭通过 SASL 鉴定。`-D` 用于声明调用操作的用户。在此输入管理员的有效 DN，就如同已经在 `slapd.conf` 中配置。在当前示例中为 `cn=admin,dc=suse,dc=de`。`-W` 用于避免在命令行中输入口令（以明文形式），改为激活单独的口令提示。此口令此前已在 `slapd.conf` 中通过 `rootpw` 确定。`-f` 用于传递文件名。请参见例 29.8 “`example.ldif` 中的 `ldapadd`” 中关于运行 `ldapadd` 的详细信息。

*29.8: example.ldif 中的 ldapadd*

```
ldapadd -x -D cn=admin,dc=suse,dc=de -W -f example.ldif
```

```
Enter LDAP password:  
adding new entry "dc=suse,dc=de"  
adding new entry "ou=devel,dc=suse,dc=de"  
adding new entry "ou=doc,dc=suse,dc=de"  
adding new entry "ou=it,dc=suse,dc=de"
```

个人的用户数据可以储存在单独的 LDIF 文件中。例29.9 “Tux 的 LDIF 数据”向新的 LDAP 目录添加了 Tux。

### 29.9: Tux 的 LDIF 数据

```
# coworker Tux  
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de  
changetype: modify  
replace: telephoneNumber  
telephoneNumber: +49 1234 567-10
```

LDIF 文件可以包含任意数目的对象。一次可以将整个目录分支都传递到服务器中，也可以仅传递部分目录，如单个对象的示例所示。如果需要对某些数据进行较为频繁的修改，建议对单个对象进行细分。

## 29.4.2 修改 LDAP 目录中的数据

使用 `ldapmodify` 工具可以修改储存数据。最简单的方法是修改相应的 LDIF 文件，然后将这个修改过的文件传递给 LDAP 服务器。若要将同事 Tux 的电话号码从 +49 1234 567-8 改为 +49 1234 567-10，请像在例29.10 “修改过的 LDIF 文件 tux.ldif” 中那样编辑 LDIF 文件。

### 29.10: 修改过的 LDIF 文件 tux.ldif

```
# coworker Tux  
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de  
changetype: modify  
replace: telephoneNumber  
telephoneNumber: +49 1234 567-10
```

使用以下命令将修改过的文件导入 LDAP 目录：



```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W -f tux.ldif
```

或者，直接将要更改的特性传递给 `ldapmodify`。该过程如下所述：

1. 启动 `ldapmodify` 并输入您的口令：

```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W Enter LDAP password:
```

2. 输入更改，同时要小心地采用如下所示的语法顺序：

```
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

有关 `ldapmodify` 及其语法的详细信息，请参见相应的手册页 (`ldapmodify(1)`)。

### 29.4.3 搜索或读取 LDAP 目录中的数据

OpenLDAP 提供 `ldapssearch` 命令行工具，用于搜索 LDAP 目录中的数据并从中读取数据。简单查询将采用以下语法：

```
ldapdelete -x -D cn=admin,dc=suse,dc=de -W cn=Tux \
Linux,ou=devel,dc=suse,dc=de
```

选项 `-b` 用于确定搜索基础——要对目录树的哪个部分执行搜索。在当前情况下为 `dc=suse,dc=de`。要在 LDAP 目录的某个子部分中执行更为精确的搜索（例如仅在 `devel` 部门内搜索），请使用 `-b` 将此部分传递给 `ldapssearch`。`-x` 用于请求激活简单鉴定。`(objectClass=*)` 用于声明应该读取目录包含的所有对象。可以在创建新目录树后使用此命令选项，用来校验是否正确记录所有项并且服务器是否能按预期响应。有关 `ldapssearch` 用法的详细信息，请参见相应的手册页 (`ldapssearch(1)`)。

### 29.4.4 删除 LDAP 目录中的数据

使用 `ldapdelete` 可以删除不需要的项。该命令的语法与上述命令的语法相似。例如，要彻底删除 `Tux Linux` 项，请发出以下命令：

```
ldapdelete -x -D cn=admin,dc=suse,dc=de -W cn=Tux \
Linux,ou=devel,dc=suse,dc=de
```

## 29.5 YaST LDAP 客户程序

YaST 包含一个用于设置基于 LDAP 的用户管理的模块。如果安装期间未启用此功能，请选择‘网络服务’→‘LDAP 客户程序’启动该模块。YaST 将自动启用 LDAP 要求的所有与 PAM 和 NSS 相关的更改（如下所述）并安装必需的文件。

### 29.5.1 标准过程

必须清楚在客户机后台运行的各个进程才能了解 YaST LDAP 客户程序模块的工作方式。如果为进行网络鉴定激活了 LDAP 或是调用了 YaST 模块，系统会安装软件包 `pam_ldap` 和 `nss_ldap` 并调整两个相应的配置文件。`pam_ldap` 是负责在登录进程和 LDAP 目录（作为鉴定数据源）之间进行协商的 PAM 模块。将安装专用模块 `pam_ldap.so` 并调整 PAM 配置（请参见例 29.11 “为适应 LDAP 而调整的 `pam_unix2.conf`”）。

#### *29.11: 为适应 LDAP 而调整的 `pam_unix2.conf`*

```
auth:use_ldap nullok account:use_ldap password:use_ldap nullok session:无
```

手工配置其他服务使用 LDAP 时，请将该服务对应的 PAM 配置文件中的 PAM LDAP 模块加入 `/etc/pam.d`。可以在 `/usr/share/doc/packages/pam_ldap/pam.d/` 中找到为适应各种服务而调整过的配置文件。将相应文件复制到 `/etc/pam.d` 中。

使用 `nss_ldap` 可以对通过 `nsswitch` 机制执行的 `glibc` 名称解析进行调整，使其适应 LDAP 的部署。安装此软件包时将在 `/etc/` 中创建新的调整过的文件 `nsswitch.conf`。有关 `nsswitch.conf` 工作方式的详细信息，请参见第 22.5.1 节“配置文件”。`nsswitch.conf` 中必须存在以下行才能进行用户管理及 LDAP 鉴定。请参见例 29.12 “`nsswitch.conf` 中的调整”。

#### *29.12: `nsswitch.conf` 中的调整*

```
passwd:compat group:compat  
  
passwd_compat:ldap group_compat:ldap
```

这些行指示 `glibc` 的解析程序库首先评估 `/etc` 中的相应文件，而后还要访问作为鉴定和用户数据来源的 LDAP 服务器。测试这种机制，例如通过使用命令 `getent passwd` 读取用户数据库中的内容。返回的结果集不仅应该包含您系统中本地用户的调查结果，还应包含所有储存在 LDAP 服务器上的用户的调查结果。

要防止通过 LDAP 管理的普通用户使用 `ssh` 或 `login` 登录该服务器，文件 `/etc/passwd` 和 `/etc/group` 都需要添加另外一行。这一行在 `/etc/passwd` 中为 `+:::/:sbin/nologin`，在 `/etc/group` 中为 `+:::`。

## 29.5.2 LDAP 客户程序的配置

在用 YaST 以所需方式修改 `nss_ldap`、`pam_ldap`、`/etc/passwd` 和 `/etc/group` 后，可以在第一个 YaST 对话框中开始实际的配置工作。请参见图 29.2 “YaST: LDAP 客户程序的配置”。

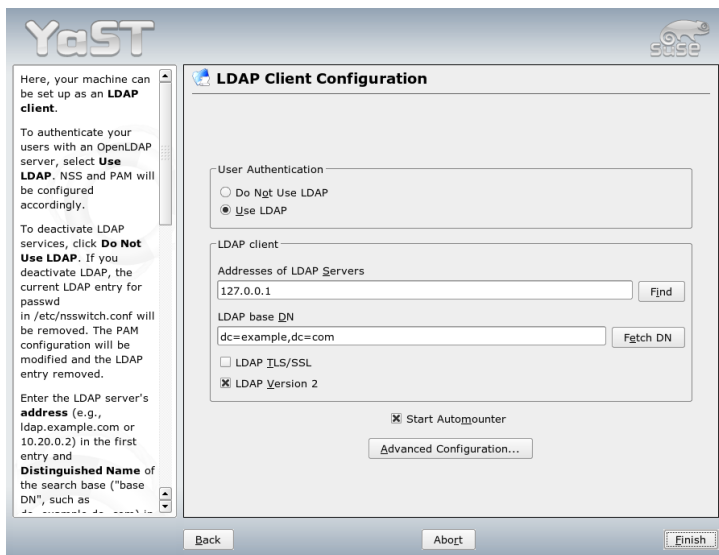


Figure 29.2: YaST: LDAP 客户程序的配置

在第一个对话框中为用户鉴定启用 LDAP。在 ‘LDAP 基本 DN’ 中输入服务器上的搜索基础，该搜索基础之下的所有数据都储存在 LDAP 服务器中。在 ‘LDAP

服务器地址’中输入该 LDAP 服务器的地址。要在远程主机上自动装入目录，请选择‘启动 Automounter’。要作为管理员修改服务器上的数据，请单击‘高级配置’。请参见图29.3 “YaST：高级配置”。

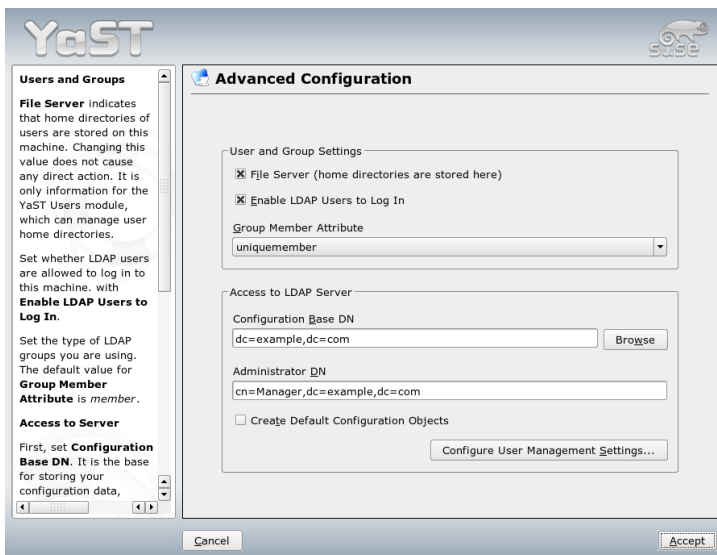


Figure 29.3: YaST：高级配置

下一个对话框包含两部分：在上半部分，设置用户和组的常规选项（如 YaST 用户模块所示）。在下半部分，提供访问 LDAP 服务器所需的数据。用户和组设置包含以下项目：

**文件服务器** 如果当前系统是文件服务器（/home 包含各个用户的目录），那么启用此选项可以确保 YaST 模块正确处理用户目录。

**允许 LDAP 用户登录** 启用此选项将允许通过 LDAP 管理的用户登录系统。

**组成员特性** 使用此选项可以指定要使用的 LDAP 组的类型：‘成员’（默认设置）或‘唯一成员’。

在此输入修改 LDAP 服务器配置所需的访问数据。包括‘配置基本 DN’（所有配置对象都储存在其下）和‘管理员 DN’。

单击‘配置用户管理设置’编辑 LDAP 服务器上的项。在随后出现的对话框中，输入 LDAP 口令，进行服务器鉴定。随后将根据服务器上储存的 ACL 和 ACI 授予对服务器上的配置模块的访问权。

## Important

### 使用 YaST 客户程序

使用 YaST LDAP 客户程序可以调整 YaST 模块使其适应用户和组管理，并按需扩展这些模块。不仅如此，还可以为各个特性定义带有默认值的模板，以简化实际注册数据的过程。在此创建的预设值将作为 LDAP 对象储存在 LDAP 目录中。用户数据的注册仍使用常用的 YaST 模块输入表单来完成。注册信息作为对象储存在 LDAP 目录中。

Important

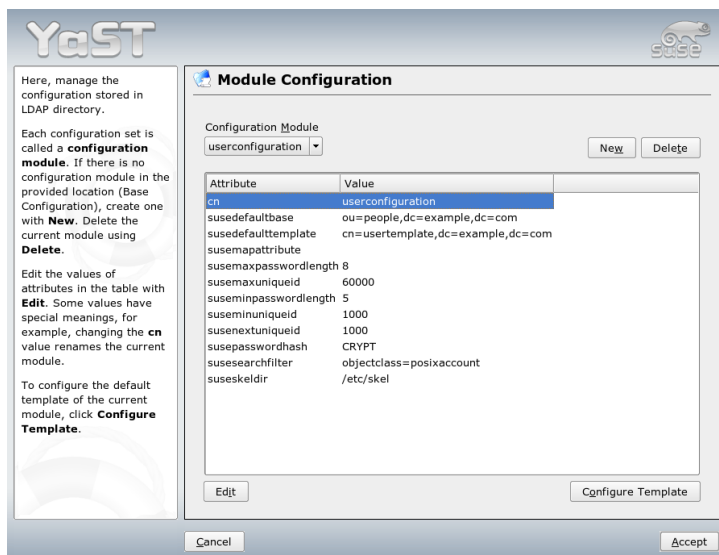


Figure 29.4: YaST: 模块配置

模块配置对话框(图29.4 “YaST: 模块配置”)允许选择和修改现有配置模块、创建新模块，以及设计和修改这些模块的模板。要修改配置模块中的值或重命名模块，请选择当前模块的内容视图之上的模块类型。内容视图随即显示一个表，

列出此模块允许使用的所有特性及其指派值。除所有已设置的特性之外，该列表还包含当前方案允许的但当前未使用的所有其他特性。

要复制模块，只需更改 `cn`。要修改各个特性值，请从内容列表中选中它们，然后单击‘编辑’。在随后打开的对话框中可以更改属于该特性的所有设置。选择‘确定’即接受更改。

如果应将新模块添加到现有模块中，请单击位于内容概览之上的‘新建’。在随后出现的对话框中，输入新模块（`suseuserconfiguration` 或 `susegroupconfiguration`）的名称和对象类。单击‘确定’关闭该对话框后，新模块即可添加到现有模块的选择列表中，可供选择或取消选择。单击‘删除’将删除当前所选模块。

用于组和用户管理的 YaST 模块内嵌带有合理标准值的模板（如果事先已使用 YaST LDAP 客户程序定义）。要按需编辑模板，请单击‘配置模板’。下拉菜单包含现有的可修改的模板或是空项。选择其中一个模板并在‘对象模板配置’表单中配置此模板的属性（请参见图29.5“YaST：对象模板的配置”）。此表单分为两个表格形式的概览窗口。上面的窗口列出所有常规模板特性。按需确定特性值，或者将某些值留空。LDAP 服务器将删除空特性值。

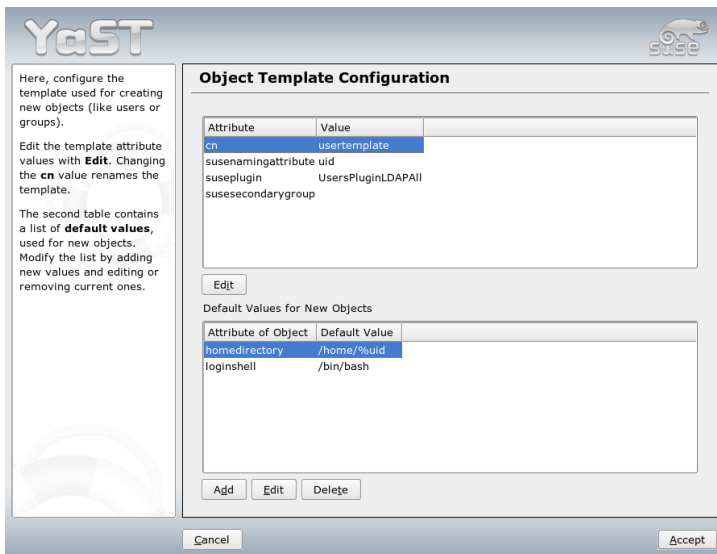


Figure 29.5: YaST：对象模板的配置

第二个视图（‘新对象的默认值’）列出相应 LDAP 对象（在此为组或用户配置）的所有特性，并且为所有特性都定义了标准值。可以添加更多特性和特性标准值、编辑现有特性和值对，或删除所有特性。通过更改 `cn` 项可以复制模板。通过将模块的 `susedefaulttemplate` 特性值设置为调整过的模板的 DN，可以将模板与模块连接起来（上文已经讨论过）。

### Tip

通过用变量样式代替绝对值的方法，可以从其他特性为某个特性创建默认值。例如，创建新用户时，将从 `sn` 和 `givenName` 的特性值自动创建 `cn=%sn %givenName`。

### Tip

在所有模块和模板经过正确配置能够运行后，可以使用 YaST 按通常方式注册新组和用户。

## 29.5.3 用户和组 — 使用 YaST 配置

用户和组数据的实际配置过程与不使用 LDAP 时的过程相差无几。下面简要说明了用户管理过程。管理组的过程与此相似。

通过‘安全性和用户’→‘用户管理’访问 YaST 用户管理。随即显示输入表单，用来注册最重要的用户数据，如用户名、登录名和口令。选择‘详细信息’可以访问用于配置组成员关系、登录 shell 和主目录的表单。默认值是在第 29.5.2 节“LDAP 客户程序的配置”所述的过程中定义的。若使用了 LDAP，此表单将转至另一个用于注册 LDAP 专有特性的表单。图 29.6 “YaST：其他 LDAP 设置”中显示了这一工具。选择所有要更改值的特性，然后单击‘编辑’。关闭使用‘继续’打开的表单，将返回最初的用户管理输入表单。

最初的用户管理输入表单提供了‘LDAP 选项’。通过此选项可以对一组现有用户应用 LDAP 搜索过滤器，或者通过选择‘LDAP 用户和组配置’转至用于配置 LDAP 用户和组的模块。

## 29.6 参考信息

本章特意排除了一些较为复杂的主题，如 SASL 配置，或如何通过建立复制 LDAP 服务器在多台从属服务器上分配工作量。有关这两个主题的详细信息，请参见《*OpenLDAP 2.2 Administrator's Guide*》（OpenLDAP 2.2 管理员指南）（请参考下文）。

OpenLDAP 项目的万维网站点为初级和高级 LDAP 用户提供了丰富的文档：

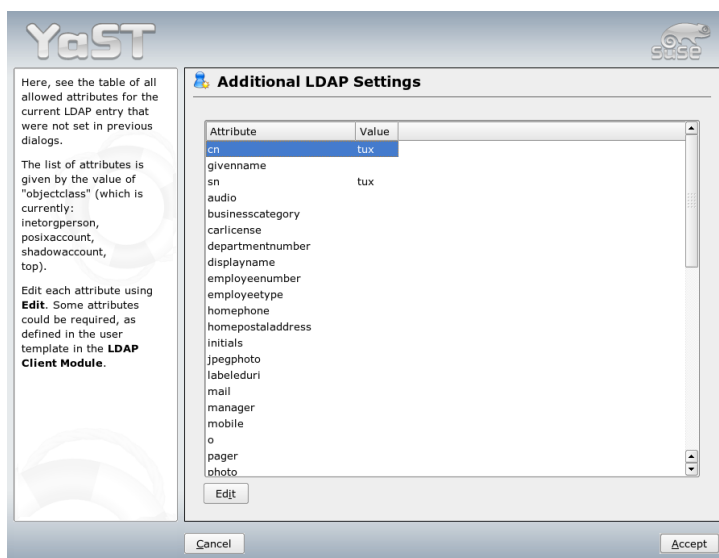


Figure 29.6: YaST: 其他 LDAP 设置

**OpenLDAP Faq-O-Matic** 涉及 OpenLDAP 的安装、配置和部署的非常详尽的问答集锦。 <http://www.openldap.org/faq/data/cache/1.html>.

**快速入门指南** 为首次安装 LDAP 服务器提供了简明的分步说明。

<http://www.openldap.org/doc/admin22/quickstart.html> 或已安装系统中的 `/usr/share/doc/packages/openldap2/admin-guide/quickstart.html`

**OpenLDAP 2.2 管理员指南** 关于 LDAP 配置的所有重要方面（包括访问控制和加密）的详细介绍。 <http://www.openldap.org/doc/admin22/> 或已安装系统中的 `/usr/share/doc/packages/openldap2/admin-guide/index.html`

以下 IBM 红皮书与 LDAP 有关：

#### Understanding LDAP (了解 LDAP)

关于 LDAP 基本原理的详尽的一般性介绍：<http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf>.



### LDAP Implementation Cookbook (LDAP 实施指南)

目标读者为 *IBM SecureWay Directory* 的管理员。不过，下面的网址也提供关于 LDAP 的重要的一般性信息：

<http://www.redbooks.ibm.com/redbooks/pdfs/sg245110.pdf>.

### LDAP 印刷文献资料：

- Howes、Smith、Good 合著：《*Understanding and Deploying LDAP Directory Services*》（了解和部署 LDAP 目录服务）。Addison-Wesley，第二版，2003。（ISBN 0-672-32316-8）
- Hodges 著：《*LDAP System Administration*》（LDAP 系统管理）。O'Reilly & Associates，2003。（ISBN 1-56592-491-6）

关于 LDAP 这一主题最后还可以参考相应的 RFC（请求注释）2251 到 2256。



# Apache 万维网服务器

Apache 所占的市场份额超过了 60%，它是世界上使用最为广泛的一种万维网服务器（来源：<http://www.netcraft.com>）。对于万维网应用程序，Apache 常与 Linux、数据库 MySQL 及编程语言 PHP 和 Perl 一起使用。这种组合通常被称为 *LAMP*。

本章对万维网服务器 Apache 进行了介绍。另外还介绍了如何执行安装和配置以及一些可用模块。本章还介绍了虚拟主机的一些变体。

30.1	基础	434
30.2	使用 YaST 设置 HTTP 服务器	435
30.3	Apache 模块	435
30.4	线程	436
30.5	安装	437
30.6	配置	438
30.7	使用 Apache	443
30.8	活动内容	443
30.9	虚拟主机	448
30.10	安全性	450
30.11	查错	451
30.12	参考信息	452

## 30.1 基础

通过本节，您可以对万维网服务器和它们使用的协议有一个基本的了解。另外还介绍了一些重要功能。

### 30.1.1 万维网服务器

万维网服务器发出客户程序请求的 HTML 页。这些页可以储存在目录中（被动页或静态页），也可以根据查询动态生成（活动内容）。

### 30.1.2 HTTP

客户程序通常是万维网浏览器，如 Konqueror 或 Mozilla。浏览器和万维网服务器之间的通讯是通过超文本传送协议 (HTTP) 进行的。当前版本 HTTP 1.1 记录在 RFC 2068 及更新文档 RFC 2616 中。可以在 <http://www.w3.org> 获得这些 RFC。

### 30.1.3 URL

客户程序使用 URL（如 <http://www.novell.com/linux/suse/>）来请求服务器中的页。URL 由以下几部分组成：

协议 常用协议：

`http://` HTTP 协议

`https://` HTTP 的安全、加密版本

`ftp://` 用于下载和上载文件的文件传送协议

**域** 在本例中，域为 `www.suse.com`。域可以分为两部分。第一部分 (`www`) 指向计算机。第二部分 (`suse.com`) 是实际的域。两部分合在一起被称为 FQDN（完全限定的域名）。

**资源** 在本例中，资源为 `index_us.html`。此部分指定指向资源的完整路径。资源可以是文件，例如，本例中的资源就是文件。但资源还可以是 CGI 脚本、JavaServer 页或其他一些资源。

相应的因特网机制（如域名系统 DNS）将查询转发给域 `www.suse.com`，从而将其定向到一个或多个储存资源的计算机。随后，Apache 从其文件目录中传递实际资源（在本例中为页 `index_us.html`）。在本例中，文件位于顶级目录。但资源还可以位于子目录中，如在 `http://support.novell.com/linux/` 中，资源便位于子目录中。

文件路径是相对于 `DocumentRoot` 的，而后者可以在配置文件中进行修改。第 30.6.2 节“`DocumentRoot`”介绍如何进行这种更改。

### 30.1.4 默认页的自动显示

如果没有指定默认页，则 Apache 会自动将一个常用的默认页名追加到 URL 后。这种页最常用的名称是 `index.html`。可以按第 30.6.2 节“`DirectoryIndex`”所述配置此功能以及服务器应使用的实际页名。在本例中，`http://www.suse.com` 足以提示服务器传递页 `http://www.novell.com/linux/suse/`。

## 30.2 使用 YaST 设置 HTTP 服务器

可以借助 YaST 方便地设置 Apache，但要以这种方式设置万维网服务器，需要有关此主题的一些知识。在选择 YaST 控制中心的‘网络服务’→‘HTTP 服务器’后，系统可能提示您安装仍缺少的一些包。安装了所有的包后，YaST 显示配置对话框（‘HTTP 服务器配置’）。

在此对话框中，首先启用‘HTTP 服务’本身。此操作还在防火墙中打开相应的端口（端口 80）（‘打开所选端口上的防火墙’）。窗口（‘设置/摘要’）下部用于设置本地 HTTP 服务器：‘监听’（默认值为 `port 80`）、‘模块’、‘默认主机’和‘主机’。‘编辑’用于更改当前所选的设置。

首先检查‘默认主机’，如有必要，请根据要求调整配置。然后，用‘模块’激活所需的模块。还可以使用另外一些对话框进行更详细的配置，特别是在创建虚拟主机时。

## 30.3 Apache 模块

通过使用模块，可以扩展 Apache，使其具有各种功能。例如，Apache 可以执行使用多种编程语言编写的访问此类模块的 CGI 脚本。除 Perl 和 PHP 外，还可

以使用其他脚本语言（如 Python 或 Ruby）。还有用于安全数据传送（安全套接层，SSL）、用户鉴定、扩展的日志记录以及更多其他功能的模块。

如果您具有所需的知识，则可以用自己编写的模块对 Apache 进行调整，使其符合各种要求和偏好。有关信息，请参考第30.12.4节“其他资源”。

可以指定多个“处理程序”（通过配置文件中的指令）来处理查询。这些处理程序可以是 Apache 的一部分，也可以是处理查询时调用的模块，因此可以用非常灵活的方式来安排此过程。还可以将自定义模块用于 Apache 以影响处理请求的方式。

Apache 中的模块化程度已达到一个很高的层次，除某些较小的任务外，任何任务都可以通过模块来处理。到目前为止，即使是 HTTP 也可以通过模块来进行处理。因此，Apache 不仅仅是一个万维网服务器。通过其他模块，它还可以用于完全不同的用途。例如，目前就有基于 Apache 的概念验证型邮件服务器 (POP3)。

Apache 模块还提供了其他一些有用功能：

**虚拟主机** 对虚拟主机的支持意味着可以将 Apache 的一个实例和一台计算机用于多个万维网站点。对用户而言，万维网服务器是作为多个独立的万维网服务器出现的。可以通过不同的 IP 地址或根据名称来配置虚拟主机。这就节省了购买额外计算机的成本并减轻了管理负担。

**灵活的 URL 重写** Apache 提供了许多方法来操纵和重写 URL。有关详细信息，请参见 Apache 文档。

**内容协商** Apache 可以传递适合客户程序（浏览器）功能的页。例如，可以为较早的浏览器或只能在文本方式下操作的浏览器（如 Lynx）传递无框架的简单页。这样，只要您愿意针对各种浏览器调整 JavaScript 代码，各种浏览器在 JavaScript 方面的不兼容性都可以通过为每个浏览器传递合适的网页来解决。

**灵活的错误处理** 灵活反应并在出现错误（如页不存在）时作出合适的响应。甚至可以动态生成（例如，通过 CGI）响应。

## 30.4 线程

线程是一种“更小”的进程。线程相对于进程的优点是它占用较少的资源。因此，使用线程而非进程可以提高性能。缺点是在线程环境中执行的应用程序必须是线程安全的。这意味着：

- 函数（或面向对象应用程序中的方法）必须是可重入的，也就是说，如果输入相同，函数始终应返回相同的结果，即使其他线程并行执行同一函数。因此，必须将函数编写成可由多个线程同时执行。
- 对资源（通常是变量）的访问必须以并发线程不会发生冲突的方式进行。

Apache 2 将查询作为单独的进程进行处理，或将进程和线程结合起来以混合方式处理查询。MPM *prefork* 用于以进程的形式执行。MPM *worker* 用于以线程的形式执行。请在安装期间选择要使用的 MPM（请参见第 30.5 节“安装”）。第三种方式 (*perchild*) 尚未完全成熟，因此不能在 SUSE LINUX 中安装。

## 30.5 安装

### 30.5.1 在 YaST 中选择包

对于基本安装，选择 Apache 包 `apache2` 就足够了。另外，可以安装其中一个 MPM（多重处理模块）包，如 `apache2-prefork` 或 `apache2-worker`。当选择 MPM 时，请记住基于线程的 `worker` MPM 不能与 `mod_php4` 一起使用，因为 `mod_php4` 的某些库仍不是线程安全的。

### 30.5.2 激活 Apache

安装后，必须在运行级别编辑器中将 Apache 作为服务激活。要在系统引导时启动它，请在运行级别编辑器中选中运行级别 3 和 5。要测试 Apache 是否正在运行，请在浏览器中转到 `http://localhost/`。如果 Apache 处于活动状态，您将看到一个示例页（假定已安装了 `apache2-example-pages`）。

### 30.5.3 用于活动内容的模块

要借助模块使用活动内容，请安装相应编程语言的模块。对于 Perl，该模块是 `apache2-mod_perl`；对于 PHP，该模块是 `mod_php4`；对于 Python，该模块是 `mod_python`。第 30.8.4 节“使用模块生成活动内容”中介绍了这些模块的用法。

## 30.5.4 其他建议使用的包

最好安装包 `apache2-doc` 中提供的文档。按照第30.5.2节“激活 Apache”所述安装了包并启动了服务器后，可以使用 URL `http://localhost/manual` 直接访问文档。

为了能够为 Apache 开发模块或编译第三方模块，另外还需要包 `apache2-devel` 及相应的开发工具。这其中还包括 `apxs` 工具，第30.5.5节“用 `apxs` 安装模块”中有所介绍。

## 30.5.5 用 `apxs` 安装模块

对模块开发人员来说，`apxs2` 是一个重要的工具。使用此程序可以通过一个命令从源代码编译并安装模块（包括对配置文件进行必需的更改）。此外，您还可以安装可用作对象文件（扩展名为 `.o`）的模块或静态库（扩展名为 `.a`）的模块。从源代码安装时，`apxs2` 将创建一个动态共享对象 (DSO)，Apache 将该对象直接作为模块使用。

用类似于 `apxs2 -c -i -a mod.foo.c` 的命令从源代码安装模块。`apxs2` 的其他选项在其手册页中有所介绍。然后，应该在 `/etc/sysconfig/apache2` 中使用选项 `APACHE_MODULES` 激活模块，如第30.6.1节“用 `SuSEconfig` 进行配置”中所述。

`apxs2` 有几个不同的版本：`apxs2`、`apxs2-prefork` 和 `apxs2-worker`。`apxs2` 安装的模块可用于所有 MPM。其他两个程序安装的模块只用于各自的 MPM（`prefork` 或 `worker`）。`apxs2` 在 `/usr/lib/apache2` 中安装模块，而 `apxs2-prefork` 在 `/usr/lib/apache2-prefork` 中安装模块。

# 30.6 配置

安装 Apache 后，只有在有特殊需要或偏好的情况下才需要进行额外更改。Apache 可以通过 `YaST` 和 `SuSEconfig` 来进行配置，也可以通过直接编辑文件 `/etc/apache2/httpd.conf` 来进行配置。

## 30.6.1 用 `SuSEconfig` 进行配置

在 `/etc/sysconfig/apache2` 中进行的设置由 `SuSEconfig` 应用于 Apache 配置文件。所提供的配置选项应足以满足大多数方案的需要。为文件中的每个变量都提供了一个注释，解释变量的作用。



## 自定义的配置文件

您可以借助变量 `APACHE_CONF_INCLUDE_FILES` 指定自己的配置文件（如 `httpd.conf.local`），而不用直接在配置文件 `/etc/apache2/httpd.conf` 中执行更改。然后，此文件将由主配置文件进行解释。这样，即使在新安装期间覆盖了文件 `/etc/apache2/httpd.conf`，也会保留对配置的更改。

## 模块

可以激活通过 YaST 安装的模块，方法是将模块的名称包括在变量 `APACHE_MODULES` 下指定的列表中。此变量在文件 `/etc/sysconfig/apache2` 中定义。

## 标志

`APACHE_SERVER_FLAGS` 可用于指定激活或取消激活配置文件某些部分的标志。如果配置文件的某一部分包含在

```
<IfDefine someflag>
.
.
.
</IfDefine>
```

中，则只有在变量 `ACTIVE_SERVER_FLAGS: ACTIVE_SERVER_FLAGS = someflag` 中设置了相应的标志，才能激活此部分。这样，可以很容易地激活或取消激活配置文件的很多部分以进行测试。

## 30.6.2 手工配置

编辑配置文件 `/etc/apache2/httpd.conf` 可以启用无法通过 `/etc/sysconfig/apache2` 中定义的设置使用的功能。以下几节介绍了可以设置的一些参数。下面以参数在文件中出现的顺序列出这些参数。

### DocumentRoot

一个基本设置是 `DocumentRoot`—Apache 在该目录下查找服务器要传递的万维网网页。对于默认虚拟主机，此目录设置为 `/srv/www/htdocs`。通常不需要更改此设置。

## Timeout

指定服务器在等待多长时间后将报告请求超时。

## MaxClients

Apache 可以同时处理的客户程序的最大数目。默认设置是 150，但此值对访问非常频繁的万维网站点来说可能太小。

## LoadModule

`LoadModule` 指令指定要装载的模块。装载顺序由模块自身确定。这些指令还指定包含模块的文件。

## Port

指定 Apache 监听查询的端口。通常是端口 80，它是 HTTP 的默认端口。通常不应更改此设置。让 Apache 监听另一个端口的原因可能是测试万维网站点的新版本。这样，可以继续通过默认端口 80 来访问万维网站点的运行版本。

另一个原因可能是您只希望在内部网中使用页，因为这些页包含的信息不是面向公众的。为此，请将端口设置为类似于 8080 之类的值，并通过防火墙阻止对此端口的外部访问。这样，服务器就可免受外部访问。

## Directory

使用此指令设置目录的访问权限和其他权限。`DocumentRoot` 也存在此类型的指令。只要更改了 `DocumentRoot`，就必须更改这里指定的目录名。

## DirectoryIndex

在此确定 Apache 应搜索哪些文件来补全未指定文件的 URL。默认设置是 `index.html`。例如，如果客户程序请求 URL `http://www.example.com/foo/bar`，而包含名为 `index.html` 的文件的目录 `foo/bar` 位于 `DocumentRoot` 下，则 Apache 会将此页返回给客户程序。

## AllowOverride

Apache 从中传递文档的每个目录都可以包含一个文件，该文件可以覆盖此目录的全局访问权限和其他设置。这些设置以递归的方式被应用到当前目录及其子目录中，直到被某个子目录中的另一个此类文件覆盖为止。因此，如果 `DocumentRoot` 中有这样一个文件，则将在全局应用该文件中指定的设置。通常，此类文件具有名称 `.htaccess`，但可以更改此名称，如第 30.6.2 节“`AccessFileName`”中所述。

使用 `AllowOverride` 确定在本地文件中指定的设置是否可以覆盖全局设置。可能值有 `None`、`All` 以及 `Options`、`FileInfo`、`AuthConfig` 和 `Limit` 的任意组合。Apache 文档详细介绍了这些值的含义。安全的默认设置是 `None`。

## Order

此选项确定应用 `Allow` 和 `Deny` 访问权限设置的顺序。默认设置是：

```
Order allow,deny
```

因此，首先应用允许访问的访问权限，之后应用拒绝访问的访问权限。底层方法基于下列各项之一：

**allow all** 允许所有访问并定义例外

**deny all** 拒绝所有访问并定义例外

`deny all` 的示例：

```
Order deny,allow
Deny from all
Allow from example.com
Allow from 10.1.0.0/255.255.0.0
```

## AccessFileName

在此设置文件的名称，这些文件可以覆盖 Apache 所提供目录的全局访问权限和其他设置（请参见第30.6.2节“`AllowOverride`”）。默认设置是 `.htaccess`。

## ErrorLog

指定 Apache 在其中记录错误讯息的文件的名称。默认设置是 `/var/log/httpd/errorlog`。除非在配置文件的 `VirtualHost` 部分中指定了特殊的日志文件，否则此文件还将记录虚拟主机的错误讯息（请参见第30.9节“虚拟主机”）。

## LogLevel

错误讯息是按照各种严重性级别进行分类的。此设置指定将记录超过哪个严重性级别的错误讯息。将其设置为某个级别会导致记录此严重性级别和更高严重性级别的错误讯息。默认设置是 `warn`。

## Alias

可以使用别名指定目录的快捷方式，通过该快捷方式可以直接访问此目录。例如，使用别名 `/manual/` 可以访问目录 `/srv/www/htdocs/manual`，即使将 `DocumentRoot` 设置为 `/srv/www/htdocs` 之外的目录也是如此（如果将 `DocumentRoot` 设置为该目录，则别名根本没有影响）。通过这个别名，`http://localhost/manual` 允许直接访问相应的目录。要定义用 `Alias` 指令指定的新目标目录的权限，可能要为其指定一条 `Directory` 指令。请参见第30.6.2节“`Directory`”。

## ScriptAlias

此指令与 `Alias` 类似。另外，此指令指出应将目标目录中的文件视为 CGI 脚本。

## Server-Side Includes

可以通过在所有可执行文件中搜索 SSI，激活服务器端包含。可以通过以下指令完成此操作：

```
<IfModule mod_include.c>
XBitHack on </IfModule>
```

要在文件中搜索 SSI，请使用命令 `chmod +x <filename>` 将此文件变为可执行文件。或者，显式指定要在哪种类型的文件中搜索 SSI。可以通过以下指令完成此操作：

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

建议不要设置 `.html`，因为这会导致 Apache 在所有页中搜索 SSI（甚至那些肯定不包含 SSI 的页），这会极大地影响性能。在 SUSE LINUX 中，配置文件中已包含了这两条指令，因此通常不需要进行更改。

## UserDir

借助模块 `mod_userdir` 和指令 `UserDir`，可以在用户的主目录中指定一个通过 Apache 发布文件的目录。可以在 `SuSEconfig` 中相应地设置变量 `HTTPD_SEC_PUBLIC_HTML` 来完成此配置。要启用文件的发布，必须将变量设置为 `yes`。此操作会在文件 `/etc/apache2/mod_userdir.conf` 中生成下列项（由 `/etc/apache2/httpd.conf` 进行解释）。

```
<IfModule mod_userdir.c>
UserDir public_html
</IfModule>
```

## 30.7 使用 Apache

要通过 Apache 显示静态万维网网页，只需将文件放在正确的目录中即可。在 SUSE LINUX 中，正确的目录是 `/srv/www/htdocs`。可能已在这里安装了一些较小的示例页。使用这些页可以检查是否已正确安装了 Apache 以及它当前是否处于活动状态。随后，只需覆盖或卸载这些页即可。自定义 CGI 脚本安装在 `/srv/www/cgi-bin` 中。

在运行期间，Apache 将日志讯息写入文件 `/var/log/httpd/access_log` 或 `/var/log/apache2/access_log` 中。这些讯息显示了在什么时间请求和提供了哪些资源以及采用了哪些方法（GET、POST 等）。错误讯息将被记录到 `/var/log/apache2`。

## 30.8 活动内容

Apache 提供了多个提供活动内容的方法。活动内容是根据来自客户程序的可变输入数据生成的 HTML 页，例如搜索引擎，对于用户输入的一个或多个搜索字符串（这些搜索字符串可能通过逻辑运算符 AND 或 OR 连结），它将返回一组包含这些搜索字符串的页。

Apache 提供三种方法来生成活动内容：

**服务器端包含 (SSI)** 服务器端包含是通过特殊注释嵌入到 HTML 页中的指令。

Apache 解释注释的内容并将结果作为该 HTML 页的一部分传递。

**通用网关接口 (CGI)** 通用网关接口是位于某些目录中的程序。Apache 将客户程序传送的参数转发给这些程序并返回程序的输出。此类编程很简单，尤其是因为现有命令程序的设计使它们可以从 Apache 接受输入并将它们的输出返回给 Apache。

**模块** Apache 提供了一些接口，可用于在请求处理的范围内执行任何模块。Apache 向这些程序授予访问重要信息（例如请求或 HTTP 报头）的权限。这些程序可以参与活动目录的生成及一些其他功能（例如鉴定）。这些模块的编程需要一些专业知识。此方法的优点是所提供的性能和功能超过了 SSI 和 CGI 所提供的性能和功能。

CGI 脚本由 Apache 在其拥有者的用户 ID 下直接执行，而模块由嵌入在 Apache 中的持久性解释器控制。这样，无需为每个请求启动和终止单独的进程（进程管理、内存管理等操作将会产生相当大的系统开销）。相反，脚本由在万维网服务器的 ID 下运行的解释器处理。

但是，此方法也有缺点。与模块相比，CGI 脚本对编程中所存在问题的容错程度要高一些。使用 CGI 脚本，错误（例如未能释放资源和内存）不会产生持续影响，因为会在处理请求后终止程序。这会清除由于编程错误而没有被程序释放的内存。如果使用模块，则会累积因编程错误而产生的影响，这是因为解释器是持久性的。如果几个月都没有重新启动服务器且解释器一直在运行，那么不能释放资源（例如数据库连接）将是一件令人头疼的事情。

### 30.8.1 服务器端包含

服务器端包含 (SSI) 是嵌入在特殊注释中并由 Apache 执行的指令。结果被嵌入在输出中。例如，可以用 `<!--#echo var="DATE_LOCAL" -->` 输出当前日期。开始注释标记 (`<!--`) 后面的 `#` 指示 Apache 这是一条 SSI 指令而不是简单的注释。

可以用多种方法激活 SSI。最简单的方法就是在所有可执行文件中搜索 SSI。另一种方法是指定特定的文件类型以搜索 SSI。第 30.6.2 节“Server-Side Includes”中解释了这两个设置。

### 30.8.2 通用网关接口

CGI 是 *common gateway interface*（通用网关接口）的英文缩写。使用 CGI，服务器不仅仅提供静态 HTML 页，它还执行生成该页的程序。这便可以生成表示计算结果（如在数据库中进行搜索的结果）的页。通过传递给被执行程序的参数，程序可以为每个请求返回单独的响应页。

CGI 的主要优点是这种技术相当简单。唯一的要求就是该程序必须存在于将由万维网服务器执行的特定目录中，就像命令行程序一样。服务器将标准输出通道 (stdout) 上的程序输出发送到客户程序。

在理论上，可以用任何编程语言编写 CGI 程序。通常，可以使用脚本语言（解释语言，如 Perl 或 PHP）来编写 CGI 程序。如果注重速度，则使用 C 或 C++ 语言更适合。

最简单的情况下，Apache 在特定目录 (cgi-bin) 中查找这些程序。可以在配置文件中设置此目录，如第 30.6 节“配置”中所述。如有必要，可以指定其他目录。如果出现这种情况，Apache 将在这些目录中搜索可执行程序。但是，这样做会产生安全风险，因为任何用户都可以让 Apache 执行程序，而其中一些操作可能是恶意的。如果限制只能执行 cgi-bin 目录下的程序，则管理员可以轻松了解哪个用户在此目录中放置了哪些脚本和程序，并检查这些脚本和程序是否会执行恶意操作。

### 30.8.3 GET 和 POST

可以通过 GET 或 POST 将输入参数传递给服务器。根据所使用的方法，服务器通过不同的方式将参数传递给脚本。对于 POST，服务器通过标准输入 (stdin) 将参数传递给程序。当从控制台启动程序时，程序以相同的方式接收其输入。对于 GET，服务器使用环境变量 QUERY\_STRING 将参数传递给程序。

### 30.8.4 使用模块生成活动内容

许多模块都可以与 Apache 一起使用。“模块”这一术语有两层含义。首先，指可以集成在 Apache 中以处理特定功能的模块，例如上面介绍的嵌入编程语言的模块。

其次，当与编程语言一起使用时，模块是指一组独立的函数、类和变量。这些模块集成在程序中以提供某项功能，例如所有脚本语言都可用的 CGI 模块。这些模块提供了各种函数（如用于读取请求参数和 HTML 输出的方法），从而简化了 CGI 应用程序的编程。

### 30.8.5 mod\_perl

Perl 是一种流行的、经证明行之有效的脚本语言。Perl 有许多模块和库，包括扩展 Apache 配置文件的库。可以在位于 <http://www.cpan.org/> 的 Comprehensive Perl Archive Network (CPAN) 中获得 Perl 的许多库。

#### 设置 mod\_perl

要在 SUSE LINUX 中设置 mod\_perl，只需安装相应的包即可（请参见第 30.5 节“安装”）。安装后，Apache 配置文件包含所需的项（请参见 `/etc/apache2/mod_perl-startup.pl`）。有关 mod\_perl 的信息，请参见 <http://perl.apache.org/>。

#### mod\_perl 与 CGI

在简单的情况下，通过用不同的 URL 请求以前的 CGI 脚本，将其作为 mod\_perl 脚本运行。配置文件包含一些别名，这些别名指向同一目录，并通过 CGI 或 mod\_perl 执行它包含的任何脚本。所有这些项已存在于配置文件中。CGI 的别名项是：

```
ScriptAlias /cgi-bin/ "/srv/www/cgi-bin/"
```

mod\_perl 的项是:

```
<IfModule mod_perl.c>
# Provide two aliases to the same cgi-bin directory,
# to see the effects of the 2 different mod_perl modes.
# for Apache::Registry Mode
ScriptAlias /perl/          "/srv/www/cgi-bin/"
# for Apache::Perlrun Mode
ScriptAlias /cgi-perl/     "/srv/www/cgi-bin/"
</IfModule>
```

mod\_perl 还需要以下项。这些项已存在于配置文件中。

```
#
# If mod_perl is activated, load configuration information
#
<IfModule mod_perl.c>
PerlRequire /usr/include/apache/modules/perl/startup.perl
PerlModule Apache::Registry

#
# set Apache::Registry Mode for /perl Alias
#
<Location /perl>
SetHandler perl-script
PerlHandler Apache::Registry
Options ExecCGI
PerlSendHeader On
</Location>

#
# set Apache::PerlRun Mode for /cgi-perl Alias
#
<Location /cgi-perl>
SetHandler perl-script
PerlHandler Apache::PerlRun
Options ExecCGI
PerlSendHeader On
</Location>

</IfModule>
```

这些项为 Apache::Registry 和 Apache::PerlRun 方式创建别名。这两种方式的區別是:

**Apache::Registry** 将编译所有脚本并将它们保存在超速缓存中。将以子例程的内容的形式应用每个脚本。尽管这种方式有利于提高性能,但也有缺点:



编写脚本时必须非常小心，因为变量和子例程会在请求间持续存在。这意味着您必须重置变量以便在下次请求时使用。例如，如果客户的信用卡号储存在在线银行脚本的变量中，当下一个客户使用此应用程序并请求同一脚本时，此号码可能再次出现。

**Apache::PerlRun** 为每个请求重新编译脚本。变量和子例程在请求之间从名称空间中消失（名称空间是在脚本存在期间内某个给定时间定义的所有变量名和例程名形成的整体）。因此，使用 `Apache::PerlRun` 时不需要进行辛苦的编程工作，因为在启动脚本时将重新初始化所有变量，不会保留以前请求中的任何值。出于这个原因，`Apache::PerlRun` 比 `Apache::Registry` 慢一些，但仍比 CGI（尽管与 CGI 有些类似）快得多，因为不需要为解释器启动单独的进程。

### 30.8.6 mod\_php4

PHP 是一种专门为万维网服务器开发的编程语言。与命令储存在不同文件（脚本）中的其他语言相比，PHP 命令嵌入在 HTML 页中（与 SSI 类似）。PHP 解释器处理 PHP 命令并将处理结果嵌入到 HTML 页中。

PHP 的主页是 <http://www.php.net/>。为了让 PHP 正常工作，请为 Apache 2 安装 `mod_php4-core` 及 `apache2-mod_php4`。

### 30.8.7 mod\_python

Python 是一种面向对象的编程语言，其语法清晰易读。这种语言有一个不寻常但很方便的功能，即程序结构是根据缩进来确定的。块不是用大括号（例如在 C 和 Perl 中）或其他划分元素（例如 `begin` 和 `end`）定义的，而是根据缩进的层次来定义的。要安装的包是 `apache2-mod_python`。

有关此语言的详细信息，可在 <http://www.python.org/> 获得。有关 `mod_python` 的详细信息，请访问 URL <http://www.modpython.org/>。

### 30.8.8 mod\_ruby

Ruby 是一种相对较新的面向对象的高级编程语言，在某些方面类似于 Perl 和 Python，是一种理想的脚本语言。与 Python 类似，它的语法清楚了。另一方面，Ruby 采用缩写形式（例如用 `$.r` 代表在输入文件中读取的最后一行的编号）—但程序员对此功能褒贬不一。Ruby 的基本概念与 Smalltalk 的基本概念非常类似。

Ruby 的主页是 <http://www.ruby-lang.org/>。有一个用于 Ruby 的 Apache 模块。主页是 <http://www.modruby.net/>。

## 30.9 虚拟主机

通过虚拟主机，一个万维网服务器可以储存多个域。采用这种方法可以节省成本并减轻管理负担，因为无需为每个域都配置一个不同的服务器。有几个与虚拟主机有关的选项：

- 基于名称的虚拟主机
- 基于 IP 的虚拟主机
- 在一台计算机上运行 Apache 的多个实例

### 30.9.1 基于名称的虚拟主机

使用基于名称的虚拟主机，Apache 的一个实例可以储存多个域。您无需为计算机设置多个 IP。这是最简单、最受欢迎的方法。反对使用基于名称的虚拟主机的原因在 Apache 文档中有所介绍。

通过配置文件 `/etc/apache2/httpd.conf` 直接配置基于名称的虚拟主机。要激活基于名称的虚拟主机，请指定合适的指令 `NameVirtualHost *`。\* 足以提示 Apache 接受所有进来的请求。随后，请配置各个主机：

```
<VirtualHost *>
    ServerName www.example.com
    DocumentRoot /srv/www/htdocs/example.com
    ServerAdmin webmaster@example.com
    ErrorLog /var/log/apache2/www.example.com-error_log
    CustomLog /var/log/apache2/www.example.com-access_log common
</VirtualHost>

<VirtualHost *>
    ServerName www.myothercompany.com
    DocumentRoot /srv/www/htdocs/myothercompany.com
    ServerAdmin webmaster@myothercompany.com
    ErrorLog /var/log/apache2/www.myothercompany.com-error_log
    CustomLog /var/log/apache2/www.myothercompany.com-access_log common
</VirtualHost>
```

还必须为最初储存在服务器中的域 (`www.example.com`) 配置 `VirtualHost` 项。在本例中，在同一服务器上储存了原始域和另一个域 (`www.myothercompany.com`)。

就像在 `NameVirtualHost` 中一样，\* 用于 `VirtualHost` 指令中。Apache 使用 HTTP 报头中的主机字段将请求连接到虚拟主机。请求将被转发到其 `ServerName` 与此字段中指定的主机名匹配的虚拟主机。

对于指令 `ErrorLog` 和 `CustomLog`，日志文件不需要包含域名。这里，使用您选择的名称。

`ServerAdmin` 指定在出现问题时可以联系的负责人的电子邮件地址。如果出现错误，Apache 在发送给客户程序的错误讯息中提供此地址。

## 30.9.2 基于 IP 的虚拟主机

这种方法要求为计算机设置多个 IP。在这种情况下，Apache 主机的一个实例储存多个域，向每个域指派一个不同的 IP。下面的示例说明了如何配置 Apache 来储存原始 IP (192.168.1.10) 及采用其他 IP (192.168.1.20 和 192.168.1.21) 的另两个域。这个特定的示例只能在内部网中使用，因为因特网上不路由从 192.168.0.0 到 192.168.255.0 的 IP。

### 配置 IP 别名

要使 Apache 储存多个 IP，底层计算机必须接受对多个 IP 的请求。这称为多 IP 储存。为此，必须在内核中激活 IP 别名。在 SUSE LINUX 中，这是默认设置。

配置内核支持 IP 别名后，可以使用命令 `ifconfig` 和 `route` 设置主机上的其他 IP。必须以 `root` 用户身份执行这些命令。对于下面的示例，假定主机已有自己的 IP (如 192.168.1.10)，而将这个 IP 指派给了网络设备 `eth0`。

输入命令 `ifconfig` 查看主机的 IP。可以使用以下命令添加其他 IP：

```
ip addr add 192.168.1.20/24 dev eth0
```

所有这些 IP 都被指派给同一个物理网络设备 (`eth0`)。

### 具有 IP 的虚拟主机

在系统上设置了 IP 别名或为主机配置了几块网卡后，就可以配置 Apache 了。请为每个虚拟服务器指定一个单独的 `VirtualHost` 块。

```
<VirtualHost 192.168.1.20>
    ServerName www.myothercompany.com
    DocumentRoot /srv/www/htdocs/myothercompany.com
    ServerAdmin webmaster@myothercompany.com
    ErrorLog /var/log/apache2/www.myothercompany.com-error_log
```

```
    CustomLog /var/log/apache2/www.myothercompany.com-access_log common
</VirtualHost>

<VirtualHost 192.168.1.21>
    ServerName www.anothercompany.com
    DocumentRoot /srv/www/htdocs/anothercompany.com
    ServerAdmin webmaster@anothercompany.com
    ErrorLog /var/log/apache2/www.anothercompany.com-error_log
    CustomLog /var/log/apache2/www.anothercompany.com-access_log common
</VirtualHost>
```

只为附加域指定了 VirtualHost 指令。原始域 (www.example.com) 通过其自己的设置（在 DocumentRoot 下）在 VirtualHost 块之外配置。

### 30.9.3 Apache 的多个实例

通过上述提供虚拟主机的方法，一个域的管理员可以读取其他域的数据。要隔离各个域，请启动 Apache 的多个实例，每个实例在配置文件中都有自己的 User、Group 及其他指令设置。

在配置文件中，使用 Listen 指令指定由相应的 Apache 实例处理的 IP。对于上面的示例，用于第一个 Apache 实例的指令是：

```
Listen 192.168.1.10:80
```

对于另两个实例：

```
Listen 192.168.1.20:80
Listen 192.168.1.21:80
```

## 30.10 安全性

### 30.10.1 最大程度减少风险

如果计算机上不需要万维网服务器，请在运行级别编辑器中取消激活 Apache、将其卸载或最初就不要安装。要最大程度减少风险，请取消激活所有不需要的服务器。这特别适用于用作防火墙的主机。如果可能，不要在这些主机上运行任何服务器。

## 30.10.2 访问权限

### DocumentRoot 应属于 root 用户

默认情况下，DocumentRoot 目录 (/srv/www/htdocs) 和 CGI 目录都属于用户 root。您不应更改此设置。如果任何用户都可写入这些目录，则任何用户都可以将文件放入这些目录中。然后，Apache 可以通过用户 wwwrun 的权限执行这些文件。同样，Apache 对它提供的数据和脚本不应有任何写权限。因此，这些数据和脚本不应属于用户 wwwrun，而是属于另一个用户，如 root 用户。

如果要让用户能够将文件放入 Apache 的文档目录中，不要让任何用户都对此目录有写权限，而应创建一个任何用户都可写入的子目录，例如 /srv/www/htdocs/miscellaneous。

### 从主目录发布文档

如果需要允许用户发布文件，则可以将用户主目录的某个子目录声明为合适万维网发布的目录。传统上将此子目录命名为 ~/public\_html。默认情况下，在 SUSE LINUX 中激活此目录。有关详细信息，请参见第30.6.2节“UserDir”。

通过在 URL 中指定用户，可以访问这些万维网网页。URL 包含元素 ~username，将其作为到用户主目录中相应目录的快捷方式。例如，在浏览器中输入 `http://localhost/~tux` 会列出用户 tux 的主目录中目录 public\_html 中的文件。

## 30.10.3 保持更新

如果运行万维网服务器，特别是如果这个万维网服务器是公共可访问的，则随时注意获取有关错误和潜在易受攻击方面的信息。第30.12.3节“安全性”中列出了可供利用的信息和修复程序。

## 30.11 查错

如果出现问题（例如，Apache 不显示页或不能正确显示页），则执行下列过程可以帮助发现问题。首先，查看错误日志，检查能否通过日志中的讯息找出错误原因。常规错误日志是 /var/log/apache2/error\_log。

一个被证明行之有效的办法是在控制台跟踪日志文件以了解服务器对访问的反应。可以在 root 控制台输入以下命令进行跟踪。

```
tail -f /var/log/apache2/*_log
```

检查位于 <http://bugs.apache.org/> 的联机错误数据库。阅读相关的邮件列表和新闻组。可以在 <http://httpd.apache.org/userslist.html> 获得用户的邮件列表。建议的新闻组是 `comp.infosystems.www.servers.unix` 和相关的组。

如果上述方法都不能解决问题，并且您确定在 Apache 中检测到一个错误，则请在 <http://www.suse.de/feedback/> 报告该错误。

## 30.12 参考信息

Apache 是一种广泛使用的万维网服务器。因此，有大量的介绍 Apache 的文档，并且许多万维网站点都提供有关 Apache 的帮助和支持。

### 30.12.1 Apache

Apache 附带有详细文档。第30.5节“安装”中介绍了此文档的安装。安装后，请在 <http://localhost/manual> 访问文档。可从位于 <http://httpd.apache.org> 的 Apache 主页获得最新文档。

### 30.12.2 CGI

可在以下页中获得有关 CGI 的更多信息：

- <http://apache.perl.org/>
- <http://perl.apache.org/>
- <http://www.modperl.com/>
- <http://www.modperlcookbook.org/>
- <http://www.fastcgi.com/>
- <http://www.boutell.com/cgic/>

### 30.12.3 安全性

可在 <http://www.novell.com/linux/security/securitysupport.html> 获得 SUSE LINUX 包的最新增补程序。请定期访问此 URL。您还可以在此注册，预订 SUSE 有关安全性声明的邮件列表。

对于 Apache 中的错误，Apache 小组推行开放信息策略。[http://httpd.apache.org/security\\_report.html](http://httpd.apache.org/security_report.html) 公布了最新的错误报告和可能受攻击方面的信息。如果您检测到安全性错误（查看上面的页确保它不是已发现的错误），请将其报告给 [security@suse.de](mailto:security@suse.de) 或 [security@apache.org](mailto:security@apache.org)。

有关 Apache（和其他因特网程序）的安全问题信息的其他来源：

- <http://www.cert.org/>
- <http://www.vnunet.com/>
- <http://www.securityfocus.com/>

### 30.12.4 其他资源

如果遇到困难，请访问位于 <http://portal.suse.com/sdb/en/index.html> 的 SUSE 支持数据库。可以在 <http://www.apacheweek.com/> 获得介绍 Apache 的联机新闻。

[http://httpd.apache.org/ABOUT\\_APACHE.html](http://httpd.apache.org/ABOUT_APACHE.html) 提供了对 Apache 历史的介绍。此页还解释此服务器为什么被称为 Apache。

可以在 <http://httpd.apache.org/docs-2.0/en/upgrading.html> 获得有关从版本 1.3 升级到 2.0 的信息。





# 文件同步

现今有很多人都在同时使用多台计算机——一台在家用，一台或多台在办公室用，还可能携带便携式计算机或 PDA 在路上用。很多文件是所有这些计算机上共同需要的。所以，您可能希望能在所有计算机上工作，修改文件，让所有计算机都能提供最新的数据。

31.1	可用的数据同步软件	456
31.2	选择程序时的决定性因素	457
31.3	Unison 简介	460
31.4	CVS 简介	462
31.5	Subversion 简介	464
31.6	rsync 简介	467
31.7	mailsync 简介	468

## 31.1 可用的数据同步软件

数据同步对于通过快速网络永久互联的计算机而言并不是个问题。在这种情况下，使用 NFS 这样的网络文件系统并将文件储存在服务器上，就可以支持所有主机通过网络访问相同的数据。但如果网络连接较差或者不是永久连接，这种方法就行不通了。使用便携式计算机在途中工作时，所有所需文件的副本都必须位于本地硬盘上。不过，您需要随后同步修改的文件。在一台计算机上修改某个文件后，一定要更新该文件在所有其他计算机上的副本。对于零星的副本，可以用 scp 或 rsync 手工更新。但如果涉及大量文件，这个过程要复杂得多，您必须小心操作，避免出现旧文件覆盖新文件之类的错误。

### Warning

#### 数据丢失风险

开始通过同步系统管理数据之前，您应该熟悉所用的程序并测试其功能。一定要对重要文件进行备份。

### Warning

使用程序可以通过各种方法自动执行数据同步，从而克服手工同步数据时既耗时又容易出错的缺点。以下概要的目的只是让您大致了解这些程序的工作原理及它们的用法。如果打算使用它们，请阅读相应的程序文档。

### 31.1.1 Unison

Unison 不是网络文件系统。相反，文件的保存和编辑都是在本地进行的。可以手工执行 Unison 程序来同步文件。首次执行同步时，两台主机上都要创建一个数据库，其中包含校验和、时间戳和选定文件的权限。下次再执行时，Unison 将能够识别哪些文件已更改，并建议将更改传送到未作更改的主机。通常可以接受所有建议。

### 31.1.2 CVS

CVS 主要用于对程序源代码进行版本管理；使用它可以在多台计算机上保留文件的副本。因此，该程序也适用于数据同步。CVS 在服务器上维护一个中央储存库，其中保存着文件和对文件的更改。本地执行的更改将提交到该储存库，并能够通过更新从其他计算机检索。这两个过程都必须由用户启动。

若多台计算机上都发生了更改，CVS 能够非常灵活地处理错误。这些更改将被合并，若发生在同一行上，则会报告冲突。发生冲突时，数据库仍保持一致状态。冲突仅显示在客户机上并在客户机上解决。

### 31.1.3 subversion

与“演进而来”的 CVS 相比，subversion 是一个在设计上始终如一的项目。subversion 实际上是在技术上经过改进的 CVS。

subversion 对 CVS 在很多方面都作了改进。由于历史原因，CVS 只维护文件而忽略了目录。而在 subversion 中，目录也有版本历史记录，并且完全可以像文件那样复制和重命名。还可以向每个文件和每个目录添加元数据。通过版本控制可以全面维护这些元数据。不同于 CVS，subversion 支持通过专用协议（如 WebDAV - 基于万维网的分布式创作和版本控制）进行透明的网络访问。WebDAV 扩展了 HTTP 协议的功能，支持对远程万维网服务器上的文件进行协作式写访问。

subversion 在很大程度上是使用已有的软件包来组装的。因此，Apache 万维网服务器和 WebDAV 扩展件始终与 subversion 一同使用。

### 31.1.4 mailsync

不同于以上各节说明的同步工具，mailsync 只在邮箱之间同步电子邮件。该过程既适用于本地邮箱文件，也适用于 IMAP 服务器上的邮箱。

根据电子邮件报头中的信件 ID，将相应地同步或删除各封信件。不论是单个邮箱之间还是各级邮箱之间都可以执行同步。

### 31.1.5 rsync

在无需版本控制但需要通过慢速网络连接同步大型目录结构时，rsync 工具可以提供较为完善的机制，仅传送文件中的更改。其中不仅涉及文本文件，还包括二进制文件。为检测文件间的差异，rsync 会将文件划分为多个块，并计算各个块的校验和。

检测更改需要消耗一定的资源。要使用 rsync，准备同步的系统应能够伸缩自如。RAM 尤为关键。

## 31.2 选择程序时的决定性因素

### 31.2.1 客户机/服务器与对等模式

在分发数据时，常用的有两个模型。在第一个模型中，所有客户机都通过中央服务器来同步文件。所有客户机都应能够访问该服务器（至少能偶尔为之）。subversion、CVS 和 WebDAV 使用的都是这个模型。

另一个模型是让所有联网主机作为同级相互同步数据。unison 遵照的就是这种概念。rsync 实际是以客户机方式工作，但任一台客户机又可以充当服务器。

### 31.2.2 可移植性

subversion、CVS 和 unison 还适用于其他很多操作系统，包括各种 Unix 和 Windows 系统。

### 31.2.3 交互与自动

在 subversion、CVS、WebDAV 和 Unison 中，数据同步是由用户手工启动的。这样可以有效控制要同步的数据并易于解决冲突。不过，如果同步间隔过长，就容易发生冲突。

### 31.2.4 冲突：冲突的发生和解决方法

在 subversion 或 CVS 中很少发生冲突，即便是多人同时在一个大型程序项目上协作时也不例外。这是因为合并文档时基于的是单个行。发生冲突时，只有一个客户机会受影响。通常很容易解决 subversion 或 CVS 中发生的冲突。

Unison 可以报告冲突，允许从同步中排除受影响的文件。不过，合并更改时并不像在 subversion 或 CVS 中那样容易。

不同于 subversion 或 CVS（这两个程序允许在发生冲突时部分接受更改），WebDAV 只有在成功进行完整修改后才执行签入。

rsync 中不提供冲突解决功能。用户自己要避免意外覆盖文件，并手工解决所有可能的冲突。为安全起见，还可以使用 RCS 之类的版本控制系统。

### 31.2.5 选择和添加文件

按照 Unison 的标准配置，它会同步整个目录树。目录树中新出现的文件将被自动纳入同步。

在 subversion 或 CVS 中，必须分别使用命令 `svn add` 或 `cvs add` 明确添加新目录和文件。这样用户可以更有效地控制要同步的文件。但另一方面，这样也容易遗漏新文件，特别是在有大量文件时，很容易忽略 `svn update` 和 `svn status` 或 `cvs update` 输出中的问号。

### 31.2.6 历史记录

subversion 或 CVS 的另一个功能是能够重建旧文件版本。每次一有更改都可以插入一个简短的编辑注释，以后根据文件内容和这些注释就很容易跟踪文件的变化。这对论文和程序文本大有帮助。

### 31.2.7 数据量和硬盘要求

所有相关主机的硬盘上都要有足够的可用于所有分发数据的空间。subversion 和 CVS 还要求服务器为储存库准备额外的空间。文件历史记录也储存在服务器上，这进一步增加了空间要求。更改文本格式的文件时，只需保存修改的那些行。而二进制文件则要求在每次更改文件时都要有与文件大小相同的额外空间。

### 31.2.8 GUI

Unison 提供图形用户界面，显示 Unison 要执行的同步过程。可以接受建议或从同步中排除单个文件。在文本方式中，交互确认各个过程。

有经验的用户通常从命令行运行 subversion 或 CVS。不过，图形用户界面也适用于 Linux（如 cervisia）以及其他操作系统（如 wincvs）。许多开发工具（如 kdevelop）及文本编辑器（如 emacs）都提供针对 CVS 或 subversion 的支持。在这些前端上解决冲突往往较为容易。

### 31.2.9 用户友好

Unison 和 rsync 较便于使用，因此适合新手使用。CVS 和 subversion 相对而言较难操作。用户应该了解储存库和本地数据之间如何交互。对数据的更改首先要在本地与储存库合并。使用命令 `cvs update` 或 `svn update` 可完成上述操作。然后必须使用命令 `cvs commit` 或 `svn commit` 将数据发回储存库。一旦了解了此过程，新手也就能毫不费力地使用 CVS 或 subversion 了。

### 31.2.10 预防攻击

在传送数据的过程中，最好防止数据被拦截或操纵。Unison、CVS、rsync 和 subversion 可以方便地通过 ssh（安全 shell）使用，从而防止遭受此类攻击。应避免通过 rsh（远程 shell）运行 CVS 或 Unison。同时也不建议在安全的网络中使用 *pserver* 机制访问 CVS。subversion 若与 Apache 同时运行就已经提供了必要的安全措施。

### 31.2.11 防止数据丢失

开发人员使用 CVS 来管理程序项目已有很长时间，所以该程序极为稳定。由于能够保存开发历史记录，CVS 甚至能够预防某些用户错误，如意外删除文件。尽管 subversion 不如 CVS 常用，但它已运用到生产环境中，例如供 subversion 项目本身使用。

Unison 仍然是个较新的程序，但它的稳定性极高。不过，它对用户错误较为敏感。一旦确认对已删除的文件进行同步，就无法再恢复该文件。

*Table 31.1: 各种文件同步工具的特点: -- = 非常差、- = 较差或不可用、o = 中等、+ = 好、++ = 极好、x = 可用*

	unison	CVS/subv.	rsync	m
客户机/服务器	同级	客户机-服务器/客户机-服务器	客户机-服务器	同
可移植性	Lin、Un*x、Win	Lin、Un*x、Win	Lin、Un*x、Win	Li
交互能力	x	x/x	x	-
速度	-	o/+	+	+
冲突	o	++/++	o	+
文件选择	目录	选择/文件、目录	目录	邮
历史记录	-	x/x	-	-
硬盘空间	o	--	o	+
GUI	+	o/o	-	-
难易程度	+	o/o	+	o
攻击	+(ssh)	+/+(ssh)	+(ssh)	+(
数据丢失	+	++/++	+	+

## 31.3 Unison 简介

Unison 是同步和传送整个目录树的绝佳解决方法。可以双向执行同步，并通过直观的图形化前端控制同步。还可以使用控制台版本。同步可以自动执行，所以不必与用户交互，但这需要一定经验。

### 31.3.1 要求

必须在客户机和服务器上同时安装 Unison。在这种上下文环境中，服务器一词指另一台远程主机（不同于 CVS，如第31.1.2节“CVS”中所述）。

下一节中将讨论如何将 Unison 与 ssh 一同使用。一同使用时，必须将 SSH 客户程序安装在客户机上，将 SSH 服务程序安装在服务器上。

### 31.3.2 使用 Unison

Unison 所用的方法是将两个目录（根目录）关联起来。这种关联是象征性的——不是在线连接。在本例中，目录布局如下所示：

---

客户机：	/home/tux/dir1
服务器：	/home/geeko/dir2

---

您需要同步这两个目录。客户机上的用户是 tux，服务器上的用户是 geeko。首先要做的是测试客户机与服务器是否能够通讯：

```
unison -testserver /home/tux/dir1 ssh://geeko@server//homes/geeko/dir2
```

最常遇到的问题包括：

- 客户机和服务器上的 Unison 版本不兼容。
- 服务器不支持 SSH 连接。
- 这两个指定路径均不存在。

如果一切正常，省略选项 `-testserver`。在首次同步期间，Unison 并不了解这两个目录之间的关系，所以会针对各个文件和目录的传送方向提出建议。‘操作’列中的箭头指示传送方向。问号表示 Unison 无法就传送方向提出建议，因为两个版本均已更改或者都是新的。

使用箭头键可为单个项设置传送方向。如果所有显示项的传送方向都正确，只需单击‘转到’。

通过在启动 Unison 时指定命令行参数，可以控制该程序的特性（例如，是否以透明方式自动执行同步）。使用 `unison --help` 可以查看所有参数的完整列表。

### 31.1: 文件 `~/unison/example.prefs`

```
root=/home/tux/dir1
root=ssh://wilber@server//homes/wilber/dir2
batch=true
```

对于每组目录，都会在用户目录 `~/unison` 中维护一个同步日志。也可以在此目录中储存配置集，如 `~/unison/example.prefs`。要启动同步，将此文件指定为命令行参数，如 `unison example.prefs` 所示。

### 31.3.3 参考信息

Unison 的正式文档非常有用。故此，本节只作简要介绍。完整的手册在 <http://www.cis.upenn.edu/~bcpierce/unison/> 和 SUSE 软件包 `unison` 中提供。

## 31.4 CVS 简介

如果经常编辑各个文件并且这些文件以 ASCII 文本或程序源代码文本之类的格式储存，则应该使用 CVS 来进行同步。用 CVS 同步其他格式的数据（如 JPEG 文件）固然可行，但这会产生大量数据，因为文件的所有变化都永久储存在 CVS 服务器中。这种情况下将无法利用 CVS 的大多数功能。只有在所有工作站都可以访问同一服务器时，才能使用 CVS 同步文件。

### 31.4.1 配置 CVS 服务器

服务器是储存所有有效文件（包括所有文件的最新版本）的主机。任何固定的工作站都可以充当服务器。如果可能，应该对 CVS 储存库的数据进行定期备份。

配置 CVS 服务器时，通过 SSH 授予用户访问服务器的权限是一种不错的方式。如果用户在服务器上的用户名为 `tux`，并且在服务器和客户机上都安装了 CVS 软件，则必须在客户端设置以下环境变量：

```
CVS_RSH=ssh CVS_ROOT=tux@server:/serverdir
```



可使用命令 `cvs init` 从客户端初始化 CVS 服务器。只需执行一次初始化。

最后，必须给同步指派名称。仅在客户机上选择或创建目录，以包含要使用 CVS 来管理的文件（该目录也可以为空）。目录的名称同时也是同步的名称。在本例中，目录名为 `synchome`。转到此目录并输入以下命令，将同步名称设置为 `synchome`：

```
cvs import synchome tux wilber
```

许多 CVS 命令都需要注释。为此，CVS 会启动一个编辑器（在环境变量 `$EDITOR` 中定义的编辑器；如果未定义任何编辑器，则使用 `vi`）。通过提前在命令行中输入注释（如下例所示），可以避免调用编辑器。

```
cvs import -m 'this is a test' synchome tux wilber
```

## 31.4.2 使用 CVS

现在，在所有主机上都可以使用 `cvs co synchome` 将该同步储存库签出。该操作将在客户机上创建新的子目录 `synchome`。要向服务器提交更改，请转到目录 `synchome`（或其子目录之一），然后输入 `cvs commit`。

默认情况下，所有文件（包括子目录）都要提交给服务器。若仅提交单个文件或目录，请按 `cvs commit file1 directory1` 中的方式进行指定。在将新文件和目录提交给服务器之前，必须使用 `cvs add file1 directory1` 之类的命令先将其添加到储存库中。随后再使用 `cvs commit file1 directory1` 命令提交新添加的文件和目录。

如果转到另一个工作站，则需要签出同步储存库（如果在同一工作站上的较早会话中尚未执行该操作，请参见上文）。

使用 `cvs update` 开始与服务器同步。如 `cvs update file1 directory1` 所示更新各个文件或目录。要查看当前文件与服务器上储存的版本的差异，请使用命令 `cvs diff` 或 `cvs diff file1 directory1`。使用 `cvs -nq update` 可以查看哪些文件将受到更新的影响。

以下是更新期间显示的一些状态符号：

- U** 已更新本地版本。这将影响服务器提供的和本地系统缺少的所有文件。
- M** 已修改本地版本。若服务器发生更改，可以将差异并入本地副本。
- P** 已使用服务器上的版本修补本地版本。
- C** 本地文件与储存库中的当前版本冲突。

? 此文件在 CVS 中不存在。

状态 M 表示本地修改的文件。可以向服务器提交本地副本，也可以在删除本地文件后再次进行更新。更新后将能够从服务器中恢复缺失的文件。如果提交了本地修改的文件但提交的这个文件中的同一行发生了更改，则可能发生冲突（由 C 表示）。

在这种情况下，查看文件中的冲突标记（ $\$$  和  $\$$ ），决定要采用哪个版本。由于这是一项令人不快的工作，您可以选择放弃更改，删除本地文件，然后输入 `cvsv up` 从服务器恢复当前版本。

### 31.4.3 参考信息

本节仅对 CVS 的多种情况进行了简要介绍。以下 URL 提供了大量的文档：

<http://www.cvshome.org/>  
<http://www.gnu.org/manual/>

## 31.5 Subversion 简介

Subversion 是一个开放源代码的版本控制系统，被公认为是 CVS 的后续版本，这意味着上文介绍的 CVS 的功能 subversion 一般都提供。在希望利用 CVS 的优点但又想避开它的缺点时，subversion 是您的首选。这其中的很多功能已经在第 31.1.3 节“subversion”中简要介绍。

### 31.5.1 安装 Subversion 服务器

在服务器上安装储存库是个相对简单的过程。Subversion 为此提供了一个专用管理工具。创建新储存库需要输入以下命令：

```
svnadmin create /path/to/repository
```

使用 `svnadmin help` 可以列出其他选项。与 CVS 不同的是，subversion 不基于 RCS，而是基于 Berkeley Database。切勿将储存库安装在远程文件系统上，如 NFS、AFS 或 Windows SMB。该数据库需要 POSIX 锁定机制，而这是这些文件系统无法支持的。

命令 `svnlook` 提供有关现有储存库的信息。

```
svnlook info /path/to/repository
```

必须配置服务器，允许不同用户访问该储存库。可以使用支持 WebDAV 的 Apache 万维网服务器进行配置，或使用 `svnserve`，即用 `subversion` 封装的服务器。一旦 `svnserve` 启动并运行，即可使用带 `svn://` 或 `svn+ssh://` 的 URL 访问该储存库。可以在 `/etc/svnserve.conf` 中设置在调用 `svn` 时应自我鉴定的用户。

使用 Apache 还是 `svnserve` 取决于很多因素。建议阅读 `subversion` 手册。有关详细信息，请参见第 31.5.3 节“参考信息”。

## 31.5.2 用法和操作

使用命令 `svn`（类似 `cvs`）可访问 `subversion` 储存库。使用以下命令之一，从任何客户机都能够访问正确配置的服务器（配有相应的储存库）提供的内容。

```
svn list http://svn.example.com/path/to/project
```

或

```
svn list svn://svn.example.com/path/to/project
```

使用命令 `svn checkout` 可以将现有项目保存到当前目录中（将其签出）：

```
svn checkout http://svn.example.com/path/to/project nameofproject
```

签出操作将在客户机上创建新的子目录 `nameofproject`。随后可对其执行各种操作（添加、复制、重命名、删除）：

```
svn add file
svn copy oldfile newfile
svn move oldfile newfile
svn delete file
```

这些命令也适用于目录。`subversion` 还可以记录文件或目录的属性：

```
svn propset license GPL foo.txt
```

上例为属性 `license` 设置了值 `GPL`。使用 `svn proplist` 可以显示属性：

```
svn proplist --verbose foo.txt
Properties on 'foo.txt':
license : GPL
```

使用 `svn commit` 可以将更改保存到服务器中。其他用户通过使用 `svn update` 与服务器同步，可以将您的更改并入他或她自己的工作目录。

不同于 CVS，不必使用 `svn status` 访问储存库即可显示 subversion 中的工作目录的状态。本地更改显示在五列中，最前面一列最为重要：

- " 无更改。
- 'A' 对象被标记为添加。
- 'D' 对象被标记为删除。
- 'M' 对象已修改。
- 'C' 对象发生冲突。
- 'I' 对象被忽略。
- '?' 对象未能由版本控制维护。
- !' 报告对象缺失。在没有使用 `svn` 命令删除或移动对象时就会出现此标志。
- ~ 对象本来作为文件维护，但之后被替换为目录，或者出现相反的情况。

第二列显示属性状态。有关其他所有列的含义，请参见 subversion 手册。

使用命令 `svn help` 可获得命令参数的说明：

```
svn help proplist
proplist (plist, pl): List all properties on files, dirs, or revisions.
usage: 1. proplist [PATH...]
       2. proplist --revprop -r REV [URL]

1. Lists versioned props in working copy.
2. Lists unversioned remote props on repos revision.
...
```

### 31.5.3 参考信息

首要参考来源是 subversion 项目的主页：<http://subversion.tigris.org/>。还有一本很值得参考的手册。在安装 subversion-doc 软件包后，该手册位于 `file:///usr/share/doc/packages/subversion/html/book.html` 目录下，也可以在 <http://svnbook.red-bean.com/svnbook/index.html> 中查阅该手册的联机版本。

## 31.6 rsync 简介

如果需要定期传送大量数据而更改的数据量不是很大，则适用 rsync。举例来说，创建备份时的情况往往就是这样。另一种应用涉及临时服务器。临时服务器是储存万维网服务器的完整目录树的服务器，这些万维网服务器定期镜像到 DMZ 中的万维网服务器。

### 31.6.1 配置和操作

rsync 有两种操作方式。可用于存档或复制数据。要执行上述操作，目标系统上只需要有远程 shell，如 ssh。不过，rsync 也可用作守护程序，为网络提供目录。

rsync 的基本操作方式不需要任何特殊配置。rsync 能直接将完整目录镜像到其他系统中。举例来说，以下命令在名为 sun 的备份服务器上为 tux 的主目录创建了备份副本。

```
rsync -baz -e ssh /home/tux/ tux@sun:backup
```

以下命令用于回放该目录：

```
rsync -az -e ssh tux@sun:backup /home/tux/
```

到目前为止，该程序的操作方式与普通的复制工具 (如 scp) 的操作方式相差无几。

应该以“rsync”方式操作 rsync，以便充分利用其所有功能。这需要在其中一个系统上启动 rsyncd 守护程序。在文件 /etc/rsyncd.conf 中配置该守护程序。例如，要使目录 /srv/ftp 可用于 rsync，请使用以下配置：

```
gid = nobody
uid = nobody
read only = true
use chroot = no
transfer logging = true
log format = %h %o %f %l %b
log file = /var/log/rsyncd.log
```

```
[FTP]
```

```
path = /srv/ftp  
comment = An Example
```

然后使用 `rcrsyncd start` 启动 `rsyncd`。`rsyncd` 也可以在引导进程中自动启动。通过在 YaST 提供的运行级别编辑器中激活此服务或通过手工输入命令 `insserv rsyncd`，都可以完成上述设置。也可以使用 `xinetd` 来启动 `rsyncd`。不过，建议只在很少使用 `rsyncd` 的服务器上采用这种启动方式。

下例还创建了一个列出所有连接的日志文件。此文件储存在 `/var/log/rsyncd.log` 中。

随后可以从客户机系统测试传送。请使用以下命令完成该操作：

```
rsync -avz sun::FTP
```

此命令列出服务器的 `/srv/ftp` 目录中现有的所有文件。此请求还记录在日志文件 `/var/log/rsyncd.log` 中。要启动实际的传送，请提供目标目录。用 `.` 来表示当前目录。例如：

```
rsync -avz sun::FTP .
```

默认情况下，使用 `rsync` 同步时不会删除任何文件。如果应强制删除，必须明确指定附加选项 `--delete`。为保证不删除任何较新的文件，可转而使用选项 `--update`。必须手动解决所有冲突。

### 31.6.2 参考信息

有关 `rsync` 的重要信息，请参见手册页 `man rsync` 和 `man rsyncd.conf`。`/usr/share/doc/packages/rsync/tech_report.ps` 专门提供了关于 `rsync` 工作原理的技术参考。在 `rsync` 的万维网站点 <http://rsync.samba.org/> 上可以找到关于该项目的最新消息。

## 31.7 mailsync 简介

`mailsync` 主要适用于以下三项任务：

- 使本地储存的电子邮件与服务器上储存的邮件同步
- 将邮箱迁移为不同的格式或迁移到其他服务器
- 对邮箱进行完整性检查或搜索重复邮件

### 31.7.1 配置和用法

mailsync 能对邮箱本身（储存区）和两个邮箱之间的连接（通道）加以区分。储存区和通道的定义储存在 `~/.mailsync` 中。以下段落解释了若干储存示例：简单的定义可能显示如下：

```
store saved-messages {
    pat Mail/saved-messages
    prefix Mail/
}
```

`Mail/` 是包含电子邮件文件夹的用户主目录中的一个子目录，其中包含文件夹 `saved-messages`。如果 `mailsync` 是通过 `mailsync -m saved-messages` 启动的，则会列出 `saved-messages` 中所有信件的索引。如果指定了以下定义

```
store localdir {
    pat Mail/*
    prefix Mail/
}
```

命令 `mailsync -m localdir` 将列出所有储存在 `Mail/` 中的信件。相比之下，命令 `mailsync localdir` 只列出文件夹的名称。IMAP 服务器上的储存区的指定方式如下所示：

```
store imapinbox {
    server {mail.edu.harvard.com/user=gulliver}
    ref {mail.edu.harvard.com}
    pat INBOX
}
```

上例仅涉及 IMAP 服务器上的主文件夹。子文件夹的储存区显示如下：

```
store imapdir {
    server {mail.edu.harvard.com/user=gulliver}
    ref {mail.edu.harvard.com}
    pat INBOX.*
    prefix INBOX.
}
```

如果 IMAP 服务器支持加密连接，服务器的指定方式将改为

```
server {mail.edu.harvard.com/ssl/user=gulliver}
```

或者，如果服务器证书不详，则改为

```
server {mail.edu.harvard.com/ssl/novalidate-cert/user=gulliver}
```

前缀将在以后说明。

现在 Mail/ 下的文件夹应能连接到 IMAP 服务器上的子目录：

```
channel folder localdir imapdir {  
msinfo .mailsync.info  
}
```

mailsync 使用 msinfo 文件跟踪已经同步的信件。

命令 mailsync folder 执行以下操作：

- 在两端扩展邮箱模式。
- 从生成的文件夹名称中删除前缀。
- 成对同步文件夹（若不存在则创建文件夹）。

相应地，IMAP 服务器上的文件夹 INBOX.sent-mail 将与本地文件夹 Mail/sent-mail 进行同步（只要存在上面说明的定义）。各个文件夹之间的同步将执行如下：

- 如果两端都存在同一封信件，则不会执行任何操作。
- 如果一端缺少信件并且是新信件（未列在 msinfo 文件中），则信件将被传送到这一端。
- 如果只有一端有信件并且是旧信件（已列在 msinfo 文件中），则信件将从这一端删除（因为另一端显然有过该信件，只不过已经删除）。

要事先了解传送哪些信件以及在同步过程中删除哪些信件，请用 mailsync folder localdir 启动带有通道和储存区的 mailsync。此命令将生成一个包含本地主机上的所有新信件的列表，以及一个包含要在同步过程中从 IMAP 端删除的所有信件的列表。同样，命令 mailsync folder imapdir 将生成一个包含 IMAP 端上的所有新信件的列表，以及一个包含要在同步过程中从本地主机删除的所有信件的列表。



### 31.7.2 可能的问题

万一数据丢失，最保险的方法是删除相关的通道日志文件 `msinfo`。这样，只在一端存在的所有信件都将被视为新信件，可以在下一次同步时传送。

只有带有信件 ID 的信件才会被同步。缺少信件 ID 的信件将被忽略，也就是不传送或删除。缺少信件 ID 通常是由于发送或撰写信件时程序出错造成的。

在特定的 IMAP 服务器上，主文件夹称作 `INBOX`，子文件夹可以是随机选择的名称（不同于 `INBOX` 和 `INBOX.name`）。因此，对于这类 IMAP 服务器，不可能为子文件夹专门指定某种模式。

将信件成功传送到 IMAP 服务器后，`mailsync` 所用的邮箱驱动程序 (`c-client`) 会设置特殊的状态标志。出于这种原因，有些电子邮件程序（如 `mutt`）无法将这些信件识别为新信件。使用选项 `-n` 可以禁止设置这种特殊状态标志。

### 31.7.3 参考信息

有关详细信息，请参见 `mailsync` 程序提供的 `/usr/share/doc/packages/mailsync/` 中的 `README`。在这种连接中，RFC 2076 “Common Internet Message Headers”（常见因特网信头）值得特别注意。



# Samba

使用 Samba，可以将 Unix 计算机配置为 DOS、Windows 和 OS/2 计算机的文件和打印服务器。本章介绍 Samba 配置的基础知识以及可用来自网络中配置 Samba 的各个 YaST 模块。

32.1	配置服务器	475
32.2	将 Samba 用作登录服务器	478
32.3	使用 YaST 配置 Samba 服务器	479
32.4	配置客户机	480
32.5	优化	481

Samba 已经发展成为一个功能完备且相当复杂的产品。本章概述其基本功能。有关详细信息，请参见附带的数字文档。在命令行输入 `apropos samba` 可显示一些手册页；如果安装了 Samba，也可以浏览 `/usr/share/doc/packages/samba` 目录获得更多的联机文档和示例。`examples` 子目录中提供了一个带注释的示例配置 (`smb.conf.SuSE`)。

附带的 `samba` 包版本 3 的重要新功能：

- 支持 Active Directory
- 增强了对 Unicode 的支持
- 全面修改了内部鉴定机制
- 增强了对 Windows 200x 和 XP 打印系统的支持
- 可以将服务器设置为 Active Directory 域的成员服务器
- 采用 NT4 域，从而支持从后者迁移到 Samba 域

---

## Tip

### 迁移到 Samba3

当从 Samba 2.x 迁移到 Samba 3 时有几点应特别注意。Samba HOWTO 文档集中有一章专门讨论此主题。安装 `samba-doc` 包后，可在 `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf` 中找到 HOWTO 文档。

Tip

Samba 使用基于 NetBIOS 服务的 SMB 协议（服务器讯息块）。迫于 IBM 的压力，Microsoft 发布了该协议，这样其他软件制造商能够与 Microsoft 域网络建立连接。使用 Samba 时，SMB 协议在 TCP/IP 协议之上工作，所以必须在所有客户机上安装 TCP/IP 协议。

NetBIOS 是为计算机之间进行通讯而设计的软件接口 (API)。这里提供了一种名称服务。它使连接到网络的计算机能够为自己保留名称，之后便可以根据名称对这些计算机进行寻址。没有任何中心进程来检查这些名称。网络上的任何计算机都可以无数目限制地保留任何未使用的名称。现在可以为不同的网络体系结构实施 NetBIOS 接口。NetBEUI 是与网络硬件结合相对密切的一种实施，但它常被称为 NetBIOS。使用 NetBIOS 实施的网络协议包括 Novell 的 IPX（通过 TCP/IP 的 NetBIOS）和 TCP/IP。

通过 TCP/IP 发送的 NetBIOS 名称与 `/etc/hosts` 中使用的名称或 DNS 定义的名称没有相同之处。NetBIOS 使用它自己的、完全独立的命名约定。但为了方便管理，仍建议您使用与 DNS 主机名对应的名称。Samba 默认采用这种方式。

所有常见操作系统（Mac OS X、Windows 和 OS/2 等）都支持 SMB 协议。必须在所有计算机上安装 TCP/IP 协议。Samba 为多种不同的 UNIX 系统提供客户机。对于 Linux，有一个用于 SMB 的内核模块，它允许在 Linux 系统级别上集成 SMB 资源。

SMB 服务器通过共享为其客户机提供硬件空间。共享包括服务器上的一个目录及其子目录。可以通过名称来导出并访问共享。可以将共享名称设置为任何名称——它不一定是导出目录的名称。也可以为打印机指派一个名称。客户机可以根据打印机的名称来访问打印机。

## 32.1 配置服务器

如果要将 Samba 用作服务器，请安装 `samba`。使用 `rcnmb start && rcsmb start` 和 `rcsmb stop && rcnmb stop` 可启动和停止 Samba 所需的服务。

Samba 的主配置文件是 `/etc/samba/smb.conf`。可以将此文件分为两个逻辑部分。`[global]` 部分包含中央和全局设置。`[share]` 部分包含各个文件和打印机共享。通过这种方式，可以在 `[global]` 部分中有区别地或全局地设置有关共享的详细设置，这样可以提高配置文件的结构透明性。

### 32.1.1 global 部分

需要对 `[global]` 部分的以下参数进行调整以满足网络设置的要求，以便其他计算机能够在 Windows 环境中通过 SMB 访问 Samba 服务器。

**workgroup = TUX-NET** 此行将 Samba 服务器指派到工作组。将 TUX-NET 替换为您的网络环境的适当工作组。您的 Samba 服务器将出现在其 DNS 名称下，除非此名称已被指派给网络中的任何其他计算机。如果 DNS 名称不可用，请使用 `netbiosname=MYNAME` 设置服务器名称。有关此参数的详细信息，请参见 `mansmb.conf`。

**os level = 2** 此参数确定您的 Samba 服务器是否会尝试成为其工作组的 LMB（本地主浏览器）。为了避免现有 Windows 网络受到配置错误的 Samba 服务器的任何影响，应选择非常低的值。有关这一重要主题的详细信息，请参见文件 `BROWSING.txt` 和 `BROWSING-Config.txt`，它们位于包文档的 `textdocs` 子目录下。

如果网络中没有任何其他 SMB 服务器（如 Windows NT 或 2000 服务器），并且您希望 Samba 服务器保留一份本地环境中存在的所有系统的列表，请将 `os level` 设置为一个较高的值（例如 65）。然后便可以选择您的 Samba 服务器作为本地网络的 LMB。

在更改此设置时，应认真考虑这样做对现有 Windows 网络环境的影响。应该首先在一个孤立网络中或一天中的非重要时间测试这些更改。

**wins support** 和 **wins server** 为了将您的 Samba 服务器集成到具有活动 WINS 服务器的现有 Windows 网络中，应启用 `wins server` 选项并将其值设置为 WINS 服务器的 IP 地址。

如果将您的 Windows 计算机连接到单独的子网，同时又希望它们互相通讯，则需要设置一个 WINS 服务器。要将 Samba 服务器转变为这样的 WINS 服务器，请设置选项 `wins support = Yes`。确保网络中只有一个 Samba 服务器启用了此设置。切勿在您的 `smb.conf` 文件中同时启用选项 `wins server` 和 `wins support`。

### 32.1.2 共享

以下示例说明了如何使 CD-ROM 驱动器和用户目录 (`homes`) 对 SMB 客户机可用。

**[cdrom]** 为了避免意外地使 CD-ROM 驱动器变得可用，应使用注释标记（在本例中是分号）取消这些行。去除第一列中的分号，以便与 Samba 共享 CD-ROM 驱动器。

#### 32.1: CD-ROM 共享

```
:[cdrom] ;          comment = Linux CD-ROM ;          path = /media/cdrom ;
```

**[cdrom]** 和 **comment** `[cdrom]` 项是网络上的所有 SMB 客户机均可看到的共享的名称。可以添加一个附加 `comment` 来进一步说明此共享。

`path = /media/cdrom` `path` 导出目录 `/media/cdrom`。

通过严格限制的默认配置，可使这种共享仅对此系统上存在的用户可用。如果应使此共享对所有用户可用，请向配置中添加一行 `guest ok = yes`。此设置为网络上的所有用户提供读权限。建议您认真处理此参数。在 `[global]` 部分使用此参数时更应如此。

[homes] [home] 共享在这里特别重要。如果用户具有 Linux 文件服务器的有效帐户和口令以及自己的主目录，则该用户可以连接到此共享。

## 32.2: 主 共 享

```
[homes] comment = Home Directories valid users = %S browseable = No read only =
```

[homes] 只要没有其他共享使用连接到 SMB 服务器的用户的共享名称，就会使用 [homes] 共享指令动态生成一个共享。所生成的共享的名称与用户名相同。

**valid users = %S** 一旦成功建立连接，就会使用共享的具体名称替换 %S。对于 [homes] 共享，此名称始终与用户名相同。这样就可以将对用户的共享的访问权限严格限制在此用户。

**browseable = No** 此设置使共享在网络环境中不可见。

**read only = No** 默认情况下，Samba 通过 `read only = Yes` 参数来禁止对任何已导出共享的写访问。要使共享可写，请设置值 `read only = No`，它与 `writable = Yes` 是等效的。

**create mask = 0640** 那些不是基于 MS Windows NT 的系统不能理解 UNIX 权限的概念，所以它们在创建文件时不能指派权限。参数 `create mask` 定义了为新创建文件指派的访问权限。这仅适用于可写共享。事实上，此设置意味着拥有者具有读写权限，且拥有者的主组的成员具有读权限。`valid users = %S` 禁止读访问，即使该组具有读权限。要使该组能够进行读或写访问，应取消 `valid users = %S` 一行。

### 32.1.3 安全性级别

SMB 协议来自 DOS 和 Windows 世界，它直接考虑了安全性问题。可以使用口令来保护每个共享访问。SMB 提供了 3 种可能的方式来检查权限：

**共享级安全性 (security = share):** 严格地为一个共享指派一个口令。任何知道此口令的用户都可以访问此共享。

**用户级安全性 (security = user):** 这里将用户的概念引入了 SMB。每个用户都必须使用自己的口令在服务器上注册。注册后，服务器可以根据用户名来授予访问各个已导出共享的权限。

**服务器级安全性 (security = server):** 从客户机来看, Samba 好像是在用户级别方式下工作。但它实际将所有口令查询传递到另一个用户级别方式下的服务器来执行鉴定。此设置需要一个附加参数 (password server =)。

共享、用户和服务级安全性之间的区分适用于整个服务器。无法既为服务器配置的某些共享提供共享级安全性, 同时又为其他共享提供用户级安全性。但是, 您可以为系统上每个已配置的 IP 地址运行单独的 Samba 服务器。

有关此主题的详细信息, 请参见 Samba HOWTO 文档集。对于一个系统上的多个服务器, 应注意选项 `interfaces` 和 `bind interfaces only`。

### Tip

对于 Samba 服务器的简单管理任务, 也可以使用 `swat` 程序。它提供了一个简单的万维网接口, 可用来方便地配置 Samba 服务器。在万维网浏览器中, 打开 `http://localhost:901` 并以 `root` 用户身份登录。但必须在文件 `/etc/xinetd.d/samba` 和 `/etc/services` 中也激活 `swat`。要在 `/etc/xinetd.d/samba` 中执行此操作, 请编辑 `disable` 行, 使其显示为 `disable = no`。有关 `swat` 的详细信息, 请参见手册页。

Tip

## 32.2 将 Samba 用作登录服务器

在主要由 Windows 客户机组成的网络中, 使用户只能使用有效帐户和口令进行注册通常是最好的选择。借助 Samba 服务器可以实现这一点。在基于 Windows 的网络中, 此任务由被配置为主域控制器 (PDC) 的 Windows NT 服务器来处理。例 32.3 “`smb.conf` 中的 `global` 部分” 中显示了必须在 `smb.conf` 的 `[global]` 部分设置的项。

### 32.3: `smb.conf` 中的 `global` 部分

```
[global] workgroup = TUX-NET domain logons = Yes domain master = Yes
```

如果将已加密口令用于校验目的 (这是保持完好的 MS Windows 9x 安装、MS Windows NT 4.0 Service Pack 3 和所有以后版本产品的默认设置), 则 Samba 服务器必须能够处理它们。 `[global]` 部分中的 `encrypt passwords = yes` 项启用了此功能 (对于 Samba 版本 3, 这是默认设置)。此外, 还需要以适合



Windows 的加密格式来准备用户帐户和口令。使用命令 `smbpasswd -a name` 可完成此任务。使用以下命令为计算机创建 Windows NT 域概念要求的域帐户：

### 32.4: 设置计算机帐户

```
useradd hostname\${ smbpasswd -a -m hostname
```

使用 `useradd` 命令可添加一个美元符号。命令 `smbpasswd` 在使用参数 `-m` 时自动插入此符号。带注释的配置示例 (`/usr/share/doc/packages/Samba/examples/smb.conf.SuSE`) 包含自动执行此任务的设置。

### 32.5: 计算机帐户的自动设置

```
add machine script = /usr/sbin/useradd -g nogroup -c "NT Machine Account" \ -s /bin/false %m\${
```

为了确保 Samba 可以正确执行此脚本，应选择具有所需管理员权限的 Samba 用户。为此，请选择一个用户并将其添加到 `ntadmin` 组。然后可以使用以下命令来为属于此 Linux 组的所有用户指派 Domain Admin 状态：

```
net groupmap add ntgroup="Domain Admins" unixgroup=ntadmin
```

有关此主题的详细信息，请参见位于 `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf` 的 Samba HOWTO 文档集的第 12 章。

## 32.3 使用 YaST 配置 Samba 服务器

通过选择将由新的 Samba 服务器控制的工作组或域来启动服务器配置。从下拉菜单‘工作组或域名’中选择一个现有的工作组/域，或输入一个新的工作组/域。下一步指定您的服务器是用作 PDC（主域控制器）还是 BDC（备份域控制器）。

在‘启动’中激活 Samba (图 32.1 “Samba 配置 — 启动”)。使用‘打开防火墙中的端口’和‘防火墙细节’来调整服务器上的防火墙，使得 `netbios-ns`、`netbios-dgm`、`netbios-ssn` 和 `microsoft-ds` 服务的端口在所有外部和内部接口上打开，从而确保 Samba 服务器的平稳运行。

在‘共享’ (图 32.2 “Samba 配置 — 共享”) 中，确定要激活的 Samba 共享。使用‘切换状态’可在‘活动’和‘不活动’之间进行切换。单击‘添加’添加新共享。

在‘标识’ (图 32.3 “Samba 配置 — 标识”) 中，确定与主机关联的域（‘基本设置’）以及是否在网络中使用替代主机名（‘NetBIOS 主机名’）。

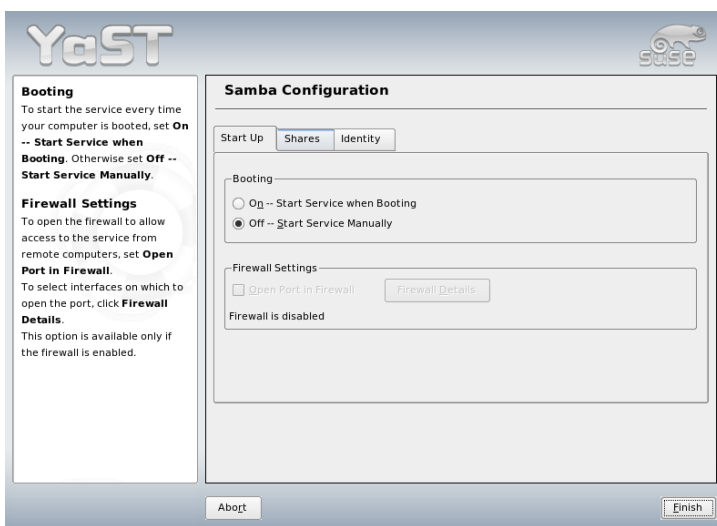


Figure 32.1: Samba 配置—启动

## 32.4 配置客户机

客户机只能通过 TCP/IP 访问 Samba 服务器。NetBEUI 和通过 IPX 的 NetBIOS 不能与 Samba 共用。

### 32.4.1 使用 YaST 配置 Samba 客户机

配置 Samba 客户机来访问 Samba 服务器上的资源（文件或打印机）。在‘Samba 工作组’对话框中输入域或工作组。单击‘浏览’来显示所有可用的组和域，然后可以用鼠标来选择它们。如果激活‘将 SMB 信息也用于 Linux 鉴定’，则用户鉴定将在 Samba 服务器上运行。在完成所有设置后，单击‘完成’完成配置。

### 32.4.2 Windows 9x 和 ME

Windows 9x 和 ME 都内置了对 TCP/IP 的支持。但默认情况下并不安装此支持。要添加 TCP/IP，请转到‘控制面板’→‘系统’，然后选择‘添加’→‘协议’→

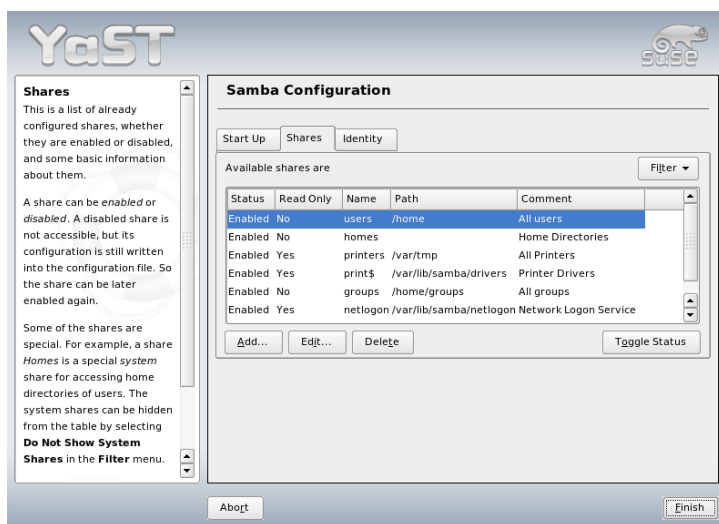


Figure 32.2: Samba 配置—共享

‘Microsoft 的 TCP/IP’。重引导您的 Windows 计算机后，双击网络环境的桌面图标便可以找到 Samba 服务器。

### Tip

要使用 Samba 服务器上的打印机，请安装对应 Windows 版本的标准或 Apple-PostScript 打印机驱动程序。最好将其链接到 Linux 打印机队列，它接受 Postscript 作为一种输入格式。

### Tip

## 32.5 优化

socket options 是您的 Samba 版本附带的示例配置中提供了一种可能的优化方法。它的默认配置涉及本地 Ethernet 网络。有关 socket options 的详细信息，请参考 smb.conf 手册页的相关部分和 socket(7) 的手册页。Samba HOWTO 文档集的 Samba 性能调优一章中提供了更多信息。

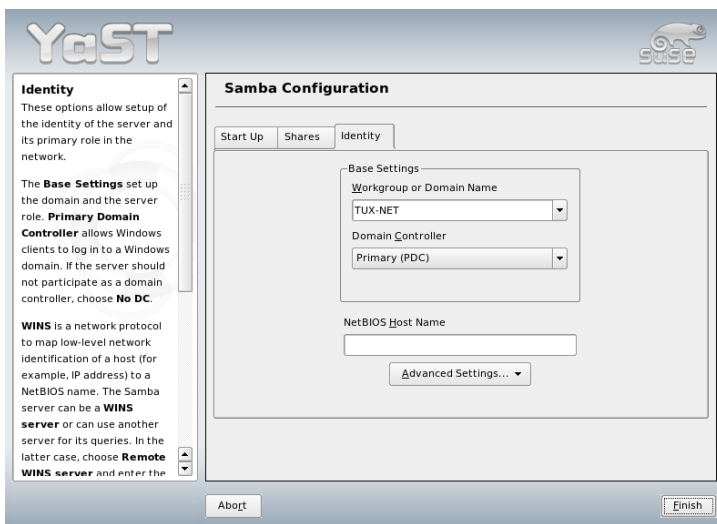


Figure 32.3: Samba 配置—标识

`/etc/samba/smb.conf` 中的标准配置旨在以 Samba 开发小组的默认设置为基础提供有用的设置。但由于网络配置和工作组名等方面的原因，还不能实现立即可用的配置。带注释的示例配置 `examples/smb.conf.SuSE` 中提供的信息可帮助您根据本地要求进行调整。

### Tip

Samba 开发小组提供的 Samba HOWTO 文档集中有一节专门介绍查错。此外，文档的第 V 部分提供了检查配置的逐步指南。

Tip

# 代理服务器 Squid

Squid 是 Linux 和 UNIX 平台广泛使用的代理超速缓存。本章将讨论其配置、运行所需的设置、如何配置系统提供透明代理、如何借助程序（如 Calamaris 和 cachemgr）收集有关使用超速缓存的统计数字，以及如何使用 squidGuard 过滤万维网内容。

33.1	作为代理超速缓存的 Squid . . . . .	484
33.2	有关代理超速缓存的一些事实 . . . . .	484
33.3	系统要求 . . . . .	485
33.4	启动 Squid . . . . .	487
33.5	配置文件 /etc/squid/squid.conf . . . . .	489
33.6	配置透明代理 . . . . .	493
33.7	cachemgr.cgi . . . . .	495
33.8	squidGuard . . . . .	497
33.9	使用 Calamaris 生成超速缓存报告 . . . . .	498
33.10	参考信息 . . . . .	499

## 33.1 作为代理超速缓存的 Squid

Squid 可以充当代理超速缓存。它将来自客户程序（这里指来自万维网浏览器）的对象请求重定向至服务器。当服务器回复所请求的对象后，它会将这些对象传递给客户程序并在硬盘超速缓存中保存对象副本。超速缓存的一个优点就是：当多个客户程序请求同一对象时，可以从硬盘超速缓存中提供该对象。这样客户程序接收数据的速度要比从因特网接收快得多。此过程还可以减少网络流量。

除实际的超速缓存外，Squid 还提供众多功能，如在代理服务器的相互通讯的层次之间分配负载、为所有访问代理的客户程序定义严格的访问控制列表、借助其他应用程序允许或拒绝访问特定的万维网网页，以及生成有关频繁访问的万维网网页的统计数字以评估用户的浏览习惯。Squid 不是通用代理。它通常只充当 HTTP 连接的代理。它确实还支持 FTP、Gopher、SSL 和 WAIS 等协议，但不支持其他的因特网协议，如 Real Audio、新闻或视频会议。由于 Squid 只支持使用 UDP 协议在不同的超速缓存间通讯，所以很多其他多媒体程序都不受支持。

## 33.2 有关代理超速缓存的一些事实

### 33.2.1 Squid 和安全性

Squid 可以与防火墙结合起来，通过使用代理超速缓存防止内部网络遭受外部攻击。防火墙会拒绝 Squid 之外的所有客户程序访问外部服务。所有万维网连接都必须通过代理方式建立。

如果防火墙配置中包含 DMZ，代理应该在此区域内操作。在这种情况下，DMZ 内的所有计算机都应将其日志文件发送到安全网络内的主机，这一点至关重要。实施透明代理的各种可能情况将在第 33.6 节“配置透明代理”中介绍。

### 33.2.2 多个超速缓存

可以对多个代理进行配置，以便在它们之间交换对象。这样会减少系统负载，同时提高找到本地网络中已有对象的几率。还可以配置超速缓存的层次，以便能够将对象请求转发给同级超速缓存或父超速缓存——使其从本地网络中的其他超速缓存或直接从数据源获取对象。

为了不给网络增加总体数据流量，为超速缓存层次选择适当的拓扑结构是十分重要的。对于超大型网络，合理的做法是：为每个子网配置一个代理服务器并将其连接至父代理，再通过父代理连接至 ISP 的代理超速缓存。

所有这些通讯都通过在 UDP 协议之上运行的 ICP（因特网超速缓存协议）来处理。超速缓存间的数据传送使用基于 TCP 的 HTTP（超文本传送协议）来处理。

要找到从中可获得对象的最合适的服务器，一个超速缓存会向所有同级代理发送 ICP 请求。同级代理会通过 ICP 响应回复请求（如果检测到对象就回复 HIT 代码，如果未检测到则回复 MISS 代码）。如果发现多个 HIT 响应，代理服务器会根据哪个超速缓存回复最快或哪个最近等因素决定从哪个服务器下载。如果没有收到满意的响应，该请求将被发送至父超速缓存。

### Tip

为了避免网络中不同超速缓存中的对象重复，还会使用其他 ICP 协议，如 CARP（超速缓存阵列路由协议）或 HTCP（超文本超速缓存协议）。网络中维护的对象越多，找到所需对象的可能性就越大。

### Tip

## 33.2.3 超速缓存因特网对象

网络中的对象并不全都是静态的。网络中有许多动态生成的 CGI 页面、访问计数器 and 加密的 SSL 内容文档。由于每次访问这类对象时它们都会更改，所以它们不会被超速缓存。

一直以来的问题是：储存在超速缓存中的所有其他对象应在其中保留多久。要确定保留时间，超速缓存中的所有对象都会被指派几种可能状态之一。万维网和代理服务器会通过为这些对象添加报头找出对象的状态（如“Last modified”或“Expires”）以及相应的日期。同时还会使用其他报头指定不能超速缓存对象。

超速缓存中的对象通常会因为缺少可用硬盘空间而使用 LRU（最近最少使用）之类的算法进行替换。一般来说，这意味着代理会销毁未被请求时间最长的对象。

## 33.3 系统要求

最重要的事情是确定系统必须承受的最大负载。由于负载峰值可能是日均值的四倍，因此要特别注意负载峰值。如果不能确定，最好高估系统要求，因为让 Squid 在接近其处理能力限值的状态下工作可能会严重影响其服务质量。以下几节按重要程度依次阐述了各个系统要素。

### 33.3.1 硬盘

速度在超速缓存过程中起到重要作用，所以此要素值得特别关注。对于硬盘，此参数通过以毫秒衡量的*随机搜索时间*来描述。Squid 从硬盘读取或写入硬盘的数据块一般都较小，因此硬盘的搜索时间比其数据吞吐量更重要。如果要考虑代理的话，高转速硬盘可能会是更好的选择，因为高转速硬盘允许读写磁头更快定位到所需位置。使系统加速的一种可能办法是同时使用多个磁盘或采用分带 RAID 阵列。

### 33.3.2 磁盘超速缓存的大小

在小型超速缓存中，HIT（在其中找到所请求的对象）的概率会很小，因为该超速缓存很容易被占满，所以较少请求的对象很快被较新的请求对象替代。例如，如果超速缓存的可用空间为 1GB，而用户每天只浏览 10MB，那么占满超速缓存至少要 100 天。

确定所需超速缓存大小的最简便方法就是参考连接的最大传送速度。1 Mbit/s（兆位/秒）连接的最大传送速度为 125 KB/s。如果所有流量都进入超速缓存，1 小时累计可达 450 MB；假设所有流量都是在 8 小时工作时间之内产生的，那么每天将达到 3.6 GB。由于连接速度一般不会达到流量上限，所以可以认为超速缓存处理的数据总量约为 2 GB。这就是为什么要在 Squid 示例中使用 2 GB 的磁盘空间来保证一天的浏览数据都能超速缓存。

### 33.3.3 RAM

Squid 所需内存 (RAM) 大小直接与超速缓存中的对象数有关。Squid 还会在主存储器中储存超速缓存对象引用和经常请求的对象，以加速对这些数据的检索。随机存储器比硬盘快得多。

除此之外，Squid 还要在内存中保存其他数据，如：所有已处理 IP 地址的表、准确域名超速缓存、最常请求的对象、访问控制列表、缓冲区等等。

拥有足够的内存对于 Squid 进程非常重要，因为如果必须交换到磁盘的话，系统性能会显著降低。可以使用 `cachemgr.cgi` 工具来管理超速缓存内存。该工具在第 33.7 节“`cachemgr.cgi`”中有详细介绍。

### 33.3.4 CPU

Squid 并不是需要大量使用 CPU 的程序。处理器的负载只会在装载或检查超速缓存内容时才会增加。使用多处理器计算机并不会提高系统性能。要提高效率，最好是购买速度更快的硬盘或增加内存。



## 33.4 启动 Squid

Squid 在 SUSE LINUX 中已预先配置，因此安装后即可启动。为保证顺利启动，应该对网络进行配置，使其至少能连接一个名称服务器和因特网。如果拨号连接使用动态 DNS 配置，则可能出现的问题。在这种情况下，至少应该明确输入名称服务器，因为如果在 `/etc/resolv.conf` 中找不到 DNS 服务器，Squid 便不会启动。

### 33.4.1 用于启动和停止 Squid 的命令

要启动 Squid，在命令行中以 root 身份输入 `rcsquid start`。如果是首次启动，必须首先在 `/var/squid/cache` 中定义目录结构。启动脚本 `/etc/init.d/squid` 会自动进行定义，该过程可能需要几秒钟甚至几分钟的时间。如果右侧显示绿色的完成，表明已成功装载 Squid。要在本地系统上测试 Squid 的功能，请在浏览器中输入 `localhost` 作为代理，输入 `3128` 作为端口。

要允许所有用户访问 Squid 并通过它访问因特网，需要将配置文件 `/etc/squid/squid.conf` 中的项 `http_access deny all` 改为 `http_access allow all`。但在这样做时，要考虑到此操作会让所有人都不受任何限制地访问 Squid。因此，应定义控制访问代理的 ACL。有关此内容的详细信息，请参考第 33.5.2 节“访问控制选项”。

修改配置文件 `/etc/squid/squid.conf` 后，Squid 必须重装该配置文件。可通过 `rcsquid reload` 执行此操作。或者，通过 `rcsquid restart` 将 Squid 彻底重启动。

可以使用命令 `rcsquid status` 来检查代理是否正在运行。使用命令 `rcsquid stop` 将关闭 Squid。这需要一些时间，因为 Squid 在断开同客户程序的连接并将其数据写入磁盘前会等候最多半分钟（`/etc/squid/squid.conf` 中的 `shutdown_lifetime` 选项）。

#### Warning

##### 终止 Squid

使用 `kill` 或 `killall` 终止 Squid 可能会损坏超速缓存。要能够重新启动 Squid，必须删除损坏的超速缓存。

#### Warning

如果 Squid 在成功启动后不久就终止，请检查名称服务器项是否有误或者是缺少 `/etc/resolv.conf` 文件。Squid 会在 `/var/squid/logs/cache.log` 文件中记录启动失败的原因。如果应该在系统引导时自动装载 Squid，请使用

YaST 运行级别编辑器激活所需的 Squid 运行级别。请参见第2.8.7节 “Systems Services (RunLevel)”

卸载 Squid 并不会删除超速缓存层次或日志文件。要删除这些内容，请手工删除 `/var/cache/squid` 目录。

### 33.4.2 本地 DNS 服务器

建立本地 DNS 服务器很有意义，即便并不用它来管理自己的域。它仅起到超速缓存专用名称服务器的作用，并且可以在无需任何特殊配置的情况下通过根名称服务器解析 DNS 请求（请参见第24.2节 “启动名称服务器 BIND”）。如何完成上述操作，取决于您在配置因特网连接的过程中是否选择了动态 DNS。

**动态 DNS** 使用动态 DNS 时，因特网服务提供商通常在建立因特网连接过程中设置 DNS 服务器，并自动调整本地文件 `/etc/resolv.conf`。这是通过将 `sysconfig` 变量 `MODIFY_RESOLV_CONF_DYNAMICALLY` 设置为 `YES` 来实现的。通过 YaST `sysconfig` 编辑器将此变量设置为 `NO`（请参见第7.8节 “YaST `sysconfig` 编辑器”）。然后在文件 `/etc/resolv.conf` 中输入本地 DNS 服务器（`localhost` 的 IP 地址是 `127.0.0.1`）。这样 Squid 一启动就能找到本地名称服务器。

为了使服务提供商的名称服务器可访问，必须在配置文件 `/etc/named.conf` 中的 `forwarders` 下输入其名称及 IP 地址。使用动态 DNS，通过将 `sysconfig` 变量 `MODIFY_NAMED_CONF_DYNAMICALLY` 设置为 `YES`，可以在建立连接时自动执行上述操作。

**静态 DNS** 使用静态 DNS 时，在建立连接时将不执行任何有关 DNS 的自动调整。因此不需要更改任何 `sysconfig` 变量，但必须根据上面的介绍在文件 `/etc/resolv.conf` 中输入本地 DNS 服务器。此外，必须在文件 `/etc/named.conf` 中的 `forwarders` 下手工输入服务提供商的静态名称服务器及其 IP 地址。

---

#### Tip

##### DNS 和防火墙

如果运行了防火墙，应确保 DNS 请求能够通过。

---

Tip

## 33.5 配置文件 /etc/squid/squid.conf

所有 Squid 代理服务器设置都在 /etc/squid/squid.conf 文件中进行。首次启动 Squid 时，不必在此文件中进行任何更改，但是外部客户程序最初不具备访问权。代理可供 localhost 使用。默认端口为 3128。预装的 /etc/squid/squid.conf 提供了有关选项的详细信息和许多示例。几乎所有项都以 # 开头（各行都标有注释）并且在行尾可找到相关说明。给定值几乎总与默认值相关，因此多数情况下，仅删除注释符号而不更改任何参数实际上没有什么影响。如果可能，保持示例不变，并将选项连同修改的参数一起插入下一行。这样就很容易解释默认值和更改了。

### Tip

#### 更新后调整配置文件

如果已从较早的 Squid 版本更新，建议编辑新的 /etc/squid/squid.conf，并只应用以前文件中的更改。如果试图执行旧的 squid.conf，配置文件可能不起作用，因为有时会修改选项并添加新的更改。

### Tip

### 33.5.1 常规配置选项（选择）

**http\_port 3128** 这是 Squid 监听客户程序请求所用的端口。默认端口为 3128，但也常使用 8080。如果需要，可指定多个以空格分隔的端口号。

**cache\_peer <hostname> <type> <proxy-port> <icp-port>**

在此输入父代理（如果您想使用 ISP 的代理）。在 <hostname> 中输入要使用代理的名称和 IP 地址，在 <type> 中输入 parent。对于 <proxy-port>，输入同样是由父代理运营商设置的在浏览器中使用的端口号，通常为 8080。如果父代理的 ICP 端口未知并且该端口的使用与提供商无关，请将 <icp-port> 设为 7 或 0。此外，端口号后应指定 default 和 no-query 以禁止使用 ICP 协议。借助提供商的代理，Squid 就可以像普通浏览器那样操作了。

**cache\_mem 8 MB** 此项定义 Squid 可用于超速缓存的内存大小。默认为 8 MB。

**cache\_dir ufs /var/cache/squid/ 100 16 256**

cache\_dir 项定义在磁盘上储存所有对象的目录。末尾的数字表示可用的最大磁盘空间（以 MB 为单位）以及第一级和第二级目录数。不要改动 ufs

参数。默认情况下，在 `/var/cache/squid` 目录内占用 100 MB 磁盘空间，并在该目录内创建 16 个子目录，每个又可以再包含 256 个子目录。指定要使用的磁盘空间时，应预留足够的磁盘空间。在此最为合理的值应该是可用磁盘空间的 50%（最小）到 80%（最大）。在增大目录的后两个数字时一定要小心，因为目录过多也可能导致性能问题。如果有多个磁盘共享超速缓存，请输入多个 `cache_dir` 行。

**cache\_access\_log /var/log/squid/access.log**

这是日志讯息的路径。

**cache\_log /var/log/squid/cache.log** 这是日志讯息的路径。

**cache\_store\_log /var/log/squid/store.log**

这是日志讯息的路径。

这三项指定 Squid 记录其所有操作的路径。通常不做任何更改。如果 Squid 因使用频繁而负担过重，则可能需要将超速缓存和日志文件分散到多个磁盘上。

**emulate\_httpd\_log off** 如果该项设置为 *on*，则可以获取可读的日志文件。但有一些评估程序不能对此作出解释。

**client\_netmask 255.255.255.255** 此项可以屏蔽日志文件中的 IP 地址，以便隐藏客户机身份。如果在此输入 `255.255.255.0`，IP 地址的最后一位将被设为 0。

**ftp\_user Squid@** 使用此项可以设置 Squid 执行匿名 FTP 登录时应使用的口令。在此可以指定有效的电子邮件地址，因为有些 FTP 服务器需要通过这种方式来验证有效性。

**cache\_mgr webmaster** 一个电子邮件地址，Squid 在意外崩溃时会向该地址发送信件。默认为 *webmaster*。

**logfile\_rotate 0** 如果运行 `squid -k rotate`，Squid 可以循环使用受保护的日志文件。在此过程中会给文件编号，并且在达到指定值后覆盖最早文件。默认值为 0，因为 SUSE LINUX 中日志文件的存档和删除是由配置文件 `/etc/logrotate/squid` 中设置的 cron job（定时执行的作业）完成的。

**append\_domain ;domain;** 使用 `append_domain` 可指定自动追加的域（如果没有指定域）。通常，在此输入的是您自己的域，所以在浏览器中输入 *www* 将访问您自己的万维网服务器。

**forwarded\_for on** 如果将此项设置为关，Squid 会将客户机的 IP 地址和系统名称从 HTTP 请求中删除。

**negative\_ttl 5 minutes; negative\_dns\_ttl 5 minutes**

一般不必更改这些值。但如果使用拨号连接，因特网有时可能无法访问。Squid 会记录失败的请求并拒绝发出新的请求，即便重新建立因特网连接也无济于事。在这种情况下，将分钟改为秒钟，然后单击浏览器中的重载，拨号进程会在几秒钟后重新启动。

**never\_direct allow** *(acl\_name)* 要防止 Squid 直接从因特网接受请求，应使用上述命令强制连接到另一个代理。事先必须已在 *cache\_peer* 中输入该代理。如果将 *(acl\_name)* 指定为 *all*，会强制所有请求直接转发给父代理。有时这可能是必要的，例如在您的提供商严格规定使用它的代理或拒绝通过其防火墙直接访问因特网时。

### 33.5.2 访问控制选项

Squid 为控制针对代理的访问提供了一套周密的系统。通过实施 ACL 可以轻松并全面地进行配置。这涉及一些依次处理的规则的列表。使用 ACL 之前必须先定义 ACL。一些默认的 ACL 已经存在，如 *all* 和 *localhost*。但是，仅仅定义 ACL 并不意味着实际应用 ACL。只有在与 *http\_access* 规则一同使用时才不是这样。

**acl** *jacl\_name* *jtype* *jdata* ACL 至少需要三个值来定义。名称 *jacl\_name* 可以任意选择。对于 *jtype*，可以在多种不同的选项中选择（在 */etc/squid/squid.conf* 文件的 *ACCESS CONTROLS* 部分中可以找到这些选项）。*jdata* 的值取决于各 ACL 的类型，并且可以从文件中读取（例如，通过主机名、IP 地址或 URL）。以下是一些简单的示例：

```
acl mysurfers srcdomain .my-domain.com acl teachers src 192.168.1.0/255.255.255
```

**http\_access allow** *jacl\_name* *http\_access* 定义谁可以使用代理，以及谁能够访问因特网上的什么内容。为此必须指定 ACL。上面已经定义了 *localhost* 和 *all*，这两个 ACL 可以通过 *deny* 或 *allow* 相应地拒绝或允许访问。可以创建一个包含任何数量 *http\_access* 项的列表，按从上到下的顺序处理各个项，并且根据出现的先后顺序允许或拒绝访问相应的 URL。最后一项应始终是 *http\_access deny all*。在下例中，*localhost* 可随意访问任何内容，而其他所有主机全部被拒绝访问。

```
http_access allow localhost http_access deny all
```

在另外一个使用这些规则的示例中，*teachers* 组总能访问因特网。*students* 组只能在星期一到星期五的午餐时间访问。

```
http_access deny localhost http_access allow teachers http_access allow
```

为提高可读性，只应该在 `/etc/squid/squid.conf` 文件的指定位置输入带有 `http_access` 项的列表。即在文本

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR # CLIENTS
```

和最后的

```
http_access deny all 之间
```

### **redirect\_program /usr/bin/squidGuard**

使用此选项可以指定重定向器（如 `squidGuard`），允许拦截不需要的 URL。通过代理鉴定和适当的 ACL 可以控制不同的用户组访问因特网。`squidGuard` 是一个可以安装和配置的独立软件包。

### **auth\_param basic program /usr/sbin/pam\_auth**

如果必须在代理上鉴定用户，请设置一个相应的程序（如 `pam_auth`）。当首次访问 `pam_auth` 时，用户会看到一个用于输入用户名和口令的登录窗口。此外，仍然需要 ACL，只允许提供有效登录信息的客户机使用因特网：

```
acl password proxy_auth REQUIRED
```

```
http_access allow password http_access deny all
```

可以用授权用户名的列表或指向此类列表的路径来替换 `proxy_auth` 后的 `REQUIRED`。

### **ident\_lookup\_access allow jacl\_name;**

使用此选项，可以为 ACL 定义的所有客户机都运行 `ident` 请求以查找各个用户的身份。如果对 `jacl_name;` 应用了 `all`，此选项对所有客户机都有效。另外，必须在所有客户机上运行 `ident` 守护程序。对于 Linux，可为此安装 `pidentd` 软件包。对于 Windows，可从因特网上下载免费软件。为确保只有成功进行 `ident` 查找的客户机才有权访问，请在此定义相应的 ACL：

```
acl identhsts ident REQUIRED
```

```
http_access allow identhsts http_access deny all
```

这里同样需要用授权用户名列表来替换 `REQUIRED`。使用 `ident` 会明显延缓访问时间，因为每个请求都要重复进行 `ident` 查找。

## 33.6 配置透明代理

以下为使用代理服务器的一般方式：万维网浏览器向代理服务器的特定端口发送请求，然后代理会提供这些所需的对象，无论它们是否在代理的超速缓存中。在网络中使用时，可能出现以下几种情况：

- 出于安全考虑，建议所有客户程序都使用代理来浏览因特网。
- 所有客户程序都必须使用代理，无论客户程序是否清楚这一点。
- 网络中的代理已转移，但是现有的客户程序应保留原有配置。

在所有这些情况下，都可以使用透明代理。原理很简单：代理截获并回复万维网浏览器的请求，所以万维网浏览器接收到所请求的页面，但并不知道它们来自何处。正如名称中指出的那样，整个处理过程完全是透明的。

### 33.6.1 内核配置

首先，确定代理服务器的内核支持透明代理。已对 SUSE LINUX 提供的内核进行了相应地配置。如果不支持，向内核添加以下选项并重新编译内核。有关详细信息，请参考第9章 Linux 内核。

### 33.6.2 /etc/squid/squid.conf 中的配置选项

要启动并运行透明代理，需在 /etc/squid/squid.conf 文件中激活的选项包括：

- `httpd_accel_host virtual`
- `httpd_accel_port 80`  
The port number where the actual HTTP server is located
- `httpd_accel_with_proxy on`
- `httpd_accel_uses_host_header on`

### 33.6.3 使用 SuSEfirewall2 配置防火墙

现在借助端口转发规则，通过防火墙将所有入站请求重定向到 Squid 端口。可使用附带的 SuSEfirewall2 工具完成此操作。可以在 `/etc/sysconfig/SuSEfirewall2` 中找到其配置文件。配置文件的项已进行适当注释。即便只设置透明代理，也必须配置一些防火墙选项：

- 指向因特网的设备： `FW_DEV_EXT="eth1"`
- 指向网络的设备： `FW_DEV_INT="eth0"`

定义防火墙上从不可信的（外部）网络（如因特网）访问的端口和服务（请参见 `/etc/services`）。在下例中，仅对外部提供万维网服务：

```
FW_SERVICES_EXT_TCP="www"
```

定义防火墙上从安全（内部）网络访问的端口或服务（请参见 `/etc/services`），包括通过 TCP 和 UDP：

```
FW_SERVICES_INT_TCP="domain www 3128" FW_SERVICES_INT_UDP="domain"
```

这会允许访问万维网服务和 Squid（Squid 的默认端口为 3128）。服务“域”代表 DNS（域名服务）。此服务很常用。如果不需要，只需将其从上面的项中删除并将下面的选项设置为 `no`：

```
FW_SERVICE_DNS="yes"
```

最重要的选项是数字 15：

#### 33.1: 防火墙配置：选项 15

```
## 15.) # Which accesses to services should be redirected to a local port # on the f
```

上面的注释显示了需要遵循的语法。首先，输入访问代理防火墙的内部网络的 IP 地址和网络掩码。其次，输入这些客户程序请求发往的 IP 地址和网络掩码。如果使用的是万维网浏览器，请指定网络 0/0（表示“至任意地址”的通配符）。之后，输入这些请求最初发送到的端口以及所有这些请求最终要重定向到的端口。由于 Squid 能够支持 HTTP 以外的更多协议，可将请求从其他端口重定向至代理，如 FTP（端口 21）、HTTPS、或 SSL（端口 443）。在本例中，万维网服务（端口 80）重定向至代理端口（端口 3128）。如果要添加更多网络或服务，必须在对应项中用空格分隔它们。



```
FW_REDIRECT_TCP="192.168.0.0/16,0/0,80,3128 192.168.0.0/16,0/0,21,3128" FW_REDIRECT_UDP="192.1
```

要启动防火墙及其新配置，请更改 `/etc/sysconfig/SuSEfirewall2` 文件的项。必须将 `START_FW` 项设置为 `"yes"`。

按第33.4节“启动 Squid”所述启动 Squid。要了解是否一切正常，请检查 `/var/log/squid/access.log` 中的 Squid 日志。

要证明所有端口都已正确配置，请从网络外的任意计算机执行本机上的端口扫描。只有万维网服务（端口 80）应该是打开的。要使用 `nmap` 扫描端口，命令语法为 `nmap -O IP_address`。

## 33.7 cachemgr.cgi

超速缓存管理器 (`cachemgr.cgi`) 是一个 CGI 实用程序，用于显示正运行的 Squid 进程占用内存的相关统计数字。这也是在不登录服务器的情况下，管理超速缓存和查看统计数字的一种更便捷的方式。

### 33.7.1 设置

首先，必须在系统上运行万维网服务器。要检查 Apache 是否已在运行，以 `root` 身份输入命令 `rcapache status`。如果显示如下讯息：

```
Checking for service httpd:OK Server uptime:1 day 18 hours 29 minutes 39 seconds
```

表示 Apache 正在该计算机上运行。如果未运行，则输入 `rcapache start` 使用 SUSE LINUX 默认设置启动 Apache。最后一个设置步骤是将文件 `cachemgr.cgi` 复制到 Apache 目录 `cgi-bin`：

```
cp /usr/share/doc/packages/squid/scripts/cachemgr.cgi /srv/www/cgi-bin/
```

### 33.7.2 `/etc/squid/squid.conf` 中的超速缓存管理器 ACL

超速缓存管理器所需的原文件中有一些默认设置。第一个 ACL 最为重要，因为超速缓存管理器要通过 `cache_object` 协议尝试与 Squid 通讯。

```
acl manager proto cache_object acl localhost src 127.0.0.1/255.255.255.255
```

还应包含以下规则：

```
http_access allow manager localhost http_access deny manager
```

随后的规则假定万维网服务器和 Squid 运行在同一台计算机上。如果超速缓存管理器与 Squid 间的通讯是另一台计算机上的万维网服务器发出的，应如例33.2“访问规则”所示包含额外的 ACL。

### 33.2: 访问规则

```
acl manager proto cache_object acl localhost src 127.0.0.1/255.255.255.255 acl webserver
```

然后添加例33.3“访问规则”中的规则。

### 33.3: 访问规则

```
http_access allow manager localhost http_access allow manager webserver http_
```

为管理器配置口令以访问更多选项，如远程关闭超速缓存或查看有关超速缓存的更多信息。为此，应配置项 `cachemgr_passwd`，设置用于管理器和可查看选项列表的口令。此列表在 `/etc/squid/squid.conf` 中显示为项注释的一部分。

每次一更改配置文件，就应重启动 Squid。使用 `rcsquid reload` 可以轻松地重启动。

### 33.7.3 查看统计数字

访问相应的万维网站点——<http://webserver.example.org/cgi-bin/cachemgr.cgi>。按“继续”来浏览不同的统计数字。有关超速缓存管理器显示的每个项的更多详细信息，请参见 <http://www.squid-cache.org/Doc/FAQ/FAQ-9.html> 上的 Squid FAQ（常见问题解答）。

## 33.8 squidGuard

本节的目的并不是要解释 squidGuard 的详细配置，而只是介绍该程序并为使用该程序提些建议。要深入了解配置问题，请参考 squidGuard 的万维网站点 <http://www.squidguard.org>。

squidGuard 是一款用于 Squid 的免费 (GPL)、灵活而快捷的过滤器、重定向器和访问控制器插件。使用它可以针对 Squid 超速缓存定义多种访问规则，对不同用户组加以不同的限制。squidGuard 使用 Squid 的标准重定向接口。

squidGuard 可以完成以下任务：

- 限制某些用户的万维网访问，使其只能访问一组可接受的或知名的万维网服务器或 URL。
- 防止某些用户访问某些列出的或在黑名单中列出的万维网服务器或 URL。
- 防止某些用户访问与一组正则表达式或单词匹配的 URL。
- 将拦截的 URL 重定向至基于 CGI 的“智能”信息页面。
- 将未注册用户重定向至注册表单。
- 将横幅重定向至空白 GIF。
- 使用基于时间、周中各天、日期等的不同访问规则。
- 对不同用户组使用不同规则。

squidGuard 和 Squid 不能用于：

- 编辑、过滤或审查文档内的文本。
- 编辑、过滤或审查 HTML 嵌入脚本语言，如 JavaScript 或 VBscript。

在使用 squidGuard 之前，请先进行安装。提供最小的配置文件，如 `/etc/squidguard.conf`。可在 <http://www.squidguard.org/config/> 中找到配置示例。以后可尝试更为复杂的配置设置。

接下来，如果客户程序请求列在黑名单中的万维网站点，则创建一个虚拟的“拒绝访问”页面或相对复杂的 CGI 页面来重定向 Squid。强烈建议使用 Apache。

现在，配置 Squid 以使用 squidGuard。使用 `/etc/squid/squid.conf` 文件中的以下项：

```
redirect_program /usr/bin/squidGuard
```

另一个名为 `redirect_children` 的选项会配置计算机上运行的“重定向”（在此为 `squidGuard`）进程的个数。`squidGuard` 运行足够快，能够处理很多请求：在一台提供 5900 个域和 7880 个 URL（总计 13,780 个）的 Pentium 500 MHz 计算机上，不到 10 秒就能够处理 100,000 次请求。因此，不建议设置四个以上的进程，因为进程的分配会过多的占用内存。

```
redirect_children 4
```

最后，通过运行 `rcsquid reload` 让 Squid 装载新配置。现在，可以通过浏览器测试这些设置。

## 33.9 使用 Calamaris 生成超速缓存报告

Calamaris 是一个 Perl 脚本，用来以 ASCII 或 HTML 格式生成超速缓存活动的报告。它可以处理本机 Squid 访问日志文件。Calamaris 的主页为 <http://Calamaris.Cord.de/>。该程序很方便使用。

以 root 身份登录，然后输入 `cat access.log.files | calamaris <options> > reportfile`。在通过管道输出一个以上日志文件时，日志文件要按时间先后排列，较早的文件先输出，这一点很重要。该程序有一些选项：

- a 输出所有可用报告
- w 以 HTML 格式输出报告
- l 在报告标题处包含讯息或图徽

有关不同选项的详细信息，可通过 `man calamaris` 在该程序的手册页中找到。典型示例如下：

```
cat access.log.2 access.log.1 access.log | calamaris -a -w \ > /usr/local/httpd/htdocs
```

这会将报告放入万维网服务器目录。需要通过 Apache 来查看这些报告。

另一个功能强大的超速缓存报告生成器工具是 SARG（Squid 分析报告生成器）。详细信息请参见 <http://web.onda.com.br/orso/>。

## 33.10 参考信息

访问 Squid 的主页 <http://www.squid-cache.org/>。在此可找到“Squid 用户指南”及有关 Squid 的大量 FAQ（常见问题解答）信息。

安装该程序后，在 `howtoen` 中有一个关于透明代理的小的使用说明文件 `/usr/share/doc/howto/en/txt/TransparentProxy.gz`。此外还通过 `squid-users@squid-cache.org` 提供 Squid 的邮件列表。其存档文件位于 <http://www.squid-cache.org/mail-archive/squid-users/> 中。



# Part IV

## 管 理





# Linux 中的安全性

伪装功能和防火墙可以有效地控制数据流和数据交换。SSH（安全 Shell）使您可以通过加密连接登录到远程主机中。对文件或整个分区加密可以在第三方未经授权访问您的系统时保护您的数据。除了这些纯粹的技术指导，最后一节还强调了 Linux 网络中的安全问题。

34.1	伪装和防火墙 . . . . .	504
34.2	SSH：安全的网络操作 . . . . .	512
34.3	对分区和文件进行加密 . . . . .	516
34.4	安全性和机密性 . . . . .	518

## 34.1 伪装和防火墙

只要在联网环境中使用 Linux，您就可以使用内核功能通过操纵网络包将内部网络区域和外部网络区域隔开。Linux netfilter 框架提供了一种建立有效防火墙的方法，可以将不同网络隔开。借助 iptables — 用于定义规则集的通用表结构 — 可以精确控制哪些包能通过网络接口。使用 SuSEfirewall2 和相应的 YaST 模块，您可以轻而易举地设置这种包过滤器。

### 34.1.1 使用 iptables 过滤包

部件 netfilter 和 iptables 负责网络包的过滤和操纵以及网络地址转换 (NAT)。过滤准则及与过滤准则关联的所有操作均储存在链中；各个网络包在到达时，必须依次与这些链进行匹配。要匹配的链储存在表中。使用 iptables 命令可以更改这些表和规则集。

Linux 内核维护以下三个表，分别对应包过滤器的不同功能：

**filter** 此表储存大多数过滤规则，因为它执行严格意义上的包过滤机制，例如，决定是让包通过 (ACCEPT) 还是将包丢弃 (DROP)。

**nat** 此表定义对包的源地址和目标地址所做的任何更改。使用这些功能还能实现伪装，这是 NAT 的一个特例，用于将专用网络与因特网链接起来。

**mangle** 此表中的规则用于操纵 IP 报头中储存的值（如服务类型）。

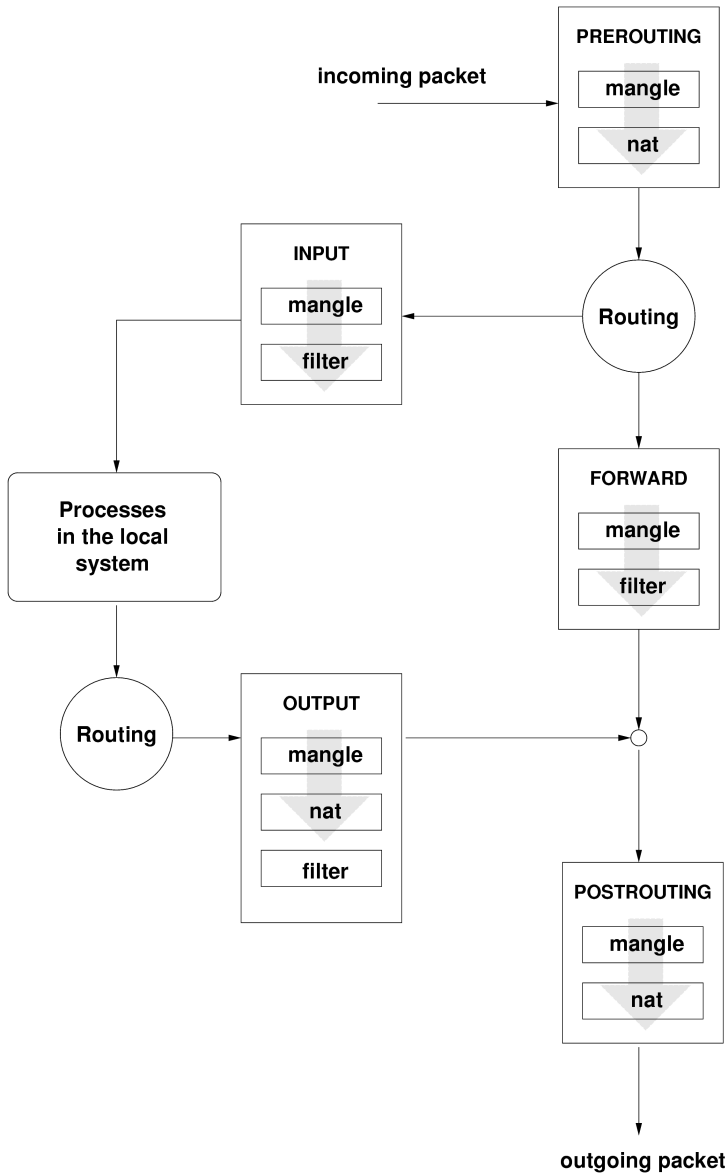


Figure 34.1: iptables: 数据包的可能路径

上述各表包含多个用于匹配包的预定义链。

**PREROUTING** 此链适用于入站包。

**INPUT** 此链适用于发往系统内部进程的包。

**FORWARD** 此链适用于仅在系统中路由的包。

**OUTPUT** 此链适用于从系统自身发出的包。

**POSTROUTING** 此链适用于所有出站包。

图34.1 “iptables: 数据包的可能路径” 演示了网络包在特定系统中传送时可能经过的路径。为了便于说明, 图中将表作为链的各个部分列出, 但实际上表本身储存了这些链。

在所有可能的情况中最简单的情况是: 发往系统本身的入站包到达 eth0 接口。数据包首先要转到 mangle 表的 PREROUTING 链, 然后转到 nat 表的 PREROUTING 链。随后的步骤 (涉及包的路由选择) 确定包的最终目标, 这是系统自身的过程。在包经过 mangle 表和 filter 表的 INPUT 链后, 只要与 filter 表的规则确实匹配, 那么包将最终到达目标。

### 34.1.2 关于伪装的基础知识

伪装是 Linux 特有的一种 NAT (网络地址转换) 形式。通过伪装可以将小型 LAN (其中的主机使用专用地址范围中的 IP 地址 — 请参见第22.1.2节 “网络掩码和路由”) 与因特网 (使用正式的 IP 地址) 连接起来。为使 LAN 主机能连接到因特网, 需要将其专用地址转换为正式地址。这种转换是在路由器上完成的, 路由器充当了 LAN 和因特网之间的网关。基本原理很简单: 路由器包含多个网络接口, 通常包含一个网卡和一个与因特网连接的独立接口。后者将路由器与外部世界链接起来, 同时, 还会有一个或多个其他网络接口将路由器与 LAN 主机链接起来。在本地网络中的这些主机连接到路由器的网卡 (如 eth0) 后, 它们就可以将发往本地网络之外的所有包发送到其默认网关或路由器。

#### Important

使用正确的网络掩码

在配置网络时, 确保所有本地主机的广播地址和网络掩码都相同。做不到这一点就会因为无法正确路由数据包而断开网络连接。

Important

如上所述，只要有某台 LAN 主机要向因特网地址发送包，这个包就会发送到默认路由器。但是，必须先配置路由器，然后才能转发这些包。出于安全考虑，SUSE LINUX 没有在默认安装中提供这种支持。要提供这种支持，请将文件 `/etc/sysconfig/sysctl` 中的变量 `IP_FORWARD` 设置为 `IP_FORWARD=yes`。连接的目标主机可以看到路由器，但对内部网络中发出包的那台主机却毫不知情。伪装技术就是因此而得名的。由于要进行地址转换，路由器自然成为所有回复包首先到达的目标。路由器必须能够识别这些入站包并转换其目标地址，这样才能将包转发给本地网络中的正确主机。

由于入站通讯数据的路由选择取决于伪装表，所以从外部根本无法打开与内部主机的连接。对于这种连接，伪装表中不会有任何对应项。此外，所有已建立的连接在该表中都被指派了一个状态项，所以其他连接无法再使用该项。

受以上各种因素影响，在使用某些应用程序协议，如 ICQ、cucme、IRC（DCC、CTCP）和 FTP（采用 PORT 方式）时，您可能会遇到一些问题。因此，Netscape（标准的 FTP 程序）和其他许多程序都使用 PASV 方式。就包过滤和伪装而言，这种被动方式不容易出问题。

### 34.1.3 防火墙基础知识

在描述不仅可以提供和管理网络间的链接，同时还能够控制网络间的数据流的机制时，防火墙可能是最常使用的术语。严格地说，本节所述的机制应该叫做包过滤器。包过滤器根据特定准则（如协议、端口和 IP 地址）来控制数据流。这样您就可以根据包的地址来拦截不应该发送到您网络中的包。举例来说，若允许对万维网服务器进行公共访问，应明确打开相应的端口。不过，包过滤器并不扫描有合法地址的包的内容（例如那些要发送到该万维网服务器的包）。例如，即使是在入站包想要破坏万维网服务器上的 CGI 程序的情况下，包过滤器仍然允许它们通过。

一种更有效但同时也更复杂的机制是将多种系统结合起来使用，例如让包过滤器与应用程序网关或代理进行交互。在这种情况下，包过滤器将拒绝所有发往禁用端口的包，而只接受发往应用程序网关的包。此网关或代理伪装成服务器的实际客户端。从某种意义上说，可以将这种代理视为应用程序使用的协议级的伪装主机。这种代理的一个示例就是 Squid（一种 HTTP 代理服务器）。要使用 Squid，必须将浏览器配置为通过代理通讯。代理超速缓存将提供请求的任何 HTTP 页，超速缓存中没有的页将由代理从因特网获取。再以 SUSE proxy-suite (proxy-suite) 为例，该程序为 FTP 协议提供了代理。

下一节着重介绍 SUSE LINUX 附带的包过滤器。有关包过滤和防火墙的详细信息，请阅读 `howto` 软件包中的 Firewall HOWTO（防火墙使用说明）。如果已安装此软件包，请使用 `less /usr/share/doc/howto/en/txt/Firewall-HOWTO.gz` 阅读 HOWTO。

## 34.1.4 SuSEfirewall2

SuSEfirewall2 是一个脚本，用来读取在 `/etc/sysconfig/SuSEfirewall2` 中设置的变量以生成一组 iptables 规则。该脚本定义了三个安全区域，但在以下示例配置中只涉及第一个和第二个安全区域：

**外部区域** 鉴于根本无法对外部网络进行控制，所以需要保护主机，使其免受外部网络的影响。在多数情况下，外部网络就是因特网，但也可能是其他不安全的网络，如 WLAN。

**内部区域** 内部网络是指专用网络，多为 LAN。如果此网络中的主机使用专用地址范围中的 IP 地址（请参见第 22.1.2 节“网络掩码和路由”），则必须启用网络地址转换 (NAT)，内部网络中的主机才能访问外部网络。

**网络隔离区 (DMZ)** 尽管从外部网络和内部网络都可以访问此区域内的主机，但这些主机本身无法访问内部网络。这种设置可用于在内部网络前再加一道防线，因为 DMZ 系统与内部网络是隔离的。

凡是过滤规则集没有明确允许通过的网络通讯数据，iptables 都一概禁止。因此，必须将入站通讯数据所流经的各个接口放入这三个区域之一。对于每个区域，都应定义所允许的服务或协议。规则集只适用于来自远程主机的包。防火墙不截获本地生成的包。

可以使用 YaST 进行上述配置（请参见第 34.1.4 节“用 YaST 进行配置”）。还可以在文件 `/etc/sysconfig/SuSEfirewall2` 中进行手工配置，该文件已作适当注释。另外，`/usr/share/doc/packages/SuSEfirewall2/EXAMPLES` 还提供了一些示例方案。

### 用 YaST 进行配置

#### Important

##### 自动配置防火墙

完成安装后，YaST 会在所有已配置接口上自动启动防火墙。如果在系统中配置并激活了某个服务器，YaST 将通过服务器配置模块中的选项‘打开防火墙中所选接口上的端口’或‘打开防火墙中的端口’修改自动生成的防火墙配置。某些服务器模块对话框包含‘防火墙细节’按钮，用于激活其他服务和端口。YaST 防火墙配置模块可用于激活、取消激活防火墙或单独对其进行重配置。

#### Important

可以从 YaST 控制中心访问图形化的 YaST 配置对话框。选择‘安全性和用户’→‘防火墙’。该配置分为 7 个部分，可以在屏幕左侧的树结构上直接访问它们。

**启动** 在该对话框中设置启动行为。在默认安装中，SuSEfirewall2 已在新安装的系统上运行。也可以在这里启动和停止防火墙。如果要测试当前防火墙设置，请使用按钮‘立即保存设置并重启动防火墙’。

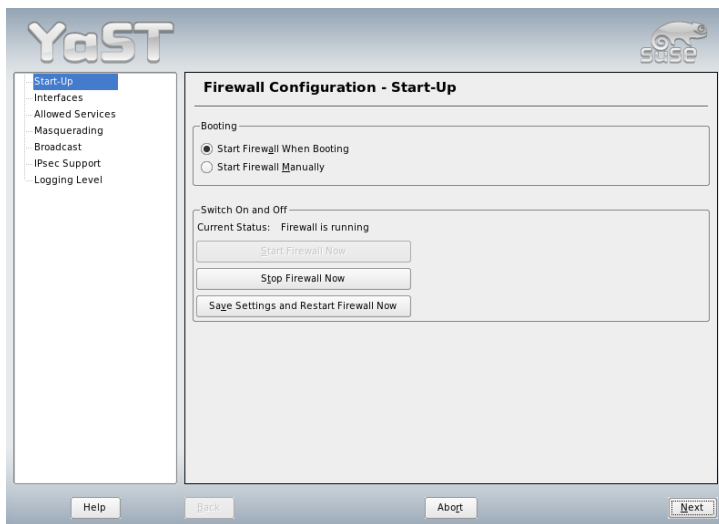


Figure 34.2: YaST 防火墙配置

**接口** 这里列出了所有已知的网络接口。要从区域中去除某接口，请选择此接口，并在按‘更改’后选择‘\_\_\_\_no\_zone\_\_\_\_’。要向区域中添加某接口，请选择此接口，并在按‘更改’后选择任一可用区域。也可以使用‘用户定义’来创建一个采用您自己的设置的特殊接口。

**允许的服务** 您需要此选项来从自己的系统向受保护区域提供服务。默认情况下只保护外部区域。在此情况下，您需要显式地允许可从外部主机看到的服务。在‘所选区域允许的服务’中选择适当的区域后，激活相应的服务。

**伪装** 伪装允许您对外部网络（例如因特网）隐藏自己的内部网络。它还使得内部网络能够透明地访问外部网络。伪装将阻塞从外部网络到内部网络的请求，而发自内部网络的请求从外部来看好像是由伪装服务器发出的。

如果需要使外部网络能够使用内部计算机的特殊服务，则可以为相应服务添加特殊重定向规则。

**广播** 在该对话框中配置允许广播的 UDP 端口。必须将所需的各个端口号或服务添加到相应区域，中间用空格分隔。另请参见文件 `/etc/services`。

可以在这里启用对不允许的广播的日志记录。这样做可能会出现问題，因为 Windows 主机使用广播来互相识别，从而生成许多不允许的包。

**IPsec 支持** 可以在此对话框中配置是否允许从外部网络使用 IPsec 服务。必须在‘细节’下配置可信的包。

**日志记录级别** 日志记录有两种规则：允许的包和不允许的包。允许的包将被接受，不允许的包则被删除或拒绝。可以为它们选择‘全部记录’、‘记录关键信息’或‘不记录任何信息’。

完成防火墙配置后，选择‘下一步’退出此对话框。然后系统将为您显示一个面向区域的防火墙配置概要，此时应再次检查所有设置。所有已允许使用的服务、端口和协议均在此概要中列出。如果要返回到配置，请选择‘后退’并按‘接受’保存设置。

## 手工配置

以下段落提供进行成功配置的分步说明。每个配置项都根据该项是与防火墙有关还是与伪装有关作了相应标记。这里未涉及配置文件中提到的与 DMZ（网络隔离区）相关的内容。这些内容只适用于大型组织中较复杂的网络基础结构（公司网络），这需要大量配置以及对此主题的深入了解。

首先，使用 YaST 模块 Systems Services (RunLevel) 以您的运行级别（很可能是 3 或 5）启用 `SuSEfirewall2`。这会在 `/etc/init.d/rc?.d/` 目录中设置 `SuSEfirewall2.*` 脚本的符号链接。

**FW\_DEV\_EXT**（防火墙，伪装） 链接到因特网的设备。对于调制解调器连接，请输入 `ppp0`。对于 ISDN 链接，请使用 `ipp0`。DSL 连接使用 `dsl0`。指定 `auto` 使用与默认路由对应的接口。

**FW\_DEV\_INT**（防火墙，伪装） 链接到内部专用网络的设备（如 `eth0`）。如果内部网络不存在，防火墙只保护它运行所在的主机，则应将此项保留为空白。

**FW\_ROUTE**（防火墙，伪装） 如果需要伪装功能，请将此项设置为 `yes`。内部主机对外将是不可见的，因为因特网路由器将忽略内部主机的专用网络地址（例如 `192.168.x.x`）。

对于未使用伪装功能的防火墙而言，只有在您希望允许访问内部网络时才应将此项设置为 `yes`。在此情况下，内部主机需要使用正式注册的 IP。但在通常情况下，应禁止从外部访问您的内部网络。



**FW.MASQUERADE** (伪装) 如果需要伪装功能, 则将此项设置为 `yes`。这实际上为内部主机提供了与因特网的直接连接。但更保险的做法是在内部网络主机和因特网主机之间设置代理服务器。代理服务器提供的服务不需要伪装。

**FW.MASQ\_NETS** (伪装) 指定要伪装的主机或网络, 各个项之间要留有空格。例如:

```
FW_MASQ_NETS="192.168.0.0/24 192.168.10.1"
```

**FW.PROTECT\_FROM\_INT** (防火墙) 将此项设置为 `yes` 可以保护防火墙主机免遭来自内部网络的攻击。只有已经显式启用服务, 才可以在内部网络中使用这些服务。另请参见 `FW.SERVICES.INT.TCP` 和 `FW.SERVICES.INT.UDP`。

**FW.SERVICES\_EXT.TCP** (防火墙) 输入应该可用的 TCP 端口。对于不应提供任何服务的家用普通工作站, 应将此项保留为空白。

**FW.SERVICES\_EXT.UDP** (防火墙) 除非您运行 UDP 服务并希望此服务对外部可用, 否则将此项保留为空白。使用 UDP 的服务包括 DNS 服务器、IPSec、TFTP、DHCP 及其他。在此情况下, 请输入要使用的 UDP 端口。

**FW.SERVICES\_INT.TCP** (防火墙) 使用此变量定义可用于内部网络的服务。变量表示法与 `FW.SERVICES_EXT.TCP` 的表示法相同, 但变量设置适用于内部网络。只有在 `FW.PROTECT_FROM_INT` 设置为 `yes` 时才需要设置此变量。

**FW.SERVICES\_INT.UDP** (防火墙) 请参见 `FW.SERVICES_INT.TCP`。

配置防火墙后, 请测试您的设置。以 `root` 身份输入 `SuSEfirewall2 start` 可创建防火墙规则集。然后, 举例来说, 可以从外部主机使用 `telnet` 查看是否确实会拒绝连接。此后, 请查看 `/var/log/messages`, 应能看到如下所示的内容:

```
Mar 15 13:21:38 linux kernel: SFW2-INext-DROP-DEFLT IN=eth0
OUT= MAC=00:80:c8:94:c3:e7:00:a0:c9:4d:27:56:08:00 SRC=192.168.10.0
DST=192.168.10.1 LEN=60 TOS=0x10 PREC=0x00 TTL=64 ID=15330 DF PROTO=TCP
SPT=48091 DPT=23 WINDOW=5840 RES=0x00 SYN URGP=0
OPT (020405B40402080A061AFEB0000000001030300)
```

用于测试您的防火墙设置的其他包有 `nmap` 或 `nessus`。在安装相应的包后, `nmap` 的文档位于目录 `/usr/share/doc/packages/nmap` 中, `nessus` 的文档位于目录 `/usr/share/doc/packages/nessus-core` 中。

### 34.1.5 参考信息

`/usr/share/doc/packages/SuSEfirewall12` 中提供了有关 SuSEfirewall12 软件包的最新信息和其他文档。netfilter 和 iptables 项目的主页 <http://www.netfilter.org> 以多种语言提供了丰富的文档资料。

## 34.2 SSH：安全的网络操作

随着在联网环境中安装的计算机越来越多，常常需要从远程位置访问主机。通常，这意味着用户要发送登录名和口令字符串进行鉴定。只要以明文形式传送这些字符串，攻击者就可能截获并恶意使用这些字符串获取对该用户帐户的访问权，而授权用户根本毫无察觉。一旦得逞，攻击者不仅可以控制所有用户文件，还可能利用此非法帐户获取管理员或 root 访问权，或侵入其他系统。过去常用 telnet 建立远程连接，但该程序没有采用加密形式或其他安全机制防止窃听。还存在其他未加保护的通讯信道，例如传统的 FTP 协议和某些远程复制程序。

SSH 套件通过对鉴定字符串（通常由登录名和口令构成）及主机间交换的所有其他数据进行加密，能够提供必要的保护。使用 SSH，虽然第三方仍可以记录数据流，但内容是经过加密的，除非了解加密钥，否则无法将其还原为明文。这样，SSH 在不安全的网络（如因特网）上实现了安全通讯。SUSE LINUX 附带的 SSH 程序是 OpenSSH。

### 34.2.1 OpenSSH 软件包

SUSE LINUX 会在默认情况下安装 OpenSSH 软件包。安装之后即可由程序 ssh、scp 和 sftp 来替代 telnet、rlogin、rsh、rcp 和 ftp。在默认配置中，只能使用 OpenSSH 实用程序访问 SUSE LINUX 系统，而且只能在防火墙允许访问的情况下访问。

### 34.2.2 ssh 程序

使用 ssh 程序可以登录到远程系统并以交互方式工作。该程序取代了 telnet 和 rlogin。slogin 程序只是指向 ssh 的符号链接。例如，使用命令 `ssh sun` 登录到主机 sun。该主机会随后提示输入 sun 上的口令。

通过鉴定后，既可以使用远程命令行操作，也可以使用 YaST 之类的交互式应用程序操作。如果本地用户名不同于远程用户名，则可以使用不同的登录名通过 `ssh -l augustine sun` 或 `ssh augustine@sun` 登录。

此外，ssh 还提供了在远程系统上运行命令的功能（也就是 rsh 提供的功能）。下例在主机 sun 上运行命令 uptime 并创建了一个名为 tmp 的目录。程序输出显示在主机 earth 的本地终端上。

```
ssh otherplanet "uptime; mkdir tmp" tux@otherplanet's password:1:21pm up 2:17, 9
```

在此需要加引号，以使用一个命令同时发送两条指令。只有加引号才能在 sun 上执行第二个命令。

### 34.2.3 scp — 安全复制

使用 scp 可以将文件复制到远程计算机。该程序取代了 rcp，是一个既安全又有加密功能的程序。例如，scp MyLetter.tex sun: 将文件 MyLetter.tex 从主机 earth 复制到主机 sun。如果 earth 上的用户名不同于 sun 上的用户名，请使用 username@host 格式指定后者的用户名。此命令没有 -l 选项。

输入正确的口令后，scp 开始传送数据，同时显示一行不断增多的星号来模拟进度条。此外，该程序还能显示到达进度条右端的估计时间。通过提供选项 -q 可以取消显示所有输出。

scp 还提供了对整个目录的递归复制功能。命令 scp -r src/ sun:backup/ 可以将目录 src 的全部内容（包含所有子目录）都复制到主机 sun 上的 backup 目录。这个子目录若不存在，就会自动创建该子目录。

选项 -p 指示 scp 不更改文件的时间戳。-C 将对传送数据进行压缩。这会最大限度地减少要传送的数据量，但同时增加了处理器的负担。

### 34.2.4 sftp — 安全的文件传送

sftp 程序可以取代 scp 来执行安全的文件传送。在 sftp 会话期间，您可以使用许多来自 ftp 的命令。sftp 程序可能要优于 scp，特别是在传送文件名未知的数据时。

### 34.2.5 SSH 守护程序 (sshd) — 服务器端

要使用 SSH 客户程序 ssh 和 scp，必须在后台运行服务程序（即 SSH 守护程序），用于监听 TCP/IP port 22 上的连接。首次启动该守护程序时将生成三个密钥对。各密钥对均由私钥和公钥组成。因此，此过程被称为基于公钥的过程。要保证通过 SSH 安全地通讯，必须限制只有系统管理员才能访问私钥文件。文件权限是在默认安装中相应设置的。只有在本地 SSH 守护程序才需要私

钥，切勿将私钥提供给其他任何人。公钥组件（可通过扩展名 .pub 识别）将被发送到请求连接的客户机。所有用户都可以读取公钥组件。

连接请求是 SSH 客户程序发出的。正在等待的 SSH 守护程序将与请求连接的 SSH 客户程序交换标识数据以比较协议和软件版本，防止通过错误的端口连接。由于请求是由最初的 SSH 守护程序的子进程回复的，所以可以同时建立多个 SSH 连接。

对于 SSH 服务器和 SSH 客户机间的通讯，OpenSSH 支持使用版本 1 和版本 2 的 SSH 协议。新安装的 SUSE LINUX 系统默认采用版本 2。要在更新后继续使用版本 1，请按照 `/usr/share/doc/packages/openssh/README.SuSE` 中的说明操作。本文档还介绍了如何只通过几个步骤就将 SSH 1 环境转换为有效的 SSH 2 环境。

使用 SSH 版本 1 时，服务器将发送其公共主机密钥和服务器密钥，SSH 守护程序每小时就重新生成一次服务器密钥。这两个密钥都允许 SSH 客户程序对自由选择的会话密钥加密（会话密钥会被发送到 SSH 服务器）。SSH 客户程序还会通知服务器使用哪种加密方法（加密法）。

版本 2 的 SSH 协议不需要服务器密钥。但服务器端和客户端都要使用符合 Diffie-Helman 的算法来交换它们的密钥。

一定要使用私用主机密钥和服务器密钥对会话密钥解密，从公钥根本无法得出这些密钥。只有建立联系的 SSH 守护程序才能使用其私钥对会话密钥解密（请参见 `man /usr/share/doc/packages/openssh/RFC.nroff`）。打开 SSH 客户程序的冗长调试选项 `-v` 可以密切监视初始阶段的连接。

默认情况下使用的是版本 2 的 SSH 协议。用 `-1` 开关可以覆盖此默认设置，转而使用该协议的版本 1。在与远程主机首次建立联系之后，客户机会将所有公共主机密钥储存在 `~/.ssh/known_hosts` 中。这会防止各种中间人攻击——外部 SSH 服务器试图使用伪造名称和伪造 IP 地址侵入系统。如果 `~/.ssh/known_hosts` 未包含某个主机密钥，或是因未能提供正确的私钥致使服务器无法对会话密钥解密，这两种情况下都可以检测到此类攻击。

建议将储存在 `/etc/ssh/` 中的私钥和公钥备份到安全的外部位置。这样就可以检测到密钥修改，并可以在重装后再次使用旧密钥。用户也就无需了解那些含糊不清的警告了。如果经过校验（尽管有警告）得知这确实是正确的 SSH 服务器，则必须从 `~/.ssh/known_hosts` 中去除有关此系统的现有项。

### 34.2.6 SSH 鉴定机制

现在开始执行实际的鉴定，最简单的鉴定形式就是上文提到的输入口令。SSH 的目的是提供安全且方便使用的软件。由于 SSH 是用来取代 `rsh` 和 `rlogin` 的，所以 SSH 必须也能提供一种适合日常使用的鉴定方法。SSH 是通过另一个密

钥对（由用户生成）来实现该功能的。为此，SSH 软件包提供了一个帮助程序：`ssh-keygen`。输入 `ssh-keygen -t rsa` 或 `ssh-keygen -t dsa` 后就会生成密钥对，同时系统会提示您输入储存密钥所用的基本文件名。

确认默认设置并回应针对通行密码的请求。即使软件建议输入空通行密码，仍建议您为此处说明的过程输入一个 10 到 30 个字符的通行密码。请不要使用简短的单词或短语。再次输入通行密码进行确认。随后，您将看到私钥和公钥的储存位置，在本例中为文件 `id_rsa` 和 `id_rsa.pub`。

使用 `ssh-keygen -p -t rsa` 或 `ssh-keygen -p -t dsa` 更改旧通行密码。将公钥组件（本例中为 `id_rsa.pub`）复制到远程计算机并保存到 `~/.ssh/authorized_keys` 中。下次建立连接时，系统将要求您使用通行密码对自身进行鉴定。如果系统没有要求鉴定，请校验这些文件的位置和内容。

从长期看，此过程比每次提供口令要麻烦。所以，SSH 软件包提供了另一种工具 `ssh-agent`，该工具可以在 X 会话期间保留私钥。整个 X 会话作为 `ssh-agent` 的子进程启动。为此，最简单的方法是将 `.xsession` 文件开始位置的变量 `usessh` 设置为 `yes` 并通过显示管理器（如 KDM 或 XDM）登录。也可以输入 `ssh-agent startx`。

现在您就可以像平常那样使用 `ssh` 或 `scp` 了。如果按上文所述分发了公钥，系统就不再提示您输入口令。终止 X 会话或用口令保护应用程序（如 `xlock`）锁定该会话时一定要小心。

文件 `/usr/share/doc/packages/openssh/README.SuSE` 还介绍了由于引入版本 2 的 SSH 协议而产生的所有相关更改。

## 34.2.7 X、鉴定和转发机制

除上述有关安全方面的改进之外，SSH 还简化了远程 X 应用程序的用法。如果运行带选项 `-X` 的 `ssh`，远程计算机上会自动设置 `DISPLAY` 变量，而且所有 X 输出都将通过现有 SSH 连接导出到远程计算机上。同时，如果未经授权，其他人将无法截获通过此方法远程启动并在本地查看的 X 应用程序。

通过添加选项 `-A`，可以将 `ssh-agent` 鉴定机制转移到下一台计算机上。这样，您就可以在不同计算机上工作而无需输入口令，但前提是：已将公钥分发给目标主机并在其上正确保存。

上述两种机制在默认设置中均处于取消激活状态，但可以随时在全系统范围的配置文件 `/etc/ssh/sshd_config` 或用户的 `~/.ssh/config` 中永久激活这两种机制。

`ssh` 还可用于重定向 TCP/IP 连接。在下例中，SSH 分别用于重定向 SMTP 和 POP3 端口：

```
ssh -L 25:sun:25 earth
```

使用此命令，可以通过加密信道将定向到 *earth* 端口 25 (SMTP) 的任何连接重定向到 *sun* 上的 SMTP 端口。如果用户所用的 SMTP 服务器不具备 SMTP-AUTH 或 POP-before-SMTP 功能，此命令特别有用。从与网络相连的任意位置都可以将电子邮件传送到“家庭”邮件服务器进行递送。同样，使用以下命令，可以将 *earth* 上的所有 POP3 请求（端口 110）转发给 *sun* 的 POP3 端口：

```
ssh -L 110:sun:110 earth
```

必须以 `root` 身份执行这两个命令，因为连接指向有特权的本地端口。普通用户通过现有 SSH 连接发送和检索电子邮件。为此，必须将 SMTP 和 POP3 主机设置为 `localhost`。有关其他信息，请参见上述每个程序的手册页以及 `/usr/share/doc/packages/openssh` 下的文件。

## 34.3 对分区和文件进行加密

### 34.3.1 应用方案

每个用户都有一些不想被第三方访问的机密数据。您连接到网络的几率越高、移动性越强，就越应该注意数据的处理。如果第三方可以通过网络连接或直接物理连接访问您的数据，则对文件或整个分区进行加密永远是明智之举。下面列出了几个可能的应用方案。

**便携式计算机** 如果您携带便携式计算机去旅行，最好对包含机密数据的硬盘分区进行加密。如果便携式计算机丢失或失窃，则驻留在加密文件系统或单个加密文件中的数据是安全的。

**可移动媒体** USB 闪存盘或外部硬盘像便携式计算机一样容易失窃。加密文件系统可以在这些情况下保护数据不被第三方访问。

### 34.3.2 使用 YaST 设置加密文件系统

YaST 可以对正在安装和已安装的系统进行文件或分区的加密。加密文件可以随时创建，原因是它可以很好地嵌入现有的分区布局中。要加密整个分区，需要在分区布局中提供一个专用于加密的分区。默认情况下，YaST 的标准分区建议并不包括加密分区。请在分区对话框中手工添加此分区。

## 在安装过程中创建加密分区

### Warning

#### 口令输入

在为加密分区创建口令时，请仔细阅读有关口令安全性的警告并记住口令。没有口令，就不可能访问加密数据。

### Warning

第2.8.5节“分区”中介绍的用于分区的 YaST 专家对话框提供了创建加密分区所需的选项。单击‘创建’，这与创建常规分区时类似。在打开的对话框中，输入新分区的分区参数，例如所需的格式和装入点。单击‘加密文件系统’完成创建。在随后出现的对话框中，设置口令，出于安全原因，再重复输入一次。通过单击‘确定’完成分区对话框之后便创建了新的加密分区。引导时，操作系统将要求用户输入口令，之后才能装入分区。

如果在启动期间不想装入加密分区，则在系统提示输入口令时按 **(Enter)** 键。然后，再次拒绝输入口令的提示。此时，未装入加密文件系统，而操作系统继续引导，这种方式可以更安全地保护您的数据。装入分区后，此分区对所有用户都可用。

如果只在必要时才装入加密文件系统，请选中‘fstab 选项’对话框中的‘不要在引导时装入’。系统启动时不会考虑装入对应的分区。之后若要使该分区可用，需要使用 `mount <name_of_partition> <mount_point>` 手工装入。在系统提示装入分区时输入口令。完成装入分区后，使用 `umount name_of_partition` 卸装分区，以阻止其他用户访问此分区。

## 在运行的系统上创建加密分区

### Warning

#### 在运行的系统中激活加密

还可以在正在运行的系统（例如在安装过程中）上创建加密分区。但是，加密现有的分区会损坏分区上的所有数据。

### Warning

在正在运行的系统上，在 YaST 控制中心中选择‘系统’→‘分区’。单击‘是’继续。请不要选择上文介绍的‘创建’，而是单击‘编辑’。其余的过程相同。

## 安装加密文件

除使用分区外，还可以在单个文件内创建存放机密数据的加密文件系统。可以从同一个 YaST 对话框创建这些文件系统。选择‘加密文件’，然后输入指向要创建的文件的的路径及其所需大小。接受建议的格式化设置和文件系统类型。然后，指定装入点并确定是否应在引导过程中装入加密文件系统。

加密文件具有一个优点，即可以添加加密文件而无须更改硬盘的分区。可以借助于环路设备装入加密文件，而其行为方式与常规的分区分类似。

## 使用 vi 加密文件

使用加密分区也有缺点，即在装入加密分区时，至少 root-用户可以访问数据。要防止出现这种情况，可以在加密方式中使用 vi。

使用 `vi -x filename` 编辑新文件。vi 将提示输入口令，然后加密文件的内容。只要您再次访问此文件，vi 就提示您输入正确的口令。

要获得真正的安全保护，甚至可以将此加密文本文件放入已得到安全保护的分区中。这种方法很有用，因为 vi 中使用的加密机制被认为不是一种强加密机制。

### 34.3.3 加密可移动媒体的内容

可移动媒体（例如外部硬盘或 USB 闪存盘）被 YaST 识别为与任何其他硬盘相同的设备。通过执行上面介绍的步骤，可以对这些媒体上的文件或分区进行加密。不要选择在引导过程中装入这些媒体，因为这些媒体通常只在系统运行时才连接上。

## 34.4 安全性和机密性

Linux 或 UNIX 系统的一个主要特点就是能够同时处理多个用户（多用户），并允许这些用户在同一计算机上同时执行多项任务（多任务）。此外，操作系统对网络是透明的。用户往往不知道他们所用的数据和应用程序是他们自己的计算机本地提供的，还是通过网络提供的。

利用多用户功能时，不同用户的数据必须分开储存。要确保安全性和私密性。数据安全性早在计算机能够联网之前就已经是个关键问题。和今天一样，那时人们最担心的问题就是如何保持数据可用，即便数据丢失或是数据媒体（通常是硬盘）损坏也不受任何影响。

本节侧重说明机密性问题以及保护用户隐私的方法；但是，全面的安全观离不开一些过程，即：随时准备好定期更新、可行的和经过测试的备份副本，这一点再怎么强调都不为过。做不到这一点，您可能很难恢复数据——不仅是在出现硬件故障时，在怀疑有人未经授权访问和篡改文件时，也很难恢复数据。



### 34.4.1 本地安全和网络安全

访问数据有以下几种方式：

- 与拥有所需信息或能访问计算机数据的人进行人际交流
- 直接从计算机控制台访问（物理访问）
- 通过串行线路
- 使用网络链接

在上述所有情况下，用户只有在通过鉴定后才能访问相关资源或数据。万维网服务器在这方面可能较为宽松，但您仍不希望它向任何访问者透露您个人的所有数据。

在上面的列表中，第一种情况需要最大程度的人际互动，例如在您联系银行职员时，您需要证明自己就是银行帐号的持有人。然后您需要提供签名、PIN 或口令，来证明您就是您自己声称的那个人。有时候，有些人可能会通过提及一些大家都知道的零散信息，利用花言巧语骗取知情人的信任，诱导知情人说出机密信息。受骗人可能会被一步步地诱骗出更多信息，而自己却一直蒙在鼓里。这种手段在黑客内部称为社会工程。要防止受骗，您只能对人们进行教育，并且学会言辞谨慎，不轻易透露信息。在闯入计算机系统之前，攻击者通常将目标锁定在接待员、公司内的服务人员，甚至也可能是家庭成员。在很多情况下，最终发现这种利用社会工程手段发起的攻击时，通常为时已晚。

企图未经授权访问您数据的人也可能使用传统方法直接攻击您的硬件。因此，应该防止有人对计算机做手脚，让任何人都无法拆除、替换或损坏其部件。这也适用于各种备份，甚至是各种网络电缆或电线。还应保护引导过程，因为已知有些按键组合会引起异常行为。通过为 BIOS 和引导装载程序设置口令可以提供这方面的保护。

串行终端连接到串行端口，目前很多地方仍在使用这种连接方式。与网络接口不同，它们不依赖网络协议与主机通讯。通过一根缆线或是红外端口就可以在设备间发送纯字符数据。缆线本身就是这种系统最大的弱点：只要连接上一台旧式打印机，就能很容易地将通过缆线传送的所有数据都记录下来。通过打印机做的事情也可以通过其他方式来完成，具体取决于攻击的强度。

在一台主机的本地读取文件需要一定的访问规则，这与打开网络连接访问其它主机上的服务器所需的规则不同。本地安全不同于网络安全。区别即在于：必须将数据放入包中才能发送到其他位置。

## 本地安全

关于本地安全，还要从计算机运行所在的位置的物理环境说起。在符合您的安全预期和需要的地点搭建计算机系统。本地安全的主要目标是将不同用户分隔开，防止某个用户假借其他用户的权限或身份。这是要遵守的一条基本规则，但这条规则对 root（根用户，对系统拥有最高权力）尤其适用。root 不必输入口令即具有其他任何本地用户的身份，还能够读取本地储存的所有文件。

## 口令

在 Linux 系统中，口令当然不是以纯文本格式储存的，而且也不能简单地将输入的文本字符串与保存的模式匹配。如果事实正好相反，只要有人取得了相应文件的访问权，就会危及系统中所有帐户的安全。所以对储存的口令加密，并且每次输入时都要再次加密，而后对两个加密字符串进行比较。只有在无法将加密口令反向计算为原始文本字符串时，这种方式才能提高安全性。

实际上，这是通过一种特殊算法（亦称活门算法）实现的，因为该算法只能单向有效。截获加密字符串的攻击者无法通过简单地再次应用同一算法来获取您的口令。这需要测试所有可能的字符组合，直到有一种组合看似加密时的口令。对于八位字符口令，需要计算的可能组合是相当多的。

20 世纪 70 年代时，人们对这种方法是否比其他方法更安全存在争议，因为其中所用算法的计算速度相对较慢，几秒钟才能加密一个口令。而与此同时，PC 的处理能力已足够大，每秒可进行几十万次甚至是几百万次的加密。正因为这样，不应让普通用户看到加密口令（普通用户不能读取 /etc/shadow）。更重要的是，口令应该是不容易猜出的，以防因意外错误而暴露口令文件。因此，将“tantalise”这样的口令“转换”为“t@nt@1ls3”实际上没有任何帮助。

用相似的数字替换单词中的某些字母是不够安全的。利用字典来猜字的口令破解程序用的也是类似的替换方法。最好是编造出没有一般意义的词，即仅对您自己有意义的词，如句中各词的词首字母，或书名，如 Umberto Eco 撰写的“The Name of the Rose”。这样可以编写以下安全的口令：“TNotRbUE9”。相比之下，像“beerbuddy”或“jasmine76”这样的口令，即便是不太了解您的人也很容易猜出。

## 引导过程

配置系统，让其无法从软盘或 CD 引导；配置方法为：彻底拆除驱动器，或设置 BIOS 口令，将 BIOS 配置为只能从硬盘引导。通常，Linux 系统要通过引导装载程序来启动，这样您就可以向引导内核传递更多选项。通过在 /boot/grub/menu.lst 中设置额外的口令可防止其他用户在引导期间使用这类参数（请参见第 8 章 引导装载程序）。这对您的系统安全举足轻重。不仅内核本身以 root 权限运行，而且内核还是在系统启动时授予 root 权限的第一个权威对象。

## 文件权限

一般来说，执行某项任务时应始终尽量使用限制性最强的特权。例如，以 root 权限读写电子邮件是完全没有必要的。如果邮件程序有错误，攻击者则可能利用该错误，以该程序启动时所具有的权限发起攻击。如若遵守上述规则，则可以尽量减少可能的损失。

为 SUSE 版本中的 200,000 多个文件指派的权限都是经过精挑细选的。系统管理员在安装附加软件或其他文件时要特别小心，尤其是在设置权限位时。经验丰富且安全意识强的系统管理员始终在命令 `ls` 后使用 `-l` 选项获取详细的文件列表，这样他们可以立即检测出有错误的文件权限。错误的文件特性不仅意味着文件可能被更改或删除，root 还可能执行这些改过的文件；或者如果是配置文件，程序则可能以 root 权限使用这些文件。这使系统很容易受到攻击。这类攻击被称为布谷鸟蛋，因为程序（蛋）由另一个用户（鸟）执行（孵化），就像是布谷鸟诱骗其他鸟代其孵蛋。

SUSE LINUX 系统中的文件 `permissions`、`permissions`、

`easy`、`permissions.secure` 和 `permissions.paranoid` 都位于 `/etc` 目录中。这些文件用于定义特殊权限（如全局可写目录），或为文件定义 `setuser` ID 位（设置了 `setuser` ID 位的程序不以启动该程序的用户的权限运行，而以文件拥有者的权限运行，这个权限多为 root）。管理员可以使用文件 `/etc/permissions.local` 添加自己的设置。

要定义 SUSE 配置程序使用上述哪个文件来相应地设置权限，请在 YaST 中选择‘安全’。要了解这一主题的详细信息，请阅读 `/etc/permissions` 中的注释，或参考 `chmod` 手册页 (`man chmod`)。

## 缓冲区溢出错误和格式字符串错误

只要程序要处理的数据可以或可能被用户更改，就应该加倍小心；但这个问题更应该是应用程序编程人员的问题而不是普通用户的问题。编程人员必须确保自己编写的应用程序以正确的方式解释数据，避免将这些数据写入内存时，内存区域过小而无法容纳。此外，程序应能够通过专用接口一致地传递数据。

如果写入内存缓冲区时未考虑该缓冲区的实际大小，则会发生缓冲区溢出。有时这些数据（即用户生成的数据）占用的空间会超过该缓冲区所能提供的空间。结果导致数据写出缓冲区，以致于在特定情况下，程序可能执行受用户（而不是编程人员）影响的程序顺序，而不只是处理用户数据。这种错误可能导致严重后果，特别是在使用特权执行程序时（请参见第 34.4.1 节“文件权限”）。

格式字符串错误的作用方式略有不同，但同样也是用户输入导致了程序出错。多数情况下，这些编程错误会被以特殊权限执行的程序（`setuid` 程序和 `setgid` 程序）利用，这也意味着您可以通过取消这些程序的相应执行特权，来防止您的数据和系统受到这种错误的影响。同样，最好的办法是使用尽可能低的特权（请参见第 34.4.1 节“文件权限”）。

鉴于缓冲区溢出错误和格式字符串错误都是与用户数据处理有关的错误，攻击者不仅可以在本地帐户具有访问权的情况下利用这些错误，还可以通过网络链接利用许多已报告的错误。因此，缓冲区溢出错误和格式字符串错误与本地安全和网络安全都有关系。

## 病毒

Linux 上确有病毒运行，这与某些人的说法正好相反。但是，已知的这些病毒是由它们的创作者为进行概念验证而发布的，目的是为证明这种技术可以发挥预期的作用。目前尚未发现上述病毒中有任何正在流行。

没有借以寄生的宿主，病毒将无法生存和传播。对于计算机病毒来说，这个“宿主”是病毒程序代码能写入的某个程序或系统中的重要储存区域（如主引导记录）。由于 Linux 支持多用户处理，所以它可以将写权限限制到特定文件，这对系统文件尤其重要。因此，以 root 权限执行一般操作无疑会增大系统受病毒感染的机率。相反，若遵循上述规则（即使用尽可能低的特权），感染病毒的机率会非常低。

除此之外，切勿在未真正了解某个因特网站点时就仓促执行其中的程序。SUSE 的 RPM 包具有加密签名，这是个数字标签，表明在生成包时已经考虑到必要的安全措施。病毒的存在是管理员或用户缺乏必要的安全意识的典型表现，病毒使系统受到威胁，即使是设计上高度安全的系统也无法逃避。

不应将病毒和蠕虫混为一谈，后者属于全球网络的问题。蠕虫的传播不需要宿主。

## 网络安全

网络安全对于保护系统免遭外部攻击至关重要。典型的登录过程（即要求提供用户名和口令以进行用户鉴定）仍然是个本地安全问题。在通过网络登录的特定情况下，应该区分这两方面的安全问题。在实际鉴定之前发生的属于网络安全问题，之后发生的属于本地安全问题。

## X 窗口系统和 X 鉴定

正如本文开头所述，网络透明特性是 UNIX 系统的核心特性之一。X（作为 UNIX 操作系统的窗口系统）能明显地利用这一特性。利用 X 系统，在远程主机上登录，然后启动一个图形化程序，令其随后通过网络发送并显示在您的计算机上，这个过程基本不成问题。

如果应该使用 X 服务器远程显示 X 客户机，X 服务器应该防止未经授权访问受其管理的资源（即显示）。更具体地说，必须给客户程序指派特定权限。在 X 窗口系统中，有两种指派权限的方法，分别为基于主机的访问控制和基于

Cookie 的访问控制。前者依赖应该运行客户程序的主机的 IP 地址。所用的控制程序为 `xhost`。使用 `xhost` 可以将合法客户程序的 IP 地址输入属于 X 服务器的小型数据库。不过，依赖 IP 地址进行鉴定不是十分安全。例如，如果另有用户在发送客户程序的主机上工作，该用户将同样能够访问 X 服务器 — 就像是有人盗取了 IP 地址。由于存在这些缺点，在此不再详述这种鉴定方法，但您可以通过 `man xhost` 了解更多相关信息。

使用基于 Cookie 的访问控制时，将生成一个只有 X 服务器和合法用户才知道的字符串，就像是某种身份证。这个 Cookie（该词并不指普通意义上的小甜饼，而是指里面有幸运签的中国幸运饼）在登录时储存在用户主目录下的文件 `.Xauthority` 中，并且可供要使用 X 服务器显示窗口的所有 X 客户机使用。用户可以使用工具 `xauth` 检查文件 `.Xauthority`。如果将 `.Xauthority` 重命名或者意外地从主目录中删除了该文件，则无法再打开任何新窗口或 X 客户机。有关 X 窗口系统安全机制的详细信息，请参见 `Xsecurity` 手册页 (`man Xsecurity`)。

SSH（安全 shell）可用于对网络连接彻底加密并将其透明地转发给 X 服务器，而用户根本察觉不到这种加密机制。这也称为 X 转发。要实现 X 转发，需要在服务器端模拟 X 服务器，并在远程主机上为 shell 设置 `DISPLAY` 变量。有关 SSH 的更多详细信息，请参见第 34.2 节“SSH：安全的网络操作”。

### Warning

如果不认为登录主机是安全主机，请不要使用 X 转发。在不安全的主机上启用 X 转发后，攻击者可能经由 SSH 连接通过鉴定，闯入您的 X 服务器并嗅探键盘输入之类的信息。

### Warning

## 缓冲区溢出错误和格式字符串错误

如第 34.4.1 节“缓冲区溢出错误和格式字符串错误”所述，缓冲区溢出错误和格式字符串错误应划归与本地安全和网络安全都有关系的问题。与这些错误的本地情况一样，若成功攻击了网络程序中的缓冲区溢出漏洞，常常可以获取 `root` 权限。即便不是这样，攻击者也可以利用该错误获取对非特权本地帐户的访问，以攻击系统中可能存在的其他任何漏洞。

一般来说，通过网络链接发起的针对缓冲区溢出错误和格式字符串错误的攻击当属远程攻击中最常见的形式。攻击的漏洞 — 用于攻击新发现的安全漏洞的程序 — 通常在安全邮件列表上发布。通过它们可以修补漏洞而不必了解代码的详细信息。多年来的经验表明：漏洞检测代码的存在成就了更为安全的操作系统，这显然是因为迫于压力，操作系统制造商不得不修复他们软件中的问题。在自由软件中，任何人都拥有访问源代码（SUSE LINUX 提供所有可用源代码），并且任何人在发现漏洞及其漏洞检测代码后都可以提交增补程序来修复相应的错误。

## DoS — 拒绝服务

这种攻击的目的是拦截服务器程序甚至是整个系统，采用的手段有以下几种：使服务器超载、让服务器忙于处理垃圾包，或攻击远程缓冲区溢出漏洞。DoS 攻击往往只有一个目的：让服务不再可用。不过，一旦某个服务不再可用，通讯就易于受到中间人攻击（嗅探、TCP 连接劫持、欺骗），或发生 DNS 中毒。

### 中间人：嗅探、劫持、欺骗

一般而言，由身处通讯主机之间的攻击者发起的所有远程攻击都称为中间人攻击。几乎所有类型的中间人攻击都有一个共同特点，即受害人通常毫无察觉。这种攻击有多种变化形式，例如，攻击者可能会截获连接请求并自行将其转发给目标计算机。现在受害者就会在不知情的情况下与错误的主机建立连接，因为连接的这一端伪装为合法的目标计算机。

最简单的中间人攻击的形式称为嗅探——攻击者“只是”监听通过网络传递的数据流。更为复杂的“中间人攻击”可能会试图接管已经建立的连接（劫持）。劫持之前，攻击者需要一段时间对数据包进行分析，以便能够推测出属于该连接的 TCP 顺序号。在攻击者最终夺取目标主机的角色后，受害人会注意到这一点，因为他们会收到一条错误讯息，说明连接因失败而终止。

由于有些协议没有通过加密来预防劫持，只是在建立连接后执行简单的鉴定过程，这给攻击者创造了可乘之机。

欺骗类型的攻击是对数据包进行修改，使其包含虚假的源数据（通常是 IP 地址）。多数较活跃的攻击都依赖于发送这种虚假数据包——在 Linux 计算机上，发送包的任务只能由超级用户（root）执行。

上述很多攻击都是与 DoS 同时进行的。一旦攻击者发现有机会让某台主机突然宕机，即便是很短的时间，攻击者也容易发起猛烈攻击，因为主机将在一段时间内无法对抗攻击。

## DNS 中毒

DNS 中毒指的是通过向 DNS 服务器回复伪造的 DNS 回复包，试图让该服务器向请求其发送信息的受害者发送特定数据，以此破坏 DNS 服务器的超速缓存。许多服务器都基于 IP 地址或主机名与其他主机保持信任关系。攻击者需要非常熟悉主机之间信任关系的实际结构，才能将自己伪装为受信主机之一。通常，攻击者会分析一些从服务器接收的包，获取必要信息。攻击者还常常需要对名称服务器适时发动 DoS 攻击。您可以使用加密连接，通过对要连接的主机的身份进行校验来保护自己。

## 蠕虫

蠕虫经常被误认为是病毒，但两者有着明显的区别。不同于病毒，蠕虫无需感染某个要寄生的主机程序。它们的特点就是尽快在网络结构中传播。以往的蠕虫，如 Ramen、Lion 或 Adore，全都利用 bind8 或 lprNG 之类的服务器程序中的已知安全漏洞。蠕虫的预防相对简单。鉴于在发现安全漏洞和蠕虫对服务器发起攻击之间有一段时间，很有可能及时提供受影响的程序的更新版本。只有管理员确实在受感染系统上安装了安全更新程序时，这种方法才有用。

### 34.4.2 一些常用的安全提示和技巧

要有效处理安全问题，关键在于随时关注安全方面的新动态并了解最新的安全问题。要保护您的系统免受各种问题的侵扰，最好的方式就是尽快获取并安装安全公告推荐的更新软件包。SUSE 使用邮件列表发布安全公告，您可以通过链接 <http://www.novell.com/linux/security/securitysupport.html> 订阅该邮件列表。suse-security-announce@suse.de 列表提供关于更新软件包的第一手信息，向其积极投稿的人当中还有 SUSE 安全小组的成员。

邮件列表 suse-security@suse.de 提供了一个不错的论坛，可以在其中讨论任何相关的安全问题。可以使用上文中为 suse-security-announce@suse.de 提供的 URL 订阅该列表。

bugtraq@securityfocus.com 是全球知名的安全邮件列表之一。建议阅读此列表，该列表每天要接收 15 到 20 条投递信息。有关详细信息，请参见 <http://www.securityfocus.com>。

下面是一些规则，可能有助于您处理基本的安全问题：

- 根据要对每个作业都尽量使用限制性最强的一组权限的规则，应避免使用 root 权限执行常规作业。这样即可降低受布谷鸟蛋或病毒攻击的风险，防止您自己犯错误。
- 如果可能，应尽量使用加密连接在远程计算机上工作。用 ssh（安全 shell）替代 telnet、ftp、rsh 和 rlogin，这应是标准做法。
- 避免使用仅基于 IP 地址的鉴定方法。
- 尽量使最重要的网络相关软件包保持最新，并且订阅相应的邮件列表，接收有关以下程序（如 bind、sendmail、ssh 等）的最新版本的公告。该做法同样适用于与本地安全相关的软件。
- 更改 /etc/permissions 文件，优化对系统安全至关重要的文件的权限。如果删除某个程序的 setuid 位，该程序很可能无法再正常执行作业。但从另一方面考虑，在多数情况下，该程序也将不再是个潜在的安全隐患。所以，您可以对全局可写目录和文件采取相同的做法。

- 禁用不是绝对需要的所有网络服务，以便服务器正常运行。这样可以使系统更加安全。使用程序 `netstat` 可以找到套接状态为 `LISTEN` 的打开端口。至于选项，建议使用 `netstat -ap` 或 `netstat -anp`。`-p` 选项允许您查看哪个进程正以什么名称占用端口。

将 `netstat` 的结果与在主机外部执行的彻底端口扫描的结果进行比较。最适合执行这项作业的程序当属 `nmap`，该程序不仅可以检测计算机的端口，而且可以对哪些服务正等待端口处理作出一些判断。不过，端口扫描可能被认为是一种入侵行为，所以不要在未经管理员明确批准的情况下在主机上执行该操作。最后，要记住不仅要扫描 `TCP` 端口，而且要扫描 `UDP` 端口（使用选项 `-sS` 和 `-sU`），这一点至关重要。

- 要以可靠方式监视系统文件的完整性，请使用 `SUSE LINUX` 版本提供的程序 `tripwire`。对 `tripwire` 创建的数据库加密，防止有人篡改。此外，在其他计算机上保留此数据库的备份副本，储存该副本的外部数据媒体不能通过网络链接连接到您的计算机。
- 安装任何第三方软件时都要小心谨慎。曾经有黑客在安全软件包的 `tar` 档案中嵌入了特洛伊木马病毒，不过幸好发现得及时。如果安装二进制软件包，则应确保下载站点是安全的。

`SUSE` 的 `RPM` 包都具有 `gpg` 签名。`SUSE` 使用以下密钥来签名：

```
ID:9C800ACA 2000-10-19 SUSE Package Signing Key jbuild@suse.dej
```

```
Key fingerprint = 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA
```

命令 `rpm --checksig package.rpm` 显示校验和及未安装软件包的签名是否正确。可以在本版本的第一张 `CD` 上和全球多数密钥服务器上找到该密钥。

- 定期检查用户和系统文件的备份副本。考虑如果不测试备份是否有效，实际上它可能毫无价值。
- 检查日志文件。尽可能编写小型脚本搜索可疑项。无可否认，这并不是是一项非常烦琐的任务。最终，只有您才知道哪些项异常，哪些项正常。
- 使用 `tcp_wrapper` 限制访问计算机上运行的各个服务，这样您就可以明确控制哪个 `IP` 地址可以连接到某个服务。有关 `tcp_wrapper` 的进一步信息，请参考 `tcpd` 和 `hosts_access` 的手册页（`man 8 tcpd`、`man hosts_access`）。
- 使用 `SUSEfirewall` 可以增强 `tcpd` (`tcp_wrapper`) 提供的安全性。
- 设计安全措施重复出现：两次显示同一讯息总比不显示任何讯息好得多。



### 34.4.3 使用中央安全报告地址

如果您发现与安全有关的问题（请首先检查可用的更新软件包），请向 `security@suse.de` 发送电子邮件。请提供问题的详细说明以及涉及的软件包的版本号。SUSE 将尽快给予答复。建议您对电子邮件讯息进行 pgp 加密。SUSE 的 pgp 密钥为：

```
ID:3D25D3D9 1999-03-06 SUSE Security Team <security@suse.de> Key fingerprint = 73 4E 2E 99 DF
```

也可以从 <http://www.novell.com/linux/security/securitysupport.html> 下载该密钥。



# Linux 中的访问控制列表

本章概述了 Linux 文件系统的 POSIX ACL（访问控制列表）的背景和功能。可以将 ACL 作为文件系统对象的传统权限概念的扩展来使用。利用 ACL，可以比传统权限概念更灵活地定义权限。

35.1	ACL 的优势 . . . . .	530
35.2	定义 . . . . .	530
35.3	处理 ACL . . . . .	531
35.4	应用程序中的 ACL 支持 . . . . .	538
35.5	参考信息 . . . . .	538

POSIX ACL 这一术语表明它是一种真正的 POSIX（可移植操作系统接口）标准。由于多种原因，相应的标准草案 POSIX 1003.1e 和 POSIX 1003.2c 已被撤消。但是，在属于 UNIX 系列的许多系统上使用的 ACL 都基于这两个草案，并且本章中介绍的文件系统 ACL 的实施也遵照这两个标准。有关它们的信息，请参见 <http://wt.xpilot.org/publications/posix.1e/>。

## 35.1 ACL 的优势

传统情况下，会为 Linux 系统上的每个文件对象定义三组权限。这三组权限包括用于每种类型用户（即文件拥有者、组和其他用户这三种用户）的读 (r)、写 (w) 和执行 (x) 权限。此外，还可以设置设置用户 ID、设置组 ID 和粘滞位。这种简缩概念完全适用于大多数实际情况。但对于较复杂的方案或高级应用程序，系统管理员在以前必须采用多种技巧来避开传统权限概念的限制。

ACL 可以用于要求扩展传统文件权限概念的情况。它们可用于向单个用户或组分配权限，即使这些权限并不与原始拥有者或所属组相对应。访问控制列表是 Linux 内核的一项功能。目前，ReiserFS、Ext2、Ext3、JFS 和 XFS 都支持访问控制列表。通过使用 ACL，无需在应用程序级别实施复杂的权限模型就可以实现复杂的方案。

在类似用 Linux 服务器替换 Windows 服务器这样的情况下，ACL 的优势就尤为明显。即使在移植后，一些已连接的工作站仍可以继续 Windows 下运行。Linux 系统利用 Samba 向 Windows 客户机提供文件和打印服务。如果 Samba 支持访问控制列表，则既可以在 Linux 服务器上配置用户权限，也可以在具有图形用户界面的 Windows（仅限 Windows NT 和更高版本）中配置用户权限。利用 winbindd，甚至可以向仅存在于 Windows 域中而在 Linux 服务器中没有任何帐户的用户分配权限。

## 35.2 定义

**用户类别** 传统的 POSIX 权限概念在文件系统中使用三个用户类别来分配权限：拥有者、所属组和其他用户。可以为每个用户类别设置三个权限位，用于分配读 (r)、写 (w) 和执行 (x) 权限。

**访问 ACL** 各种文件系统对象（文件和目录）的用户和组访问权限均通过访问 ACL 来确定。

**默认 ACL** 默认 ACL 只能应用于目录。它们确定文件系统对象在创建时从其父目录继承的权限。

**ACL 项** 每个 ACL 都包含一组 ACL 项。ACL 项中包含一个类型（参见表35.1 “ACL 项类型”）、一个此项所关联的用户或组的限定符和一组权限。对于某些项类型，未定义组或用户的限定符。

## 35.3 处理 ACL

表35.1 “ACL 项类型”总结了 ACL 项 6 种可能出现的类型，每种类型都定义了一个用户或一组用户的权限。*拥有者*项定义了拥有该文件或目录的用户的权限。*所属组*项定义了文件所属组的权限。超级用户可以使用 `chown` 或 `chgrp` 更改拥有者或所属组，而在这种情况下，拥有者和所属组项表示新的拥有者和所属组。每个*已命名用户*项定义了在该项的限定符字段中指定的用户的权限，该限定符字段即表35.1 “ACL 项类型”中以文本形式显示的中间字段。每个*已命名组*项定义了在该项的限定符字段中指定的组的权限。只有已命名用户和已命名组项具有非空的限定符字段。*其他*项定义了所有其他用户的权限。

通过定义这些项中的有效权限和要屏蔽的权限，*掩码*项进一步限制了*已命名用户*、*已命名组*和*所属组*项授予的权限。如果权限同时存在于上述项之一和掩码中，它们就是有效的。仅包含在掩码或实际项中的权限是无效的，表示未授予这些权限。*拥有者*和*所属组*项中定义的所有权限始终有效。表35.2 “屏蔽访问权限”中的示例说明了这种机制。

ACL 有两种基本类别：一种是最小 ACL，仅包含用于类型*拥有者*、*所属组*和其他的项，对应于文件和目录的传统权限位。另一种是扩展 ACL，它比前一种要复杂得多。它必须包含一个*掩码*项，并可能包含若干*已命名用户*和*已命名组*类型的项。

Table 35.1: ACL 项类型

类型	文本形式
拥有者	<code>user::rwx</code>
已命名用户	<code>user:name:rwx</code>
所属组	<code>group::rwx</code>
已命名组	<code>group:name:rwx</code>
掩码	<code>mask::rwx</code>
其他	<code>other::rwx</code>

Table 35.2: 屏蔽访问权限

项类型	文本形式	权限
已命名用户	user:geeko:r-x	r-x
掩码	mask::rw-	rw-
	有效权限:	r--

### 35.3.1 ACL 项和文件方式权限位

图35.1 “最小 ACL: ACL 项 (对照权限位)” 和图 图35.2 “扩展 ACL: ACL 项 (对照权限位)” 说明了最小 ACL 和扩展 ACL 这两种情况。这些图分为三块 — 左边一块显示 ACL 项的类型规范，中间一块显示一个示例 ACL，右边一块显示对应于传统权限概念的各个权限位 (如 `ls -l` 所显示的)。在这两种情况下，*拥有者* 权限均被映射到 ACL *拥有者* 项。其他类别权限也被映射到各自的 ACL 项。但是，*组类别* 权限的映射在这两种情况中是不同的。

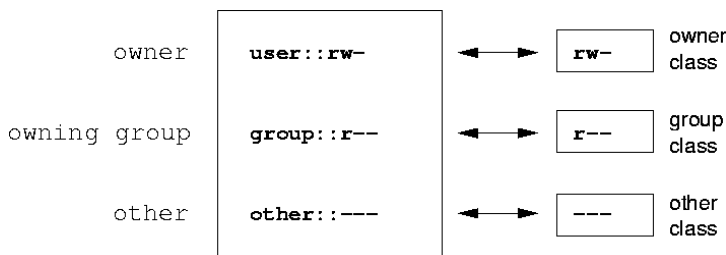


Figure 35.1: 最小 ACL: ACL 项 (对照权限位)

对于最小 ACL (没有掩码)，*组类别* 权限被映射到 ACL 的*所属组* 项。图35.1 “最小 ACL: ACL 项 (对照权限位)” 中显示了这一工具。对于扩展 ACL (具有掩码)，*组类别* 权限被映射到掩码项。图35.2 “扩展 ACL: ACL 项 (对照权限位)” 中显示了这一工具。

不管应用程序是否具有 ACL 支持，这种映射方式都可以确保应用程序的流畅交互。通过权限位方式分配的访问权限表示通过 ACL 所进行的所有其他“微调”的上限。对权限位的更改将由 ACL 反映出来，反之亦然。

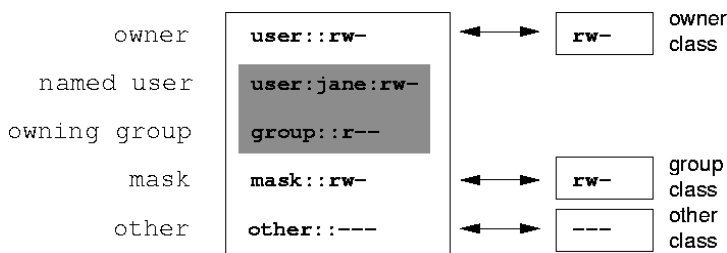


Figure 35.2: 扩展 ACL: ACL 项 (对照权限位)

### 35.3.2 具有访问 ACL 的目录

以下示例演示了访问 ACL 的处理:

在创建目录之前, 使用 `umask` 命令来定义每次创建文件对象时应屏蔽哪些访问权限。命令 `umask 027` 设置了默认权限, 即为拥有者分配全部权限 (0), 拒绝组写访问 (2), 并且不为其他用户分配任何权限 (7)。umask 实际上屏蔽了相应的权限位或将它们关闭。有关详细信息, 请参考相应的手册页 (`man umask`)。

`mkdir mydir` 应创建具有由 `umask` 设置的默认权限的 `mydir` 目录。使用 `ls -dl mydir` 来检查是否已正确分配所有权限。该示例的输入为:

```
drwxr-x--- ... tux project3 ... mydir
```

使用 `getfacl mydir`, 检查 ACL 的初始状态。这样会得出如下信息:

```
# file: mydir
# owner: tux
# group: project3
user::rwx
group::r-x
other::---
```

`getfacl` 的输出准确地反映了第 35.3.1 节“ACL 项和文件方式权限位”中描述的权限位与 ACL 项之间的映射。输出的前三行显示了目录的名称、拥有者和所属组。随后三行包含三个 ACL 项, 即拥有者、所属组和其他。事实上, 对于最小 ACL, `getfacl` 命令不会生成您使用 `ls` 所不能获得的任何信息。

使用以下命令修改 ACL, 为附加用户 `geeko` 和附加组 `mascots` 指派读、写和执行权限:

```
setfacl -m user:geeko:rwx,group:mascots:rwx mydir
```

选项 `-m` 提示 `setfacl` 修改现有的 ACL。以下参数指示要修改的 ACL 项（各项之间用逗号隔开）。最后部分指定了应该对其应用这些修改的目录的名称。使用 `getfacl` 命令来查看所生成的 ACL。

```
# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other::---
```

除了为用户 `geeko` 和组 `mascots` 创建的项外，还生成了一个掩码项。系统自动设置此掩码项，以便使所有权限生效。`setfacl` 自动使现有的掩码项与已修改的设置相适应，但前提是不要用 `-n` 取消此功能。掩码为组类别中的所有项定义了最大有效访问权限。其中包括已命名用户、已命名组和所属组。由 `ls -dl mydir` 显示的组类别权限位现在与掩码项相对应。

```
drwxrwx---+ ... tux project3 ... mydir
```

输出的第一栏现在包含一个附加的 `+`，表明此项存在一个扩展 ACL。

根据 `ls` 命令的输出，掩码项的权限包含写访问权限。传统情况下，这样的权限位意味着所属组（这里是 `project3`）也具有对 `mydir` 目录的写访问权限。但是，所属组的有效访问权限对应于为所属组和掩码定义的权限的重叠部分——在我们的示例中是 `r-x`（参见表 35.2 “屏蔽访问权限”）。对本例中的所属组的有效权限而言，即使是在添加了 ACL 项之后，也未发生任何改变。

用 `setfacl` 或 `chmod` 编辑掩码项。例如，使用 `chmod g-w mydir`。 `ls -dl mydir` 输出以下结果：

```
drwxr-x---+ ... tux project3 ... mydir
```

`getfacl mydir` 提供以下输出：

```
# file: mydir
# owner: tux
# group: project3
```



```
user::rwx
user:geeko:rwx      # effective: r-x
group::r-x
group:mascots:rwx   # effective: r-x
mask::r-x
other::---
```

执行 `chmod` 命令将写权限从组类别位去除后，从 `ls` 命令的输出就可看出掩码位肯定已被相应地更改了：写权限再次限于 `mydir` 的拥有者。`getfacl` 的输出证实了这一点。这个输出包含了对有效权限位与原始权限不对应的所有项的注释，因为已根据掩码项对它们进行了过滤。可以随时用 `chmod g+w mydir` 来恢复原始权限。

### 35.3.3 具有默认 ACL 的目录

目录可以具有默认 ACL，这是一种特殊的 ACL，它定义的是此目录下的对象在创建时继承的访问权限。默认 ACL 影响子目录和文件。

#### 默认 ACL 的效果

将目录的默认 ACL 的权限传递到其中的文件和子目录时，有两种不同的方式：

- 子目录继承父目录的默认 ACL 作为其默认 ACL 和访问 ACL。
- 文件继承默认 ACL 作为其访问 ACL。

创建文件系统对象的所有系统调用都使用 `mode` 参数，该参数定义新创建的文件系统对象的访问权限。如果父目录没有默认 ACL，则从 `mode` 参数传递的权限中去除 `umask` 定义的权限位，同时将结果分配到新对象。如果父目录存在默认 ACL，则分配到新对象的权限位对应于 `mode` 参数的权限和默认 ACL 中定义的权限的重叠部分。这种情况下忽略了 `umask`。

#### 默认 ACL 的应用

以下三个示例说明了目录和默认 ACL 的主要操作：

1. 将默认 ACL 添加到现有目录 `mydir`，语句为：

```
setfacl -d -m group:mascots:r-x mydir
```

setfacl 命令的 -d 选项使 setfacl 在默认 ACL 中执行以下修改（选项 -m）。

仔细查看此命令的结果：

```
getfacl mydir

# file: mydir
# owner: tux
# group: project3
user::rwx
user:geeko:rwx
group::r-x
group:mascots:rwx
mask::rwx
other:---
default:user::rwx
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other:---
```

getfacl 返回访问 ACL 和默认 ACL。默认 ACL 由以 default 开头的所有行组成。虽然您只是对 mascots 组的一个项执行 setfacl 命令来创建默认 ACL，但 setfacl 将自动复制访问 ACL 中的所有其他项来创建有效的默认 ACL。默认 ACL 对访问权限没有直接效果。它们只在创建文件系统对象时起作用。这些新对象只从其父目录的默认 ACL 中继承权限。

2. 在下一个示例中，我们将使用 mkdir 在 mydir 中创建一个子目录，它将继承默认 ACL。

```
mkdir mydir/mysubdir

getfacl mydir/mysubdir

# file: mydir/mysubdir
# owner: tux
# group: project3
user::rwx
group::r-x
group:mascots:r-x
mask::r-x
other:---
default:user::rwx
```

```
default:group::r-x
default:group:mascots:r-x
default:mask::r-x
default:other:---
```

根据预期，新创建的子目录 `mysubdir` 具有父目录的默认 ACL 的权限。`mysubdir` 的访问 ACL 准确反映了 `mydir` 的默认 ACL。该目录将向其从属对象传递的默认 ACL 也是相同的。

3. 使用 `touch` 在 `mydir` 目录中创建一个文件，例如 `touch mydir/myfile`。`ls -l mydir/myfile` 输出以下结果：

```
-rw-r-----+ ... tux project3 ... mydir/myfile
```

`getfacl mydir/myfile` 的输出是：

```
# file: mydir/myfile
# owner: tux
# group: project3
user::rw-
group::r-x          # effective:r--
group:mascots:r-x   # effective:r--
mask::r--
other:---
```

当创建新文件时，`touch` 使用值为 `0666` 的 `mode`，这意味所创建的新文件具有用于所有用户类别的读和写权限，前提是 `umask` 或默认 ACL 中不存在任何其他限制（请参见第 35.3.3 节“默认 ACL 的效果”）。实际上，这意味着 `mode` 值中不包含的所有访问权限均将从各自的 ACL 项中去除。虽然没有从组类别的 ACL 项中去除任何权限，但仍修改了掩码项来屏蔽不在 `mode` 中设置的权限。

这种方式确保应用程序（如编译器）与 ACL 的流畅交互。您可以创建具有有限访问权限的文件，然后将其标记为可执行文件。`mask` 机制确保适当的用户和组可以在需要时执行它们。

### 35.3.4 ACL 检查算法

在为任何进程或应用程序授予访问受 ACL 保护的文件系统对象的权限之前，将应用检查算法。作为一条基本规则，将按以下顺序检查 ACL 项：*拥有者*、*已命名用户*、*所属组*或*已命名组*，以及其他。访问将根据最适合进程的项进行处理。权限不能累加。

如果某个进程属于多个组并且潜在适合多个组项，情况会更为复杂。这时将从具有所需权限的合适项中随机选择一个。它与是哪些项触发了最终结果“已授权访问”无关。同样，如果合适的组项中没有包含所需的权限，则随机选择的项将触发最终结果“访问被拒绝”。

## 35.4 应用程序中的 ACL 支持

ACL 可用于实施非常复杂的权限方案以满足目前应用程序的要求。可以用一种智能方式将传统权限概念和 ACL 结合在一起。像 Samba 一样，基本的文件命令 (`cp`、`mv`、`ls` 等) 支持 ACL。

遗憾的是，许多编辑器和文件管理器仍缺少 ACL 支持。例如，当用 Konqueror 复制文件时，这些文件的 ACL 会丢失。当用编辑器修改文件时，文件的 ACL 有时会被保留，有时则会丢失，这取决于所使用编辑器的备份方式。如果编辑器将更改写入原始文件，访问 ACL 就会被保留。如果编辑器将已更新的内容保存到一个新文件，然后将此文件重命名为旧文件名，则 ACL 就可能丢失，除非此编辑器支持 ACL。除了 star 存档程序外，当前没有任何其他备份应用程序保留 ACL。

## 35.5 参考信息

有关 ACL 的详细信息，请参见 <http://acl.bestbits.at/>。另请参见 `getfacl(1)`、`acl(5)` 和 `setfacl(1)` 的手册页。

# 系统监视实用程序

一些程序和机制（在此对其中的某些进行了介绍）可用于检查系统的状态。同时介绍了一些用于日常工作的实用程序和它们最重要的参数。

36.1	打开的文件的列表：lsof . . . . .	540
36.2	访问文件的用户：fuser . . . . .	541
36.3	文件属性：stat . . . . .	541
36.4	USB 设备：lsusb . . . . .	542
36.5	关于 SCSI 设备的信息：scsiinfo . . . . .	542
36.6	进程：top . . . . .	543
36.7	进程列表：ps . . . . .	544
36.8	进程树：pstree . . . . .	545
36.9	哪些用户在执行哪些操作：w . . . . .	546
36.10	内存使用：free . . . . .	547
36.11	内核信号缓冲区：dmesg . . . . .	547
36.12	文件系统及其使用：mount、df 和 du . . . . .	548
36.13	/proc 文件系统 . . . . .	549
36.14	vmstat、iostat 和 mpstat . . . . .	551
36.15	procinfo . . . . .	551
36.16	PCI 资源：lspci . . . . .	552
36.17	程序运行的系统调用：strace . . . . .	553
36.18	程序运行的库调用：ltrace . . . . .	554
36.19	指定必需的库：ldd . . . . .	554
36.20	有关 ELF 二进制文件的其他信息 . . . . .	555
36.21	进程间通讯：ipcs . . . . .	555
36.22	使用 time 进行时间度量 . . . . .	556

对于所介绍的每个命令，都将提供相关输出的示例。在这些示例中，第一行是命令本身（在美元符号提示符后）。使用方括号([...])表示注释，必要时对较长的行进行换行。较长的行的换行符由反斜线(\)表示。

```
$ command -x -y
output line 1
output line 2
output line 3 is annoyingly long, so long that \
    we have to break it
output line 3
[...]
output line 98
output line 99
```

这里尽量缩短对每个实用程序的说明，从而介绍尽可能多的实用程序。手册页中提供了所有命令的详细信息。大多数命令还接受参数 `--help`，该参数将生成可能参数的简要列表。

## 36.1 打开的文件的列表：lsdf

要查看为具有进程 ID (PID) 的进程打开的所有文件的列表，请使用 `-p`。例如，要查看当前 shell 使用的所有文件，请输入：

```
$ lsdf -p $$
COMMAND PID USER  FD  TYPE DEVICE  SIZE  NODE NAME
zsh      4694  jj   cwd  DIR   0,18   144  25487368 /suse/jj/t (totan:/real-home/jj)
zsh      4694  jj   rtd  DIR   3,2    608           /
zsh      4694  jj   txt  REG   3,2   441296   20414 /bin/zsh
zsh      4694  jj   mem  REG   3,2  104484   10882 /lib/ld-2.3.3.so
zsh      4694  jj   mem  REG   3,2  11648    20610 /usr/lib/zsh/4.2.0/zsh/rlimits.so
[...]
zsh      4694  jj   mem  REG   3,2  13647    10891 /lib/libdl.so.2
zsh      4694  jj   mem  REG   3,2  88036    10894 /lib/libnsl.so.1
zsh      4694  jj   mem  REG   3,2  316410   147725 /lib/libncurses.so.5.4
zsh      4694  jj   mem  REG   3,2  170563   10909 /lib/tls/libm.so.6
zsh      4694  jj   mem  REG   3,2  1349081  10908 /lib/tls/libc.so.6
zsh      4694  jj   mem  REG   3,2    56      12410 /usr/lib/locale/de_DE.utf8/LC_TELEPHONE
[...]
zsh      4694  jj   mem  REG   3,2    59      14393 /usr/lib/locale/en_US/LC_NUMERIC
zsh      4694  jj   mem  REG   3,2  178476   14565 /usr/lib/locale/en_US/LC_CTYPE
zsh      4694  jj   mem  REG   3,2  56444    20598 /usr/lib/zsh/4.2.0/zsh/computil.so
zsh      4694  jj    0u  CHR 136,48    50 /dev/pts/48
zsh      4694  jj    1u  CHR 136,48    50 /dev/pts/48
zsh      4694  jj    2u  CHR 136,48    50 /dev/pts/48
zsh      4694  jj   10u  CHR 136,48    50 /dev/pts/48
```

使用了特殊 shell 变量 `$$`，它的值是 shell 的进程 ID。

如果不带任何参数使用命令 `lsdf`，它将列出当前打开的所有文件。由于有数千个打开的文件，大多数情况下不必列出所有这些文件。但是，所有文件的列表可以与搜索功能组合在一起产生有用的列表。例如，列出所有使用过的字符设备：

```
$ lsof | grep CHR
sshd      4685    root  mem   CHR   1,5      45833 /dev/zero
sshd      4685    root  mem   CHR   1,5      45833 /dev/zero
sshd      4693    jj    mem   CHR   1,5      45833 /dev/zero
sshd      4693    jj    mem   CHR   1,5      45833 /dev/zero
zsh       4694    jj    0u    CHR  136,48    50 /dev/pts/48
zsh       4694    jj    1u    CHR  136,48    50 /dev/pts/48
zsh       4694    jj    2u    CHR  136,48    50 /dev/pts/48
zsh       4694    jj    10u   CHR  136,48    50 /dev/pts/48
X         6476    root  mem   CHR   1,1      38042 /dev/mem
lsof      13478   jj    0u    CHR  136,48    50 /dev/pts/48
lsof      13478   jj    2u    CHR  136,48    50 /dev/pts/48
grep      13480   jj    1u    CHR  136,48    50 /dev/pts/48
grep      13480   jj    2u    CHR  136,48    50 /dev/pts/48
```

## 36.2 访问文件的用户：fuser

它可用于确定当前哪些进程或用户正在访问特定的文件。例如，假定您需要卸装已装入 /mnt 的文件系统。umount 返回“设备正忙”。然后可使用 fuser 命令确定哪些进程正在访问该设备：

```
$ fuser -v /mnt/*

          USER          PID ACCESS COMMAND
/mnt/notes.txt
          jj            26597 f....  less
```

在终止 less 进程之后（该进程在另一个终端上运行），便可以成功卸装该文件系统了。

## 36.3 文件属性：stat

命令 stat 显示文件属性：

```
$ stat xml-doc.txt
  File: 'xml-doc.txt'
  Size: 632          Blocks: 8          IO Block: 4096   regular file
Device: eh/14d Inode: 5938009  Links: 1
Access: (0644/-rw-r--r--)  Uid: (11994/   jj)   Gid: (   50/   suse)
Access: 2004-04-27 20:08:58.000000000 +0200
Modify: 2003-06-03 15:29:34.000000000 +0200
Change: 2003-07-23 17:48:27.000000000 +0200
```

参数 --filesystem 将生成指定文件所在文件系统的属性详细信息：

```
$ stat . --filesystem
File: "."
ID: 0          Namelen: 255      Type: ext2/ext3
Blocks: Total: 19347388  Free: 17831731  Available: 16848938  Size: 4096
Inodes: Total: 9830400  Free: 9663967
```

如果您使用 z shell (zsh), 则必须输入 `/usr/bin/stat`, 因为 z shell 具有 shell 内置 `stat`, 后者带有不同的选项和不同的输出格式:

```
% type stat
stat is a shell builtin
% stat .
device 769
inode 4554808
mode 16877
nlink 12
uid 11994
gid 50
rdev 0
size 4096
atime 1091536882
mtime 1091535740
ctime 1091535740
blksize 4096
blocks 8
link
```

## 36.4 USB 设备: `lsusb`

`lsusb` 命令可列出所有 USB 设备。使用 `-v` 选项, 可打印更加详细的列表。详细信息从目录 `/proc/bus/usb/` 中读取。下面是在连接一个 USB 存储器后的 `lsusb` 输出。最后一行指示存在新设备。

```
Bus 004 Device 001: ID 0000:0000
Bus 003 Device 001: ID 0000:0000
Bus 002 Device 001: ID 0000:0000
Bus 001 Device 001: ID 0000:0000
Bus 001 Device 018: ID 0402:5634 ALi Corp.
```

## 36.5 关于 SCSI 设备的信息: `scsiinfo`

`scsiinfo` 命令可列出关于 SCSI 设备的信息。使用选项 `-l`, 可列出系统已知的所有 SCSI 设备 (通过 `lsscsi` 命令可获取类似的信息)。下面是 `scsiinfo -i /dev/sda` 的输出, 它提供关于一个硬盘的信息。选项 `-a` 可提供更加详细的信息。



```

Inquiry command
-----
Relative Address                0
Wide bus 32                     0
Wide bus 16                     1
Synchronous neg.               1
Linked Commands                 1
Command Queueing                1
SftRe                           0
Device Type                     0
Peripheral Qualifier            0
Removable?                      0
Device Type Modifier            0
ISO Version                     0
ECMA Version                    0
ANSI Version                    3
AENC                            0
TrmIOP                          0
Response Data Format             2
Vendor:                          FUJITSU
Product:                         MAS3367NP
Revision level:                  0104A0K7P43002BE

```

其中有一个缺陷列表，它包括两个硬盘坏区表：第一个表由厂商提供（制造商表），第二个表列出操作中出现的坏区（增长表）。如果增长表中的项目数增加，则最好更换硬盘。

## 36.6 进程：top

top 命令（代表“进程表”）可显示进程的列表，该列表每两秒钟刷新一次。要终止程序，请按 **Ⓢ** 键。参数 `-n 1` 在显示一次进程列表后终止程序。下面是 `top -n 1` 命令的示例输出：

```

top - 14:19:53 up 62 days, 3:35, 14 users, load average: 0.01, 0.02, 0.00
Tasks: 102 total, 7 running, 93 sleeping, 0 stopped, 2 zombie
Cpu(s): 0.3% user, 0.1% system, 0.0% nice, 99.6% idle
Mem: 514736k total, 497232k used, 17504k free, 56024k buffers
Swap: 1794736k total, 104544k used, 1690192k free, 235872k cached

  PID USER      PR  NI  VIRT  RES  SHR  S %CPU  %MEM    TIME+  Command
 1426 root        15   0  116m  41m  18m  S  1.0   8.2   82:30.34 X
20836 jj          15   0   820   820  612  R  1.0   0.2    0:00.03 top
     1 root        15   0   100    96   72  S  0.0   0.0    0:08.43 init

```

```

  2 root      15  0      0      0      0 S  0.0  0.0    0:04.96 keventd
  3 root      34 19      0      0      0 S  0.0  0.0    0:00.99 ksoftirqd_CPU0
  4 root      15  0      0      0      0 S  0.0  0.0    0:33.63 kswapd
  5 root      15  0      0      0      0 S  0.0  0.0    0:00.71 bdflush
    [...]
1362 root      15  0    488  452  404 S  0.0  0.1    0:00.02 nscd
1363 root      15  0    488  452  404 S  0.0  0.1    0:00.04 nscd
1377 root      17  0     56   4     4 S  0.0  0.0    0:00.00 mingetty
1379 root      18  0     56   4     4 S  0.0  0.0    0:00.01 mingetty
1380 root      18  0     56   4     4 S  0.0  0.0    0:00.01 mingetty

```

如果当 `top` 正在运行时按 `F6` 键，则将打开一个菜单，使用该菜单可以对输出的格式进行全面更改。

参数 `-U UID` 只监视与特定用户关联的进程。⟨*UID*⟩ 应替换为用户的用户 ID。`top -U $(id -u username)` 根据用户名返回用户的 UID 并显示该用户的进程。

## 36.7 进程列表：ps

命令 `ps` 生成进程的列表。如果添加了参数 `r`，则只显示当前正在使用计算时间的进程：

```

$ ps r
  PID TTY          STAT TIME COMMAND
22163 pts/7        R    0:01 -zsh
 3396 pts/3        R    0:03 emacs new-makedoc.txt
20027 pts/7        R    0:25 emacs xml/common/utilities.xml
20974 pts/7        R    0:01 emacs jj.xml
27454 pts/7        R    0:00 ps r

```

书写此参数时一定不能带减号。不同的参数在书写时有时带减号，有时不带减号。手册页中有关此问题的复杂说明很容易让潜在用户望而止步，但幸好 `ps --help` 命令可以生成简要的帮助页。

要检查有多少 `emacs` 进程正在运行，请使用：

```

$ ps x | grep emacs
 1288 ?          S    0:07 emacs
 3396 pts/3        S    0:04 emacs new-makedoc.txt
 3475 ?          S    0:03 emacs .Xresources
20027 pts/7        S    0:40 emacs xml/common/utilities.xml
20974 pts/7        S    0:02 emacs jj.xml

```

```

$ pidof emacs
20974 20027 3475 3396 1288

```

参数 `-p` 通过进程 ID 选择进程：

```
$ ps www -p $(pidof xterm)
  PID TTY          STAT       TIME COMMAND
  9025 ?            S           0:01 xterm  -g 100x45+0+200
  9176 ?            S           0:00 xterm  -g 100x45+0+200
 29854 ?            S           0:21 xterm  -g 100x75+20+0 -fn \
    -B&H-LucidaTypewriter-Medium-R-Normal-Sans-12-120-75-75-M-70-iso10646-1
  4378 ?            S           0:01 xterm  -bg MistyRose1 -T root -n root -e su -l
 25543 ?            S           0:02 xterm  -g 100x45+0+200
 22161 ?            R           0:14 xterm  -g 100x45+0+200
 16832 ?            S           0:01 xterm  -bg MistyRose1 -T root -n root -e su -l
 16912 ?            S           0:00 xterm  -g 100x45+0+200
 17861 ?            S           0:00 xterm  -bg DarkSeaGreen1 -g 120x45+40+300
 19930 ?            S           0:13 xterm  -bg LightCyan
 21686 ?            S           0:04 xterm  -g 100x45+0+200 -fn \
lucidasanstypewriter-12
 23104 ?            S           0:00 xterm  -g 100x45+0+200
 26547 ?            S           0:00 xterm  -g 100x45+0+200
```

可以根据需要设置进程列表的格式。选项 `-L` 返回所有关键字的列表。输入以下命令可以生成所有进程按内存使用量排序的列表：

```
$ ps ax --format pid,rss,cmd --sort rss
  PID  RSS CMD
    2    0 [ksoftirqd/0]
    3    0 [events/0]
   17    0 [kblockd/0]
[... ]
10164 5260 xterm
31110 5300 xterm
17010 5356 xterm
 3896 29292 /usr/X11R6/bin/X -nolisten tcp -br vt7 -auth /var/lib/xdm/authdir/au
```

## 36.8 进程树： `ps tree`

命令 `ps tree` 生成树结构的进程列表：

```
$ ps tree
init--+-atd
  | -3*[automount]
  | -bdflush
  | -cron
[... ]
  | -usb-storage-1
```

```

|-usb-storage-2
|-10*[xterm---zsh]
|-xterm---zsh---mutt
|-2*[xterm---su---zsh]
|-xterm---zsh---ssh
|-xterm---zsh---pstree
|-ypbind---ypbind---2*[ypbind]
`-zsh---startx---xinit4-+-X
                                `-ctwm+-xclock
                                    |-xload
                                        `-xosview.bin

```

参数 `-p` 将进程 ID 添加到给定的名称。要让命令行也显示出来，请使用 `-a` 参数：

```

$ pstree -pa
init,1
  |-atd,1255
  [...]
  `zsh,1404
    `startx,1407 /usr/X11R6/bin/startx
      `xinit4,1419 /suse/jj/.xinitrc [...]
        |-X,1426 :0 -auth /suse/jj/.Xauthority
          `ctwm,1440
            |-xclock,1449 -d -geometry -0+0 -bg grey
            |-xload,1450 -scale 2
            `xosview.bin,1451 +net -bat +net

```

## 36.9 哪些用户在执行哪些操作：w

使用命令 `w`，可以查看哪些用户登录到系统上以及每个用户正在执行哪些操作。例如：

```

$ w
15:17:26 up 62 days,  4:33, 14 users,  load average: 0.00, 0.04, 0.01
USER      TTY      LOGIN@  IDLE   JCPU   PCPU WHAT
jj        pts/0    30Mar04  4days 0.50s  0.54s xterm -e su -l
jj        pts/1    23Mar04  5days 0.20s  0.20s -zsh
jj        pts/2    23Mar04  5days 1.28s  1.28s -zsh
jj        pts/3    23Mar04  3:28m  3.21s  0.50s -zsh
[...]
jj        pts/7    07Apr04  0.00s  9.02s  0.01s w
jj        pts/9    25Mar04  3:24m  7.70s  7.38s mutt
[...]
jj        pts/14   12:49   37:34  0.20s  0.13s ssh totan

```

最后一行显示用户 `jj` 建立了与计算机 `totan` 的安全 shell (ssh) 连接。如果其他系统的任何用户远程登录，则参数 `-f` 将显示这些用户从其上建立连接的计算机。

## 36.10 内存使用：free

实用程序 `free` 检查 RAM 使用情况。将显示有关可用内存和已使用内存（以及交换区域）的详细信息：

```
$ free
              total          used          free      shared    buffers     cached
Mem:          514736        273964        240772           0        35920        42328
-/+ buffers/cache:    195716        319020
Swap:         1794736        104096        1690640
```

使用 `-m`，所有大小将以 MB 为单位表示：

```
$ free -m
              total          used          free      shared    buffers     cached
Mem:           502            267            235           0           35           41
-/+ buffers/cache:    191            311
Swap:          1752            101            1651
```

下面一行中包含了非常有趣的信息：

```
-/+ buffers/cache:    191            311
```

这将计算缓冲区和超速缓存所占用的内存量。参数 `-d delay` 可以确保显示每  $\langle delay \rangle$  秒刷新一次。例如，`free -d 1.5` 每 1.5 秒进行一次更新。

## 36.11 内核信号缓冲区：dmesg

Linux 内核在信号缓冲区中保存某些讯息。要查看这些讯息，请输入命令 `dmesg`：

```
$ dmesg
[...]
sdc : READ CAPACITY failed.
sdc : status = 1, message = 00, host = 0, driver = 08
```

```
Info fld=0xa00 (nonstd), Current sd00:00: sense key Not Ready
sdc : block size assumed to be 512 bytes, disk size 1GB.
sdc: test WP failed, assume Write Enabled
sdc: I/O error: dev 08:20, sector 0
I/O error: dev 08:20, sector 0
I/O error: dev 08:20, sector 2097144
I/O error: dev 08:20, sector 2097144
I/O error: dev 08:20, sector 0
I/O error: dev 08:20, sector 0
unable to read partition table
I/O error: dev 08:20, sector 0
nfs: server totan not responding, still trying
nfs: server totan OK
```

最后一行指出 NFS 服务器 totan 中存在临时问题。到此位置为止的这些行是由 USB 闪存盘的插入触发的。以前的事件记录在文件 `/var/log/messages` 和 `/var/log/warn` 中。

## 36.12 文件系统及其使用： mount、df 和 du

命令 `mount` 显示在哪个装入点装入哪个文件系统（设备和类型）：

```
$ mount
/dev/hdb2 on / type ext2 (rw)
proc on /proc type proc (rw)
devpts on /dev/pts type devpts (rw,mode=0620,gid=5)
/dev/hdal on /data type ext2 (rw)
shmfs on /dev/shm type shm (rw)
usbdevfs on /proc/bus/usb type usbdevfs (rw)
automount(pid1012) on /suse type autofs \
    (rw,fd=5,pgrp=1012,minproto=2,maxproto=3)
totan:/real-home/jj on /suse/jj type nfs \
    (rw,nosuid,rsize=8192,wsiz=8192,hard,intr,nolock,addr=10.10.0.1)
```

使用命令 `df` 可以获得有关文件系统全部使用情况的信息。参数 `-h`（或 `--human-readable`）将输出转换为普通用户可以理解的形式。

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/hdb2       7.4G  5.1G  2.0G  73% /
/dev/hdal       74G   5.8G   65G   9% /data
shmfs           252M   0    252M   0% /dev/shm
totan:/real-home/jj 350G 324G  27G  93% /suse/jj
```

NFS 文件服务器 totan 的用户应该立即清除他们的主目录。使用命令 `du` 可以显示给定目录及其子目录中所有文件的总大小。使用参数 `-s` 将不输出详细信息。`-h` 再次将数据转换为普通用户可以理解的形式。使用此命令：

```
$ du -sh ~
361M    /suse/jj
```

查看您自己的主目录占用了多少空间。

## 36.13 /proc 文件系统

`/proc` 文件系统是一个假文件系统，在该文件系统中，内核以虚拟文件的形式保留重要信息。例如，使用以下命令显示 CPU 类型：

```
$ cat /proc/cpuinfo
processor       : 0
vendor_id     : AuthenticAMD
cpu family    : 6
model        : 8
model name    : AMD Athlon(tm) XP 2400+
stepping     : 1
cpu MHz      : 2009.343
cache size   : 256 KB
fdiv_bug     : no
[...]
```

可以使用以下命令查询中断的分配和使用：

```
$ cat /proc/interrupts
          CPU0
0: 537544462          XT-PIC  timer
1:   820082          XT-PIC  keyboard
2:           0          XT-PIC  cascade
8:           2          XT-PIC  rtc
9:           0          XT-PIC  acpi
10:       13970          XT-PIC  usb-uhci, usb-uhci
11: 146467509          XT-PIC  ehci_hcd, usb-uhci, eth0
12:   8061393          XT-PIC  PS/2 Mouse
14:   2465743          XT-PIC  ide0
15:    1355           XT-PIC  ide1
NMI:           0
LOC:           0
ERR:           0
MIS:           0
```

一些重要的文件及其内容如下：

`/proc/devices` 可用设备

`/proc/modules` 装载的内核模块

`/proc/cmdline` 内核命令行

`/proc/meminfo` 有关内存使用的详细信息

`/proc/config.gz` 当前运行的内核的 `gzip` 压缩配置文件

文本文件 `/usr/src/linux/Documentation/filesystems/proc.txt` 中提供了详细信息。`/proc/<NNN>` 目录中提供了当前运行进程的信息，其中 `<NNN>` 是相关进程的进程 ID (PID)。每个进程都可以在 `/proc/self/` 中找到它自己的特性：

```
$ ls -l /proc/self
lrwxrwxrwx 1 root root 64 Apr 29 13:52 /proc/self -> 27585
```

```
$ ls -l /proc/self/
total 0
dr-xr-xr-x 2 jj suse 0 Apr 29 13:52 attr
-r----- 1 jj suse 0 Apr 29 13:52 auxv
-r--r--r-- 1 jj suse 0 Apr 29 13:52 cmdline
lrwxrwxrwx 1 jj suse 0 Apr 29 13:52 cwd -> /suse/jj/t
-r--r--r-- 1 jj suse 0 Apr 29 13:52 delay
-r----- 1 jj suse 0 Apr 29 13:52 environ
lrwxrwxrwx 1 jj suse 0 Apr 29 13:52 exe -> /bin/ls
dr-x----- 2 jj suse 0 Apr 29 13:52 fd
-rw----- 1 jj suse 0 Apr 29 13:52 mapped_base
-r--r--r-- 1 jj suse 0 Apr 29 13:52 maps
-rw----- 1 jj suse 0 Apr 29 13:52 mem
-r--r--r-- 1 jj suse 0 Apr 29 13:52 mounts
lrwxrwxrwx 1 jj suse 0 Apr 29 13:52 root -> /
-r--r--r-- 1 jj suse 0 Apr 29 13:52 stat
-r--r--r-- 1 jj suse 0 Apr 29 13:52 statm
-r--r--r-- 1 jj suse 0 Apr 29 13:52 status
dr-xr-xr-x 3 jj suse 0 Apr 29 13:52 task
-r--r--r-- 1 jj suse 0 Apr 29 13:52 wchan
```

`maps` 文件中包含可执行文件和库的地址指派：

```
$ cat /proc/self/maps
08048000-0804c000 r-xp 00000000 03:02 22890 /bin/cat
0804c000-0804d000 rw-p 00003000 03:02 22890 /bin/cat
0804d000-0806e000 rwxp 0804d000 00:00 0
40000000-40016000 r-xp 00000000 03:02 10882 /lib/ld-2.3.3.so
40016000-40017000 rw-p 00015000 03:02 10882 /lib/ld-2.3.3.so
40017000-40018000 rw-p 40017000 00:00 0
4002b000-40135000 r-xp 00000000 03:02 10908 /lib/tls/libc.so.6
40135000-4013d000 rw-p 0010a000 03:02 10908 /lib/tls/libc.so.6
4013d000-40141000 rw-p 4013d000 00:00 0
bffffe000-c0000000 rw-p bffffe000 00:00 0
fffffe000-fffff000 ---p 00000000 00:00 0
```



## 36.14 vmstat、iostat 和 mpstat

实用程序 `vmstat` 可报告虚拟内存统计信息。它读取文件 `/proc/meminfo`、`/proc/stat` 和 `/proc/*/stat`。它可用于识别系统性能的瓶颈。

`iostat` 命令报告有关 CPU 以及设备与分区输入输出的统计信息。所显示的信息取自文件 `/proc/stat` 和 `/proc/partitions`。该输出可用于更好地平衡硬盘之间的输入和输出负载。`mpstat` 命令报告与 CPU 相关的统计信息。

## 36.15 procinfo

命令 `procinfo` 对 `/proc` 文件系统中的重要信息进行了总结：

```
$ procinfo
Linux 2.6.4-54.5-default (geeko@buildhost) (gcc 3.3.3 ) #1 lCPU [roth.suse.de]

Memory:      Total      Used      Free      Shared    Buffers
Mem:         516696    513200    3496      0          43284
Swap:        530136    1352     528784

Bootup: Wed Jul 7 14:29:08 2004      Load average: 0.07 0.04 0.01 1/126 5302

user  :      2:42:28.08  1.3%  page in :      0
nice  :      0:31:57.13  0.2%  page out:      0
system:  0:38:32.23  0.3%  swap in  :      0
idle  :  3d 19:26:05.93 97.7%  swap out:      0
uptime: 4d 0:22:25.84      context :207939498

irq 0: 776561217 timer          irq 8:      2 rtc
irq 1: 276048 i8042             irq 9:     24300 VIA8233
irq 2: 0 cascade [4]          irq 11:  38610118 acpi, eth0, uhci_hcd
irq 3: 3                       irq 12:  3435071 i8042
irq 4: 3                       irq 14:  2236471 ide0
irq 6: 2                       irq 15:   251 ide1
```

要查看所有信息，请使用参数 `-a`。参数 `-nN` 每  $\langle N \rangle$  秒更新一次信息。在这种情况下，按 `Ⓢ` 键终止程序。

默认情况下显示累积值。使用参数 `-d` 将生成差异值。`procinfo -dn5` 显示最近 5 秒内更改的值：

```
Memory:      Total      Used      Free      Shared    Buffers    Cached
Mem:         0          2         -2         0          0          0
Swap:        0          0          0          0          0          0
```

```

Bootup: Wed Feb 25 09:44:17 2004      Load average: 0.00 0.00 0.00 1/106 31902

user  :      0:00:00.02   0.4%  page in :      0  disk 1:      0r      0w
nice  :      0:00:00.00   0.0%  page out:      0  disk 2:      0r      0w
system:      0:00:00.00   0.0%  swap in :      0  disk 3:      0r      0w
idle  :      0:00:04.99  99.6%  swap out:      0  disk 4:      0r      0w
uptime: 64d  3:59:12.62      context :      1087

irq 0:      501 timer                irq 10:      0  usb-uhci, usb-uhci
irq 1:      1  keyboard              irq 11:      32  ehci_hcd, usb-uhci,
irq 2:      0  cascade [4]          irq 12:      132 PS/2 Mouse
irq 6:      0                        irq 14:      0  ide0
irq 8:      0  rtc                  irq 15:      0  ide1
irq 9:      0  acpi

```

## 36.16 PCI 资源: lspci

命令 `lspci` 列出 PCI 资源:

```

$ lspci
00:00.0 Host bridge: VIA Technologies, Inc. \
    VT8366/A/7 [Apollo KT266/A/333]
00:01.0 PCI bridge: VIA Technologies, Inc. \
    VT8366/A/7 [Apollo KT266/A/333 AGP]
00:0b.0 Ethernet controller: Digital Equipment Corporation \
    DECchip 21140 [FasterNet] (rev 22)
00:10.0 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.1 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.2 USB Controller: VIA Technologies, Inc. USB (rev 80)
00:10.3 USB Controller: VIA Technologies, Inc. USB 2.0 (rev 82)
00:11.0 ISA bridge: VIA Technologies, Inc. VT8235 ISA Bridge
00:11.1 IDE interface: VIA Technologies, Inc. VT82C586/B/686A/B \
    PIPC Bus Master IDE (rev 06)
00:11.5 Multimedia audio controller: VIA Technologies, Inc. \
    VT8233 AC97 Audio Controller (rev 50)
01:00.0 VGA compatible controller: Matrox Graphics, Inc. \
    MGA G550 AGP (rev 01)

```

使用 `-v` 可以生成更加详细的列表:

```

$ lspci -v
[...]
01:00.0 \
    VGA compatible controller: Matrox Graphics, Inc. MGA G550 AGP (rev 01) \
    (prog-if 00 [VGA])
    Subsystem: Matrox Graphics, Inc. Millennium G550 Dual Head DDR 32Mb
    Flags: bus master, medium devsel, latency 32, IRQ 10
    Memory at d8000000 (32-bit, prefetchable) [size=32M]
    Memory at da000000 (32-bit, non-prefetchable) [size=16K]
    Memory at db000000 (32-bit, non-prefetchable) [size=8M]

```

```
Expansion ROM at <unassigned> [disabled] [size=128K]
Capabilities: <available only to root>
```

文件 `/usr/share/pci.ids` 中提供有关设备名称解析的信息。此文件中未列出的 PCI ID 被标为“未知设备”。

参数 `-vv` 生成程序可查询的所有信息。要查看纯数字值，应该使用参数 `-n`。

## 36.17 程序运行的系统调用：strace

实用程序 `strace` 使您可以跟踪当前运行的进程的所有系统调用。以正常方式输入命令，在行开头添加 `strace`：

```
$ strace -e open ls

execve("/bin/ls", ["ls"], [/* 88 vars */]) = 0
uname({sys="Linux", node="edison", ...}) = 0
brk(0) = 0x805b000
old_mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) \
    = 0x40017000
open("/etc/ld.so.preload", O_RDONLY) = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY) = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=76333, ...}) = 0
old_mmap(NULL, 76333, PROT_READ, MAP_PRIVATE, 3, 0) = 0x40018000
[...]
ioctl(1, SNDCTL_TMR_TIMEBASE or TCGETS, {B38400 opost isig icanon echo ...}) = 0
ioctl(1, TIOCGWINSZ, {ws_row=53, ws_col=110, ws_xpixel=897, ws_ypixel=693}) = 0
open(".", O_RDONLY|O_NONBLOCK|O_LARGEFILE|O_DIRECTORY) = 3
fstat64(3, {st_mode=S_IFDIR|0755, st_size=144, ...}) = 0
fcntl64(3, F_SETFD, FD_CLOEXEC) = 0
getdents64(3, /* 5 entries */, 4096) = 160
getdents64(3, /* 0 entries */, 4096) = 0
close(3) = 0
fstat64(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(136, 48), ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) \
    = 0x40018000
write(1, "ltrace-ls.txt myfile.txt strac"... , 41) = 41
munmap(0x40018000, 4096) = 0
exit_group(0)
```

例如，要跟踪打开特定文件的所有尝试，请使用以下命令：

```
$ strace -e open ls myfile.txt

open("/etc/ld.so.preload", O_RDONLY) = -1 ENOENT (No such file or directory)
open("/etc/ld.so.cache", O_RDONLY) = 3
open("/lib/tls/librt.so.1", O_RDONLY) = 3
```

```

open("/lib/libacl.so.1", O_RDONLY) = 3
open("/lib/libselinux.so.1", O_RDONLY) = 3
open("/lib/tls/libc.so.6", O_RDONLY) = 3
open("/lib/tls/libpthread.so.0", O_RDONLY) = 3
open("/lib/libattr.so.1", O_RDONLY) = 3
open("/proc/mounts", O_RDONLY) = 3
[...]
open("/proc/filesystems", O_RDONLY) = 3
open("/proc/self/attr/current", O_RDONLY) = 4

```

要跟踪所有子进程，请使用参数 `-f`。可以在很大程度上控制 `strace` 的行为和输出格式。有关信息，请参见 `man strace`。

## 36.18 程序运行的库调用：ltrace

命令 `ltrace` 使您可以跟踪进程的库调用。此命令的使用方式与 `strace` 类似。参数 `-c` 输出所发生的库调用的次数和持续时间：

```

$ ltrace -c find /usr/share/doc
% time      seconds  usecs/call   calls   errors syscall
-----
 86.27      1.071814    30          35327   write
10.15      0.126092    38          3297   getdents64
 2.33      0.028931    3          10208   lstat64
 0.55      0.006861    2           3122   1 chdir
 0.39      0.004890    3           1567   2 open
[...]
 0.00      0.000003    3            1   uname
 0.00      0.000001    1            1   time
-----
100.00      1.242403    58269      3 total

```

## 36.19 指定必需的库：ldd

命令 `ldd` 可用于找出哪些库将装载以参数形式指定的动态可执行文件：

```

$ ldd /bin/ls
linux-gate.so.1 => (0xffffe000)
librt.so.1 => /lib/tls/librt.so.1 (0x4002b000)
libacl.so.1 => /lib/libacl.so.1 (0x40033000)
libselinux.so.1 => /lib/libselinux.so.1 (0x40039000)
libc.so.6 => /lib/tls/libc.so.6 (0x40048000)

```

```
libpthread.so.0 => /lib/tls/libpthread.so.0 (0x4015d000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
libattr.so.1 => /lib/libattr.so.1 (0x4016d000)
```

静态二进制文件不需要任何动态库：

```
$ ldd /bin/sash
        not a dynamic executable
$ file /bin/sash
/bin/sash: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), \
for GNU/Linux 2.2.5, statically linked, stripped
```

## 36.20 有关 ELF 二进制文件的其他信息

可以使用 `readelf` 实用程序读取二进制文件的内容。这甚至可用于为其他硬件体系结构生成的 ELF 文件：

```
$ readelf --file-header /bin/ls
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00
  Class:                               ELF32
  Data:                                   2's complement, little endian
  Version:                               1 (current)
  OS/ABI:                               UNIX - System V
  ABI Version:                           0
  Type:                                   EXEC (Executable file)
  Machine:                               Intel 80386
  Version:                               0x1
  Entry point address:                   0x8049b40
  Start of program headers:              52 (bytes into file)
  Start of section headers:              76192 (bytes into file)
  Flags:                                  0x0
  Size of this header:                   52 (bytes)
  Size of program headers:               32 (bytes)
  Number of program headers:             9
  Size of section headers:               40 (bytes)
  Number of section headers:             29
  Section header string table index:     26
```

## 36.21 进程间通讯： `ipcs`

命令 `ipcs` 生成当前正在使用的 IPC 资源的列表：

```

$ ipcs
----- Shared Memory Segments -----
key      shmid      owner      perms      bytes      nattch     status
0x000027d9 5734403    toms       660        64528      2
0x00000000 5767172    toms       666        37044      2
0x00000000 5799941    toms       666        37044      2

----- Semaphore Arrays -----
key      semid      owner      perms      nsems
0x000027d9 0          toms       660        1

----- Message Queues -----
key      msqid      owner      perms      used-bytes  messages

```

## 36.22 使用 time 进行时间度量

可以使用 `time` 实用程序确定命令执行所花的时间。此实用程序以两个版本提供：作为 shell 内置程序以及作为程序 (`/usr/bin/time`)。

```

$ time find . > /dev/null

real    0m4.051s
user    0m0.042s
sys     0m0.205s

```

# Part V

## 附 錄







# 信息源和文档

适用于 SUSE LINUX 系统的信息源的范围十分广泛。有些信息源专门针对 SUSE，但多数都是更为通用的信息源。有些已存在于您的系统或安装媒体中，其他则可以通过因特网访问。

## SUSE 文档

有关详细信息，请参见 RPM 包中以 HTML 或 PDF 格式提供的指南（`suselinux-userguide_en` 和 `suselinux-adminguide_en`）。在标准安装过程中，这些指南安装在 `/usr/share/doc/manual/` 目录下。通过 SUSE 帮助中心可以访问这些信息。

## Linux 文档计划 (LDP)

Linux 文档计划（请参见 <http://www.tldp.org/>）是由多名志愿者组成的小组，负责编写有关 Linux 的文档。LDP 提供 HOWTO、常见问题解答及各种指南，所有这些均已免费发布，不受许可证限制。

HOWTO 即分步说明，供最终用户、系统管理员和编程人员使用。例如，HOWTO 中可以说明 DHCP 服务器的创建过程及要注意的几点事项，但并不说明如何安装 Linux 系统本身。作为一条准则，这种文档应尽量保持通用，以便适用于所有版本。`howto` 包提供 ASCII 格式的 HOWTO 文件。倾向于使用 HTML 格式的用户应安装 `howtoenh`。

FAQ（frequently asked questions，常见问题解答）是经常在邮件列表中提出的关于某些方面的问题和解答的汇总。常见的问题如“什么是 LDAP？”或“什么是 RAID？”。这类文字通常很简短。

指南是指比 HOWTO 和常见问题解答更为详细地讨论某个主题的文档。例如：内核编程和网络管理等方面的文档。指南旨在为读者提供更为详细的信息。

有些 LDP 文档还提供其他格式，如 PDF、单独的 HTML 页和多个 HTML 页、PostScript，还可以作为 SGML 或 XML 源。有时还提供其他语言版本。

## 手册页和信息页

手册页是命令、系统调用、文件格式或类似项目的帮助文本。手册页通常分为多个部分，如名称、语法、说明、选项和文件。

要显示手册页，请输入 `man`，后接命令名称，如 `man ls`；该命令将显示 `ls` 命令的帮助文本。使用光标键可移动可见区域。使用 `Q` 可退出 `man`。要打印手册页（例如为命令 `ls` 打印手册页），可输入命令 `man -Tps | lpr`。有关 `man` 命令的更多帮助，请使用 `--help` 选项或 `man(man man)` 的手册页。

有些文档还能以 `info` 格式提供，例如，可以为 `grep` 提供 `info` 格式的文档。使用 `info grep` 可访问该文档。

信息页比手册页更为详细。信息页划分为不同的节点—即可以使用信息阅读器（与万维网浏览器非常类似）阅读的页面。使用 `P`（上一页）和 `N`（下一页）可以在信息页中导航。使用 `Q` 可退出 `info`。其他键在 `info` 文档中列出 (`info info`)。

在 Konqueror 中既可阅读手册页也可以阅读信息页。在 URL 行中输入 `man:<命令>` 或 `info:<命令>` 可打开所需的文档。

## 标准和规范

有很多信息源都提供有关标准和规范的信息。

**www.linuxbase.org** 自由标准组织 (Free Standards Group) 是一个独立的旨在促进自由软件和开放源代码软件发布的非营利组织。该组织致力于制定独立于各版本的标准，以期实现上述目标。包括重要的 LSB (Linux Standard Base, Linux 标准库) 在内的若干标准均由该组织负责维护。

**http://www.w3.org** 万维网联合会 (World Wide Web Consortium, W3C) 当属最知名的标准化组织之一。该组织由 Tim Berners-Lee 在 1994 年 10 月创办，主要致力于万维网技术的标准化。W3C 提倡发布开放、不受许可证限制并且与制造商无关的规范，如 HTML、XHTML 和 XML。这些万维网标准由多个工作组分四个阶段完成，最后以 W3C 建议 (REC) 的形式公诸于世。

<http://www.oasis-open.org> OASIS (Organization for the Advancement of Structured Information Standards, 结构化信息标准促进组织) 是一个国际联盟, 专门负责开发万维网安全、电子商务、交易事务处理、物流和多个市场之间的互操作性等方面的标准。

<http://www.ietf.org> 因特网工程任务组 (Internet Engineering Task Force, IETF) 是一个十分活跃的国际合作组织, 由众多研究人员、网络设计人员、供应商和用户组成。该组织侧重于开发因特网体系结构以及借助协议确保因特网平稳运行。

每个 IETF 标准均作为 RFC (Request for Comments, 请求注释) 发布, 并且免费提供。共有六种 RFC: 建议标准、草拟标准、因特网标准、试用协议、信息文档及历史标准。从狭义上讲, 只有前三种 (建议、草拟和完整标准) 才属于 IETF 标准 (请参见 <http://www.ietf.org/rfc/rfc1796.txt>)。

<http://www.ieee.org> 电气与电子工程师协会 (Institute of Electrical and Electronics Engineers, IEEE) 是负责在信息技术、电信、医药保健、运输和其他领域内制定标准的组织。IEEE 标准需付费才可获得。

<http://www.iso.org> ISO (International Organization for Standards, 国际标准化组织) 委员会是全球最大的标准开发组织, 负责维护由 140 多个国家/地区的国家/地区标准化协会构成的庞大网络。ISO 标准需付费才可获得。

<http://www.din.de>, <http://www.din.com>

德国标准化协会 (Deutsches Institut für Normung, DIN) 是经注册的科技领域内的协会。DIN 宣称该组织是“负责德国标准化的协会, 并且在欧洲和全球标准化组织中代表德国的利益”。

该协会汇集了制造商、客户、贸易专业组织、服务公司、科学家和其他对制定标准有兴趣的人员。该协会制定的标准需付费才可获得, 可通过 DIN 主页订购。



# File System Checking

## reiserfsck 的手册页

REISERFSCK(8)

REISERFSCK(8)

### 名称

reiserfsck - 检查 Linux Reiserfs 文件系统

### 语法

```
reiserfsck [ -afprVy ] [ --rebuild-sb | --check | --fix-  
fixable | --rebuild-tree | --clean-attributes ] [ -j |  
--journal device ] [ -z | --adjust-size ] [ -n | --nolog ]  
[ -l | --logfile file ] [ -q | --quiet ] [ -y | --yes ] [  
-S | --scan-whole-partition ] [ --no-journal-available ]  
device
```

### 说明

Reiserfsck 搜索设备上的 Reiserfs 文件系统、重放任何必要的事务，以及检查或修复文件系统。

设备是与设备或分区（例如，/dev/hdXX 对应于 IDE 磁盘分区，/dev/sdXX 对应于 SCSI 磁盘分区）对应的特殊文件。

### 选项

--rebuild-sb

此选项恢复 Reiserfs 分区上的超级块。通常，只有当装入报告“read\_super\_block: can't find a reiserfs file system”并且您确定存在 Reiserfs 文件系统时，才需要此选项。

--check

此默认操作检查文件系统一致性并报告，但不对所找到的任何损坏进行修复。此选项可用于只读文件系统装入。

`--fix-fixable`  
 此选项恢复不需要重建整个文件系统树 (`--rebuild-tree`) 的损坏类型。通常, 只有当 `--check` 选项报告“corruption that can be fixed with `--fix-fixable`”时才需要此选项。这包括: 将无效的数据块指针归零、更正目录的 `st_size` 和 `st_blocks`, 以及删除无效的目录项。

`--rebuild-tree`  
 此选项使用设备上找到的页节点重建整个文件系统树。通常, 只有当 `--check` 选项报告“corruption that can be fixed only during `--rebuild-tree`”时才需要此选项。强烈建议您在尝试使用 `--rebuild-tree` 选项前制作整个分区的备份副本。

`--clean-attributes`  
 此选项清除 `Stat-Data` 项的保留字段。

`--journal device, -j device`  
 此选项提供当前文件系统日记的设备名。当日记驻留在与主数据设备不同的设备上时 (虽然可以使用专家选项 `--no-journal-available` 避免这种情况), 需要使用此选项。

`--adjust-size, -z`  
 此选项使 `reiserfsck` 更正大于所发现的最后一个字节的偏移量的文件大小。这意味着将去除文件末尾的孔。小于所发现的最后一个字节的偏移量的文件大小由 `--fix-fixable` 进行更正。

`--logfile file, -l file`  
 此选项使 `reiserfsck` 将它找到的任何损坏报告到指定的日志文件而不是 `stderr`。

`--nolog, -n`  
 此选项禁止 `reiserfsck` 报告任何类型的损坏。

`--quiet, -q`  
 此选项禁止 `reiserfsck` 报告其进度。

`--yes, -y`  
 此选项禁止 `reiserfsck` 在通知您它将执行的操作后要求您进行确认, 假定为“是”。出于安全原因, 它与 `--rebuild-tree` 选项一起使用。

`-a, -p` 这些选项通常是在自动检查 `/etc/fstab` 中列出的那些分区期间由 `fsck -A` 传递的。这些选项使 `reiserfsck` 输出有关指定文件系统的一些信息、检查是否在超级块中设置了错误标志并执行某些轻量检查。如果这些检查发

现损坏或发现超级块中设置有指示（可能是可修复的）损坏的标志，则 `reiserfsck` 切换到 `fix-fixable` 方式。如果发现超级块中设置有指示致命损坏的标志，则 `reiserfsck` 结束并返回一个错误。

`-V` 此选项输出 `reiserfsprogs` 版本并退出。

`-r`, `-f` 这些选项被忽略。

#### 专家选项

除非您了解正在执行的操作，否则不要使用这些选项。如果因为使用这些选项导致数据丢失，我们概不负责。

`--no-journal-available`

此选项允许 `reiserfsck` 在日记设备不可用时继续执行。

当日记位于主数据设备上时，此选项不起作用。

注意：在此操作后，必须使用 `reiserfstune` 指定一个新的日记设备。

`--scan-whole-partition, -S`

此选项使 `--rebuild-tree` 扫描整个分区，而不仅仅是分区上使用的空间。

#### 使用示例

1. 您认为 `/dev/hda1` 上的 `reiserfs` 分区可能出现了问题或者只是想执行定期磁盘检查。
2. 运行 `reiserfsck --check --logfile check.log /dev/hda1`。  
如果 `reiserfsck --check` 退出并返回状态 0，则表示没有发现任何错误。
3. 如果 `reiserfsck --check` 退出并返回状态 1（以及有关可修复损坏的报告），则表示您应该运行  
`reiserfsck --fix-fixable --logfile fixable.log /dev/hda1`。
4. 如果 `reiserfsck --check` 退出并返回状态 2（以及有关致命损坏的报告），则表示您应该运行  
`reiserfsck --rebuild-tree`。如果 `reiserfsck --check` 在某方面失败，则您还应该运行 `reiserfsck --rebuild-tree`，但我们还建议您将此作为错误报告提交。
5. 在运行 `reiserfsck --rebuild-tree` 之前，请制作整个分区的备份。然后运行  
`reiserfsck --rebuild-tree --logfile rebuild.log /dev/hda1`。
6. 如果 `--rebuild-tree` 步骤失败或未恢复为所预期的状态，请将此作为错误报告提交。请尽量提供尽可能多的信息，我们将尽力帮助您解决问题。

#### 退出代码

reiserfsck 使用以下退出代码:

- 0 - 没有错误。
- 1 - 已更正文件系统错误。
- 4 - 未更正文件系统致命错误, 需要启动 reiserfsck --rebuild-tree。
- 6 - 未更正文件系统可修复错误, 需要启动 reiserfsck --fix-fixable。
- 8 - 操作错误。
- 16 - 用法或语法错误。

#### 作者

此版本的 reiserfsck 由 Vitaly Fertman 编写, 电子邮件为 <vitaly@namesys.com>。

#### BUGS

有可能存在一些错误。请将错误报告给 ReiserFS 邮件列表 <reiserfs-list@namesys.com>。

#### TODO

更快的恢复、信号处理、i/o 错误处理等

#### 另请参见

mkreiserfs(8)、reiserfstune(8)、resize\_reiserfs(8)、  
debu greiserfs(8)

Reiserfsprogs-3.6.9

2003 年 4 月

REISERFSCK(8)

## e2fsck 手册页

E2FSCK(8)

E2FSCK(8)

#### 名称

e2fsck - 检查 Linux 的第二扩展文件系统

#### 语法

e2fsck [ -pacnyrdfvstDFSV ] [ -b superblock ] [ -B block size ] [ -l|-L bad\_block ]

#### 说明

e2fsck 用于检查 Linux 的第二扩展文件系统 (ext2fs)。E2fsck 也支持包含日记的 ext2 文件系统。device 是储存文件系统的设备文件 (如 /dev/hdc1)。



## OPTIONS

- a 此选项与 `-p` 选项等效。该选项仅用于向后兼容；建议用户尽量使用 `-p` 选项。
- b `superblock` 不使用通常的超级块，而使用由 `superblock` 指定的备用超级块。通常在主超级块损坏时才使用带 `-n` 选项的 `mke2fs` 程序可以输出超级块的创建位置，从而确定更多的备用超级块。必须为 `mke2fs` 指定如果指定了备用超级块而文件系统不是以只读方式打开的，`e2fsck` 将在文件系统检查完毕后确保正确更新主超级块。
- B `blocksize` 通常，`e2fsck` 会搜索各种块大小的超级块，以便找到正确的块大小。这种搜索有时可能并不准确。
- c 此选项将促使 `e2fsck` 运行 `badblocks(8)` 程序，查找文件系统上所有已损坏的块，找到后使通过将这些块
- C `fd` 此选项促使 `e2fsck` 将完成信息写入指定的文件描述符，以便监视文件系统的检查进度。此选项通常供运
- d 打印调试输出（除非调试 `e2fsck`，否则没有用处）。
- D 优化文件系统目录。此选项促使 `e2fsck` 尽量优化所有目录。所用方法为：对于支持目录索引的文件系统
- E `extended_options` 设置 `e2fsck` 扩展选项。各扩展选项之间用逗号分隔，而且可以带有使用等号（`'='`）  
`ea_ver=extended_attribute_version` 假定文件系统中的扩展属性块使用指定版本号的格式。版本号可为 1
- f 即便文件系统看起来是清洁的，也强制执行检查。
- F 开始之前先清理文件系统设备的缓冲区超速缓存。仅对执行 `e2fsck` 计时测试真正有用。
- j `external-journal` 设置可以找到此文件系统的 `external-journal`（外部日记）的路径名。
- l `filename` 将由 `filename` 指定的文件所列的块编号添加到坏块列表中。此文件的格式与 `badblocks(8)` 和
- L `filename` 将坏块列表设置为 `filename` 指定的块列表。（此选项与 `-l` 选项等效，只不过在将文件所列的块
- n 以只读方式打开文件系统，并且对所有问题都默认回答“否”。允许以非交互方式使用 `e2fsck`。（注意：如果
- p 自动修复（“整理”）文件系统，不回答任何问题。
- r 使用此选项不能执行任何操作；仅用于向后兼容。
- s 此选项将对文件系统执行字节交换，以便系统使用规范化的标准字节顺序（即 `i386` 或 `little endian`）。
- S 此选项将对文件系统执行字节交换，不论当前采用哪种字节顺序。
- t 输出 `e2fsck` 的计时统计数字。如果两次使用此选项，附加的计时统计数字将按先后通过的顺序输出。
- v 冗长方式。
- V 输出版本信息，然后退出。
- y 对所有问题默认回答“是”；允许以非交互方式使用 `e2fsck`。

#### EXIT CODE

`e2fsck` 返回的退出代码是以下条件之合：0 - 无错误 1 - 已更正文件系统错误 2 - 已更正文件系统

#### SIGNALS

将以下信号发送到 `e2fsck` 时有以下影响。

#### SIGUSR1

此信号促使 `e2fsck` 开始显示进度条。（请参见 `-C` 选项的说明。）

#### SIGUSR2

此信号促使 `e2fsck` 停止显示进度条。

#### 报告错误

几乎所有软件都不可避免地存在错误。如果您发现某个文件系统导致 `e2fsck` 崩溃或者存在 `e2fsck` 问题，请向作者报告。

请在错误报告中包含尽量详尽的信息。最好包含一份有关 `e2fsck` 运行状况的完整记录，以便我能清楚

发送 `dumpe2fs(8)` 的输出也十分有用。如果 `e2fsck` 的错误看起来是由某个或某些 `inode` 造成的，

始终包含 `e2fsck` 运行时所显示的完整版本字符串，这样方便我了解您运行的版本。

作者 本版本的 `e2fsck` 由 Theodore Ts'o 编写，电子邮件为 `<tytso@mit.edu>`。

另请参见 `mke2fs(8)`、`tune2fs(8)`、`dumpe2fs(8)`、`debugfs(8)`

E2fsprogs 版本 1.34

2003 年 7 月

E2FSCK(8)

## Manual Page of xfs\_check

`xfs_check(8)`

`xfs_check(8)`

#### NAME

`xfs_check` - check XFS filesystem consistency

#### SYNOPSIS

`xfs_check` [ `-i` `ino` ] ... [ `-b` `bno` ] ... [ `-s` ] [ `-v` ] `xfs_special`

`xfs_check` `-f` [ `-i` `ino` ] ... [ `-b` `bno` ] ... [ `-s` ] [ `-v` ] `file`

#### DESCRIPTION

`xfs_check` checks whether an XFS filesystem is consistent.

It is normally run only when there is reason to believe that the filesystem has a consistency problem. The filesystem to be checked is specified by the `xfs_special` argument, which should be the disk or volume device for the filesystem. Filesystems stored in files can also be checked, using the `-f` flag. The filesystem should normally be unmounted or read-only during the execution of `xfs_check`. Otherwise, spurious problems are reported.

The options to `xfs_check` are:

- `-f` Specifies that the special device is actually a file (see the `mkfs.xfs -d file` option). This might happen if an image copy of a filesystem has been made into an ordinary file.
- `-s` Specifies that only serious errors should be reported. Serious errors are those that make it impossible to find major data structures in the filesystem. This option can be used to cut down the amount of output when there is a serious problem, when the output might make it difficult to see what the real problem is.
- `-v` Specifies verbose output; it is impossibly long for a reasonably-sized filesystem. This option is intended for internal use only.
- `-i ino` Specifies verbose behavior for a specific inode. For instance, it can be used to locate all the blocks associated with a given inode.
- `-b bno` Specifies verbose behavior for a specific filesystem block. For instance, it can be used to determine what a specific block is used for. The block number is a "file system block number". Conversion between disk addresses (i.e. addresses reported by `xfs_bmap`) and file system blocks may be accomplished using `xfs_db`'s `convert` command.

Any non-verbose output from `xfs_check` means that the filesystem has an inconsistency. The filesystem can be repaired using either `xfs_repair(8)` to fix the filesystem in place, or by using `xfsdump(8)` and `mkfs.xfs(8)` to dump the filesystem, make a new filesystem, then use `xfsrestore(8)` to restore the data onto the new filesystem. Note that `xfsdump` may fail on a corrupt filesystem. However, if the filesystem is mountable, `xfsdump` can be used to try and save important data before repairing the filesystem with `xfs_repair`. If the filesystem is not mountable though, `xfs_repair` is the

only viable option.

#### DIAGNOSTICS

Under one circumstance, `xfs_check` unfortunately might dump core rather than produce useful output. If the filesystem is completely corrupt, a core dump might be produced instead of the message `xxx is not a valid filesystem`

If the filesystem is very large (has many files) then `xfs_check` might run out of memory. In this case the message out of memory is printed.

The following is a description of the most likely problems and the associated messages. Most of the diagnostics produced are only meaningful with an understanding of the structure of the filesystem.

`agf_freeblks n, counted m in ag a`

The freeblocks count in the allocation group header for allocation group `a` doesn't match the number of blocks counted free.

`agf_longest n, counted m in ag a`

The longest free extent in the allocation group header for allocation group `a` doesn't match the longest free extent found in the allocation group.

`agi_count n, counted m in ag a`

The allocated inode count in the allocation group header for allocation group `a` doesn't match the number of inodes counted in the allocation group.

`agi_freecount n, counted m in ag a`

The free inode count in the allocation group header for allocation group `a` doesn't match the number of inodes counted free in the allocation group.

`block a/b expected inum 0 got i`

The block number is specified as a pair (allocation group number, block in the allocation group). The block is used multiple times (shared), between multiple inodes. This message usually follows a message of the next type.

`block a/b expected type unknown got y`

The block is used multiple times (shared).

`block a/b type unknown not expected`

The block is unaccounted for (not in the freelist and not in use).

link count mismatch for inode nnn (name xxx), nlink m,  
counted n

The inode has a bad link count (number of references  
in directories).

rtblock b expected inum 0 got i

The block is used multiple times (shared), between  
multiple inodes. This message usually follows a mes-  
sage of the next type.

rtblock b expected type unknown got y

The real-time block is used multiple times (shared).

rtblock b type unknown not expected

The real-time block is unaccounted for (not in the  
freelist and not in use).

sb\_fdblocks n, counted m

The number of free data blocks recorded in the  
superblock doesn't match the number counted free in  
the filesystem.

sb\_frextents n, counted m

The number of free real-time extents recorded in the  
superblock doesn't match the number counted free in  
the filesystem.

sb\_icount n, counted m

The number of allocated inodes recorded in the  
superblock doesn't match the number allocated in the  
filesystem.

sb\_ifree n, counted m

The number of free inodes recorded in the superblock  
doesn't match the number free in the filesystem.

#### SEE ALSO

mkfs.xfs(8), xfsdump(8), xfsrestore(8), xfs\_ncheck(8),  
xfs\_repair(8), xfs(5).

xfs\_check(8)

# Manual Page of jfs\_fsck

jfs\_fsck(8)            JFS utility - file system check            jfs\_fsck(8)

## NAME

jfs\_fsck - initiate replay of the JFS transaction log, and check and repair a JFS formatted device

## SYNOPSIS

```
jfs_fsck [ -afnpvV ] [ -j journal_device ] [ --omit_journal_replay ] [ --replay_journal_only ] device
```

## DESCRIPTION

jfs\_fsck is used to replay the JFS transaction log, check a JFS formatted device for errors, and fix any errors found.

device is the special file name corresponding to the actual device to be checked (e.g. /dev/hdb1).

jfs\_fsck must be run as root.

## WARNING

jfs\_fsck should only be used to check an unmounted file system or a file system that is mounted READ ONLY. Using jfs\_fsck to check a file system mounted other than READ ONLY could seriously damage the file system!

## OPTIONS

If no options are selected, the default is -p.

- a        Autocheck mode - Replay the transaction log. Do not continue fsck processing unless the aggregate state is dirty or the log replay failed. Functionally equivalent to -p. Autocheck mode is typically the default mode used when jfs\_fsck is called at boot time.
- f        Replay the transaction log and force checking even if the file system appears clean. Repair all problems automatically.
- j journal\_device        Specify the journal device.
- n        Open the file system read only. Do not replay the transaction log. Report errors, but do not repair them.

- `--omit_journal_replay`  
Omit the replay of the transaction log. This option should not be used unless as a last resort (i.e. the log has been severely corrupted and replaying it causes further problems).
- `-p` Automatically repair ("preen") the file system. Replay the transaction log. Do not continue fsck processing unless the aggregate state is dirty or the log replay failed. Functionally equivalent to `-a`.
- `--replay_journal_only`  
Only replay the transaction log. Do not continue with a full file system check if the replay fails or if the file system is still dirty even after a journal replay. In general, this option should only be used for debugging purposes as it could leave the file system in an unmountable state. This option cannot be used with `-f`, `-n`, or `--omit_journal_replay`.
- `-v` Verbose messaging - print details and debug statements to stdout.
- `-V` Print version information and exit (regardless of any other chosen options).

#### EXAMPLES

Check the 3rd partition on the 2nd hard disk, print extended information to stdout, replay the transaction log, force complete jfs\_fsck checking, and give permission to repair all errors:

```
jfs_fsck -v -f /dev/hdb3
```

Check the 5th partition on the 1st hard disk, and report, but do not repair, any errors:

```
jfs_fsck -n /dev/hda5
```

#### EXIT CODE

The exit code returned by `jfs_fsck` represents one of the following conditions:

- 0 No errors
- 1 File system errors corrected and/or transaction log replayed successfully

- 2 File system errors corrected, system should be rebooted if file system was mounted
- 4 File system errors left uncorrected
- 8 Operational error
- 16 Usage or syntax error
- 128 Shared library error

#### REPORTING BUGS

If you find a bug in JFS or `jfs_fsck`, please report it via the bug tracking system ("Report Bugs" section) of the JFS project web site:  
<http://oss.software.ibm.com/jfs>

Please send as much pertinent information as possible, including the complete output of running `jfs_fsck` with the `-v` option on the JFS device.

#### SEE ALSO

`fsck(8)`, `jfs_mkfs(8)`, `jfs_fscklog(8)`, `jfs_tune(8)`, `jfs_log-dump(8)`, `jfs_debugfs(8)`

#### AUTHORS

Barry Arndt ([barndt@us.ibm.com](mailto:barndt@us.ibm.com))  
William Braswell, Jr.

`jfs_fsck` is maintained by IBM.  
See the JFS project web site for more details:  
<http://oss.software.ibm.com/jfs>

October 29, 2002

`jfs_fsck(8)`





# GNU 通用公共许可证

## GNU 通用公共许可证

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307, USA

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 675 Mass Ave, Cambridge, MA 02139, USA

允许任何人复制和分发此许可证文档的逐字副本，但禁止对其进行更改。

### 前言

大多数软件许可证的目的是为了剥夺您共享和更改软件的自由。与之相反，*GNU 通用公共许可证*旨在保证您共享和更改自由软件的自由，即确保软件对其所有用户都是自由的。本*通用公共许可证*适用于*自由软件基金会*的大多数软件以及由作者提交使用的任何其他程序。（其他一些*自由软件基金会*软件遵照*GNU 库通用公共许可证*。）您也可以将它应用于自己的程序。

我们所说的“自由”软件，是指自由而非价格。我们的通用公共许可证旨在确保您能够自由分发自由软件副本（以及根据需要对此服务收费）、获得所需的源代码、更改软件或将其某些部分用在新的自由程序中；并使您知道自己可以做这些事情。

为了保护您的权利，我们需要作出某些限制，以防止任何人剥夺或要求您放弃这些权利。如果您分发软件的副本或修改它，这些限制同样会转化为您的某些责任。

例如，无论您免费或有偿分发这样的程序副本，都必须向接收者提供您拥有的所有相关权利。您也必须确保他们收到或能够获得源代码。而且您必须向他们提供这些条款，以便他们能够了解其权利。

我们以两个步骤来保护您的权利：(1) 取得软件的版权，并 (2) 为您提供这个许可证，使您获得复制、分发和/或修改此软件的法律权限。

此外，为了保护每个作者和我们自己，我们希望所有人都明确地知道这个自由软件不提供任何担保。如果此软件经过某人的修改后分发给别人，我们希望接收者了解他们收到的不是原始软件，以便由其他人带来的任何问题不会影响原作者的声誉。

最后，任何自由程序都会经常受到软件专利的威胁。自由程序的再分发者可能会个人获得专利许可证，从而使该程序变为专有程序，我们希望避免这样的危险。为了防止这一点，我们必须明确，申请任何专利必须是为了所有人能够自由使用这些自由程序，否则不要为这些自由程序申请专利。

下面是有关复制、分发和修改的具体条款和条件。

## GNU 通用公共许可证

### 复制、分发和修改的条款和条件

0. 此许可证适用的对象：由版权所有者在其中明确声明可按照此通用公共许可证条款分发的任何程序和其他作品。下文中的“本程序”指任何这样的程序或作品，**基于本程序的作品**是指本程序或依据版权法创作的任何衍生作品：也就是包含本程序或其一部分（逐字逐句或经过修改和/或翻译为另一种语言）的作品。（在下文中，“修改”一词将没有限制地包含翻译的意思。每个获得许可证的人都被称为“您”。

本许可证不适用于复制、分发和修改之外的其他活动；它们超出了本许可证的适用范围。运行本程序的行为不受限制，对于本程序生成的输出，仅当其内容构成了基于本程序的作品时才在此许可证的适用范围之内（与是否是通过运行本程序生成的无关）。判断其内容是否构成基于本程序的作品依据是本程序的用途。

1. 您可以采用任何媒体来复制和分发本程序源代码的逐字副本，前提是您必须：**显著而适当地**在其每个副本上提供适当的版权声明和免责声明；保持涉及本许可证和提供任何担保的所有声明的完整性；并随本程序为任何其他接收者提供本许可证的副本。

您可以为分发副本的实际行为收取费用，也可以选择为提供担保保护而收取费用。

2. 您可能会修改本程序的任何副本或任何部分，并由此形成基于本程序的作品，此时您可以按照第 1 节的规定复制和分发这些修改或作品，但必须满足以下所有条件：



1. 您必须在被修改的文件中提供显著声明，指出您更改了这些文件并给出更改日期。
2. 如果要分发或发布全部或部分包含本程序或其任何部分、以及从本程序或其任何部分衍生的任何作品，都必须在本许可证条款的规定下以一个整体为所有第三方免费提供许可证。
3. 如果经过修改的程序在运行时通常以交互方式读取命令，在为了实现这种交互使用而以最普通的方式开始运行它时，必须使它输出和显示一条声明，其中应包含适当的版权声明和不提供任何担保的声明（或者声明您提供担保），并告知用户可以在这些条款和条件下再分发此程序以及如何查看本许可证的副本。（例外：如果本程序本身是交互式但通常不输出这样的声明，则您的基于本程序的作品就不需要输出或显示这样的声明）。

这些要求也适用于作为一个整体经过修改的作品。如果作品可以确认的组成部分不是从本程序中衍生的，并且有充分理由将其本身视为独立和单独的作品，则当您将这些组成部分作为单独的作品分发时，就不受本许可证及其条款的约束。但是，如果您分发的作品的某些组成部分与基于本程序的作品整体的一部分相同时，则此作品的分发必须在整体上遵照本许可证的条款，它与其他许可证获得者提供的权限将延伸到整个作品，并由此延伸到每一部分，而不考虑每一部分的作者是谁。

因此，本节的意图不是主张拥有或争夺完全由您创作的作品的权利；相反，我们的意图是行使控制基于本程序的衍生性或集合性作品分发的权利。

此外，如果只是将不基于本程序的另一作品和本程序（或基于本程序的作品）一起储存在某个储存或分发媒体上，则另一作品并不受本许可证的约束。

3. 您可以在第 1 节和 2 节条款的规定下以结果代码或可执行表单的形式复制和分发本程序（或基于它的作品，按照第 2 节的规定），但前提是必须满足以下要求之一：

1. 随本程序提供完整的、对应的计算机可读源代码，且这些源代码必须按照第 1 节和第 2 节条款的规定在某种通常用于软件交换的媒体上分发。
2. 随本程序为任何第三方提供一个有效期至少为三年的书面收费报价，该报价不得高于进行源代码分发而实际发生的成本，并提供对应源代码的完整的计算机可读副本，同时，该副本必须按照第 1 节和第 2 节条款的规定在某种通常用于软件交换的媒体上分发。
3. 随本程序提供您收到的有关分发对应源代码的报价信息。（仅当您的分发给非商业性、并且您收到了带有这一报价的采用结果代码或可执行表单形式的程序（与上述 b 小节一致）时，才能使用此选择。）

作品的源代码表明了对它进行修改时的首选作品形式。对于可执行作品，“完整源代码”表示它包含的所有模块的所有源代码、任何相关接口定义文件以及用于控制可执行文件的编译和安装脚本报本。但有一种特殊情况，要分发的源代码不需要包含通常要与可执行文件在其上运行的操作系统的主要部件（编译器、内核等）一起（以源代码或二进制形式）分发的任何对象，除非这些部件本身带有可执行文件。

如果采用提供指定地址供接收者复制的方式来分发可执行文件或结果代码，则分发源代码就必须为第三方提供等效的访问权限在相同的地址复制源代码，虽然并不要求第三方在复制结果代码时一定要复制源代码。

4. 除本许可证中有明确规定外，您不能对本程序进行复制、修改、分授许可或分发。在本许可证规定外对本程序所进行的任何复制、修改、分授许可或分发都是无效的，并且将自动终止您在本许可证下所拥有的权利。但是，在本许可证的规定下从您这里获得副本或权利的各方，只要他们完全遵守本许可证的规定，其许可证将不会被终止。

5. 由于您还没有对本许可证签名，我们不要求您接受本许可证。但是，除本许可证之外，没有任何东西可授予您修改或分发本程序或其衍生作品的权利。如果您不接受本许可证，则这些活动就被法律所禁止。因此，如果您修改或分发了本程序（或基于本程序的任何作品），即表明您接受了本许可证来进行这样的活动，也就是接受了它所有的关于复制、分发或修改本程序或基于本程序的作品条款和条件。

6. 每次您再分发本程序（或任何基于本程序的作品）时，接收者应自动收到来自原始许可人的许可证，接收者可按照这些条款和条件复制、分发或修改本程序。您不能对接收者行使在许可证中获得的权利施加任何其他限制。您不负责强制第三方遵守本许可证。

7. 如果由于法庭判决或专利侵权诉讼或任何其他原因（并不限于专利事宜），而造成向您施加的条款和条件（无论是法院指令、协议或其他）与本许可证的条款和条件冲突，它们并不能使您免受本许可证的条款和条件的约束。如果在分发本程序时不能同时满足本许可证规定的义务和任何其他相关义务，则您就根本不能分发本程序。例如，如果某专利许可证禁止通过您直接或间接受到本程序副本的所有人对本程序进行无版税再分发，则要同时满足此规定和本许可证条款和条件的唯一方法就是您完全不分发本程序。

如果本节的任何部分在任何特定情形下变得无效或无法执行，则本节的其他部分仍适用，而且本节作为一个整体仍适用于其他情形。

本节的目的在于诱使您侵犯专利或其他财产权主张，或争论这种主张的有效性。本节的唯一目的在于保护自由软件分发体系的完整性，它是通过通用许可证的应用来实现的。许多人相信这一体系的一致应用，并且慷慨地贡献了许多软件来通过这一体系进行分发。作者/捐赠者拥有全部权利来决定他/她是否愿意通过任何其他体系分发软件，许可证获得者不能强制他们作出选择。



本节的目的在于明确说明本许可证其余部分所产生的结果。

8. 如果本程序的分发和/或使用在某些国家/地区受到专利或受版权保护接口的限制，那么将本程序置于本许可证约束下的原始版权所有者可以明确添加分发地理限制来排除这些国家/地区，从而使得分发只能在未排除的国家/地区进行。在这种情况下，本许可证将包括这些限制，其效力同写入本许可证正文的条款和条件一样。

9. 自由软件基金会有时会发布通用公共许可证的修订版和/或新版本。这些新版本的主旨和精神与当前版本是一致的，但在解决新问题的具体细节方面可能有所不同。

每个版本都有一个不同的版本号。如果本程序指定了它所应用的本许可证的某个版本和“任何以后版本”，则您可以选择遵照此版本或自由软件基金会发布的任何以后版本的条款和条件。如果本程序没有指定本许可证的版本号，您可以选择自由软件基金会发布的任何版本的许可证。

10. 如果希望将本程序的某些部分包含在其分发条款和条件不同的其他自由程序中，请致函作者来获得许可。对于受自由软件基金会版权保护的软件，应致函自由软件基金会；我们有时对此有例外处理。我们的决定以两个目标为导向，第一个目标是保持我们的自由软件的所有衍生作品的自由状态，第二个目标是广泛地促进软件的共享和重用。

## 不提供担保

11. 因为本程序免费提供许可证，所以在适用法律许可的限度内对它不提供任何担保。除非另有书面声明，版权所有者和/或其他方是“按原样”提供程序的，且不作任何明示或暗示的担保，包括但不限于对任何特定目的的适销性及适用性的暗示的担保。程序的所有质量和性能风险完全由您自行承担。如果证实程序有缺陷，您需要承担所有必需的维护、修复或更正的成本。

12. 除非有适用法律要求或书面协定，任何版权所有人或按照本许可证的许可而修改和/或再分发程序的任何其他方均不对您的损失负责，这些损失包括由于使用或不能使用程序而引起的任何一般、特殊、偶发或连带损失（包括但不限于数据丢失、数据转换错误或您或第三方遭受的损失或程序未能与其他任何程序共用的损失），即使这些版权所有者或其他方已被告知可能出现这些损失。

条款和条件结束

## 如何将这些条款应用于您的新程序

如果您开发了一个新程序并希望使它最大限度地对公众有用，最好的方法是使它成为自由软件，这样每个人都可以在这这些条款的规定下再分发和更改它。

要做到这一点，请为您的程序附加以下声明。最安全的方式是将它们附加在每个源文件的开头，以便最有效地表明此程序不提供担保；而且每个文件至少有“版权声明”行以及指向完整声明所在位置的指示。

```
<one line to give the program's name and a brief idea of what it does.>
```

```
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
```

```
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License along with this program; if not, write to the
```

```
Free Software Foundation, Inc.,  
59 Temple Place, Suite 330,  
Boston, MA 02111-1307,  
USA.
```

另外还要添加您的电子邮件和邮政地址等联系信息。

如果此程序是交互式的，使它在交互方式下启动时输出如下简短声明：

```
Gnomovision version 69, Copyright (C) <year> <name of author>
```

```
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.
```

假设的命令 `show w` 和 `show c` 应显示通用公共许可证的相应部分。当然，您使用的命令也可以不称为 `show w` 和 `show c`；它们甚至可以是鼠标点击或菜单项——即可以是适合您的程序的任何形式。

如果需要，您应让您的雇主（如果您是程序员）或学校（如果有）为该程序签署一份“版权免责声明”。下面是一个示例（可更改其名称）：



Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision'  
signature of Ty Coon, 1st April 1989 Ty Coon, President of Vice

本通用公共许可证不允许将您的程序包含在专有程序中。如果您的程序是一个子  
例程库，您可能认为允许专有应用程序链接此库会更有用。如果您希望这样做，  
请使用 GNU 库通用公共许可证来代替本许可证。





# 词汇表

## 访问权限

文件的访问权限决定用户或组是否有权读取、写入或执行某个文件或目录。访问权限通常由系统管理员设置。

## 帐户

帐户是由用户名或登录名称和口令定义的。帐户对应于用户 ID (UID)。

## ACL ( Access Control List, 访问控制列表)

文件和目录的常规权限概念的扩展。使用 ACL 可以对访问权限进行更进一步的控制。

## ADSL ( Asymmetric Digital Subscriber Line, 非对称数字用户线路)

使用电话网络的快速传送协议。

## AGP ( Accelerated Graphics Port, 加速图形接口)

一种高速图形卡插槽，提供比 PCI 更高的带宽。AGP 图形卡不必绕经处理器即可直接返回随机存储器。

## ATAPI ( Advanced Technology Attachment Packet Interface, 高级技术附加数据包接口)

ATAPI 是一种连接到 (E)IDE 控制器的 CD-ROM 驱动器。除 ATAPI 驱动器之外，还有 SCSI CD-ROM 驱动器，后者通过 SCSI 控制器处理。

## 备份

备份即数据副本，用于恢复损坏或丢失的数据。应定期备份所有重要数据。

### 带宽

数据传送通道的最大传送速率。通常用在网络连接中。

### BIOS ( Basic Input/Output System, 基本输入/输出系统)

开机或重引导计算机后启动的小程序。该程序负责硬件部件的初始化。多数 BIOS 允许通过交互式设置程序对低级系统参数进行修改。程序代码驻留在只读存储器 (ROM) 芯片中。

### 书签 (适用于浏览器)

URL 集合中的一个项目。

### 引导

从开机直到系统可用所经历的一系列计算机操作。

### 浏览器

显示本地文件或万维网网页内容的程序。

### 客户机

在网络环境中连接到服务器并向其请求信息的程序或计算机。

### 命令行

向计算机发布命令的基于文本的方式。

### 控制台

以前是终端的同义词。在 Linux 中, 虚拟控制台允许在同一屏幕上显示若干独立、并行的工作会话, 而无需运行任何图形显示界面。

### CPU ( Central Processing Unit, 中央处理单元)

请参见处理器。

### 光标

光标是一个块字符或下划线字符, 用来标记文本的输入位置。

### 守护程序

守护程序 (磁盘和执行监视程序) 是在后台运行的程序, 在需要时即可自动激活。例如, HTTP 守护程序 (httpd) 可以响应 HTTP 请求。

### DDC ( Direct Display Channel, 直接显示通道)

监视器和图形卡之间的通讯标准, 允许向图形卡传送特定参数, 如监视器名称或分辨率。

**目录**（在文件系统中）

包含文件或更多目录（子目录）的结构。文件系统中的目录构成了树形结构，用来组织文件。

**DNS**（**Domain Name System**，域名系统）

在基于名称的地址和 IP 地址之间相互转换所用的协议。

**驱动程序**

操作系统中负责与硬件部件通讯的部分。

**电子邮件**

在用户之间以电子方式通过网络传送邮件的方式。电子邮件地址的格式为 `username@domain.org`。

**EIDE**（**Enhanced Integrated Drive Electronics**，增强型集成驱动电子设备）

经过改进的 IDE 标准，支持容量超过 512 MB 的硬盘。

**环境**

shell 保留的一组环境变量及变量值。用户可以更改（或取消设置）现有环境变量的值，并设置新变量。永久指派通过 shell 的配置文件来完成。

**环境变量**

shell 环境中的一个元素。

**Ethernet**

计算机网络中的一种数据传送标准。

**EXT2**（**Second Extended File System**，第二扩展文件系统）

Linux 支持的一种文件系统。

**FAQ**（**Frequently Asked Questions**，常见问题解答）

首字母缩略语，代表为常见问题提供解答的文档。

**防火墙**

一种过滤网络流量的机制，旨在保护本地网络免遭外部未经授权的访问的攻击。

**FTP**（**File Transfer Protocol**，文件传送协议）

一种基于 TCP/IP 跨网络传送文件的协议。

**GNOME** ( **GNU Network Object Model Environment**, **GNU** 网络对象模型环境)

用于 Linux 的图形桌面环境。

**GNU (GNU is Not Unix)**

GNU 是自由软件基金会 (Free Software Foundation, FSF) 的一个项目。GNU 项目旨在创建完善而自由的 Unix 式操作系统。Free 指的并非免费, 而是自由: 即有权获取、修改和更改软件。现已成为经典的 GNU 宣言 (<http://www.gnu.org/gnu/manifesto.html>) 对此进行了详细解释。依照法律规定, GNU 软件受 GPL 和 LGPL 的共同保护。GPL 即 GNU General Public License (通用公共许可证, 参见 <http://www.gnu.org/copyleft/gpl.html>); LGPL 即 GNU Lesser General Public License (较宽松公共许可证, 参见 <http://www.gnu.org/copyleft/lgpl.html>)。受 GPL 保护的 Linux 内核受益于该项目 (特别是相关的工具), 但不应将 Linux 内核等同于 GNU 项目。

**GPL (GNU General Public License, GNU 通用公共许可证)**

请参见 GNU。

**用户主目录**

文件系统中属于特定用户的私人目录 (通常位于 `/home/<username>` 中)。除超级用户 (根用户) 之外, 只有用户本人才能不受限制地访问其主目录。

**主机名**

计算机的名称。根据该名称通常就可以在网络中找到该计算机。

**HTML (Hypertext Markup Language, 超文本标记语言)**

万维网中使用的适用于文本文档的标记语言。通常使用浏览器来查看 HTML 文档。

**HTTP (Hypertext Transfer Protocol, 超文本传送协议)**

定义如何在万维网中请求和传送文档的网络协议。这些文档通常是服务器提供的 HTML 网页, 用户通过浏览器请求这些网页。

**IDE (Integrated Drive Electronics, 集成驱动电子设备)**

用于挂接硬盘的标准。

## 因特网

基于 TCP/IP 的全球计算机网络。

## IP 地址

TCP/IP 网络中的唯一（32 位）计算机地址。经常书写为用句点分隔的四个十进制数（例如 192.168.10.1）。

## IRQ（Interrupt Request，中断请求）

可以由硬件或软件触发的申请执行某种操作的（异步）请求。多数 IRQ 都由操作系统来处理。

## ISDN（Integrated Services Digital Network，综合业务数字网）

通过电话网络传送数字数据的标准。

## KDE（K Desktop Environment，K 桌面环境）

用于 Linux 的图形桌面环境。

## 内核

内核是操作系统的核心组成部分。内核负责管理内存和文件系统、包含与硬件设备通讯的驱动程序，并且负责处理进程和联网任务。

## LAN（Local Area Network，局域网）

LAN 是一种通常范围较小的本地网络。

## LILO（Linux Loader，Linux 装载程序）

安装在硬盘引导区中的小程序，可以启动 Linux 或其他操作系统。

## 链接

链接（在文件系统中）即文件的指针。有硬链接和符号链接之分。硬链接指在文件系统中的确切位置，而符号链接仅指向对应的名称。

## Linux

在 GPL (GNU) 授权下自由发布的类似 UNIX 的高性能操作系统内核。该名称是个缩写词 (*Linus' uniX*)，意指其创建者 Linus Torvalds。尽管该名称从严格意义上来说仅指内核本身，但按照一般理解，*Linux* 这个词常指整个系统。

## 登录

依据用户名和口令鉴定用户，通过鉴定的用户即可访问计算机系统或网络。

## 注销

关闭 Linux 交互会话的过程。

## 主存储器

支持几乎不带延迟的随机访问的物理存储器。通常称为 RAM，即随机存储器。

## 手册页

Unix 系统的传统格式的文档，可以使用 `man` 命令阅读。手册页通常书写为参考格式。

## MBR (Master Boot Record, 主引导记录)

硬盘的第一个物理扇区，硬盘内容就是从这个位置被装载到主存储器中再由 BIOS 执行的。此代码随后会从某个硬盘分区装载操作系统或是装载更复杂的引导装载程序，如 LILO 或 GRUB。

## MD5

用于生成哈希值的算法（文件的 MD5 校验和）。这些校验和的生成方式决定：几乎不可能创建一个具有特定 MD5 校验和而内容不同于原文件的文件。

## 装入

将文件系统连接到系统目录树的进程。

## MP3

适用于音频文件的损耗较大的压缩算法，使用该算法压缩的数据要比未压缩的音频文件小十倍。

## 多任务

（几乎）可以并行运行多个进程的操作系统功能。

## 多用户

支持多个用户在一台计算机上并行操作的操作系统功能。

## 网络

多台计算机之间的互联互通，旨在支持相互之间的数据传送。通过网络发送请求的计算机通常称为客户机，响应请求（例如通过传递文档）的计算机称为服务器。

## **NFS ( Network File System, 网络文件系统)**

通过网络访问文件系统所用的协议。

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

网络中的集中式用户管理系统。通过 NIS 可以在整个网络范围内管理用户名和口令。

## **操作系统**

请参见内核。

## **分区**

硬盘上的扇区，其中包含文件系统或交换空间。

## **路径**

关于文件在文件系统中的位置的唯一说明。

## **即插即用**

自动硬件检测和配置协议。

## **进程**

运行的程序。有时称为任务。

## **处理器**

处理器，即 CPU ( Central Processing Unit, 中央处理单元 )，是用于执行主存储器中储存的代码的微型芯片。它是计算机的大脑。

## **提示符**

在每个命令行开头显示的简短 ( 可配置 ) 字符串。通常包含当前的工作目录。

## **协议**

为硬件、软件或网络定义接口和通讯方法的标准。如 HTTP 协议和 FTP 协议。

## **代理**

通常指为因特网上传送的数据充当中间储存区的计算机。若多次请求同一文档，对后续请求的响应速度会大大提高。准备利用代理功能的计算机必须配置为通过代理发出自身的请求。

**RAM ( Random Access Memory, 随机存储器 )**

请参见主存储器。

**ReiserFS**

支持快速修复潜在不一致性的文件系统类型。关闭操作系统前若未卸载文件系统（如出现电源故障）就可能造成这种不一致性。

**根用户**

超级用户帐户。超级用户具有所有权限。此帐户仅可用于管理性任务，不应用于常规任务。

**根目录**

文件系统层次中的基目录。在 UNIX 中，根目录用 / 来表示。

**SCSI ( Small Computer Systems Interface, 小型计算机系统接口 )**

用于挂接硬盘和其他设备（如扫描仪和磁带）的标准。

**服务器**

专门用于提供服务（通常通过网络）的计算机或程序。服务类型如 HTTP、DNS 和 FTP。

**shell**

支持发布命令的交互式程序。shell 有多种类型，如 Bash、zsh 和 tcsh。每种 shell 都各有特定的编程语言。

**SMTP ( Simple Mail Transfer Protocol, 简单邮件传送协议 )**

通过网络传送电子邮件的协议。

**SSL ( Secure Socket Layer, 安全套接层 )**

用于传送 HTTP 数据的加密协议。

**超级用户**

请参见根用户。

**系统管理员**

请参见根用户。

**任务**

请参见进程。



## TCP/IP

用于因特网的通讯协议，也用于多数本地网络。

## Telnet

Telnet 是一种用于与远程主机通讯的协议。就远程登录而言，Telnet 基本上已被提供加密连接的 SSH 所取代。

## 终端

以前指与中央计算机相连的特定键盘与监视器组合。目前这个术语多指模拟实际终端的程序（如 xterm）。

## Tux

Linux 企鹅标志的名字。请参见 <http://www.sjbaker.org/tux/>。

## UNIX

UNIX 是一种操作系统，同时也是注册商标。

## URL (Uniform Resource Locator, 统一资源定位器)

网络中资源的规范，由协议（如 <http://>）、主机名和域名（如 [www.suse.de](http://www.suse.de)）及文档（如 [/us/company/index.html](http://www.suse.de/us/company/index.html)）构成。上面几个示例构成的完整 URL 为：<http://www.suse.de/us/company/index.html>。

## 用户目录

请参见主目录。

## VESA (Video Electronics Standard Association, 视频电子标准协会)

定义视频标准之类内容的业界联盟。

## 通配符

代表一个字符（符号：?）或多个字符（符号：\*）的占位符，是正则表达式的组成部分。

## 窗口管理器

运行在 X 窗口系统顶部的程序，支持执行调整窗口大小或移动窗口等操作。窗口管理器还负责对窗口标题和边框等进行装饰。用户可以自定义窗口管理器的行为和外观。

**WWW (World Wide Web, 万维网)**

基于 HTTP 协议，万维网是可以使用万维网浏览器查看的各种文档、文件和图象的超链接集合。

**X 窗口系统**

X 窗口系统是一种可以在众多计算机上运行的基于网络的窗口系统。它提供一些基元类型，如用于绘制线条或矩形的类型。它是硬件和窗口管理器之间的中间层。

**X11**

X 窗口管理器的第 11 版。

**YaST (Yet another Setup Tool)**

SUSE LINUX 系统的辅助工具。

**YP**

请参见 NIS。

# Index

## symbols

.local 作为顶级域 ..... 100  
万维网服务器  
- Apache ..... *see* Apache  
主引导记录 ..... *see* MBR  
主机名 ..... 51  
代理 ..... 52, *see* Squid  
- 优点 ..... 484  
- 超速缓存 ..... 484  
- 透明 ..... 493  
伪装 ..... 506  
- 用 SuSEfirewall2 进行配置 ..... 508  
便携式计算机 ..... 228–233  
- 电源管理 ..... 228, 253–262  
- 硬件 ..... 228  
- IrDA ..... 291–293  
- PCMCIA ..... 228  
- SCPM ..... 229, 243  
- SLP ..... 230  
信息页 ..... 176  
修复系统 ..... 121  
内存  
- RAM ..... 177  
内核 ..... 166–171  
- 参数 ..... 166  
- 守护程序 ..... 169  
- 安装 ..... 170–171  
- 模块 ..... 168–169  
  · 编译 ..... 170  
  · 网卡 ..... 342  
  · modprobe.conf ..... 98  
- 模块装载程序 ..... 169  
- 源 ..... 166

- 版本 2.6 ..... 98  
- 编译 ..... 166, 170  
- 超速缓存 ..... 178  
- 配置 ..... 166–167  
- 错误讯息 ..... 170  
- 限制 ..... 317  
- kmod ..... 169  
- modprobe.conf ..... 169  
冷插拔 ..... 300  
分区  
- 交换 ..... 61  
- 分区表 ..... 148  
- 创建 ..... 9, 58, 60  
- 加密 ..... 516  
- 参数 ..... 60  
- 类型 ..... 9  
- 调整 Windows 分区的大小 ..... 14  
- fstab ..... 62  
- LVM ..... 61  
- RAID ..... 61  
加密  
- 分区 ..... 516  
- 文件 ..... 516  
加密文件系统 ..... 516  
包  
- 使用 build 编译 ..... 118  
- 包管理器 ..... 110  
- 卸载 ..... 111  
- 安装 ..... 111  
- 构建 ..... 97  
- 校验 ..... 110  
- 编译 ..... 116  
- LSB ..... 110

- RPM .....	110
包过滤器 .....	<i>see</i> 防火墙
区域设置	
- UTF-8 .....	100
协议	
- FTP .....	434
- HTTP .....	434
- HTTPS .....	434
- IPv6 .....	335
- LDAP .....	411
- SLP .....	363
- SMB .....	474
卡	
- 图形 .....	188
· 驱动程序 .....	198
- 声音 .....	48
- 无线电 .....	49
- 电视 .....	49
卸载	
- GRUB .....	160
- Linux .....	160
变量	
- 环境 .....	182
可插拔鉴定模块 .....	<i>see</i> PAM
可移动媒体	
- subfs .....	103
同步数据 .....	231
- 电子邮件 .....	231
- Evolution .....	234
- Kontact .....	234
- KPilot .....	234
名称服务器 .....	<i>see</i> DNS
命令	
- 热插拔 .....	297
- chown .....	100
- e2fsck .....	566
- fonts-config .....	199
- free .....	177
- getfacl .....	533
- grub .....	149
- head .....	100
- hwinfo .....	299
- ldapadd .....	421
- ldapdelete .....	423
- ldapmodify .....	423
- ldapsearch .....	423
- lp .....	216
- nice .....	100
- rpm .....	110
- rpmbuild .....	110

- scp .....	513
- setfacl .....	534
- sftp .....	513
- slptool .....	365
- smbpasswd .....	479
- sort .....	100
- ssh .....	512
- ssh-agent .....	515
- ssh-keygen .....	514
- tail .....	100
- udev .....	303
因特网	
- 万维网服务器 .....	<i>see</i> Apache
- 拨号 .....	359-361
- cinternet .....	360
- DSL .....	349
- ISDN .....	346
- kinternet .....	360
- qinternet .....	360
- smpppd .....	359-361
- TDSL .....	351
国际化 .....	182
图形	
- 卡	
· 驱动程序 .....	198
- 图形卡	
· 3D .....	204-206
- 3D .....	204-206
· 安装支持 .....	205
· 支持 .....	204
· 查错 .....	205
· 测试 .....	205
· 诊断 .....	205
· 驱动程序 .....	204
· 3Ddiag .....	205
· SaX .....	204
- GLIDE .....	204-206
- OpenGL .....	204-206
· 测试 .....	205
· 驱动程序 .....	204
图形用户界面 .....	186-193
地址	
- IP .....	332
- MAC .....	332
域名系统 .....	<i>see</i> DNS
声卡	
- 字体 .....	48
声音	
- 在 YaST 中配置 .....	48
备份 .....	43

- 使用 YaST 创建	56
- 恢复	57
多路广播 DNS	100
字体	199
- CID-keyed	203
- TrueType	199
- X11 核心	202
- Xft	199
安全性	518-527
- 串行终端	519
- 加密文件系统	232
- 口令	520
- 工程	519
- 引导	519-520
- 报告问题	527
- 提示和技巧	525
- 攻击	524-525
- 本地	520-522
- 权限	521
- 病毒	522
- 网络	522-525
- 蠕虫	525
- 配置	52-56
- 错误	521, 523
- 防火墙	55, 504
- DNS	524
- RPM 签名	526
- Samba	477
- Squid	484
- SSH	512-516
- tcpd	526
- X	522
安装	
- 包	111
- 媒体检查	44
- 引导装载程序	76
- 文本方式	74-75
- GRUB	149
- VNC	73
- YaST	3-28
安装支持	
- 3D 图形卡	205
屏幕	
- 分辨率	197
崩溃	563, 566
帮助	
- 信息页	176
- 手册页	176
- X	198
应急系统	124

- 使用	125
- 启动	125
引导	133, 563, 566
- 图形	76, 161
- 禁用	76, 161
- 引导扇区	148
- 引导管理器	148
- 方法	76
- 日志	64
- 管理	148
- 系统冻结	76
- 装载程序	159
- 位置	159
- 软盘, 从	79
- 配置	19
- YaST	157-160
- CD, 从	4
- CD 2, 从	80
- GRUB	76, 147, 149-163
- initrd	
- 创建	135
- LILO	76
- USB 存储器	148
引导磁盘	148
- 使用 dd 创建	78
- 使用 rawrite 创建	78
- 创建	
- DOS	77
- CD	148
手册页	176
手工安装	108
手提电话	234
打印	207, 211-213
- 使用 YaST 进行配置	211
- 命令行	216
- 应用程序, 从	215
- 查错	
- 网络	222
- 测试页	212
- 端口	211
- 网络	
- 查错	222
- 连接	211
- 队列	211
- 驱动程序	212
- CUPS	216
- foomatic-filters	97
- GDI 打印机	221
- Ghostscript 驱动程序	212
- IrDA	292

- kprinter ..... 216
- LPRng ..... 97
- PPD 文件 ..... 212
- Samba ..... 475
- xpp ..... 216
- 扫描**
  - 查错 ..... 47
  - 配置 ..... 46
- 控制台**
  - 切换 ..... 181
  - 图形
    - 禁用 ..... 76, 161
  - 指派 ..... 181
- 支持** ..... 64
- 数据安全性** ..... 232
- 数码相机** ..... 233
- 文件**
  - 加密 ..... 516
  - 同步 ..... 455-471
    - CVS ..... 456, 462-464
    - mailsync ..... 457, 468-471
    - rsync ..... 457
    - subversion ..... 457
    - Unison ..... 456, 460-462
  - 查找 ..... 176
- 文件服务器** ..... 51
- 文件系统** ..... 310-318
  - 修复 ..... 126
  - 加密 ..... 516
  - 支持的 ..... 315-316
  - 术语 ..... 310
  - 选择 ..... 310
  - 限制 ..... 317
  - ACL ..... 530-538
  - e2fsck ..... 566
  - Ext2 ..... 311-312
  - Ext3 ..... 312-313
  - FAT ..... 15
  - JFS ..... 314
  - LFS ..... 316
  - NTFS ..... 15, 16
  - Reiser4 ..... 313-314
  - ReiserFS ..... 310-311
  - reiserfsck ..... 563
  - sysfs ..... 296
  - XFS ..... 315
- 无线连接**
  - 蓝牙 ..... 283
- 日志文件** ..... 175
  - 日志 ..... 55
- 讯息 ..... 65, 375, 511
- apache2 ..... 443, 451
- boot.msg ..... 64, 256
- httpd ..... 441, 443, 451
- Squid ..... 487, 490, 495
- Unison ..... 462
- XFree86 ..... 205
- 日志记录**
  - 登录尝试 ..... 55
  - logrotate
    - 配置 ..... 175
- 时区** ..... 64
- 更新** ..... 93-97, 119
  - 增补程序 CD ..... 41
  - 混音器 ..... 107
  - 联机 ..... 39-41
  - 问题 ..... 94
  - passwd 和 group ..... 94
  - YaST ..... 95
- 服务位置协议** ..... *see* SLP
- 本地化** ..... 182
- 权限**
  - 文件权限 ..... 176
  - ACL ..... 530-538
- 核心文件** ..... 176
- 游戏杆**
  - 配置 ..... 194
- 源**
  - 编译 ..... 116
- 热插拔** ..... 295-301
  - 事件 ..... 297
  - 事件记录器 ..... 301
  - 代理 ..... 297
    - 接口 ..... 298
    - 设备 ..... 297
    - PCI ..... 299
    - USB ..... 299
  - 储存设备 ..... 298
  - 日志文件 ..... 300
  - 映射文件 ..... 299
  - 模块
    - 自动装载 ..... 299
  - 白名单 ..... 299
  - 网络设备 ..... 298
  - 设备名 ..... 296
  - 错误分析 ..... 300
  - 黑名单 ..... 299
  - PCI ..... 300
- 用户**
  - /etc/passwd ..... 322, 425

- 使用 YaST 管理 .....	52	- 安全性 .....	54
电子邮件 .....		- 应急 .....	124
- 同步 .....	231, 457	- 更新 .....	42, 93–97, 119
· mailsync .....	468–471	- 本地化 .....	182
- 配置 .....	50	- 语言 .....	64
电源管理 .....	228, 253–269	- 配置 .....	31–65
- 休眠 .....	254	- 限制资源使用 .....	176
- 待机 .....	254	系统服务 .....	51
- 暂停 .....	254	系统监视 .....	230
- 电池监视 .....	255	- KPowerSave .....	230
- 电量水平 .....	265	- KSysguard .....	230
- ACPI .....	253, 256–261, 265	线程包 .....	
- APM .....	253, 255–256, 265	- NPTL .....	99
- cpufrequency .....	262	组 .....	
- cpuspeed .....	262	- 管理 .....	53
- powersave .....	262	编码 .....	
- YaST .....	269	- ISO-8859-1 .....	183
电视 .....		- UTF-8 .....	100
- 卡配置 .....	49	编辑器 .....	
电话交换机 .....	347	- Emacs .....	178–179
监视器设置 .....	186	- vi .....	179
硬件 .....		网卡 .....	
- 信息 .....	45	- 网络 .....	342
- 硬盘控制器 .....	45	· 测试 .....	342
- CD-ROM .....	44	网络 .....	329
- ISDN .....	346	- 基本网络地址 .....	334
- SCSI 设备 .....	81	- 广播地址 .....	334
硬盘 .....		- 无线 .....	232
- DMA .....	45	- 本地主机 .....	334
磁盘 .....		- 网络掩码 .....	333
- 应急 .....	57	- 蓝牙 .....	232, 286
- 引导 .....	57	- 路由 .....	332, 333
· 创建 .....	160	- 路由选择 .....	52
- 软盘 .....		- 配置 49–52, <i>hyperpage</i> 351, 342 — —	351
· 格式化 .....	78	· IPv6 .....	340
移动能力 .....	227–234	- 配置文件 .....	353–359
- 便携式计算机 .....	228	- DHCP .....	50, 397
- 外部硬盘 .....	233	- DNS .....	341
- 手提电话 .....	234	- IrDA .....	232
- 数据安全 .....	232	- SLP .....	363
- 数码相机 .....	233	- TCP/IP .....	330
- 防火墙 (IEEE1394) .....	233	- WLAN .....	232
- PDA .....	234	- YaST .....	342
- USB .....	233	网络信息服务 .....	<i>see</i> NIS
端口 .....		网络文件系统 .....	<i>see</i> NFS
- 扫描 .....	495	网络鉴定 .....	
- 53 .....	377	- Kerberos .....	108
笔记本电脑 .....	<i>see</i> 便携式计算机	脚本 .....	
系统 .....		- boot.udev .....	307
- 冻结 .....	76	- init.d .....	137, 139–142, 359

· 引导	140
· boot.local	140
· boot.setup	141
· halt	141
· network	359
· nfsserver	359, 395
· portmap	359, 395
· rc	139, 141
· sendmail	359
· squid	487
· xinetd	359
· ypbind	359
· ypserv	359
- irda	292
- mkinitrd	135
- modify_resolvconf	178, 355
- SuSEconfig	143-145
· 禁用	145
蓝牙	232, 283
- 网络	286
- hciconfig	287
- hcitool	287
- opd	289
- pand	288
- sdptool	288
虚拟内存	61
虚拟控制台	
- 切换	63
解析程序库	
- .local 作为顶级域	100
许可证	<i>see</i> GPL
设备节点	
- udev	303
语言	64
调制解调器	
- 电缆	349
- YaST	344
路由	332
- 网络掩码	333
路由选择	52, 354
- 伪装	506
- 路由	354
- 静态	354
软 RAID	<i>see</i> RAID
软件	
- 去除	33-38
- 安装	33-38
- 编译	116
软盘	
- 引导	148

轻量级目录访问协议	<i>see</i> LDAP
输入方法	
- CJK	182
运行级别	63, 137-139
- 切换	63
- 在 YaST 中编辑	142
- 更改	138-139
逻辑卷管理器	<i>see</i> LVM
配置	143
- 便携式计算机	237-242
- 图形卡	188
- 声卡	48
- 安全性	52-56
- 打印	211-213
- 扫描仪	46
- 无线电	49
- 时区	64
- 游戏杆	194
- 用户	52
- 电子邮件	50
- 电缆调制解调器	349
- 电视	49
- 硬件	44-49
- 硬盘	
- DMA	45
- 硬盘控制器	45
- 系统	31-65
- 系统服务	51
- 组	53
- 网络	49-52, 342
- 手工	351
- 语言	64
- 调制解调器	344
- 路由选择	52, 354
- 软件	33-43
- 键盘	194
- 防火墙	55
- 鼠标	194
- Apache	438-442
- CD-ROM	44
- DNS	51, 367
- DSL	349
- GRUB	149, 155
- IPv6	340
- IrDA	292
- ISDN	346
- NFS	51
- NTP	
- 客户机	51
- PAM	109



- Samba	475–478	- irda	292
· 客户机	52, 480	- logrotate.conf	175
· 服务器	52	- menu.lst	150
- Squid	489	- modprobe.conf	48, 98, 169
- SSH	512	- modules.conf	98
- T-DSL	351	- modules.dep	169
- X	186	- named.conf	373, 376–382, 488
配置文件	353	- networks	356
- .bashrc	174, 177	- nscd.conf	358
- .emacs	178	- nsswitch.conf	357, 424
- .mailsync	469	- pam_unix2.conf	424
- .profile	174	- passwd	94
- .xsession	515	- powersave	256
- 内核	135	- powersave.conf	106
- 导出	395, 494	- resolv.conf	178, 354, 373, 487
- 服务	478	- samba	478
- 权限	525	- slapd.conf	416
- 热插拔	296	- smb.conf	474, 475
- 网络	354	- smppd.conf	360
- 语言	182, 184	- smpppd-c.conf	361
- 路由	354	- squid.conf	487, 489, 491, 493, 495, 497
- 配置文件	174, 177, 184	- squidguard.conf	497
- acpi	257	- sshd_config	515
- apache2	438	- suseconfig	145
- asound.conf	48	- sysconfig	63, 143–145
- config	166	- termcap	182
- crontab	174	- wireless	354
- csh.cshrc	184	- XF86Config	<i>see</i> 配置文件, xorg.conf
- dhclient.conf	399	- xml/catalog	97
- dhcp	354	- xml/suse-catalog.xml	97
- dhcpd.conf	400	- xorg.conf	109, 194
- foomatic/filter.conf	97	· Device	197
- fstab	62, 125	· Monitor	198
- group	94	· Screen	196
- grub.conf	155	鉴定	
- gshadow	101	- PAM	319–325
- host.conf	356	错误讯息	
· alert	356	- 权限被拒绝	62
· multi	356	- 错误解释器	62
· nospoof	356	键盘	
· order	356	- 亚洲字符	182
· trim	356	- 布局	181
- HOSTNAME	359	- 映射	181
- hosts	51, 342, 355	· 组合	182
- httpd.conf	438, 439	· multikey	182
- hwinfo	299	- 配置	194
- hwup	298	- X 键盘扩展	182
- ifcfg-*	353	- XKB	182
- inittab	136–139, 181	闪存盘	233
- inputrc	182	- 引导	148

防火墙 .....	55, 504
- 包过滤器 .....	504, 507
- Squid .....	494
- SuSEfirewall2 .....	504, 508
防火墙 (IEEE1394)	
- 硬盘 .....	233
音效	
- 混音器 .....	107
鼠标	
- 配置 .....	194
64 位 Linux .....	129
- 内核规范 .....	132
- 软件开发 .....	130
- 运行时支持 .....	130

## A

ACL .....	529–538
- 处理 .....	531
- 定义 .....	530
- 掩码 .....	534
- 支持 .....	538
- 效果 .....	535
- 权限位 .....	532
- 检查算法 .....	537
- 结构 .....	531
- 访问 .....	530, 533
- 默认 .....	530, 535
ACPI	
- 禁用 .....	5
Apache .....	51, 433–453
- 内容协商 .....	436
- 启动 .....	437
- 安全性 .....	450–451
- 安装 .....	437–438
- 日志记录 .....	441, 443
- 权限 .....	440, 451
- 查错 .....	451
- 标志 .....	439
- 模块 .....	435
· 激活 .....	439
· 装载 .....	440
· mod_perl .....	445
· mod_php4 .....	447
· mod_python .....	447
· mod_ruby .....	447
- 线程 .....	436
- 虚拟主机 .....	436, 448–450
- 配置 .....	438–442
- 默认页 .....	435
- apxs .....	438

- CGI .....	444
- DocumentRoot .....	439
- Squid .....	495
- SSI .....	442, 444

## B

Bash	
- .bashrc .....	174
- .profile .....	174
- 配置文件 .....	174
BIND .....	372–382
BIOS	
- 引导顺序 .....	4
- 病毒防护 .....	76
booting .....	568, 572

## C

CD	
- 引导 .....	4, 148
CD-ROM 驱动器	
- 支持的 .....	80
chown .....	100
CJK .....	182
commands	
- jfs_fsck .....	572
- xfs_check .....	568
cpuspeed .....	262
crashes .....	568, 572
cron .....	174
CVS .....	456, 462–464

## D

deltarpm .....	113
depmod .....	169
DHCP .....	50, 397–404
- 使用 YaST 进行配置 .....	398
- 包 .....	399
- 服务器 .....	400–403
- 静态地址指派 .....	403
- dhcpd .....	400–403
DNS .....	341
- 区域	
· 文件 .....	380
- 反向查找 .....	382
- 名称服务器 .....	354
- 启动 .....	375
- 域 .....	354
- 安全性 .....	524
- 日志记录 .....	378
- 查错 .....	375

- 转发 .....	375
- 选项 .....	377
- 邮件交换器 .....	342
- 配置 .....	51, 367
- 顶级域 .....	341
- BIND .....	372-382
- NIC .....	342
- Squid .....	488
DOS	
- 共享文件 .....	473

<b>E</b>	
e2fsck .....	566
Emacs .....	178-179
- .emacs .....	178
- default.el .....	178
Evolution .....	234

<b>F</b>	
FAT 文件系统 .....	15
file systems	
- checking .....	563-574
- jfs_fsck .....	572
- xfs_check .....	568

<b>G</b>	
GPL .....	575
GRUB .....	147-163
- 主引导记录 (MBR) .....	148
- 分区名 .....	151
- 卸载 .....	160
- 命令 .....	149-157
- 引导 .....	149
- 引导口令 .....	156
- 引导扇区 .....	148
- 引导管理 .....	148
- 引导菜单 .....	150
- 查错 .....	162
- 菜单编辑器 .....	153
- 设备名 .....	151
- 通配符 .....	154
- 限制 .....	149
- device.map .....	149, 155
- GRUB Geom 错误 .....	162
- GRUB shell .....	156
- grub.conf .....	149, 155
- JFS 和 GRUB .....	162
- menu.lst .....	149, 150

<b>H</b>	
hciconfig .....	287

hcitool .....	287
head .....	100
hwinfo .....	299

<b>I</b>	
I18N .....	182
inetd .....	51, 96
init .....	136-137
- 添加脚本 .....	141
- 脚本 .....	139-142
- inittab .....	136
insmod .....	169
installing	
- network, from .....	80

IP 地址	
- 专用 .....	334
- 伪装 .....	506
- 动态指派 .....	397
- 类别 .....	333
- IPv6 .....	335
- 配置 .....	340
IrDA .....	232, 291-293
- 停止 .....	292
- 启动 .....	292
- 查错 .....	292
- 配置 .....	292

<b>J</b>	
jade .....	<i>see</i> SGML, openjade
jade.dsl .....	96
jfs_fsck .....	572

<b>K</b>	
Kmod .....	<i>see</i> 内核, 模块装载程序
Kontakt .....	234
KPilot .....	234
KPowersave .....	230
KSysguard .....	230

<b>L</b>	
L10N .....	182
LDAP .....	51, 411-431
- 修改数据 .....	422
- 删除数据 .....	423
- 搜索数据 .....	423
- 服务器配置 .....	416
- 添加数据 .....	420
- 目录树 .....	413
- 管理用户 .....	429

- 管理组 .....	429
- 访问控制 .....	419
- ACL .....	417
- ldapadd .....	420
- ldapdelete .....	423
- ldapmodify .....	422
- ldapsearch .....	423
- YaST	
· 模块 .....	425
· 模板 .....	425
- YaST LDAP 客户程序 .....	424
LFS .....	316
LILO	
- 配置 .....	76
Linux	
- 与其他操作系统共享文件 .....	473
- 卸载 .....	160
- 网络 .....	329
linuxrc .....	72
- 手工安装 .....	108
linuxthreads .....	99
locate .....	176
logrotate .....	175
LSB	
- 安装包 .....	110
lsmmod .....	169
LVM	
- YaST .....	81
<b>M</b>	
MBR .....	148
modinfo .....	169
modprobe .....	169
mountd .....	395
<b>N</b>	
NAT .....	<i>see</i> 伪装
NetBIOS .....	474
NFS .....	391
- 客户机 .....	51, 392
- 导入 .....	392
- 导出 .....	394
- 服务器 .....	51, 393
- 权限 .....	395
- 装入 .....	392
nfsd .....	395
NGPT .....	99
nice .....	100
NIS .....	51, 385-389
- 主服务器 .....	386-388

- 从属服务器 .....	386-388
- 客户程序 .....	388
NPTL .....	99
NSS .....	357
- 数据库 .....	357
NTFS 文件系统 .....	15
NTP	
- 客户机 .....	51
nVidia .....	96

## O

opd .....	289
OpenSSH .....	<i>see</i> SSH
OS/2	
- 共享文件 .....	473

## P

PAM .....	319-325
- 配置 .....	109
pand .....	288
PCMCIA .....	228, 236
- 卡管理器 .....	236
- 实用程序 .....	239
- 查错 .....	239
- 网卡 .....	238
- 调制解调器 .....	238
- 配置 .....	237
- IrDA .....	291-293
- ISDN .....	238
- SCSI .....	238
PDA .....	234
PostgreSQL	
- 更新 .....	95
powersave .....	262
- 配置 .....	263

## R

RAID	
- YaST .....	87
reiserfsck .....	563
RFC .....	330
rmmmod .....	169
RPM .....	110-119
- 依赖性 .....	111
- 卸载 .....	112
- 增补程序 .....	112
- 安全性 .....	526
- 工具 .....	119
- 数据库	
· 重建 .....	112, 116

- 更新 .....	111
- 查询 .....	114
- 校验 .....	110, 116
- 版本 .....	97
- deltarpm .....	113
- rpmnew .....	111
- rpmorig .....	111
- rpmsave .....	111
- SRPMS .....	117
rpmbuild .....	97, 110
rsync .....	457, 467

## S

Samba .....	473–482
- 优化 .....	481
- 停止 .....	475
- 共享 .....	475, 476
- 名称 .....	475
- 启动 .....	475
- 安全性 .....	477–478
- 安装 .....	475
- 客户机 .....	52, 475, 480–481
- 帮助 .....	482
- 打印 .....	481
- 打印机 .....	475
- 服务器 .....	52, 475–478
- 权限 .....	477
- 登录 .....	478
- 配置 .....	475–478
- SMB .....	474
- swat .....	478
- TCP/IP 和 .....	474
SaX .....	186
- 多头 .....	191
SCPM .....	63, 243
- 便携式计算机 .....	229
- 切换配置文件 .....	246
- 启动 .....	245
- 管理配置文件 .....	245
- 资源组 .....	245
- 高级设置 .....	246
SCSI 设备 .....	
- 文件名, 指派 .....	81
- 配置 .....	81
sdptool .....	288
SGML .....	
- 目录 .....	103
- openjade .....	96
SLP .....	230, 363
- 注册服务 .....	364
- 浏览器 .....	365
- Konqueror .....	365
- slptool .....	365
SMB .....	see Samba
sort .....	100
spm .....	116
Squid .....	483
- 停止 .....	487
- 功能 .....	484
- 卸载 .....	488
- 启动 .....	487
- 安全性 .....	484
- 对象状态 .....	485
- 报告 .....	498
- 损坏的超速缓存 .....	487
- 日志文件 .....	487, 490, 495
- 权限 .....	487, 491
- 查错 .....	487
- 目录 .....	487
- 系统要求 .....	485
- 统计数字 .....	495, 496
- 访问控制 .....	495
- 超速缓存 .....	484
- 大小 .....	486
- 透明代理 .....	493, 495
- 配置 .....	489
- 防火墙 .....	494
- ACL .....	491
- Apache .....	495
- cachemgr.cgi .....	495, 496
- Calamaris .....	498
- CPU .....	486
- DNS .....	488
- RAM .....	486
- squidGuard .....	497
SSH .....	512–516
- 守护程序 .....	513
- 密钥对 .....	514, 515
- 鉴定机制 .....	514
- scp .....	513
- sftp .....	513
- ssh .....	512
- ssh-agent .....	515
- ssh-keygen .....	515
- sshd .....	513
- X .....	515
subfs .....	
- 可移动媒体 .....	103
subversion .....	457, 464
SUSE LINUX .....	

- 安装 ..... 72  
sx ..... 97

## T

tail ..... 100  
TCP/IP ..... 330  
- 包 ..... 330, 331  
- 层次模型 ..... 330  
- ICMP ..... 330  
- IGMP ..... 330  
- TCP ..... 330  
- UDP ..... 330

## U

udev ..... 303  
- 关键字 ..... 305  
- 启动脚本 ..... 307  
- 大容量储存 ..... 306  
- 硬盘 ..... 307  
- 自动操作 ..... 305  
- 规则 ..... 304  
- 通配符 ..... 305  
- sysfs ..... 306  
- udevinfo ..... 306  
ulimit ..... 176  
- 选项 ..... 176  
USB  
- 硬盘 ..... 233  
- 闪存盘 ..... 233  
UTF-8  
- 编码 ..... 100

## V

VNC  
- 安装 ..... 73  
- 管理 ..... 52

## W

whois ..... 342  
Windows  
- 共享文件 ..... 473  
WLAN ..... 232

## X

X ..... 185  
- 优化 ..... 194–199  
- 多头 ..... 191  
- 字体 ..... 199  
- 字体系统 ..... 199  
- 字符集 ..... 199

- 安全性 ..... 522  
- 帮助 ..... 198  
- 虚拟屏幕 ..... 197  
- 配置 ..... 186  
- 驱动程序 ..... 198  
- 3D ..... 190  
- CID-keyed 字体 ..... 203  
- SaX2 ..... 195  
- SSH ..... 515  
- TrueType 字体 ..... 199  
- X11 核心字体 ..... 202  
- xfb6config ..... 195  
- Xft ..... 199  
- xft ..... 199

X 窗口系统 ..... *see* X

X 键盘扩展 ..... *see* 键盘, X 键盘扩展

X.Org ..... *see* 键盘, X 键盘扩展

xfs.check ..... 568

Xft ..... 199

xinetd ..... 96

XKB ..... *see* 键盘, X 键盘扩展

XML

- 目录 ..... 103

- 编目 ..... 97

- openjade ..... 96

xorg.conf

- 文件 ..... 195

- 方式行 ..... 195

- 监视器 ..... 195

- 颜色深度 ..... 197

- Depth ..... 197

- Device ..... 197

- Display ..... 197

- InputDevice ..... 195

- Modeline ..... 197

- Modes ..... *see* 195, 197

- Monitor ..... 197

- ServerFlags ..... 195

## Y

YaST  
- 主机名 ..... 51  
- 修复系统 ..... 121  
- 分区 ..... 9, 58  
- 包依赖性 ..... 18  
- 包管理器 ..... 34  
- 启动 ..... 4, 32  
- 图形卡 ..... 186, 188  
- 图形用户界面 ..... 186–193  
- 声卡 ..... 48

- 备份 .....	43, 56	- 语言 .....	64
- 媒体检查 .....	44	- 语言选择 .....	7, 32
- 安全性 .....	52-56	- 调制解调器 .....	344
- 安全设置 .....	6	- 路由选择 .....	52
- 安装 .....	3-28	- 软件 .....	33-43
- 安装建议 .....	7	- 软件更新程序 .....	23
- 安装方式 .....	7	- 运行级别 .....	142
- 安装源 .....	39	- 配置 .....	31-65
- 安装范围 .....	18	- 配置文件管理器 .....	63
- 引导方式 .....	19	- 键盘布局 .....	8, 194
- 引导配置 .....	157	- 防火墙 .....	55
- 打印 .....	211-213	- 驱动程序 CD .....	65
- 扫描仪 .....	46	- 鼠标 .....	8, 194
- 控制中心 .....	32	- 3D .....	204
- 支持请求 .....	64	- CD-ROM .....	44
- 文本方式 .....	65-70, 74-75	- DHCP .....	398
· 查错 .....	75	- DMA .....	45
· 模块 .....	68	- DNS .....	51
- 无线网卡 .....	49	- DSL .....	349
- 时区 .....	64	- ISDN .....	346
- 更新 .....	41, 42, 95	- LDAP 客户程序 .....	424
- 根口令 .....	21	- LVM .....	58, 81
- 游戏杆 .....	194	- ncurses .....	65
- 用户管理 .....	52	- NFS 客户机 .....	51
- 电子邮件 .....	50	- NFS 服务器 .....	51
- 电源管理 .....	269	- NIS 客户机 .....	25
- 电缆调制解调器 .....	349	- NIS 客户程序 .....	388
- 电视卡 .....	49	- NTP .....	
- 监视器设置 .....	186	· 客户机 .....	51
- 硬件 .....	44-49	- RAID .....	87
- 硬件信息 .....	45	- Samba .....	
- 硬盘控制器 .....	45	· 客户机 .....	52, 480
- 磁盘创建 .....	57	· 服务器 .....	52
- 磁盘空间 .....	10	- SCPM .....	63
- 系统启动 .....	4	- sendmail .....	50
- 系统安全性 .....	54	- SLP 浏览器 .....	365
- 组管理 .....	53	- sysconfig 编辑器 .....	63, 145
- 网卡 .....	342	- T-DSL .....	351
- 网络配置 .....	21, 49-52	- YOU .....	39-41
- 联机更新 .....	39-41, 68	YP .....	see NIS