管理指南

SUSE Linux Enterprise Server 12 SP5

管理指南

SUSE Linux Enterprise Server 12 SP5

讲述系统管理任务，如维护、监视和自定义初始安装的系统。

出版日期： 2019 年 12 月 02 日

SUSE LLC
10 Canal Park Drive
Suite 200
Cambridge MA 02141
USA
https://www.suse.com/documentation

版权所有 © 2006– 2019 SUSE LLC 和贡献者。保留所有权利。

根据 GNU 自由文档许可证 (GNU Free Documentation License) 版本 1.2 或（根据您的选择）版本 1.3 中的条款，在此授予您复制、分发和/或修改本文档的许可权限；本版权声明和许可证附带不可变部分。许可证版本 1.2 的副本包含在题为“GNU 自由文档许可证”的部分。

有关 SUSE 商标，请参见 http://www.suse.com/company/legal/。所有其它第三方商标是其各自所有者的财产。商标符号（®、™ 等）代表 SUSE 及其附属公司的商标。星号 (*) 代表第三方商标。

本指南力求涵盖所有详细信息。但这并不确保本指南准确无误。SUSE LLC 及其附属公司、作者和译者对于可能出现的错误或由此造成的后果皆不承担责任。

目录

关于本指南 xxi

I 常用任务 1

1 Bash 和 Bash 脚本 2

1.1 什么是“外壳”？ 2

了解 Bash 配置文件 2 • 目录结构 3

1.2 编写外壳脚本 7

1.3 重定向命令事件 8

1.4 使用别名 9

1.5 在 Bash 中使用变量 10

使用自变量 11 • 使用变量替换 11

1.6 将命令分组和组合 12

1.7 使用通用流程构造语句 13

if 控制命令 13 • 使用 `for` 命令创建循环 14

1.8 更多信息 14

2 sudo 16

2.1 `sudo` 基本用法 16

运行单个命令 16 • 启动外壳 17 • 环境变量 18

2.2 配置 `sudo` 18

编辑配置文件 19 • sudoers 基本配置语法 19 • sudoers 中的规则 21

2.3 常见使用情况 23
在无需 root 口令的情况下使用 sudo 23 • 对 X.Org 应用程序使用 sudo 24
2.4 更多信息 25
3 YaST 联机更新 26
3.1 联机更新对话框 27
3.2 安装增补程序 28
3.3 自动联机更新 29
4 YaST 31
4.1 高级组合键 31
5 文本方式的 YaST 33
5.1 在模块中导航 34
5.2 高级组合键 35
5.3 组合键的限制 36
5.4 YaST 命令行选项 36
启动单个模块 37 • 从命令行安装包 37 • YaST 模块的命令行参数 37
6 使用命令行工具管理软件 39
6.1 使用 Zypper 39
一般使用 39 • 使用 Zypper 安装和删除软件 40 • 使用 Zypper 更新软件 45 • 识别使用已删除文件的进程和服务 49 • 用 Zypper 管理安装源 51 • 用 Zypper 查询储存库和包 53 • 配置 Zypper 55 • 查错 55 • Btrfs 文件系统上的 Zypper 回滚功能 55 • 更多信息 56

6.2 RPM — 包管理器 56
校验包真实性 56 • 管理包：安装、更新和卸装 57 • 增量 RPM 包 58 • RPM 查询 59 • 安装和编译源包 61 • 使用 build 编译 RPM 包 63 • 用于 RPM 存档和 RPM 数据库的工具 64

7 通过 Snapper 进行系统恢复和快照管理 65

7.1 默认设置 65
快照类型 66 • 快照中排除的目录 67 • 自定义设置 68

7.2 使用 Snapper 撤销更改 72
撤销 YaST 和 Zypper 更改 73 • 使用 Snapper 恢复文件 78

7.3 通过从快照引导来执行系统回滚 79
回滚后的快照 81 • 访问和识别快照引导项 82 • 限制 83

7.4 在用户主目录中启用 Snapper 84
安装 pam_snapper 和创建用户 85 • 去除用户 86 • 手动启用主目录中的快照 86

7.5 创建并修改 Snapper 配置 86
管理现有配置 88

7.6 手动创建和管理快照 91
快照元数据 91 • 创建快照 93 • 修改快照元数据 94 • 删除快照 94

7.7 自动清理快照 96
清理编号快照 96 • 清理时间线快照 98 • 清理没有差异的快照对 99 • 清理手动创建的快照 100 • 添加磁盘定额支持 100

7.8 常见问题 101

8 使用 VNC 远程访问 103

8.1 `vncviewer` 客户端 103
使用 vncviewer CLI 进行连接 103 • 使用 vncviewer GUI 进行连接 104 • 未加密连接通知 104

8.2 Remmina：远程桌面客户端 104
安装 104 • 主窗口 105 • 添加远程会话 105 • 启动远程会话 106 • 编辑、复制和删除保存的会话 108 • 从命令行运行远程会话 108
8.3 一次性 VNC 会话 109
可用配置 110 • 启动一次性 VNC 会话 110 • 配置一次性 VNC 会话 111
8.4 持续 VNC 会话 111
使用 vncserver 启动的 VNC 会话 112 • 使用 vncmanager 启动的 VNC 会话 113
8.5 加密 VNC 通讯 117
9 使用 RSync 复制文件 119
9.1 概念概述 119
9.2 基本语法 119
9.3 在本地复制文件和目录 120
9.4 远程复制文件和目录 121
9.5 配置和使用 Rsync 服务器 121
9.6 更多信息 124
II 引导 LINUX 系统 125
10 引导过程简介 126
10.1 术语 126
10.2 Linux 引导进程 126
初始化和引导加载程序阶段 127 • 内核阶段 128 • init on initramfs 阶段 131 • systemd 阶段 133

11 UEFI（统一可扩展固件接口） 134

11.1 安全引导 134

在 SUSE Linux Enterprise Server 上实施 135 • MOK（机器拥有者密钥） 137 • 引导自定义内核 138 • 使用非内置驱动程序 140 • 功能和限制 141

11.2 更多信息 142

12 引导加载程序 GRUB 2 143

12.1 GRUB Legacy 与 GRUB 2 之间的主要差异 143

12.2 配置文件结构 143

文件 /boot/grub2/grub.cfg 144 • 文件 /etc/default/grub 145 • /etc/grub.d 中的脚本 147 • BIOS 驱动器与 Linux 设备之间的映射 149 • 在引导过程中编辑菜单项 149 • 设置引导口令 151

12.3 使用 YaST 配置引导加载程序 152

引导加载程序位置和引导代码选项 153 • 调整磁盘顺序 155 • 配置高级选项 155

12.4 IBM Z 上终端使用方式的差异 158

限制 158 • 组合键 159

12.5 有用的 GRUB 2 命令 161

12.6 更多信息 162

13 systemd 守护程序 163

13.1 systemd 概念 163

systemd 是什么 163 • 单元文件 164

13.2 基本用途 164

管理正在运行的系统中的服务 165 • 永久启用/禁用服务 167

13.3 系统启动和目标管理 168
目标与运行级别的比较 168 • 调试系统的启动 172 • System V 兼容性 175
13.4 使用 YaST 管理 服务 176
13.5 systemd 自定义 177
自定义服务文件 177 • 创建“插入式”文件 178 • 创建自定义目标 178
13.6 高级用途 179
清理临时目录 179 • 系统日志 180 • 快照 180 • 装载内核模块 180 • 装载服务之前执行必要操作 181 • 内核控制组 (cgroups) 182 • 终止服务（发送信号） 183 • 有关 D-Bus 服务的重要说明 184 • 调试服务 184
13.7 更多信息 186
III 系统 187
14 64 位系统环境中的 32 位和 64 位应用程序 188
14.1 运行时支持 188
14.2 内核规范 189
15 journalctl: 查询 systemd 日记 190
15.1 将日记设为永久 190
15.2 journalctl 的有用开关 191
15.3 过滤日记输出 192
根据引导编号过滤 192 • 根据时间间隔过滤 193 • 根据字段过滤 193
15.4 调查 systemd 错误 194
15.5 Journald 配置 196
更改日记大小限制 196 • 将日记转发到 /dev/ttyX 196 • 将日记转发到 Syslog 工具 196

15.6 使用 YaST 过滤 systemd 日记 197

16 基本联网知识 198

16.1 IP 地址和路由 200

IP 地址 200 • 网络掩码和路由 201

16.2 IPv6 — 下一代的因特网 203

优点 203 • 地址类型和结构 205 • IPv4 与 IPv6 并存 208 • 配置 IPv6 209 • 更多信息 209

16.3 名称解析 210

16.4 使用 YaST 配置网络连接 211

使用 YaST 配置网卡 212 • IBM Z：配置网络设备 222

16.5 手动配置网络连接 224

wicked 网络配置 224 • 配置文件 231 • 测试配置 242 • 单元文件和启动脚本 245

16.6 路由器基本设置 246

16.7 设置绑定设备 248

绑定从属的热插拔 250

16.8 设置小组设备以进行网络协作 252

使用案例：使用网络协作实现负载平衡 255 • 使用案例：使用网络协作实现故障转移 256 • 用例：组合设备上的 VLAN 257

16.9 采用 Open vSwitch 的软件定义网络 259

Open vSwitch 的优点 259 • 安装 Open vSwitch 260 • Open vSwitch 守护程序和实用程序概述 260 • 使用 Open vSwitch 创建网桥 261 • 将 Open vSwitch 直接与 KVM 配合使用 262 • 将 Open vSwitch 与 libvirt 搭配使用 264 • 更多信息 265

17 打印机操作 266

17.1 CUPS 工作流程 267

17.2 连接打印机的方法和协议 268

17.3 安装软件 268

17.4 网络打印机 269

17.5 使用命令行工具配置 CUPS 270

17.6 从命令行打印 271

17.7 SUSE Linux Enterprise Server 中的特殊功能 271

CUPS 和防火墙 272 • 浏览网络打印机 272 • 多种包中的 PPD 文件 273

17.8 查错 273

打印机没有标准打印机语言支持 273 • 没有合适的 PPD 文件可用于 PostScript 打印机 274 • 网络打印机连接 274 • 打印件有问题但没有错误消息 277 • 禁用的队列 277 • CUPS 浏览：删除打印任务 277 • 有问题的打印任务和数据传送错误 278 • 调试 CUPS 278 • 更多信息 279

18 X Window 系统 280

18.1 安装和配置字体 280

显示安装的字体 281 • 查看字体 282 • 查询字体 282 • 安装字体 283 • 配置字体的外观 284

18.2 更多信息 292

19 使用 FUSE 访问文件系统 294

19.1 配置 FUSE 294

19.2 装入 NTFS 分区 294

19.3 更多信息 295

20 管理内核模块 296

20.1 使用 lsmod 和 modinfo 列出装载的模块 296

20.2 添加和去除内核模块 297
引导时自动装载内核模块 297 • 使用 modprobe 将内核模块列入黑名单 298

21 使用 udev 进行动态内核设备管理 300

21.1 /dev 目录 300

21.2 内核 uevents 和 udev 300

21.3 驱动程序、内核模块和设备 301

21.4 引导和启动设备设置 301

21.5 监视正在运行的 udev 守护程序 302

21.6 使用 udev 规则影响内核设备事件处理 303
在 udev 规则中使用运算符 305 • 在 udev 规则中使用替换项 305 • 使用 udev 匹配键 306 • 使用 udev 指派键 307

21.7 永久设备命名 309

21.8 udev 使用的文件 310

21.9 更多信息 311

22 使用 kGraft 在线增补 Linux 内核 312

22.1 kGraft 的优势 312

22.2 kGraft 的底层函数 313

22.3 安装 kGraft 增补程序 313
激活 SLE Live Patching 314 • 更新系统 314

22.4 增补程序生命周期 315

22.5 去除 kGraft 增补程序 315

22.6 阻塞的内核执行线程 316

22.7 kgr 工具 316

22.8 kGraft 技术的应用范围 316

22.9 SLE Live Patching 的应用范围 317

22.10 使用支持流程与我们交互 317

23 特别的系统功能组件 318

23.1 特殊软件包的相关信息 318

bash 包和 /etc/profile 318 • cron 包 319 • 停止 Cron 状态讯息 320 • 日志文件：包 logrotate 320 • locate 命令 320 • ulimit 命令 320 • free 命令 322 • 手册页和信息页 322 • 使用 man 命令选择手册页 322 • GNU Emacs 的设置 323

23.2 虚拟控制台 324

23.3 键盘映射 324

23.4 语言和国家/地区特定的设置 325

一些示例 326 • ~/.i18n 中的语言环境设置 327 • 语言支持的设置 327 • 更多信息 328

24 持久内存 329

24.1 简介 329

24.2 术语 330

24.3 使用案例 332

将 PMEM 与 DAX 搭配使用 332 • 将 PMEM 与 BTT 搭配使用 332 • BLK 储存 333

24.4 用于管理持久内存的工具 333

24.5 设置持久内存 334

查看可用的 NVDIMM 储存 334 • 将储存配置为使用 DAX 的单个 PMEM 名称空间 336 • 创建使用 BTT 的 PMEM 名称空间 338 • 创建 BLK 名称空间 340

24.6 查错 342
查找故障模块 342 • 测试持久内存 343
24.7 更多信息 344

IV 服务 346

25 使用 NTP 同步时间 347
25.1 使用 YaST 配置 NTP 客户端 347
基本配置 347 • 更改基本配置 348
25.2 手动配置网络中的 NTP 350
25.3 运行时动态时间同步 351
25.4 设置本地参考时钟 352
25.5 与外部时间参考 (ETR) 的时钟同步 352

26 域名系统 353
26.1 DNS 术语 353
26.2 安装 354
26.3 使用 YaST 进行配置 354
向导配置 354 • 专家配置 357
26.4 启动 BIND 名称服务器 364
26.5 /etc/named.conf 配置文件 366
重要的配置选项 367 • 日志记录 368 • 区域项 369
26.6 区域文件 370
26.7 区域数据的动态更新 374
26.8 安全事务 374
26.9 DNS 安全性 376

26.10 更多信息 376

27 DHCP 377

27.1 使用 YaST 配置 DHCP 服务器 378
初始配置（向导） 378 • DHCP 服务器配置（专家） 383

27.2 DHCP 软件包 388

27.3 DHCP 服务器 dhcpd 389
具有固定 IP 地址的客户端 390 • SUSE Linux Enterprise Server 版本 391

27.4 更多信息 392

28 通过 NFS 共享文件系统 393

28.1 概述 393

28.2 安装 NFS 服务器 394

28.3 配置 NFS 服务器 394
使用 YaST 导出文件系统 395 • 手动导出文件系统 396 • 采用 Kerberos 的 NFS 399

28.4 配置客户端 399
使用 Yast 导入文件系统 399 • 手动导入文件系统 400 • 并行 NFS (pNFS) 402

28.5 更多信息 403

29 Samba 404

29.1 术语 404

29.2 安装 Samba 服务器 405

29.3 启动和停止 Samba 405

29.4 配置 Samba 服务器 406
使用 YaST 配置 Samba 服务器 406 • 手动配置服务器 408

29.5 配置客户端 412
使用 YaST 配置 Samba 客户端 412
29.6 将 Samba 用作登录服务器 413
29.7 带有 Active Directory 的网络中的 Samba 服务器 414
29.8 高级主题 415
Btrfs 上的透明文件压缩 415 • 快照 416
29.9 更多信息 425
30 使用 Autofs 按需装入 426
30.1 安装 426
30.2 配置 426
Master 映射文件 426 • 映射文件 428
30.3 操作和调试 429
控制 autofs 服务 429 • 调试自动装入器问题 430
30.4 自动装入 NFS 共享 431
30.5 高级主题 432
/net 安装点 432 • 使用通配符自动装入子目录 433 • 自动装入 CIFS 文件系统 433
31 SLP 434
31.1 SLP 前端 slptool 434
31.2 通过 SLP 提供服务 435
设置 SLP 安装服务器 437
31.3 更多信息 437

32 Apache HTTP 服务器 438

32.1 快速入门 438

要求 438 • 安装 438 • 开始 439

32.2 配置 Apache 439

Apache 配置文件 440 • 手动配置 Apache 443 • 使用 YaST 配置 Apache 448

32.3 启动和停止 Apache 453

32.4 安装、激活和配置模块 455

模块安装 456 • 激活和停用 456 • 基础模块和扩展模块 456 • 多处理模块 459 • 外部模块 460 • 编译 461

32.5 启用 CGI 脚本 462

Apache 配置 462 • 运行示例脚本 463 • CGI 查错 464

32.6 使用 SSL 设置安全性 Web 服务器 464

创建 SSL 证书 465 • 使用 SSL 配置 Apache 468

32.7 在同一服务器上运行多个 Apache 实例 470

32.8 避免安全性问题 473

最新软件 473 • DocumentRoot 权限 474 • 文件系统访问权 474 • CGI 脚本 474 • 用户目录 475

32.9 查错 475

32.10 更多信息 476

Apache 2.4 476 • Apache 模块 476 • 开发 477 • 其他来源 477

33 使用 YaST 设置 FTP 服务器 478

33.1 启动 FTP 服务器 478

33.2 FTP 常规设置 479

33.3 FTP 性能设置 480

33.4 身份验证 480

33.5 专家设置 481

33.6 更多信息 481

34 代理服务器 Squid 482

34.1 有关代理缓存的一些事实 482

Squid 和安全性 483 • 多个缓存 483 • 缓存因特网对象 484

34.2 系统要求 484

RAM 484 • CPU 485 • 磁盘缓存的大小 485 • 硬盘/SSD 体系结构 485

34.3 Squid 的基本用法 486

启动 Squid 486 • 检查 Squid 是否正在工作 486 • 停止、重新装载和重启动 Squid 488 • 去除 Squid 489 • 本地 DNS 服务器 489

34.4 YaST Squid 模块 490

34.5 Squid 配置文件 491

一般配置选项 491 • 访问控制选项 494

34.6 配置透明代理 497

34.7 使用 Squid 超速缓存管理器 CGI 接口 (cachemgr.cgi) 500

34.8 squidGuard 502

34.9 使用 Calamaris 生成缓存报告 503

34.10 更多信息 504

35 使用 SFCB 的基于 Web 的企业管理 505

35.1 简介和基本概念 505

35.2 设置 SFCB 506

安装其他提供程序 508 • 启动、停止 SFCB 和检查其状态 508 • 确保安全访问 509

35.3 SFCB CIMOM 配置 511
环境变量 512 • 命令行选项 513 • SFCB 配置文件 514
35.4 高级 SFCB 任务 526
安装 CMPI 提供程序 526 • 测试 SFCB 530 • 命令行 CIM 客户端：wbemcli 532
35.5 更多信息 535

V 移动计算机 536

36 Linux 中的移动计算 537
36.1 便携式计算机 537
省电 537 • 在变化的操作环境中集成 538 • 软件选择 540 • 数据安全性 544
36.2 移动硬件 545
36.3 移动设备（智能手机和平板电脑） 545

37 使用 NetworkManager 547
37.1 NetworkManager 的用例 547
37.2 启用或禁用 NetworkManager 547
37.3 配置网络连接 548
管理有线网络连接 550 • 管理无线网络连接 550 • 将 Wi-Fi/蓝牙网卡配置为接入点 551 • NetworkManager 和 VPN 551
37.4 NetworkManager 和安全性 552
用户和系统连接 553 • 储存密码和身份凭证 553
37.5 常见问题 (FAQ) 553
37.6 查错 555
37.7 更多信息 555

38 电源管理 557
38.1 省电功能 557
38.2 高级配置和电源接口 (ACPI) 558
控制 CPU 性能 558 • 故障诊断 559
38.3 硬盘的休眠 560
38.4 查错 562
CPU 频率不工作 562
38.5 更多信息 562

VI 查错 563
39 帮助和文档 564
39.1 文档目录 564
SUSE 手册 565 • 包文档 565
39.2 手册页 566
39.3 信息页 567
39.4 联机资源 567
40 收集用于支持的系统信息 569
40.1 显示当前系统信息 569
40.2 使用 Supportconfig 收集系统信息 570
创建服务请求编号 570 • 使用 YaST 创建 Supportconfig 存档 570 • 从命令行创建 Supportconfig 存档 573 • 常用的 supportconfig 选项 573
40.3 将信息提交到全球技术支持 574
40.4 分析系统信息 576
SCA 命令行工具 576 • SCA 设备 578 • 开发自定义分析模式 589
40.5 在安装过程中收集信息 590

40.6 内核模块支持 590
技术背景 591 • 使用不支持的模块 591
40.7 更多信息 592
41 常见问题及其解决方案 594
41.1 查找和收集信息 594
41.2 安装问题 597
检查媒体 597 • 没有可用于引导的 DVD 驱动器 598 • 从安装媒体引导失败 598 • 无法引导 600 • 无法启动图形安装程序 602 • 只启动简陋的引导屏幕 603 • 日志文件 604
41.3 引导问题 604
GRUB 2 引导加载程序无法装载 604 • 无图形登录 604 • 无法装入 Btrfs 根分区 605 • 强制检查根分区 605
41.4 登录问题 605
有效的用户名和口令组合失败 606 • 有效的用户名和口令不被接受 607 • 登录至加密的主分区失败 609 • 登录成功但 GNOME 桌面发生故障 609
41.5 网络问题 610
NetworkManager 问题 614
41.6 数据问题 614
管理分区映像 614 • 使用救援系统 615
41.7 IBM Z：将 initrd 用作救援系统 621
A 网络示例 623
B GNU Licenses 624
B.1 GNU Free Documentation License 624

关于本指南

本指南的目标用户为专业网络管理员和系统管理员，供其在操作 SUSE® Linux Enterprise 的过程中使用。因此，本指南的重点只在于确保 SUSE Linux Enterprise 的配置正确，并且网络上的所需服务都可使用，使其在初始安装后即可正常工作。本指南不包含用于确保 SUSE Linux Enterprise 与用户企业的应用程序软件兼容或者其核心功能符合那些要求的过程。本指南假定已经进行了全面的要求审核，已经请求进行安装或者已经请求进行用于此类审核的测试安装。

本指南包含如下内容：

支持任务和常见任务

SUSE Linux Enterprise 提供了大量工具，用于自定义系统的各个方面。本部分介绍其中几个。一个可用设备技术、高可用性配置及高级管理功能的明细表向管理员介绍了该系统。

系统

通过研究本部分了解关于底层操作系统的更多信息。SUSE Linux Enterprise 支持若干种硬件体系结构，您可以参考此信息来调整自己的应用程序，以便在 SUSE Linux Enterprise 上运行。引导加载程序和引导过程信息有助于您了解 Linux 系统的工作方式以及您自己的自定义脚本和应用程序与该系统的调和方式。

服务

SUSE Linux Enterprise 被设计为一个网络操作系统。它提供各种各样的网络服务，例如 DNS、DHCP、Web、代理和身份验证服务。它还可以很好地集成到异构环境中，其中包括 MS Windows 客户端和服务器。

移动计算机

需要特别注意便携式计算机、移动设备（如 PDA 或手机）和 SUSE Linux Enterprise 之间的通讯。要注意省电以及将不同设备集成到不断变化的网络环境中。同时要了解提供所需功能的后台技术。

查错

概述了当您需要更多信息或要执行特定任务时，如何查找帮助和其他文档。还提供了最常见的问题及其解决方法的列表。

1 可用文档

注意：在线文档和最新更新

我们的产品文档可从 https://documentation.suse.com/ 获取，您也可以在此处找到最新更新，以及浏览或下载各种格式的文档。

此外，您安装的系统的 `/usr/share/doc/manual` 下通常会提供产品文档。

针对本产品提供的文档如下：

《安装快速入门》文章

列出系统要求，并指导您从 DVD 或 ISO 映像逐步安装 SUSE Linux Enterprise Server。

《部署指南》

显示如何安装单个或多个系统，以及如何利用产品继承功能建立部署基础结构。有各种方法可供选择，可以选择使用本地安装或网络安装服务器，也可以选择使用远程控制、高度自定义的自动安装技术进行大规模部署。

管理指南

讲述系统管理任务，如维护、监视和自定义初始安装的系统。

《Virtualization Guide》

概述虚拟化技术，并介绍虚拟化的统一接口 libvirt，以及有关特定超级管理程序的详细信息。

《储存管理指南》

提供有关如何在 SUSE Linux Enterprise Server 上管理储存设备的信息。

《AutoYaST》

AutoYaST 系统使用包含安装和配置数据的 AutoYaST 配置文件，让您以无人照管方式批量部署 SUSE Linux Enterprise Server 系统。该手册将引导您完成自动安装的基本步骤，包括准备、安装和配置。

《Security Guide》

介绍系统安全的基本概念，包括本地安全方面和网络安全方面。说明如何使用产品固有的安全软件（例如 AppArmor），或者能够可靠收集有关任何安全相关事件的信息的审核系统。

《Security and Hardening Guide》

处理安装和设置安全 SUSE Linux Enterprise Server 的特定事项以及进一步确保和强化安装所需的额外安装后步骤。支持管理员选择与安全相关的选项并做出决策。

《System Analysis and Tuning Guide》

关于问题检测、解决和优化的管理员指南。了解如何使用监视工具检查和优化系统以及如何有效管理资源。还包含常见问题和解决方法的概述以及其他帮助和文档资源。

《Subscription Management Tool for SLES 12 SP5》

订阅管理工具管理员指南。订阅管理工具是用于 SUSE Customer Center 并包含储存库和注册目标的代理系统。了解如何安装和配置本地 SMT 服务器、镜像和管理储存库、管理客户端计算机，以及配置客户端以使用 SMT。

《GNOME 用户指南》

介绍 SUSE Linux Enterprise Server 的 GNOME 桌面。指导您使用和配置桌面并帮助您执行关键任务。它主要面向想要有效使用 GNOME 作为其默认桌面的最终用户。

2 反馈

提供了多种反馈渠道：

错误和增强请求

有关产品可用的服务和支持选项，请参见 http://www.suse.com/support/。

有关 openSUSE 的帮助由社区提供。有关更多信息，请参考https://en.opensuse.org/Portal:Support。

要报告产品组件的 Bug，请访问 https://scc.suse.com/support/requests 并登录，然后单击新建。

用户意见

我们希望收到您对本手册和本产品中包含的其他文档的意见和建议。请使用每个标题旁的“报告问题”链接通过 SUSE Bugzilla 提供反馈。

邮件

如有对本产品文档的反馈，也可以发送邮件至 `doc-team@suse.com`。请确保反馈中含有文档标题、产品版本和文档发布日期。要报告错误或给出增强建议，请提供问题的简要说明并指出相应章节编号和页码（或 URL）。

3 文档约定

本文档中使用了以下通知和排版约定：

- `/etc/passwd`：目录名称和文件名
- `PLACEHOLDER`：`PLACEHOLDER` 将会替换为实际的值
- `PATH`：环境变量 PATH
- `ls`、`--help`：命令、选项和参数
- `user`：用户和组
- `package name`：包名称
- `Alt`、`Alt`-`F1`：按键或组合键；这些键以大写形式显示，如在键盘上一样
- 文件、文件 › 另存为：菜单项，按钮
- `x86_64` 本段内容仅与 AMD64/Intel 64 体系结构相关。箭头标记文本块的开始位置和结束位置。

  `System z, POWER` 本段内容仅与 IBM 体系结构 `Z` 和 `POWER` 相关。箭头标记文本块的开始位置和结束位置。
- 跳舞的企鹅（企鹅一章，↑其他手册）：此内容参见自其他手册中的一章。
- 必须使用 `root` 特权运行的命令。您往往还可以在这些命令前加上 `sudo` 命令，以非特权用户身份来运行它们。

  ```
  root # command
  tux > sudo command
  ```
- 可以由非特权用户运行的命令。

  ```
  tux > command
  ```
- 注意

警告：警告通知

在继续操作之前，您必须了解的不可或缺的信息。向您指出有关安全问题、潜在数据丢失、硬件损害或物理危害的警告。

重要：重要通知

在继续操作之前，您必须了解的重要信息。

注意：注意通知

额外信息，例如有关软件版本差异的信息。

提示：提示通知

有用信息，例如指导方针或实用性建议。

4 关于本文档的制作

本文档是以 GeekoDoc (https://github.com/openSUSE/geekodoc) （DocBook 5 (http://www.docbook.org) 的子集）编写的。XML 源文件用 `jing`（请参见 https://code.google.com/p/jing-trang/）加以验证、用 `xsltproc` 进行处理，并用 Norman Walsh 的样式表的自定义版本转换为 XSL-FO。最终的 PDF 通过 Apache Software Foundation (https://xmlgraphics.apache.org/fop) 的 FOP 排版。用于制作本文档的开放源代码工具和环境由 DocBook Authoring and Publishing Suite (DAPS) 提供。项目的主页可以在 https://github.com/openSUSE/daps 中找到。

本文档的 XML 源代码可以在 https://github.com/SUSE/doc-sle 中找到。

I 常用任务

1 Bash 和 Bash 脚本 2

2 sudo 16

3 YaST 联机更新 26

4 YaST 31

5 文本方式的 YaST 33

6 使用命令行工具管理软件 39

7 通过 Snapper 进行系统恢复和快照管理 65

8 使用 VNC 远程访问 103

9 使用 RSync 复制文件 119

1 Bash 和 Bash 脚本

现今，许多人都在使用装有 GNOME 之类图形用户界面 (GUI) 的计算机。尽管它们提供了大量功能，但使用它们执行自动任务时，还是会有限制。外壳是对 GUI 的很好补充，本章提供关于外壳（在本例中为 Bash）的某些方面的概述。

1.1 什么是“外壳”？

通常来说，外壳就是指 Bash（Bourne again 外壳）。在本章中提到“外壳”时，指的是 Bash。事实上，除了 Bash 还存在很多其他外壳（ash、csh、ksh、zsh 等），每种外壳都具备不同的功能和特征。如果需要关于其他外壳的更多信息，请在 YaST 中搜索外壳。

1.1.1 了解 Bash 配置文件

外壳可调用为：

1. 交互式登录外壳： 当登录计算机时需要使用此方式，即使用 `--login` 选项调用 Bash 或通过 SSH 登录到远程计算机时。
2. “普通”交互式外壳： 这通常在启动 xterm、konsole、gnome 终端或类似工具时使用。
3. 非交互式外壳： 当在命令行调用外壳脚本时使用。

根据所用外壳的类型，会读取不同的配置文件。下表显示登录和非登录外壳的配置文件。

表 1.1 登录外壳的 BASH 配置文件

文件	描述
`/etc/profile`	不要修改此文件，否则在下一次更新时可能损坏您的修改！
`/etc/profile.local`	如果扩展 `/etc/profile`，请使用此文件
`/etc/profile.d/`	包含特定程序的系统范围配置文件

文件	描述
`~/.profile`	在此处插入特定于用户的登录外壳配置

请注意，登录外壳还会获取表 1.2 “非登录外壳的 Bash 配置文件”中所列的配置文件。

表 1.2 非登录外壳的 BASH 配置文件

`/etc/bash.bashrc`	不要修改此文件，否则在下一次更新时可能损坏您的修改！
`/etc/bash.bashrc.local`	使用此文件插入系统范围的修改（仅 Bash）
`~/.bashrc`	在此处插入特定于用户的配置

此外，Bash 还使用更多文件：

表 1.3 用于 BASH 的特殊文件

文件	描述
`~/.bash_history`	包含已键入的所有命令的列表
`~/.bash_logout`	注销时执行
`~/.别名`	用户为常用命令定义的别名。有关如何定义别名的详细信息，请参见 `man 1 alias`。

1.1.2 目录结构

下表简要介绍 Linux 系统上最重要的较高级别目录。以下列表中是关于这些目录和重要子目录的更多详细信息。

表 1.4 标准目录树概述

目录	内容
`/`	根目录 — 目录树的起点。

目录	内容
/bin	基本二进制文件，例如系统管理员和普通用户都需要的命令。通常还包含外壳，如 Bash。
/boot	引导加载程序的静态文件。
/dev	访问特定于主机的设备所需的文件。
/etc	特定于主机的系统配置文件。
/home	储存系统上具有帐户的所有用户的用户主目录。但是，root 的主目录不在 /home 中，而是在 /root 中。
/lib	基本共享库和内核模块。
/media	可卸媒体的安装点。
/mnt	临时装入文件系统的安装点。
/opt	附加应用程序软件包。
/root	超级用户 root 的主目录。
/sbin	基本系统二进制文件。
/srv	系统提供的服务的数据。
/tmp	临时文件。
/usr	具有只读数据的辅助层次结构。
/var	变量数据，如日志文件。
/windows	只在系统上同时安装了 Microsoft Windows* 和 Linux 时可用。包含 Windows 数据。

以下列表提供有关这些目录中有哪些文件和子目录的更多详细信息，并给出一些示例：

`/bin`

包含 `root` 和其他用户都可使用的基本 shell 命令。这些命令包括 `ls`、`mkdir`、`cp`、`mv`、`rm` 和 `rmdir`。`/bin` 也包含 Bash，它是 SUSE Linux Enterprise Server 中的默认外壳。

`/boot`

包含引导所需的数据，如引导加载程序、内核以及内核开始执行用户模式程序之前使用的其他数据。

`/dev`

储存代表硬件组件的设备文件。

`/etc`

包含控制诸如 X Window 系统等程序操作的本地配置文件。`/etc/init.d` 子目录包含引导过程中可执行的 LSB init 脚本。

`/home/USERNAME`

储存在系统中建立帐户的所有用户的私人数据。这里的文件只能由其拥有者或系统管理员修改。默认情况下，电子邮件目录和个人桌面配置以隐藏文件和目录的形式储存在此处，例如 `.gconf/` 和 `.config`。

注意：网络环境中的用户主目录

如果在网络环境中工作，则您的用户主目录可能映射到文件系统中除 `/home` 之外的其他目录中。

`/lib`

包含引导系统和运行 root 文件系统中的命令所需的基本共享库。共享库相当于 Windows 中的 DLL 文件。

`/media`

包含 CD-ROM、闪存盘和数码相机（如果它们使用 USB）等可卸媒体的安装点。`/media` 通常包含除系统硬盘之外的各类驱动器。可卸媒体插入或连接到系统并装入之后，您便可从此处访问该媒体。

`/mnt`

此目录提供临时装入的文件系统的安装点。`root` 可以在此处装入文件系统。

`/opt`

保留用于安装第三方软件。在此处可以找到可选软件和较大附加产品程序包。

`/root`

`root` 用户的主目录。`root` 的个人数据储存在此处。

`/run`

`systemd` 和各个组件使用的 tmpfs 目录。`/var/run` 是 `/run` 的符号链接。

`/sbin`

如 `s` 所表明的，该目录储存超级用户的实用程序。`/sbin` 包含 `/bin` 中的二进制文件以及引导和恢复系统所需的其他二进制文件。

`/srv`

储存系统提供的服务（如 FTP 和 HTTP）的数据。

`/tmp`

此目录由需要临时储存文件的程序使用。

重要：在引导时清理 /tmp

系统重引导后，之前储存在 `/tmp` 中的数据无法保证仍然存在。这视情况而定，例如，取决于 `/etc/tmpfiles.d/tmp.conf` 中所做的设置。

`/usr`

`/usr` 与用户无关，而是 Unix 系统资源 (Unix system resource) 的缩写。`/usr` 中的数据是可以在符合 文件系统层次标准 (FHS) 的各个主机之间共享的静态只读数据。此目录包含所有应用程序（包括 GNOME 等图形桌面），并且会在文件系统中创建次要层次。`/usr` 储存了多个子目录，例如 `/usr/bin`、`/usr/sbin`、`/usr/local` 和 `/usr/share/doc`。

`/usr/bin`

包含一般可访问的程序。

`/usr/sbin`

包含为系统管理员保留的程序，例如维修功能。

`/usr/local`

在此目录中，系统管理员可以安装本地的独立于分发包的扩展。

`/usr/share/doc`

储存系统的各种文档文件和发行描述。在 `manual` 子目录中可以找到此手册的联机版本。如果安装了多种语言，则此目录可能包含这些手册不同语言的版本。

在 `packages` 下可以找到系统上安装的软件包中包含的文档。对于每个包，都会创建一个子目录 `/usr/share/doc/packages/PACKAGENAME`，通常用其储存该包的自述文件，有时储存示例、配置文件或附加脚本。

如果系统上安装了操作指南，`/usr/share/doc` 中还会包含 `howto` 子目录，该目录中提供与 Linux 软件安装和操作相关的许多任务的附加文档。

`/var`

`/usr` 用于储存静态只读的数据，而 `/var` 用于在系统操作期间写入并成为变量数据的数据，例如日志文件或假脱机数据。有关最重要日志文件的概述，可以在 `/var/log/` 下找到，请参见见表 41.1 “日志文件”。

1.2 编写外壳脚本

使用外壳脚本可以方便地完成各种任务：收集数据、在文本中搜索单词或短语，以及执行其他有用的操作。以下示例显示用于打印文本的小外壳脚本：

例 1.1 用于打印文本的外壳脚本

```
#!/bin/sh ❶
# Output the following line: ❷
echo "Hello World" ❸
```

❶ 第一行开头是 Shebang 字符 (`#!`)，它表示此文件是一个脚本。该脚本使用 Shebang 后的指定解释程序执行，在本示例中为 `/bin/sh`。

❷ 第二行是一个以哈希符号开头的注释。建议对较难理解的行进行注释以记住它们的作用。

❸ 第三行使用内置命令 `echo` 打印相应文本。

可以运行该脚本之前，需要一些先决条件：

1. 每个脚本都应包含 Shebang 行（如上面的示例所示）。如果该行缺失，您需要手动调用解释器。
2. 可以将该脚本保存在任何位置。但是，建议将其保存在外壳可以找到的目录中。外壳中的搜索路径由环境变量 `PATH` 确定。通常，一般用户不具有对 `/usr/bin` 的写权限。因此，建议将脚本保存在用户目录 `~/bin/` 中。在上例中使用名称 `hello.sh`。
3. 该脚本需要可执行权限。使用以下命令设置权限：

```
chmod +x ~/bin/hello.sh
```

如果已满足上述所有先决条件，则可以按如下方式执行此脚本：

1. 作为绝对路径： 可以使用绝对路径执行脚本。在本例中为 `~/bin/hello.sh`。
2. 所有位置： 如果 `PATH` 环境变量包含脚本所在目录，则可以使用 `hello.sh` 来执行该脚本。

1.3 重定向命令事件

每个命令都可以使用三个通道输入或输出：

- 标准输出： 这是默认的输出通道。在命令打印某些内容时都会使用标准输出通道。
- 标准输入： 如果一个命令需要用户或其他命令输入，则使用此通道。
- 标准错误： 命令使用此通道报告错误。

要重定向这些通道，有以下可行的操作方式：

`命令 > 文件`

将该命令的输出保存为文件，将删除现有文件。例如，`ls` 命令会将其输出写入文件 `listing.txt`：

```
ls > listing.txt
```

`命令 >> 文件`

将命令输出追加到文件。例如，`ls` 命令会将其输出追加到文件 `listing.txt`：

```
ls >> listing.txt
```

`命令 < 文件`

读取该文件作为给定命令的输入。例如，`read` 命令会将此文件的内容读入变量：

```
read a < foo
```

`命令 1 | 命令 2`

将左侧命令的输出重定向为右侧命令的输入。例如，`cat` 命令会输出文件 `/proc/cpuinfo` 的内容。`grep` 使用此输出仅过滤出包含 `cpu` 的行：

```
cat /proc/cpuinfo | grep cpu
```

每个通道都有一个文件描述符：0（零）表示标准输入，1 表示标准输出，2 表示标准错误。允许在 `<` 或 `>` 字符前插入此文件描述符。例如，以下行搜索以 `foo` 开头的文件，但通过将错误重定向到 `/dev/null` 而抑制错误。

```
find / -name "foo*" 2>/dev/null
```

1.4 使用别名

别名是一个或多个命令的快捷方式定义。别名的语法为：

```
alias NAME=DEFINITION
```

例如，下行定义了一个别名 `lt`，它会输出一份较长的列表（选项 `-l`），将其按修改时间排序（`-t`），并按排好序的倒序打印（`-r`）：

```
alias lt='ls -ltr'
```

要查看所有别名定义，请使用 `alias`。使用 `unalias` 和相应的别名删除别名。

1.5 在 Bash 中使用变量

外壳变量可以是全局变量，也可以是局部变量。全局变量（或环境变量）可以在所有外壳中访问。而局部变量仅在当前外壳中可见。

要查看所有环境变量，请使用 `printenv` 命令。如果需要知道变量的值，请将变量名称作为自变量插入：

```
printenv PATH
```

也可以使用 `echo` 查看变量（无论是全局或本地变量）：

```
echo $PATH
```

要设置局部变量，请使用变量名后加等号和值：

```
PROJECT="SLED"
```

不要在等号两边插入空格，否则会出错。要设置环境变量，请使用 `export`：

```
export NAME="tux"
```

要删除变量，请使用 `unset`：

```
unset NAME
```

下表包含外壳脚本中可以使用的一些常见环境变量：

表 1.5 有用的环境变量

HOME	当前用户的用户主目录
HOST	当前主机名
LANG	当一个工具本地化后，它使用此环境变量中的语言。英语也可以设置为 C
PATH	外壳的搜索路径，冒号分隔的目录列表
PS1	指定在每个命令前打印的普通提示符

PS2	指定在执行多行命令时打印的辅助提示符
PWD	当前工作目录
USER	当前用户

1.5.1 使用自变量

例如，如果具有脚本 `foo.sh`，则可以如下执行：

```
foo.sh "Tux Penguin" 2000
```

要访问传递给脚本的所有自变量，您需要定位参数。`$1` 表示第一个自变量，`$2` 表示第二个自变量，依此类推。至多可以有九个参数。要获取脚本名称，请使用 `$0`。

以下脚本 `foo.sh` 打印从 1 到 4 的所有自变量：

```
#!/bin/sh
echo \"$1\" \"$2\" \"$3\" \"$4\"
```

如果使用上述自变量执行此脚本，将获取：

```
"Tux Penguin" "2000" "" ""
```

1.5.2 使用变量替换

变量替换将一个模式应用于变量的内容（从左侧或从右侧）。以下列表包含可能的语法格式：

`${VAR#pattern}`

从左侧删除可能的最短匹配：

```
file=/home/tux/book/book.tar.bz2
echo ${file#*/}
home/tux/book/book.tar.bz2
```

`${VAR##pattern}`

从左侧删除可能的最长匹配：

```
file=/home/tux/book/book.tar.bz2
echo ${file##*/}
book.tar.bz2
```

`${VAR%pattern}`

从右侧删除可能的最短匹配：

```
file=/home/tux/book/book.tar.bz2
echo ${file%.*}
/home/tux/book/book.tar
```

`${VAR%%pattern}`

从右侧删除可能的最长匹配：

```
file=/home/tux/book/book.tar.bz2
echo ${file%%.*}
/home/tux/book/book
```

`${VAR/pattern_1/pattern_2}`

将来自 `PATTERN_1` 的 `VAR` 的内容替代为 `PATTERN_2`：

```
file=/home/tux/book/book.tar.bz2
echo ${file/tux/wilber}
/home/wilber/book/book.tar.bz2
```

1.6 将命令分组和组合

外壳允许您对命令执行连接和分组以有条件地执行。每个命令都返回一个退出码，该退出码确定操作是成功还是失败。如果是 0，则命令成功，任何其他值都表示特定于该命令的一个错误。

以下列表显示可以如何将命令分组：

`命令 1 ; 命令 2`

顺序地执行这些命令。不检查退出码。以下行使用 `cat` 显示文件的内容，然后使用 `ls` 打印其文件属性，而不考虑退出码：

```
cat filelist.txt ; ls -l filelist.txt
```

`命令 1 && 命令 2`

如果左侧命令成功，则运行右侧命令（逻辑运算符 AND）。仅当上一个命令成功时，以下行才显示文件的内容并打印其文件属性（将其与列表中的上一项相比较）：

```
cat filelist.txt && ls -l filelist.txt
```

`命令 1 || 命令 2`

当左侧命令失败时运行右侧命令（逻辑运算符 OR）。以下行仅当在 `/home/tux/foo` 中创建目录失败时才会在 `/home/wilber/bar` 中创建目录：

```
mkdir /home/tux/foo || mkdir /home/wilber/bar
```

`funcname(){ ... }`

创建外壳函数。您可以使用定位参数访问其自变量。以下行定义用于打印短消息的函数 `hello`：

```
hello() { echo "Hello $1"; }
```

您可以如下调用此函数：

```
hello Tux
```

它会打印：

```
Hello Tux
```

1.7 使用通用流程构造语句

为了控制脚本的流程，外壳有 `while`、`if`、`for` 和 `case` 等构造语句。

1.7.1 if 控制命令

`if` 命令用于检查表达式。例如，以下代码测试当前用户是否是 Tux：

```
if test $USER = "tux"; then
  echo "Hello Tux."
```

```
else
  echo "You are not Tux."
fi
```

测试表达式既可以复杂也可以简单。以下表达式检查文件 `foo.txt` 是否存在：

```
if test -e /tmp/foo.txt ; then
  echo "Found foo.txt"
fi
```

测试表达式也可以缩写在角括号中：

```
if [ -e /tmp/foo.txt ] ; then
  echo "Found foo.txt"
fi
```

在 http://www.cyberciti.biz/nixcraft/linux/docs/uniqlinuxfeatures/lsst/ch03sec02.html 上可以找到更多有用表达式。

1.7.2 使用 `for` 命令创建循环

`for` 循环允许您对一系列项执行命令。例如，以下代码打印关于当前工作目录中 PNG 文件的某些信息：

```
for i in *.png; do
 ls -l $i
done
```

1.8 更多信息

关于 Bash 的重要信息在手册页 `man bash` 中提供。可以在以下列表中找到关于此主题的更多信息：

- http://tldp.org/LDP/Bash-Beginners-Guide/html/index.html — Bash 入门者指南
- http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html — BASH 编程 - 简介操作指南

- http://tldp.org/LDP/abs/html/index.html — 高级 Bash 脚本编写指南
- http://www.grymoire.com/Unix/Sh.html — Sh - Bourne 外壳

2 sudo

许多命令和系统实用程序都需要以 `root` 身份运行才能修改文件和/或执行只有超级用户方能执行的任务。为了确保安全和避免发生意外运行危险命令的情况，通常建议不要直接以 `root` 身份登录。建议的做法是以非特权的普通用户身份工作，并使用 `sudo` 命令来运行需要较高特权的命令。

在 SUSE Linux Enterprise Server 上，`sudo` 默认配置为与 su 的工作方式类似。但是，`sudo` 可让用户以高度可配置的方式使用任何其他用户的特权来运行命令。这样，便可为某些用户和组指派具有特定特权的角色。举例来说，可以允许组 `users` 的成员使用 `wilber` 的特权运行命令。例如，通过禁止指定任何命令选项，可以进一步限制对命令的权限。虽然 su 始终需要 `root` 口令才能使用 PAM 进行身份验证，但是您可以将 `sudo` 配置为使用您自己的身份凭证进行身份验证。这样就不需要共享 `root` 口令，从而提高了安全性。例如，您可以允许 `users` 组的成员以 `wilber` 身份运行 `frobnicate` 命令，但限制其不能指定自变量。这样，便可为某些用户和组指派具有特定能力的角色。

2.1 sudo 基本用法

虽然 `sudo` 简单易用，功能却十分强大。

2.1.1 运行单个命令

以普通用户身份登录后，您可以在命令前加上 `sudo` 以 `root` 身份运行任何命令。按照提示输入 root 口令后，如果身份验证成功，您便能以 `root` 身份运行命令：

```
tux > id -un❶
tux
tux > sudo id -un
root's password:❷
root
tux > id -un
tux❸
tux > sudo id -un
❹
```

```
root
```

1. `id -un` 命令会打印当前用户的登录名。
2. 在输入过程中不会显示口令，无论是明文还是密文均不显示。
3. 只有以 `sudo` 开头的命令才会使用较高的特权运行。如果是不带 `sudo` 前缀的相同命令，仍会使用当前用户的特权运行。
4. 在限定时间内，您无需再次输入 `root` 口令。

提示：I/O 重定向

I/O 重定向的工作方式与您预期的可能不同：

```
tux > sudo echo s > /proc/sysrq-trigger
bash: /proc/sysrq-trigger: Permission denied
tux > sudo cat < /proc/1/maps
bash: /proc/1/maps: Permission denied
```

只有 `echo`/`cat` 二进制会使用较高特权运行，重定向则由用户外壳使用用户特权执行。您可以按第 2.1.2 节 “启动外壳”中所述启动外壳，也可以使用 `dd` 实用程序来启动：

```
echo s | sudo dd of=/proc/sysrq-trigger
sudo dd if=/proc/1/maps | cat
```

2.1.2 启动外壳

必须在每条命令前加上 `sudo` 可能很繁琐。虽然可以将外壳指定为命令 `sudo bash`，但还是建议您使用以下其中一种内置机制来启动外壳：

`sudo -s (<命令>)`

启动 `SHELL` 环境变量所指定的外壳或目标用户的默认外壳。如果给定了命令，则会将该命令传递给外壳（使用 `-c` 选项），否则外壳会以交互模式运行。

```
tux:~ > sudo -i
root's password:
```

```
root:/home/tux # exit
tux:~ >
```

`sudo -i (<命令>)`

与 `-s` 类似，但是会将外壳启动为登录外壳。也就是说，系统会对外壳的启动文件（`.profile` 等）进行处理，并会将当前的工作目录设置为目标用户的主目录。

```
tux:~ > sudo -i
root's password:
root:~ # exit
tux:~ >
```

2.1.3 环境变量

默认情况下，`sudo` 不会传播环境变量：

```
tux > ENVVAR=test env | grep ENVVAR
ENVVAR=test
tux > ENVVAR=test sudo env | grep ENVVAR
root's password:
❶
tux >
```

❶ 输出为空即表明在使用 `sudo` 运行的命令的环境中不存在环境变量 `ENVVAR`。

此行为可通过 `env_reset` 选项进行更改，请参见表 2.1 “有用的标志和选项”。

2.2 配置 sudo

`sudo` 是一个非常灵活的工具，提供各种配置选项。

注意：无法使用 sudo

如果您不小心将自己锁定在 `sudo` 之外，则可以使用 `su -` 及 `root` 口令来获取 root 外壳。要修复该错误，请运行 `visudo`。

2.2.1 编辑配置文件

`sudo` 的主要策略配置文件为 `/etc/sudoers`。如果此文件中存在错误，您可能便会无法进入系统，因此强烈建议您使用 `visudo` 来编辑配置文件。此举可防止同时更改打开的文件，并会在保存修改之前检查语法错误。

您还可以通过设置 `EDITOR` 环境变量来使用除 vi 以外的编辑器（不论名字如何），例如：

```
sudo EDITOR=/usr/bin/nano visudo
```

不过，`/etc/sudoers` 文件本身是由系统包提供的，更新时这些修改可能会取消。因此，建议您将自定义配置放到 `/etc/sudoers.d/` 目录下的文件中。该目录下的任何文件都会自动纳入系统中。要在该子目录下创建或编辑文件，请运行：

```
sudo visudo -f /etc/sudoers.d/NAME
```

或者，使用其他编辑器（例如 `nano`）：

```
sudo EDITOR=/usr/bin/nano visudo -f /etc/sudoers.d/NAME
```

注意：/etc/sudoers.d 中忽略的文件

`/etc/sudoers` 中的 `#includedir` 命令（用于 `/etc/sudoers.d`）会忽略以 `~`（波浪号）结尾或包含 `.`（点）的文件。

关于 `visudo` 命令的详细信息，请运行 `man 8 visudo`。

2.2.2 sudoers 基本配置语法

在 sudoers 配置文件中，有两种类型的选项：字符串和标志。字符串可以包含任何值，而标志则只能在“ON”或“OFF”之间切换。sudoers 配置文件最重要的语法构造为：

```
# Everything on a line after a # gets ignored ❶
Defaults !insults # Disable the insults flag ❷
Defaults env_keep += "DISPLAY HOME" # Add DISPLAY and HOME to env_keep
tux ALL = NOPASSWD: /usr/bin/frobnicate, PASSWD: /usr/bin/journalctl ❸
```

❶ `#include` 和 `#includedir` 这两个普通命令例外。其后跟数字，用于指定 UID。

❷ 去除 `!` 可将指定的标志设置为“ON”。

❸ 请参见第 2.2.3 节 “sudoers 中的规则”。

表 2.1 有用的标志和选项

选项名称	说明	示例
`targetpw`	此标志控制调用用户是需要输入目标用户（例如 `root`）的口令 (ON) 还是需要输入调用用户的口令 (OFF)。	`Defaults targetpw # Turn targetpw flag ON`
`rootpw`	如果设置了该选项，`sudo` 将提示输入 `root` 口令，而非目标用户或调用命令的用户的口令。默认值为 OFF。	`Defaults !rootpw # Turn rootpw flag OFF`
`env_reset`	如果设置了该选项，`sudo` 会构造一个仅包含 `TERM`、`PATH`、`HOME`、`MAIL`、`SHELL`、`LOGNAME`、`USER`、`USERNAME` 和 `SUDO_*` 集的最小环境。此外，会从调用环境导入 `env_keep` 中列出的变量。默认值为“ON”。	`Defaults env_reset # Turn env_reset flag ON`

`env_keep`	`env_reset` 标志设为“ON”时要保留的环境变量列表。	`# Set env_keep to contain EDITOR and PROMPT`
		`Defaults env_keep = "EDITOR PROMPT"`
		`Defaults env_keep += "JRE_HOME" # Add JRE_HOME`

选项名称	说明	示例
		`Defaults env_keep -= "JRE_HOME" # Remove JRE_HOME`

`env_delete`	`env_reset` 标志设为“OFF”时要去除的环境变量列表。	`# Set env_delete to contain EDITOR and PROMPT`
		`Defaults env_delete = "EDITOR PROMPT"`
		`Defaults env_delete += "JRE_HOME" # Add JRE_HOME`
		`Defaults env_delete -= "JRE_HOME" # Remove JRE_HOME`

还可以使用 `Defaults` 令牌为用户、主机和命令集合创建别名。并且，可以仅将选项应用到特定用户集。

关于 `/etc/sudoers` 配置文件的详细信息，请参见 `man 5 sudoers`。

2.2.3 sudoers 中的规则

sudoers 配置中的规则可能会非常复杂，因此本节仅涉及基本内容。每个规则都遵循基本模式（`[]` 标记的是可选部分）：

```
#Who      Where          As whom      Tag                 What
User_List Host_List = [(User_List)] [NOPASSWD:|PASSWD:] Cmnd_List
```

SUDOERS 规则的语法

`User_List`

一个或多个（用 `,` 分隔）标识符：用户名、格式为 `%GROUPNAME` 的组或格式为 `#UID` 的用户 ID。可以使用 `!` 前缀来取反。

`Host_List`

一个或多个（用 `,` 分隔）标识符：（完全限定的）主机名或 IP 地址。可以使用 `!` 前缀来取反。`Host_List` 的惯常选项为 `ALL`。

`NOPASSWD:|PASSWD:`

如果用户在 `NOPASSWD:` 后面运行的命令与 `CMDSPEC` 匹配，系统不会提示用户输入口令。

`PASSWD` 为默认选项，仅当两个选项位于同一行时才需要指定它：

```
tux ALL = PASSWD: /usr/bin/foo, NOPASSWD: /usr/bin/bar
```

`Cmnd_List`

一个或多个（用 `,` 分隔）区分符：可执行文件的路径，后跟允许使用的自变量或什么也不跟。

```
/usr/bin/foo     # Anything allowed
/usr/bin/foo bar # Only "/usr/bin/foo bar" allowed
/usr/bin/foo ""  # No arguments allowed
```

`ALL` 可以用作 `User_List`、`Host_List` 和 `Cmnd_List`。

允许 `tux` 在无需输入口令的情况下以 root 身份运行所有命令的规则：

```
tux ALL = NOPASSWD: ALL
```

允许 `tux` 运行 `systemctl restart apache2` 的规则：

```
tux ALL = /usr/bin/systemctl restart apache2
```

允许 `tux` 在不带自变量的情况下以 `admin` 身份运行 `wall` 的规则：

```
tux ALL = (admin) /usr/bin/wall ""
```

警告：危险构造

以下类型的构造

```
ALL ALL = ALL
```

在没有 `Defaults targetpw` 的情况下切勿使用，否则任何人都能以 `root` 身份运行命令。

2.3 常见使用情况

尽管默认配置对于简单的设置和桌面环境通常已经够用，但是自定义配置非常有用。

2.3.1 在无需 root 口令的情况下使用 sudo

在具有特殊限制（“用户 X 只能以 `root` ” 身份运行命令 Y）的情况下，无法实现此目的。在其他情况下，还是建议进行某种分隔。按照惯例，组 `wheel` 的成员能以 root 身份运行所有带有 `sudo` 的命令。

1. 将自己添加到 `wheel` 组
   如果您自己的用户帐户尚不是 `wheel` 组的成员，请添加该帐户，具体做法是运行 `sudo usermod -a -G wheel 用户名` 然后注销并再次登录。运行 `groups 用户名` 以确认更改是否成功。

2. 将使用调用用户的口令进行身份验证的选项设为默认设置。
   使用 `visudo` 创建文件 `/etc/sudoers.d/userpw`（请参见第 2.2.1 节 “编辑配置文件”）并添加：

   ```
   Defaults !targetpw
   ```

3. 选择新默认规则。
   根据是否想要用户重新输入口令，取消对 `/etc/sudoers` 中特定行的注释，并将默认规则注释掉。

   ```
   ## Uncomment to allow members of group wheel to execute any command
   # %wheel ALL=(ALL) ALL

   ## Same thing without a password
   # %wheel ALL=(ALL) NOPASSWD: ALL
   ```

4. 提高默认规则的限制性

   将 `/etc/sudoers` 中允许一切操作的规则注释掉或去除：

```
ALL     ALL=(ALL) ALL   # WARNING! Only use this together with 'Defaults
 targetpw'!
```

警告：sudoers 中的危险规则

切勿漏掉这一步，否则任何用户都能以 `root` 身份执行任何命令。

5. 测试配置

   尝试以 `wheel` 的成员和非成员身份运行 `sudo`。

```
tux:~ > groups
users wheel
tux:~ > sudo id -un
tux's password:
root
wilber:~ > groups
users
wilber:~ > sudo id -un
wilber is not in the sudoers file.  This incident will be reported.
```

2.3.2 对 X.Org 应用程序使用 sudo

在使用 `sudo` 启动图形应用程序时，可能会出现以下错误：

```
tux > sudo xterm
xterm: Xt error: Can't open display: %s
xterm: DISPLAY is not set
```

YaST 会选择 ncurses 界面而非图形界面。

要在通过 `sudo` 启动的应用程序中使用 X.Org，需要传播环境变量 `DISPLAY` 和 `XAUTHORITY`。要进行此项配置，请创建文件 `/etc/sudoers.d/xorg`（请参见第 2.2.1 节 “编辑配置文件”）并添加下面一行：

```
Defaults env_keep += "DISPLAY XAUTHORITY"
```

如尚未设置 `XAUTHORITY` 变量，请按如下方式设置：

```
export XAUTHORITY=~/.Xauthority
```

现在，X.Org 应用程序便可正常运行：

```
sudo yast2
```

2.4 更多信息

使用 `sudo --help` 可检索有关可用命令行开关的简要概述。如需说明和其他重要信息，请参见手册页：`man 8 sudo`，配置相关信息详见 `man 5 sudoers`。

3 YaST 联机更新

SUSE 持续为您的产品提供软件安全更新。默认情况下，用更新小程序来保持您的系统是最新的。有关更新小程序的更多信息请参考《部署指南》, 第 13 章 “安装或删除软件”, 第 13.5 节 “保持系统最新”。本章介绍用于更新软件包的备用工具：YaST 联机更新。

更新软件储存库提供了 SUSE® Linux Enterprise Server 的最新增补程序。如果安装时已注册您的产品，则更新安装源已配置。如果您尚未注册 SUSE Linux Enterprise Server，可通过在 YaST 中启动产品注册来完成注册。或者，可以从信任的源中手动添加更新安装源。要添加或去除储存库，请使用 YaST 中的软件 › 软件储存库来启动储存库管理器。请在《部署指南》, 第 13 章 “安装或删除软件”, 第 13.4 节 “管理软件储存库和服务”中了解更多有关储存库管理器的内容。

注意：访问更新编目时出错

如果您不能访问更新编目，可能是由于订阅已过期。通常，SUSE Linux Enterprise Server 会附带一年或三年的订阅，在此期间您可以访问更新编目。订阅结束后，将拒绝您访问更新编目。

如果访问更新编目时遭到拒绝，您将看到一条警告讯息，提示您访问 SUSE Customer Center 并查看您的订阅。可通过 https://scc.suse.com// 访问 SUSE Customer Center。

SUSE 提供了不同相关级别的更新：

安全性更新
: 修复严重的安全性危害，请务必安装。

推荐更新
: 修复可能危及计算机安全的问题。

可选更新
: 修复非安全性相关的问题或提供增强功能。

3.1 联机更新对话框

要打开 YaST 联机更新对话框，请启动 YaST 并选择软件 › 联机更新。也可以从命令行输入 `yast2 online_update` 来启动该对话框。

联机更新窗口由四部分组成。

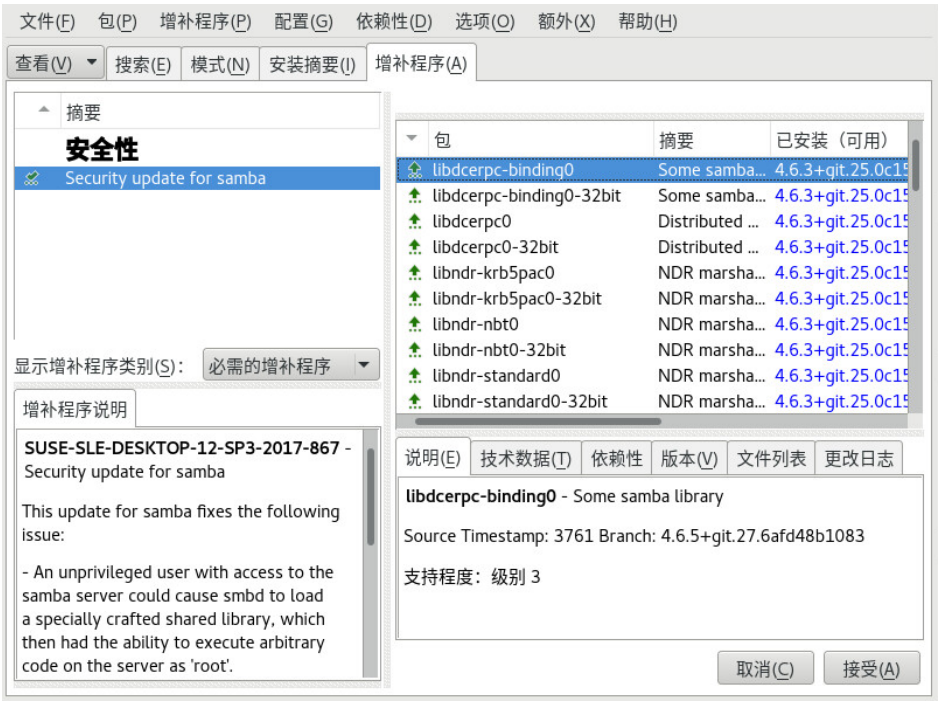


图 3.1 YAST 联机更新

左侧的摘要部分列出了 SUSE Linux Enterprise Server 可用的增补程序。增补程序按安全相关性（安全性、推荐和可选）排序。您可以从显示增补程序类别中选择以下某个选项来更改摘要部分的视图：

必需的增补程序（默认视图）

当前未安装的适用于系统上已安装的包的增补程序。

不需要的增补程序

适用于系统上未安装的包的增补程序，或要求已满足的增补程序（因为已从另一源对相关包进行了更新）。

所有增补程序

SUSE Linux Enterprise Server 的所有可用增补程序。

摘要部分的每个列表项都由符号和增补程序名称组成。如需了解可能符号及其含义的概述，请按 Shift-F1。 安全性 和 建议 增补程序需要的操作是自动预设置的。这些操作有自动安装、自动更新和自动删除。

如果从非更新储存库的某个储存库安装最新包，此安装可能满足此包的某个增补程序的要求。在这种情况下，在增补程序摘要前会显示一个复选标记。该增补程序将显示在列表中，直到将其标记用于安装。这实际上不会安装增补程序（因为该包已经是最新的），而是将该增补程序标记为已安装。

在摘要部分选择一个项，可在对话框左下角的增补程序描述中查看简短描述。右上部分列出所选增补程序中包含的包（一个增补程序可以由多个包组成）。单击右上部分中的项可以查看有关增补程序中包含的各个包的细节。

3.2 安装增补程序

在 YaST 的“联机更新”对话框中，您可以一次性安装所有可用的增补程序，也可以手动选择所需的增补程序。还可以还原已应用于系统的增补程序。

默认情况下，您的系统当前可用的所有新增补程序（ 可选 增补程序除外）都已标记为可安装。一旦您单击接受或应用，将自动应用它们。如果一个或多个增补程序需要重引导系统，在开始安装增补程序之前，系统会发出相关通知。此时，您可以选择继续安装所选增补程序、跳过需要重引导的所有增补程序的安装并安装剩余的增补程序，或者返回增补程序手动选择屏幕。

过程 3.1 使用 YAST 联机更新应用增补程序

1. 启动 YaST 并选择软件 › 联机更新。
2. 要自动应用您的系统当前可用的所有新增补程序（ 可选 增补程序除外），请按应用或接受。
3. 首先请修改要应用的增补程序选择：
    a. 使用界面中提供的相应过滤器和视图。有关详细信息，请参见第 3.1 节 “联机更新对话框”。
    b. 根据您的需要和喜好选择或取消选择增补程序，方法是右键单击增补程序并从上下文菜单中选择相应操作。

重要：始终应用安全性更新

除非很有必要，否则请不要取消选择任何 安全性 相关的增补程序。因为这些增补程序可修复严重的安全性危害，防止系统遭受攻击。

c. 多数增补程序包含几个包的更新。如果要更改单个包的操作，请右键单击包视图中的包，并选择一项操作.

d. 要确认您的选择并应用所选增补程序，请单击应用或接受以继续。

4. 安装完成后，单击完成退出 YaST 联机更新。您的系统现在已是最新的了。

3.3 自动联机更新

YaST 还提供设置每日、每周或每月自动更新的选项。要使用相应模块，需要先安装 `yast2-online-update-configuration` 包。

默认情况下，更新将以增量 RPM 的形式供您下载。由于基于增量 RPM 重构建 RPM 包需要占用大量内存和处理器资源，出于性能考虑，某些设置或硬件配置可能要求您禁用增量 RPM。

某些增补程序（如需要许可协议的内核更新或包）需要用户交互，这可能会导致自动更新过程停止。您可以配置为跳过需要用户交互的增补程序。

过程 3.2 配置自动联机更新

1. 安装后，启动 YaST 并选择软件 › 联机更新配置。
   也可以从命令行输入 `yast2 online_update_configuration` 来启动该模块。
2. 激活自动联机更新。
3. 选择更新间隔：每日、每周或每月。
4. 要自动接受任何许可协议，请激活同意许可证。
5. 如果希望更新过程完全自动执行，请选择是否要跳过交互式增补程序。

重要：跳过增补程序

如果选择跳过任何需要交互的包，请同样不定期运行手动联机更新以安装那些增补程序。否则可能会错过重要的增补程序。

6. 要自动安装更新包推荐的所有软件包，请激活包含推荐的软件包。
7. 要禁用增量 RPM（出于性能方面的考虑），请停用使用增量 RPM。
8. 要按照类别（例如安全性或推荐）过滤增补程序，请激活按类别过滤，并从列表中添加适当的增补程序类别。只会安装选中类别的增补程序，而将跳过其他类别的增补程序。
9. 单击确定确认您的配置。

之后，自动联机更新将不会自动重启动系统。如果有的包更新需要重引导系统，您需要手动重引导。

4 YaST

YaST 是 SUSE Linux Enterprise Server 的安装和配置工具。它具有图形界面，并且能够在安装期间和安装之后快速自定义系统。它可用于设置硬件、配置网络、系统服务和调整安全性设置。

4.1 高级组合键

YaST 具有一套高级组合键。

Print Screen

截图并保存。当 YaST 在某些桌面环境下运行时可能不可用。

Shift-F4

启用/禁用为视力受损的用户专门优化的配色。

Shift-F7

启用/禁用记录调试讯息。

Shift-F8

打开一个文件对话框，以将日志文件保存到非标准位置。

Ctrl-Shift-Alt-D

发送一个调试事件。YaST 模块可执行特殊的调试操作来对此作出反应。结果取决于具体的 YaST 模块。

Ctrl-Shift-Alt-M

启动/停止宏记录器。

Ctrl-Shift-Alt-P

重新播放宏。

Ctrl-Shift-Alt-S

显示样式表编辑器。

Ctrl-Shift-Alt-T

将控件树转储到日志文件。

Ctrl–Shift–Alt–X

打开一个终端窗口 (xterm)。当通过 VNC 安装时很有用。

Ctrl–Shift–Alt–Y

显示控件树浏览器。

5 文本方式的 YaST

本章所针对的读者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启动和操作 YaST 有关的基本信息。

文本模式的 YaST 使用 ncurses 库提供简单的伪图形用户界面。默认情况下已安装 ncurses 库。用于运行 YaST 的终端仿真器支持的最小大小为 80x25 个字符。

图 5.1 文本方式下 YAST 的主窗口

以文本模式启动 YaST 时，会显示 YaST 控制中心（请参见图 5.1）。该窗口包含三个区域。左侧方框中显示各种模块所属的类别。此方框在 YaST 启动后处于活动状态，因此以白色粗边框进行标记。活动类别处于选中状态。右侧方框提供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。

启动 YaST 控制中心时，会自动选择软件类别。使用 ↓ 和 ↑ 可更改类别。要从类别中选择某个模块，请使用 → 激活右侧方框，然后使用 ↓ 和 ↑ 选择该模块。按住箭头键在可用模块列表中滚动。选择模块后，按 Enter 将其启动。

模块中的各种按钮和选择字段包含一个高亮显示的字母（默认为黄色）。使用 Alt–highlighted_letter 可直接选择按钮，而无需使用 →| 键导航。要退出 YaST 控制中心，请按 Alt–Q，或者选择退出并按 Enter。

提示：刷新 YaST 对话框

如果 YaST 对话框损坏或变形（例如在调整窗口大小时），请按 Ctrl–L 来刷新并恢复其内容。

5.1 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 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 5.3 节 “组合键的限制”。

在按钮和选择列表中导航

使用 →| 键在按钮和包含选择列表的框架之间导航。要以相反顺序导航，请使用 Alt–→| 或 Shift–→| 组合键。

在选择列表中导航

使用方向键（↑ 和 ↓）可浏览包含选择列表的活动方框中的各个元素。如果方框内的项超出了方框宽度，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 → 或 ← 会导致更改活动方框或当前选择列表（如同在控制中心内），也可以使用此组合键。

按钮、单选项按钮和复选框

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。或者，可以使用 Alt–highlighted_letter 直接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 →| 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。

功能密钥

功能键（F1 ... F12）可让您快速访问各种按钮。YaST 屏幕底部的行中显示了可用的功能键组合（FX）。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（细节、信息、添加、删除等）。可以将 F10 用作接受、确定、下一步和完成。按 F1 可访问 YaST 帮助。

在 ncurses 方式中使用导航树

某些 YaST 模块使用窗口左侧的导航树选择配置对话框。使用方向键（ ↑ 和 ↓ ）可在树中导航。使用 Space 可打开或关闭树中的项。在 ncurses 模式下，在导航树中进行选择之后必须按 Enter 才能显示所选对话框。这是一种有意行为，目的是在浏览导航树时避免耗时的重绘。

在软件安装模块中选择软件

使用左侧的过滤器可以限制显示的包数。已安装的包标有字母 i。要更改包的状态，请按 Space 或 Enter。或者，也可以使用操作菜单选择所需的状态更改 (安装、删除、更新、禁止或锁定)。

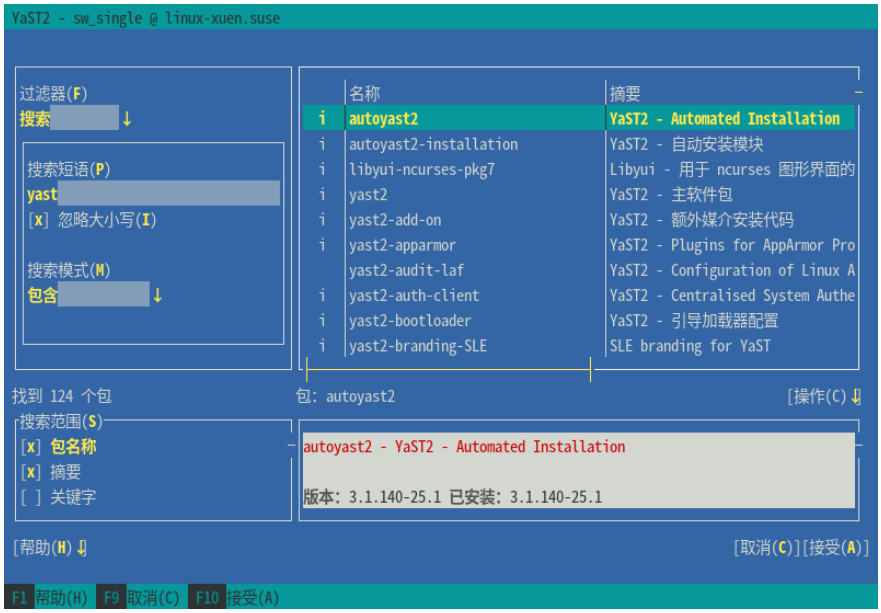


图 5.2 软件安装模块

5.2 高级组合键

文本模式的 YaST 具有一套高级组合键。

Shift-F1

显示高级热键的列表。

Shift-F4

更改颜色方案。

Ctrl-\

退出应用程序。

Ctrl-L

刷新屏幕。

Ctrl-D F1

显示高级热键的列表。

Ctrl-D Shift-D

以屏幕截图的形式将对话框转储到日志文件。

Ctrl-D Shift-Y

打开 YDialogSpy 以查看控件层次结构。

5.3 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 这样的键也可能会被终端设置占用。

使用 Esc 取代 Alt

可以代替 Alt 而使用 Esc Alt 快捷键。例如，Esc-H 可代替 Alt-H。（首先按 Esc，然后按 H 键。）

使用 Ctrl-F 和 Ctrl-B 执行向后和向前导航

如果 Alt 和 Shift 组合键由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 (F1 ... F12) 也用于执行多种功能。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。

5.4 YaST 命令行选项

除了文本模式界面以外，YaST 还提供了一个纯命令行界面。要获取 YaST 命令行选项列表，请输入：

```
yast -h
```

5.4.1 启动单个模块

为了节省时间，可以直接启动单个 YaST 模块。要启动模块，请输入：

```
yast <module_name>
```

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`。例如，要启动网络模块，请输入 `yast lan`。

5.4.2 从命令行安装包

如果知道包名称且包是由您的任何活动安装源提供的，则可以使用命令行选项 `-i` 安装该包：

```
yast -i <package_name>
```

或

```
yast --install <package_name>
```

`PACKAGE_NAME` 可以是通过依赖项检查安装的单个简短包名称（例如 `gvim`），也可以是并非通过依赖项检查安装的 RPM 包的完整路径。

如果需要具有 YaST 未提供的功能的，基于命令行的软件管理实用程序，请考虑使用 Zypper。此实用程序使用相同的软件管理库，这也是 YaST 包管理器的基础。第 6.1 节 “使用 Zypper”中介绍了 Zypper 的基本用法。

5.4.3 YaST 模块的命令行参数

为了在脚本中使用 YaST 功能，YaST 提供了对单个模块的命令行支持。并非所有模块都具有命令行支持。要显示某个模块的可用选项，请输入：

```
yast <module_name> help
```

如果模块不提供命令行支持，将以文本方式启动，并显示以下消息：

```
This YaST module does not support the command line interface.
```

6 使用命令行工具管理软件

本章描述 Zypper 和 RPM，这是两个用于管理软件的命令行工具。有关此环境中使用的术语定义（例如，*储存库*、*增补程序*或*更新*），请参见《部署指南》，第 13 章 “安装或删除软件”，第 13.1 节 “术语定义”。

6.1 使用 Zypper

Zypper 是一个命令行包管理器，用于安装、更新和去除包及管理储存库。这一点对于完成远程软件管理任务或从外壳脚本管理软件尤其有用。

6.1.1 一般使用

Zypper 的常用语法为：

```
zypper [--global-options] COMMAND  [--command-options] [arguments]
```

不需要括在括号中的组件。有关常规选项和所有命令的列表，请参见 `zypper help`。要获取有关特定命令的帮助，请键入 `zypper help` *命令*。

Zypper 命令

执行 Zypper 最简单的方式是，键入其名称后跟一个命令。例如，要将所有需要的增补程序应用于系统，请使用：

```
tux > sudo zypper patch
```

全局选项

此外，您还可以选择使用一个或多个全局选项，只需在命令前面键入它们即可：

```
tux > sudo zypper --non-interactive patch
```

在上面的示例中，选项 `--non-interactive` 表示在不询问任何问题的情况下运行命令（自动应用默认回答）。

命令特定的选项

要使用特定于某个命令的选项，请紧接在该命令后面键入这些选项：

```
tux > sudo zypper patch --auto-agree-with-licenses
```

在上面的示例中，`--auto-agree-with-licenses` 用于将所有需要的增补程序应用于系统，不要求您确认任何许可条款，而是自动接受许可条款。

自变量

某些命令需要一个或多个自变量。例如，使用 `install` 命令时，需要指定您要安装的一个或多个包：

```
tux > sudo zypper install mplayer
```

某些选项还需要单个自变量。用以下命令可列出所有已知模式：

```
tux > zypper search -t pattern
```

您可以组合上述所有模式。例如，下面的命令在冗长模式下运行时将安装 `aspell-de` 和 `aspell-fr` 包（来自 `factory` 储存库）：

```
tux > sudo zypper -v install --from factory aspell-de aspell-fr
```

`--from` 选项确保了在从指定储存库请求包时保留所有储存库的启用状态（用于解析任何依赖项）。

多数 Zypper 命令都有 `dry-run` 选项，它模拟给定的命令。它可用于测试。

```
tux > sudo zypper remove --dry-run MozillaFirefox
```

Zypper 支持 `--userdata 字符串` 全局选项。您可以使用此选项指定一个将会写入 Zypper 的日志文件和插件（例如 Btrfs 插件）的字符串。它可以用于标记和标识日志文件中的事务。

```
tux > sudo zypper --userdata STRING patch
```

6.1.2 使用 Zypper 安装和删除软件

要安装或去除包，请使用以下命令：

```
tux > sudo zypper install PACKAGE_NAME
sudo zypper remove PACKAGE_NAME
```

警告：不要去除必需的系统包

不要去除必需的系统包，例如 `glibc` 、 `zypper` 、 `kernel` 。如果去除这些包，系统可能会变得不稳定，或完全停止工作。

6.1.2.1 选择要安装或去除的包

可以使用 `zypper install` 和 `zypper remove` 命令通过多种方法来找到包。

按确切的包名称

```
tux > sudo zypper install MozillaFirefox
```

按确切的包名称和版本号

```
tux > sudo zypper install MozillaFirefox-52.2
```

按储存库别名和包名称

```
tux > sudo zypper install mozilla:MozillaFirefox
```

其中 `mozilla` 是用于安装的储存库别名。

使用通配符按包名称

您可以选择名称以特定字符串开头或结尾的所有包。使用通配符要小心，特别是去除包的时候。以下命令将安装名称以“Moz”开头的所有包：

```
tux > sudo zypper install 'Moz*'
```

提示：去除所有 -debuginfo 包

在调试问题时，您有时需要临时安装大量的 `-debuginfo` 包，以获取有关正在运行的进程的详细信息。在调试会话完成后，如果您需要清理环境，请运行以下命令：

```
tux > sudo zypper remove '*-debuginfo'
```

按功能

例如，如果您要安装 Perl 模块但不知道包名称，功能就可以派上用场：

```
tux > sudo zypper install firefox
```

按功能、硬件体系结构或版本

可以结合功能指定硬件体系结构和版本：

- 所需硬件体系结构的名称需要追加在功能的后面，两者以句点分隔。例如，要指定 Intel 64/AMD64 体系结构（在 Zypper 中命名为 `x86_64`），请使用：

  ```
  tux > sudo zypper install 'firefox.x86_64'
  ```

- 版本必须追加到字符串的末尾，并且前面必须带有一个运算符：`<`（小于）、`<=`（小于等于）、`=`（等于）、`>=`（大于等于）或 `>`（大于）。

  ```
  tux > sudo zypper install 'firefox>=52.2'
  ```

- 还可以指定硬件体系结构与版本组合要求：

  ```
  tux > sudo zypper install 'firefox.x86_64>=52.2'
  ```

按 RPM 文件的路径

您还可以指定包的本地或远程路径：

```
tux > sudo zypper install /tmp/install/MozillaFirefox.rpm
tux > sudo zypper install http://download.example.com/MozillaFirefox.rpm
```

6.1.2.2 同时安装和去除包

要同时安装和去除包，请使用 `+/-` 修饰符。要安装 `emacs` 并同时去除 `vim`，请使用：

```
tux > sudo zypper install emacs -vim
```

要去除 `emacs` 并同时安装 `vim` ，请使用：

```
tux > sudo zypper remove emacs +vim
```

为避免 `-` 开头的包名称被解释为命令行选项，要始终把它用作第二个自变量。如果做不到这点，在它之前加上 `--`：

```
tux > sudo zypper install -emacs +vim       # Wrong
tux > sudo zypper install vim -emacs        # Correct
tux > sudo zypper install -- -emacs +vim    # Correct
tux > sudo zypper remove emacs +vim         # Correct
```

6.1.2.3 清理已去除包的依赖项

如果您想将在指定的包去除后不再需要的所有包（随指定的包）自动去除，请使用 `--clean-deps` 选项：

```
tux > sudo zypper rm PACKAGE_NAME --clean-deps
```

6.1.2.4 在脚本中使用 Zypper

默认情况下，在安装或删除选定包之前或发生问题时，Zypper 会要求确认。您可以使用 `--non-interactive` 选项覆盖此行为。必须在实际命令（`install`、`remove` 和 `patch`）的前面指定此选项，如下所示：

```
tux > sudo zypper --non-interactive install PACKAGE_NAME
```

该选项允许在脚本和 cron 任务中使用 Zypper。

6.1.2.5 安装或下载源包

要安装某个包的对应源代码包，请使用：

```
tux > zypper source-install PACKAGE_NAME
```

以 root 身份执行时，源包的默认安装位置为 /usr/src/packages/；以用户身份运行时，则为 ~/rpmbuild。可以在本地 rpm 配置中更改这些值。

使用此命令还会安装指定包的版本依赖项。如果不想执行此操作，请添加开关 -D：

```
tux > sudo zypper source-install -D PACKAGE_NAME
```

要只安装版本依赖项，请使用 -d。

```
tux > sudo zypper source-install -d PACKAGE_NAME
```

当然，只有当储存库列表中启用了含有源包的储存库时，才能这样做（默认添加但不启用它）。请参见第 6.1.5 节 “用 Zypper 管理安装源” 了解有关储存库管理的细节。

可使用以下方法来获取储存库中所有源包的列表：

```
tux > zypper search -t srcpackage
```

您也可以将所有已安装软件包的源包下载到本地目录。要下载源包，请使用：

```
tux > zypper source-download
```

默认的下载目录是 /var/cache/zypper/source-download。您可以使用 --directory 选项更改下载目录。若只想显示缺失或多余的包而不进行下载或删除任何内容，请使用 --status 选项。要删除多余的源包，请使用 --delete 选项。要禁用删除，请使用 --no-delete 选项。

6.1.2.6 从禁用的储存库安装包

通常，您只能安装或刷新来自启用的储存库的包。--plus-content 标记选项可帮助您指定要刷新的、要在当前 Zypper 会话期间暂时启用的，以及要在会话完成后禁用的储存库。

例如，要启用可以提供其他 -debuginfo 或 -debugsource 包的储存库，请使用 --plus-content debug。可以多次指定此选项。

要暂时启用此类“调试”储存库以安装特定的 -debuginfo 包，请按如下所示使用该选项：

```
tux > sudo zypper --plus-content debug \
   install "debuginfo(build-id)=eb844a5c20c70a59fc693cd1061f851fb7d046f4"
```

对于缺少的 debuginfo 包，`gdb` 将会报告 `build-id` 字符串。

6.1.2.7 实用程序

要校验所有依赖项是否仍然满足，并修复缺少的依赖项，请使用：

```
tux > zypper verify
```

除了依赖项必须满足外，某些包还“推荐”其他包。只有在实际可用并可安装时才会安装这些推荐包。如果推荐的包在推荐它们的包已安装（通过添加其他包或硬件）之后才可用，请使用以下命令：

```
tux > sudo zypper install-new-recommends
```

此命令在插入网络摄像头或 Wi-Fi 设备后非常有用。如果可用，它将安装设备驱动程序和相关软件。只有在满足特定硬件依赖项后，才可安装驱动程序和相关软件。

6.1.3 使用 Zypper 更新软件

用 Zypper 更新软件有三种方式：安装包、安装包的新版本或更新整个分发包。最后一种方式可通过 `zypper dist-upgrade` 来实现。中介绍了如何升级 SUSE Linux Enterprise Server《部署指南》, 第 19 章 “升级 SUSE Linux Enterprise”。

6.1.3.1 安装全部所需的增补程序

要安装所有适用于您系统的正式发布的增补程序，请运行：

```
tux > sudo zypper patch
```

系统将会检查您计算机上配置的储存库中提供的所有增补程序是否与您的安装相关。如果相关（未分为 可选 或 功能 类别），则会立即安装这些增补程序。请注意，正式的更新储存库仅在注册 SUSE Linux Enterprise Server 安装后才可用。

如果即将安装的增补程序所包含的更改要求重引导系统，您会在重引导前收到警告。

单纯使用 `zypper patch` 命令不会应用来自第三方储存库的包。要同时更新第三方储存库，请使用 `with-update` 命令选项，如下所示：

```
tux > sudo zypper patch --with update
```

要额外安装可选增补程序，请使用：

```
tux > sudo zypper patch --with-optional
```

要安装与特定 Bugzilla 问题相关的所有增补程序，请使用：

```
tux > sudo zypper patch --bugzilla=NUMBER
```

要安装与特定 CVE 数据库项相关的所有增补程序，请使用：

```
tux > sudo zypper patch --cve=NUMBER
```

例如，要安装 CVE 编号为 `CVE-2010-2713` 的安全增补程序，请执行：

```
tux > sudo zypper patch --cve=CVE-2010-2713
```

如果只想安装影响 Zypper 和包管理本身的增补程序，请使用：

```
tux > sudo zypper patch --updatestack-only
```

请记住，如果您使用了 `updatestack-only` 命令选项，将会丢弃原本还会更新其他储存库的其他命令选项。

6.1.3.2 列出增补程序

为了让您确定增补程序是否可用，Zypper 允许您查看以下信息：

所需增补程序的数目

要列出所需增补程序（适用于您的系统但尚未安装的增补程序）的数目，请使用 `patch-check`：

```
tux > zypper patch-check
```

```
Loading repository data...
Reading installed packages...
5 patches needed (1 security patch)
```

可以结合 `--updatestack-only` 选项使用此命令，以便仅列出影响 Zypper 和包管理本身的增补程序。

所需增补程序的列表

要列出全部所需的增补程序（适用于您的系统但尚未安装的增补程序），请使用 `list-patches`：

```
tux > zypper list-patches
Loading repository data...
Reading installed packages...

Repository     | Name         | Version | Category | Status  | Summary
---------------+--------------+---------+----------+---------+---------
SLES12-Updates | SUSE-2014-8 | 1       | security | needed  | openssl: Update for OpenSSL
```

所有增补程序的列表

要列出 SUSE Linux Enterprise Server 可用的所有增补程序，而不管它们是否已安装或适用于您的安装，请使用 `zypper patches`。

还可以列出并安装与特定问题相关的增补程序。要列出特定的增补程序，请使用带以下选项的 `zypper list-patches` 命令：

按 Bugzilla 问题

要列出与 Bugzilla 问题相关的全部所需增补程序，请使用 `--bugzilla` 选项。

要列出针对特定 Bug 的增补程序，您也可以指定 Bug 编号：`--bugzilla=编号`。要搜索与多个 Bugzilla 问题相关的增补程序，请在 bug 编号之间添加逗号，例如：

```
tux > zypper list-patches --bugzilla=972197,956917
```

按 CVE 编号

要列出与 CVE（公共漏洞和披露）数据库中某个项相关的全部所需增补程序，请使用 `--cve` 选项。

要列出针对特定 CVE 数据库项的增补程序，您也可以指定 CVE 编号：`--cve=编号`。要搜索与多个 CVE 数据库项相关的增补程序，请在 CVE 编号之间添加逗号，例如：

```
tux > zypper list-patches --bugzilla=CVE-2016-2315,CVE-2016-2324
```

要列出所有增补程序而不管是否需要安装它们，请另外使用 `--all` 选项。例如，要列出指派有 CVE 编号的所有增补程序，请使用：

```
tux > zypper list-patches --all --cve
Issue | No.           | Patch             | Category    | Severity  | Status
------+---------------+-------------------+-------------+-----------+----------
cve   | CVE-2015-0287 | SUSE-SLE-Module.. | recommended | moderate  | needed
cve   | CVE-2014-3566 | SUSE-SLE-SERVER.. | recommended | moderate  | not needed
[...]
```

6.1.3.3 安装新的包版本

如果某个安装源只包含新包，但未提供增补程序，则 `zypper patch` 不会产生任何作用。要使用可用的较新版本更新所有已安装的包（同时还要保持系统完整性），请使用：

```
tux > sudo zypper update
```

要更新个别包，请用更新或安装命令指定包：

```
tux > sudo zypper update PACKAGE_NAME
sudo zypper install PACKAGE_NAME
```

可使用此命令来获取所有新的可安装包的列表：

```
tux > zypper list-updates
```

请注意，此命令只会列出符合以下准则的包：

- 与已安装的包拥有相同的供应商，
- 由至少与已安装包拥有相同优先级的储存库提供，
- 可安装（满足所有依赖项）。

所有新的可用包（无论是否可安装）的列表可通过以下方式获取：

```
tux > sudo zypper list-updates --all
```

要找出新包无法安装的原因，请使用上面所述的 `zypper install` 或 `zypper update` 命令。

6.1.3.4 识别孤立的包

每当您从 Zypper 中去除某个储存库或者升级系统时，某些包可能会进入“孤立”状态。这些孤立的包不再属于任何活动储存库。以下命令可以列出这些包：

```
tux > sudo zypper packages --orphaned
```

借助此列表，您可以确定是否仍然需要某个包，或者是否可以安全去除某个包。

6.1.4 识别使用已删除文件的进程和服务

在增补、更新或去除包时，系统上可能有一些正在运行的进程会继续使用更新或去除后已被删除的文件。运行 `zypper ps` 可以列出使用已删除文件的进程。如果此类进程属于某个已知的服务，则会列出服务名称，方便您重启动该服务。默认情况下，`zypper ps` 会显示一个表：

```
tux > zypper ps
PID   | PPID | UID | User   | Command      | Service      | Files
------+------+-----+--------+--------------+--------------+-------------------
814   | 1    | 481 | avahi  | avahi-daemon | avahi-daemon | /lib64/ld-2.19.s->
      |      |     |        |              |              | /lib64/libdl-2.1->
      |      |     |        |              |              | /lib64/libpthrea->
      |      |     |        |              |              | /lib64/libc-2.19->
[...]
```

`PID`：进程的 ID

`PPID`：父进程的 ID

`UID`：运行进程的用户的 ID

`User`：运行进程的用户的登录名

`Command`：用于执行进程的命令

`Service`：服务名称（仅当命令与系统服务关联时才显示）

`Files`：已删除文件的列表

通过如下方式可控制 `zypper ps` 的输出格式：

`zypper ps -s`

创建一份简短表格，其中不会显示已删除的文件。

```
tux > zypper ps -s
PID   | PPID | UID  | User    | Command      | Service
------+------+------+---------+--------------+--------------
814   | 1    | 481  | avahi   | avahi-daemon | avahi-daemon
817   | 1    | 0    | root    | irqbalance   | irqbalance
1567  | 1    | 0    | root    | sshd         | sshd
1761  | 1    | 0    | root    | master       | postfix
1764  | 1761 | 51   | postfix | pickup       | postfix
1765  | 1761 | 51   | postfix | qmgr         | postfix
2031  | 2027 | 1000 | tux     | bash         |
```

`zypper ps -ss`

仅显示与系统服务关联的进程。

```
PID   | PPID | UID  | User    | Command      | Service
------+------+------+---------+--------------+--------------
814   | 1    | 481  | avahi   | avahi-daemon | avahi-daemon
817   | 1    | 0    | root    | irqbalance   | irqbalance
1567  | 1    | 0    | root    | sshd         | sshd
1761  | 1    | 0    | root    | master       | postfix
1764  | 1761 | 51   | postfix | pickup       | postfix
1765  | 1761 | 51   | postfix | qmgr         | postfix
```

`zypper ps -sss`

仅显示使用已删除文件的系统服务。

```
avahi-daemon
irqbalance
postfix
sshd
```

`zypper ps --print "systemctl status %s"`

显示用于检索可能需要重启动的服务状态信息的命令。

```
systemctl status avahi-daemon
systemctl status irqbalance
systemctl status postfix
```

```
systemctl status sshd
```

有关服务处理的详细信息，请参见第 13 章 “systemd 守护程序”。

6.1.5 用 Zypper 管理安装源

Zypper 的所有安装或增补程序命令均基于已知安装源列表。要列出系统已知的所有储存库，请使用命令：

```
tux > zypper repos
```

结果将类似于与以下输出：

例 6.1 ZYPPER — 已知储存库的列表

```
tux > zypper repos
# | Alias        | Name          | Enabled | Refresh
--+--------------+---------------+---------+--------
1 | SLEHA-12-GEO | SLEHA-12-GEO  | Yes     | No
2 | SLEHA-12     | SLEHA-12      | Yes     | No
3 | SLES12       | SLES12        | Yes     | No
```

当在各个命令中指定储存库时，可以使用别名、URI 或 `zypper repos` 命令输出中的储存库编号。储存库别名是用于储存库处理命令中的储存库名称的简短版本。请注意，在修改储存库列表后，储存库编号可能会更改。别名本身不会更改。

默认情况下不显示储存库的 URI 或优先级之类的细节。用以下命令可以列出所有细节：

```
tux > zypper repos -d
```

6.1.5.1 添加安装源

要添加安装源，请运行

```
tux > sudo zypper addrepo URI ALIAS
```

`URI` 可以是因特网储存库、网络资源、目录、CD 或 DVD（有关细节请参见 http://en.opensuse.org/openSUSE:Libzypp_URIs↗）。`ALIAS` 是储存库的唯一简写标识符。您可以随意选择别名，前提是它必须唯一。如果指定的别名已在使用，Zypper 将发出警告。

6.1.5.2 刷新储存库

`zypper` 可让您从配置的储存库中提取包的更改。要提取更改，请运行：

```
tux > sudo zypper refresh
```

注意：`zypper` 的默认行为

有些命令默认会自动执行 `refresh`，因此您不需要明确运行该命令。

使用 `refresh` 命令时搭配 `--plus-content` 选项还可查看已禁用储存库中的更改：

```
tux > sudo zypper --plus-content refresh
```

该选项虽然会提取储存库中的更改，但会使禁用储存库的状态保持不变，即仍为禁用。

6.1.5.3 删除储存库

要从列表中去除某个储存库，请将命令 `zypper removerepo` 与要删除的储存库的别名或编号结合使用。例如，要从例 6.1 “Zypper — 已知储存库的列表”中去除储存库 `SLEHA-12-GEO`，请使用下列命令之一：

```
tux > sudo zypper removerepo 1
tux > sudo zypper removerepo "SLEHA-12-GEO"
```

6.1.5.4 修改储存库

用 `zypper modifyrepo` 启用或禁用储存库。您还可以用该命令更改储存库的属性（例如刷新行为、名称或优先级）。以下命令将会启用名为 `updates` 的储存库、打开自动刷新并将其优先级设置为 20：

```
tux > sudo zypper modifyrepo -er -p 20 'updates'
```

修改储存库并不局限于单个储存库 — 您也可以对组执行该操作：

`-a`：所有储存库

`-l`：本地储存库

`-t`：远程储存库

`-m 类型`：特定类型的储存库（其中 `类型` 可以是以下之一：`http`、`https`、`ftp`、`cd`、`dvd`、`dir`、`file`、`cifs`、`smb`、`nfs`、`hd` 和 `iso`）

要重命名安装源别名，请使用 `renamerepo` 命令。以下示例将别名从 `Mozilla Firefox` 更改为 `firefox`：

```
tux > sudo zypper renamerepo 'Mozilla Firefox' firefox
```

6.1.6 用 Zypper 查询储存库和包

Zypper 提供各种查询储存库或包的方式。要获取所有可用的产品、模式、包或增补程序的列表，请使用以下命令：

```
tux > zypper products
tux > zypper patterns
tux > zypper packages
tux > zypper patches
```

要查询特定包的所有储存库，请使用 `search`。要获得有关特定包的信息，请使用 `info` 命令。

6.1.6.1 `zypper search` 用法

`zypper search` 命令可对包名或（视情况）对包摘要和说明执行搜索。括在 `/` 中的字符串会解译为正则表达式。默认情况下搜索不区分大小写。

执行简单搜索来查找包含 `fire` 的包名称

```
tux > zypper search "fire"
```

执行简单搜索来查找确切的包 `MozillaFirefox`

```
tux > zypper search --match-exact "MozillaFirefox"
```

同时在包描述和摘要中搜索

```
tux > zypper search -d fire
```

仅显示尚未安装的包

```
tux > zypper search -u fire
```

显示包含字符串 `fir` 且该字符串后面不是 `e` 的包

```
tux > zypper se "/fir[^e]/"
```

6.1.6.2 `zypper what-provides` 用法

要搜索提供特殊功能的包，请使用命令 `what-provides`。例如，如果您想知道哪个包提供 Perl 模块 `SVN::Core`，请使用以下命令：

```
tux > zypper what-provides 'perl(SVN::Core)'
```

`what-provides` *包名* 与 `rpm -q --whatprovides` *包名* 类似，不过 RPM 只能查询 RPM 数据库（即所有已安装的包的数据库）。另一方面，Zypper 将告诉您任意储存库的功能的提供商，而非仅已安装的储存库功能的提供商。

6.1.6.3 `zypper info` 用法

要查询个别包，请使用 `info` 命令，并用完整包名称作为自变量。这会显示有关某个包的详细信息。如果包名与储存库中的所有包名都不匹配，该命令会输出非包匹配项的详细信息。如果您请求特定类型（通过使用 `-t` 选项），但该类型不存在，该命令会输出其他可用的匹配项，但不提供详细信息。

如果您指定源包，该命令会显示基于该源包构建的二进制包。如果您指定二进制包，该命令会输出用来构建该二进制包的源包。

如果还要显示该包必需/推荐的包，则使用选项 `--requires` 和 `--recommends`：

```
tux > zypper info --requires MozillaFirefox
```

6.1.7 配置 Zypper

Zypper 现在随附配置文件，允许您永久更改 Zypper 的行为（系统范围或用户特定）。要进行系统范围更改，请编辑 `/etc/zypp/zypper.conf`。要进行用户特定的更改，请编辑 `~/.zypper.conf`。如果 `~/.zypper.conf` 尚不存在，您可以使用 `/etc/zypp/zypper.conf` 作为模板：将其复制到 `~/.zypper.conf` 并根据您的喜好进行调整。请参见文件中的注释，获取有关可用选项的帮助。

6.1.8 查错

如果您在访问配置的储存库中的包时遇到问题（例如，尽管您知道某个包在某个储存库中，但 Zypper 找不到该包），刷新储存库或许可以解决问题：

```
tux > sudo zypper refresh
```

如果不起作用，则尝试

```
tux > sudo zypper refresh -fdb
```

这会强制完全刷新和重构建数据库，包括强制下载原始元数据。

6.1.9 Btrfs 文件系统上的 Zypper 回滚功能

如果根分区上使用的是 Btrfs 文件系统，且系统中安装了 `snapper`，当 Zypper 提交对文件系统所做的更改以创建相应的文件系统快照时，会自动调用 `snapper`。这些快照可用于还原 Zypper 进行的任何更改。有关详细信息，请参见第 7 章 “通过 Snapper 进行系统恢复和快照管理”。

6.1.10 更多信息

有关从命令行管理软件的详细信息，请输入 `zypper help`、`zypper help 命令`，或参见 `zypper(8)` 手册页。有关详尽的命令参考、最重要的命令的速查表，以及如何在脚本和应用程序中使用 Zypper 的信息，请参见 http://en.opensuse.org/SDB:Zypper_usage。最新 SUSE Linux Enterprise Server 版本的软件更改列表可在 http://en.opensuse.org/openSUSE:Zypper versions 中找到。

6.2 RPM — 包管理器

RPM（RPM 程序包管理器）用于管理软件包。其主要程命令为 `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` 有五种模式：安装、卸装（或更新）软件包、重构建 RPM 数据库、查询 RPM 库或独立 RPM 存档、包的完整性检查以及对包签名。`rpmbuild` 可用于从原始源构建可安装的包。

用特殊的二进制格式对可安装 RPM 存档进行打包。这些存档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 存档通常具有扩展名 `.rpm`。

提示：软件开发包

对于一些包，软件开发所需的组件（库、报头、包含文件等）已纳入独立的包中。只有当您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` 和 `gimp-devel`。

6.2.1 校验包真实性

RPM 包具有 GPG 签名。要校验 RPM 包的签名，请使用 `rpm --checksig PACKAGE`-1.2.3.rpm 命令确定该包是来自 SUSE 还是另一个可信机构。特别建议对来自因特网的更新包使用此命令。

修复操作系统中的问题时，您可能需要将问题临时修复 (PTF) 安装到生产系统中。SUSE 提供的包已使用特殊的 PTF 密钥签名。但是，与 SUSE Linux Enterprise 11 不同，SUSE Linux Enterprise 12 系统上默认不会导入此密钥。要手动导入该密钥，请使用以下命令：

```
tux > sudo rpm --import \
/usr/share/doc/packages/suse-build-key/suse_ptf_key.asc
```

导入该密钥后，您可以在系统上安装 PTF 包。

6.2.2 管理包：安装、更新和卸装

安装 RPM 存档的步骤通常十分简单，执行运行：`rpm -i 包`.rpm。使用此命令可以安装包，但前提是满足其依赖项并且不与其他包冲突。如果出现错误消息，`rpm` 将请求那些需要安装的包以满足依赖项要求。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` 或 `--upgrade` 以及 `-F` 或 `--freshen` 可用于更新包（例如，`rpm -F PACKAGE`.rpm）。此命令将删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，而 `-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。
- 如果配置文件在更新前曾被系统管理员更改，则 `rpm` 会以扩展名 `.rpmorig` 或 `.rpmsave`（备份文件）保存更改的文件，并安装新包中的版本。仅当原先安装的文件和较新的版本不同时，才执行此操作。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。之后，请删除所有 `.rpmorig` 和 `.rpmsave` 文件，以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删除，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过任何更改的信息。`/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`。

要去除包，请输入 `rpm -e PACKAGE`。仅当不存在未解决的依赖项问题时，此命令才会删除包。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。如果出于任何原因无法进行此删除操作（即使不存在其他依赖项），则最好使用选项 `--rebuilddb` 重构建 RPM 数据库。

6.2.3 增量 RPM 包

增量 RPM 包包含旧版本和新版本的 RPM 包之间的差别。在旧 RPM 上应用增量 RPM 将得到全新的 RPM。不需要旧 RPM 的副本，因为增量 RPM 也可以与已安装的 RPM 一起工作。增量 RPM 包的大小甚至比增补程序 RPM 小，这有利于通过因特网传送更新包。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。

`makedeltarpm` 和 `applydelta` 二进制文件是增量 RPM 套件（包 `deltarpm`）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名为 `new.delta.rpm` 的增量 RPM。以下命令假设 `old.rpm` 和 `new.rpm` 是存在的：

```
tux > sudo makedeltarpm old.rpm new.rpm new.delta.rpm
```

如果旧包已经安装，则使用 `applydeltarpm` 可以从文件系统重新构建新的 RPM：

```
tux > sudo applydeltarpm new.delta.rpm new.rpm
```

如果不访问文件系统而从旧 RPM 得到它，请使用 `-r` 选项：

```
tux > sudo applydeltarpm -r old.rpm new.delta.rpm new.rpm
```

关于技术详细信息，请参见 `/usr/share/doc/packages/deltarpm/README`。

6.2.4 RPM 查询

带 `-q` 选项的 `rpm` 将启动查询，如此用户便可查看 RPM 存档（通过添加选项 `-p`）并查询已安装包的 RPM 数据库。可以使用多个开关指定所需信息的类型。请参见表 6.1 “最重要的 RPM 查询选项”。

表 6.1 最重要的 RPM 查询选项

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

例如，命令 `rpm -q -i wget` 显示例 6.2 “`rpm -q -i wget`” 中所示的信息。

例 6.2 `rpm -q -i wget`

```
Name        : wget
Version     : 1.14
Release     : 17.1
Architecture: x86_64
Install Date: Mon 30 Jan 2017 14:01:29 CET
Group       : Productivity/Networking/Web/Utilities
Size        : 2046483
```

```
License     : GPL-3.0+
Signature   : RSA/SHA256, Thu 08 Dec 2016 07:48:44 CET, Key ID 70af9e8139db7c82
Source RPM  : wget-1.14-17.1.src.rpm
Build Date  : Thu 08 Dec 2016 07:48:34 CET
Build Host  : sheep09
Relocations : (not relocatable)
Packager    : https://www.suse.com/
Vendor      : SUSE LLC <https://www.suse.com/>
URL         : http://www.gnu.org/software/wget/
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.
Distribution: SUSE Linux Enterprise 12
```

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：

```
tux > rpm -q -f /bin/rpm /usr/bin/wget
rpm-4.11.2-15.1.x86_64
wget-1.14-17.1.x86_64
```

如果只知道部分文件名，则可以使用外壳脚本，如例 6.3 “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脚本。

例 6.3 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
```

`rpm -q --changelog PACKAGE` 命令会按日期排序显示有关特定包的详细更改信息列表。

借助已安装的 RPM 数据库，可以进行校验检查。使用 `-V` 或 `--verify` 启动这些检查。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：

表 6.2 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 包）的更改：

```
tux > 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 重构建数据库。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_RPMDB_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 1 MB。

6.2.5 安装和编译源包

所有源包都带有 .src.rpm 扩展名（源 RPM）。

注意：已安装的源包

源包可以从安装媒体复制到硬盘并使用 YaST 解压缩。但是，在包管理器中它们不会被标记为已安装（[i]）。这是因为源包不是在 RPM 数据库中输入的。只有已安装的操作系统软件列在 RPM 数据库中。安装“源包时，只将源代码添加到系统中。”

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
: 在此目录中解压缩、增补和编译所有源

RPMS
: 储存完整的二进制包的位置

SRPMS
: 这里是源 RPM

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

警告：系统完整性

不要对系统组件（glibc、rpm 等）进行试验，因为这样做会影响系统的稳定性。

下面的示例使用 wget.src.rpm 包。安装源包后，应具有类似以下列表中的文件：

```
/usr/src/packages/SOURCES/wget-1.11.4.tar.bz2
/usr/src/packages/SOURCES/wgetrc.patch
/usr/src/packages/SPECS/wget.spec
```

rpmbuild -bX /usr/src/packages/SPECS/wget.spec 会启动编译。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）。以下内容只是简要描述：

-bp
: 在 /usr/src/packages/BUILD 中准备源：解压和打增补程序。

`-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。注意：如果包不支持 BuildRoot 功能，则可能会重写配置文件。

`-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。

请记住，从 SUSE Linux Enterprise Server 12 开始，已弃用规范文件中的 `BuildRoot` 指令。如果您仍然需要此功能，请使用 `--buildroot` 选项作为替代方法。有关更详细的背景信息，请参见 https://www.suse.com/support/kb/doc?id=7017104 上的支持数据库。

6.2.6 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`，它将创建构建包的已定义环境。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 或从 DVD 使用硬盘上的此树。使用 `build --rpms DIRECTORY` 设置位置。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 构建 `wget`（如上例所示），请以 `root` 用户身份使用以下命令：

```
root # cd /usr/src/packages/SOURCES/
root # mv ../SPECS/wget.spec .
root # 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` 脚本提供多个其他选项。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。

6.2.7 用于 RPM 存档和 RPM 数据库的工具

Midnight Commander (`mc`) 可以显示 RPM 存档的内容并复制部分内容。它将存档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 键查看存档结构。使用 F5 键复制部分存档。

拥有全部功能的包管理器将作为 YaST 模块提供。有关细节，请参见《部署指南》, 第 13 章 “安装或删除软件”。

7 通过 Snapper 进行系统恢复和快照管理

能够生成文件系统快照以便在 Linux 上实现回滚，这是过去常常要求提供的功能。如今，Snapper 与 `Btrfs` 文件系统或瘦配置的 LVM 卷相结合，填补了这一空白。

`Btrfs` 是全新的 Linux 写入时复制文件系统。它支持为子卷（每个物理分区中的一或多个单独的可装入文件系统）生成文件系统快照（复制子卷在某个时间点的状态）。XFS、Ext4 或 Ext3 格式的精简 LVM 卷上也支持快照。您可以使用 Snapper 创建并管理这些快照。Snapper 有一个命令行和一个 YaST 界面。从 SUSE Linux Enterprise Server 12 开始，还可以从 `Btrfs` 快照引导。有关详细信息，请参见第 7.3 节 “通过从快照引导来执行系统回滚”。

您可以使用 Snapper 执行以下任务：

- 撤销 `zypper` 和 YaST 所做的系统更改。有关详细信息，请参见第 7.2 节 “使用 Snapper 撤销更改”。
- 通过先前的快照恢复文件。有关详细信息，请参见第 7.2.2 节 “使用 Snapper 恢复文件”。
- 从快照引导以执行系统回滚。有关详细信息，请参见第 7.3 节 “通过从快照引导来执行系统回滚”。
- 以手动方式即时创建快照并管理现有快照。有关详细信息，请参见第 7.6 节 “手动创建和管理快照”。

7.1 默认设置

SUSE Linux Enterprise Server 中设置 Snapper 的目的是提供系统更改的“撤销和恢复工具”。默认情况下，SUSE Linux Enterprise Server 的根分区 (`/`) 使用 `Btrfs` 格式。如果根分区 (`/`) 足够大（约为 16GB 以上），则创建快照功能会自动启用。默认不允许创建除 `/` 以外的分区的快照。

提示：在已安装系统中启用 Snapper

如果您在安装期间禁用了 Snapper，以后随时都可启用它。要执行此操作，请运行以下命令创建根文件系统的默认 Snapper 配置：

```
tux > sudo snapper -c root create-config /
```

之后，按第 7.1.3.1 节 “禁用/启用快照”中所述启用不同的快照类型。

请记住，要使用快照，需按照安装程序的建议设置一个包含子卷的 Btrfs 根文件系统，并且需有一个大小至少为 16 GB 的分区。

创建快照时，快照与原始点都会指向文件系统中的同一个块。因此一开始时快照并不占用额外的磁盘空间。但如果修改了原始文件系统中的数据，则会复制已更改的数据块，同时将旧的数据块作为快照保留。因此，快照就将占用与已修改数据相同的空间。所以久而久之，分配给快照的空间便会不断增长。其结果是，即便从包含快照的 `Btrfs` 文件系统删除文件可能也不会释放磁盘空间！

注意：快照存储位置

快照始终存放在创建快照的那个分区或子卷中，而无法将快照存储到其他分区或子卷。

因此，包含快照的分区需要比“常规”分区大才行。确切的空间大小主要取决于要保留的快照数量以及数据更改量。一般来说，您应考虑使用两倍于常规使用磁盘空间的空间大小。为了防止磁盘上的空间耗尽，系统会自动清理旧快照。有关详细信息，请参见第 7.1.3.4 节 “控制快照存档”。

7.1.1 快照类型

尽管快照本身在技术方面并无区别，但我们根据触发它们的事件将其分成三类：

时间线快照

每小时创建一个快照，且旧的快照会自动删除。默认情况下，会保留最近十天、最近十个月以及最近十年的首张快照。时间线快照默认为禁用状态。

安装快照

每次使用 YaST 或 Zypper 安装一个或多个包时，会创建一对快照：一个是在安装开始前（“前”），另一个是在安装完成后（“后”）。如果安装了内核等重要的系统组件，快照对会标记为重要（`important=yes`）。旧的快照会自动删除。默认情况下，会保留最近十个重要快照和最近十个“普通”快照（包括管理快照）。安装快照默认为启用状态。

管理快照

每次使用 YaST 管理系统时都会创建一对快照：一个是在 YaST 模块启动之前（“前”），另一个是在该模块关闭之后（“后”）。旧的快照会自动删除。默认情况下，会保留最近十个重要快照和最近十个“普通”快照（包括安装快照）。管理快照默认为启用状态。

7.1.2 快照中排除的目录

出于多种不同的理由，有些目录需要排除在快照之外。以下列表显示了排除的所有目录：

`/boot/grub2/i386-pc`、`/boot/grub2/x86_64-efi`、`/boot/grub2/powerpc-ieee1275`、`/boot/grub2/s390x-emu`

不能回滚引导加载程序配置。上面列出的目录是架构专属目录。前两个目录位于 AMD64/Intel 64 计算机上，后两个目录分别位于 IBM POWER 和 IBM Z 上。

`/home`

如果独立的分区中没有 `/home`，便会将该目录排除以免在回滚时发生数据丢失。

`/opt`、`/var/opt`

第三方产品通常安装到 `/opt` 下。排除此目录是为了防止在回滚时卸装这些应用程序。

`/srv`

包含 Web 和 FTP 服务器的数据。排除此目录是为了防止在回滚时发生数据丢失。

`/tmp`、`/var/tmp`、`/var/cache`、`/var/crash`

包含临时文件和超速缓存的所有目录都会排除在快照范围之外。

`/usr/local`

在手动安装软件时会用到此目录。系统会将该目录排除以免在回滚时卸载这些安装的软件。

`/var/lib/libvirt/images`

使用 libvirt 管理的虚拟机映像的默认位置。为确保回滚期间虚拟机映像不会替换为旧版本而被排除。默认情况下，此子卷是使用 写入时不复制 选项创建的。

`/var/lib/mailman`、`/var/spool`

包含邮件或邮件队列的目录会排除，以免在回滚后造成邮件丢失。

`/var/lib/bind`

包含 DNS 服务器的区域数据。排除该目录是为了确保回滚后名称服务器仍能运作。

`/var/lib/mariadb`、`/var/lib/mysql`、`/var/lib/pgqsl`

这些目录包含数据库数据。默认情况下，这些子卷是使用 写入时不复制 选项创建的。

`/var/log`

日志文件所在的位置。排除该目录是为了在对受损的系统进行回滚后能够对日志文件进行分析。`/var/log` 默认会设置 NoCOW 属性，禁止写入时复制，从而改进了性能并减少了重复的块数量。使用 `lsattr` 进行校验：

```
tux > lsattr -l /var/
/var/log      No_COW
```

7.1.3 自定义设置

SUSE Linux Enterprise Server 自带的默认设置经过多方面的考虑，适合多数使用情况。不过，您可以根据自己的需要对创建自动快照以及保留快照的各个方面进行配置。

7.1.3.1 禁用/启用快照

这三种快照类型（时间线、安装、管理）都可以单独启用或禁用。

禁用/启用时间线快照

启用： `snapper-c root set-config "TIMELINE_CREATE=yes"`

禁用： `snapper -c root set-config "TIMELINE_CREATE=no"`

时间线快照默认会启用，但根分区除外。

禁用/启用安装快照

启用： 安装 `snapper-zypp-plugin` 包

禁用： 卸装 `snapper-zypp-plugin` 包

安装快照默认为启用状态。

禁用/启用管理快照

启用： 在 `/etc/sysconfig/yast2` 中将 `USE_SNAPPER` 设置为 `yes`。

禁用： 在 `/etc/sysconfig/yast2` 中将 `USE_SNAPPER` 设置为 `no`。

管理快照默认为启用状态。

7.1.3.2 控制安装快照

使用 YaST 或 Zypper 安装包时所创建的快照会由 `snapper-zypp-plugin` 进行处理。何时创建快照由 XML 配置文件 `/etc/snapper/zypp-plugin.conf` 定义。默认情况下，该文件如下所示：

```
 1 <?xml version="1.0" encoding="utf-8"?>
 2 <snapper-zypp-plugin-conf>
 3  <solvables>
 4   <solvable match="w" ❶ important="true" ❷>kernel-* ❸</solvable>
 5   <solvable match="w" important="true">dracut</solvable>
 6   <solvable match="w" important="true">glibc</solvable>
 7   <solvable match="w" important="true">systemd*</solvable>
 8   <solvable match="w" important="true">udev</solvable>
 9   <solvable match="w">*</solvable> ❹
10  </solvables>
11 </snapper-zypp-plugin-conf>
```

❶ match 属性定义模式是 Unix 外壳风格的通配符 (`w`) 还是 Python 正则表达式 (`re`)。

❷ 如果匹配指定模式而且对应的包标记为 important（例如内核包），则快照也会标记为 important。

❸ 用于匹配包名称的模式。根据 `match` 属性的设置，特殊字符可能会被解析为外壳通配符或是正则表达式。此模式匹配名称以 `kernel-` 开头的所有包。

❹ 此行无条件匹配所有包。

在这样的快照配置下，只要安装了包，就会创建快照对（第 9 行）。如果标记为 important 的内核、dracut、glibc、systemd 或 udev 包已安装，快照对也会标记为 important（第 4 行到第 8 行）。所有规则都会进行评估。

要禁用某规则，可以删除该规则或通过 XML 注释的方式将其停用。举例来说，如果不希望系统在每次安装包时创建快照对，可以将第 9 行注释掉：

```
 1 <?xml version="1.0" encoding="utf-8"?>
 2 <snapper-zypp-plugin-conf>
 3  <solvables>
 4   <solvable match="w" important="true">kernel-*</solvable>
 5   <solvable match="w" important="true">dracut</solvable>
 6   <solvable match="w" important="true">glibc</solvable>
 7   <solvable match="w" important="true">systemd*</solvable>
 8   <solvable match="w" important="true">udev</solvable>
 9   <!-- <solvable match="w">*</solvable> -->
10  </solvables>
11 </snapper-zypp-plugin-conf>
```

7.1.3.3 创建和装入新子卷

系统支持在 `/` 层次下创建新的子卷，并永久性装入该卷。此类子卷将从快照中排除。切勿在现有快照中创建此类子卷，因为在回滚之后，您将无法再删除快照。

SUSE Linux Enterprise Server 上配置了 `/@/` 子卷，该子卷充当永久性子卷（例如 `/opt`、`/srv`、`/home` 等）的独立根目录。您创建和永久装入的任何新子卷都需要在这个初始根文件系统中创建。

为此，请运行以下命令。在此示例中，从 `/dev/sda2` 创建了一个新子卷 `/usr/important`。

```
tux > sudo mount /dev/sda2 -o subvol=@ /mnt
tux > sudo btrfs subvolume create /mnt/usr/important
tux > sudo umount /mnt
```

`/etc/fstab` 中的相应项需类似于：

```
/dev/sda2 /usr/important btrfs subvol=@/usr/important 0 0
```

提示：禁用写入时复制 (cow)

子卷可能包含经常更改的文件，例如虚拟化的磁盘映像、数据库文件或日志文件。如果是这样，可考虑对此卷禁用写入时复制功能，以免复制磁盘块。可在 `/etc/fstab` 中使用 `nodatacow` 装入选项来实现此目的：

```
/dev/sda2 /usr/important btrfs nodatacow,subvol=@/usr/important 0 0
```

或者，要为单个文件或目录禁用写入时复制功能，请使用命令 `chattr +C 路径`。

7.1.3.4 控制快照存档

快照会占用磁盘空间。为了防止磁盘空间耗尽而导致系统中断，旧的快照会自动删除。默认情况下，将保留最多 10 个重要的安装快照与管理快照，以及最多 10 个普通的安装快照与管理快照。如果这些快照占用的空间超过根文件系统大小的 50%，将会删除其他快照。系统始终会至少保留 4 个重要快照和 2 个普通快照。

有关如何更改这些值的指导，请参考第 7.5.1 节 “管理现有配置”。

7.1.3.5 在精简的 LVM 卷上使用 Snapper

除了在 `Btrfs` 文件系统上生成快照之外，Snapper 还支持在 XFS、Ext4 或 Ext3 格式的精简 LVM 卷（不支持在常规 LVM 卷上生成快照）上生成快照。有关 LVM 卷的详细信息和设置指导，请参考《部署指南》，第 12 章 “高级磁盘设置”，第 12.2 节 “LVM 配置”。

若要在瘦配置的 LVM 卷上使用 Snapper，您需要为其创建 Snapper 配置。在 LVM 上要使用 `##fstype=lvm(文件系统)` 指定文件系统。`文件系统` 的有效值为 `ext3`、`etx4` 或 `xfs`。示例：

```
tux > sudo snapper -c lvm create-config --fstype="lvm(xfs)" /thin_lvm
```

您可以按照第 7.5.1 节 “管理现有配置”中的说明根据需要调整此配置。

7.2 使用 Snapper 撤销更改

SUSE Linux Enterprise Server 上的 Snapper 经过预配置，可以用来撤销 `zypper` 和 YaST 所做的更改。要实现此功能，请对 Snapper 进行配置，让其在每次运行 `zypper` 和 YaST 前后创建一个快照对。您也可以使用 Snapper 来恢复被意外删除或修改的系统文件。要实现此目的，需要对根分区启用时间线快照 — 有关细节，请参见第 7.1.3.1 节 “禁用/启用快照”。

默认情况下，上述的自动快照针对根分区及其子卷所配置。若希望针对 `/home` 等其他分区生成快照，您可以创建自定义配置。

重要：撤消更改与回滚的比较

通过快照来恢复数据时，必须知道，Snapper 可以处理两种完全不同的情形：

撤销更改

在如下文中所述撤销更改时，系统会对两个快照进行比较，并撤销两个快照之间的更改。借助这种方式可以明确地选择要恢复的文件。

回滚

在如第 7.3 节 “通过从快照引导来执行系统回滚”中所述进行回滚时，系统会重设置为创建快照时的状态。

撤销更改时，可以将快照与现有系统进行比较。如果比较后恢复所有发生变化的文件，那么结果会和回滚完全相同。但是，还是建议使用第 7.3 节 “通过从快照引导来执行系统回滚”中介绍的方法进行回滚，因为回滚操作的速度更快，而且您可以在进行回滚前查看系统。

警告：数据一致性

在创建快照时并没有能确保数据一致性的机制。如果在创建快照的同时写入某个文件（例如数据库），将导致文件损坏或写入不完整。恢复此类文件会产生问题。而且，有些系统文件（例如 `/etc/mtab`）甚至永远都无法恢复。因此强烈建议您要始终仔细查看已更改文件及其差异的列表。只恢复您要还原的操作真正包含的文件。

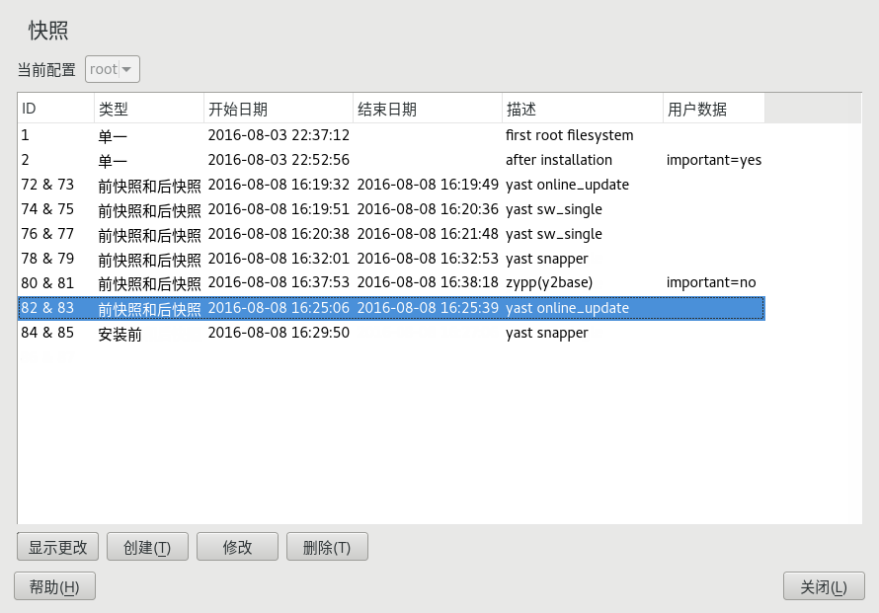
7.2.1 撤销 YaST 和 Zypper 更改

如果在安装时使用 `Btrfs` 设置根分区，则将自动安装 Snapper（经过预配置，可以回滚 YaST 或 Zypper 所做的更改）。每次启动 YaST 模块或 Zypper 事务时，会创建两个快照：即截获模块启动之前文件系统状态的“前快照”以及截获模块完成之后状态的“后快照”。

您可以使用 YaST Snapper 模块或 `snapper` 命令行工具，通过从“前快照”恢复文件来撤销 YaST 或 Zypper 所做的更改。您也可以使用该工具比较这两张快照，以查看更改了哪些文件。您还可以显示文件的两个版本之间的差异 (diff)。

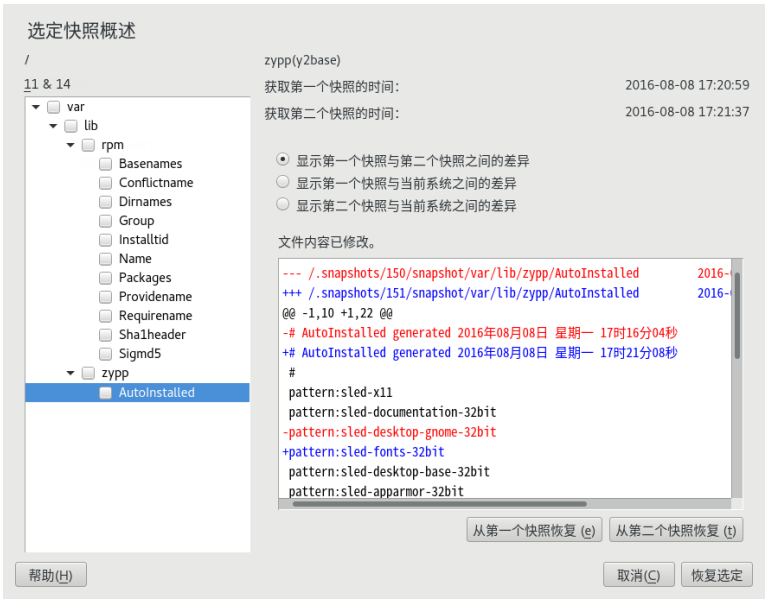
过程 7.1 使用 YAST SNAPPER 模块撤销更改

1. 从 YaST 中的其他部分或通过输入 yast2 snapper 来启动 `Snapper` 模块。
2. 务必将当前配置设置为根。除非手动添加自己的 Snapper 配置，否则请始终做此设置。
3. 从列表中选择前后快照对。YaST 和 Zypper 快照对都属于前后类型。在说明栏中，YaST 快照以 `zypp(y2base)` 标记；Zypper 快照以 `zypp(zypper)` 标记。



4. 单击显示更改，会打开一个文件列表显示两张快照之间的差异。

5. 查看文件列表。要显示文件的前后版本之间的“差异”，请从列表中选中该文件。



6. 要恢复一个或多个文件，请通过勾选相应的复选框选择相关的文件或目录。单击恢复选定，然后单击是以确认该操作。

要恢复单一文件，请单击其名称以激活该文件的差异视图。单击从第一个快照恢复，然后单击是予以确认。

过程 7.2 使用 `snapper` 命令撤销更改

1. 运行 `snapper list -t pre-post` 以获取 YaST 与 Zypper 快照的列表。在说明列中，YaST 快照标为 `yast MODULE_NAME`；Zypper 快照标为 `zypp(zypper)`。

```
tux > sudo snapper list -t pre-post
Pre # | Post # | Pre Date                      | Post Date                     | Description
------+--------+-------------------------------+-------------------------------+--------------
311   | 312    | Tue 06 May 2014 14:05:46 CEST | Tue 06 May 2014 14:05:52 CEST | zypp(y2base)
340   | 341    | Wed 07 May 2014 16:15:10 CEST | Wed 07 May 2014 16:15:16 CEST | zypp(zypper)
342   | 343    | Wed 07 May 2014 16:20:38 CEST | Wed 07 May 2014 16:20:42 CEST | zypp(y2base)
344   | 345    | Wed 07 May 2014 16:21:23 CEST | Wed 07 May 2014 16:21:24 CEST | zypp(zypper)
346   | 347    | Wed 07 May 2014 16:41:06 CEST | Wed 07 May 2014 16:41:10 CEST | zypp(y2base)
348   | 349    | Wed 07 May 2014 16:44:50 CEST | Wed 07 May 2014 16:44:53 CEST | zypp(y2base)
350   | 351    | Wed 07 May 2014 16:46:27 CEST | Wed 07 May 2014 16:46:38 CEST | zypp(y2base)
```

2. 使用 `snapper status PRE..POST` 命令以获取快照对的已更改文件列表。文件内容发生了更改会以 `c` 标记、添加了文件会以 `+` 标记、删除了文件会以 `-` 标记。

```
tux > sudo snapper status 350..351
+..... /usr/share/doc/packages/mikachan-fonts
```

```
+..... /usr/share/doc/packages/mikachan-fonts/COPYING
+..... /usr/share/doc/packages/mikachan-fonts/dl.html
c..... /usr/share/fonts/truetype/fonts.dir
c..... /usr/share/fonts/truetype/fonts.scale
+..... /usr/share/fonts/truetype/みかちゃん-p.ttf
+..... /usr/share/fonts/truetype/みかちゃん-pb.ttf
+..... /usr/share/fonts/truetype/みかちゃん-ps.ttf
+..... /usr/share/fonts/truetype/みかちゃん.ttf
c..... /var/cache/fontconfig/7ef2298fde41cc6eeb7af42e48b7d293-
x86_64.cache-4
c..... /var/lib/rpm/Basenames
c..... /var/lib/rpm/Dirnames
c..... /var/lib/rpm/Group
c..... /var/lib/rpm/Installtid
c..... /var/lib/rpm/Name
c..... /var/lib/rpm/Packages
c..... /var/lib/rpm/Providename
c..... /var/lib/rpm/Requirename
c..... /var/lib/rpm/Sha1header
c..... /var/lib/rpm/Sigmd5
```

3. 要显示某份文件的差异，请运行 `snapper diff PRE..POST 文件名`。如果没有指定 `文件名`，则会显示所有文件的差异。

```
tux > sudo snapper diff 350..351 /usr/share/fonts/truetype/fonts.scale
--- /.snapshots/350/snapshot/usr/share/fonts/truetype/fonts.scale
 2014-04-23 15:58:57.000000000 +0200
+++ /.snapshots/351/snapshot/usr/share/fonts/truetype/fonts.scale
 2014-05-07 16:46:31.000000000 +0200
@@ -1,4 +1,4 @@
-1174
+1486
 ds=y:ai=0.2:luximr.ttf -b&h-luxi mono-bold-i-normal--0-0-0-0-c-0-
iso10646-1
 ds=y:ai=0.2:luximr.ttf -b&h-luxi mono-bold-i-normal--0-0-0-0-c-0-iso8859-1
[...]
```

4. 要恢复一或多份文件，请运行 `snapper -v undochange` *PRE*..*POST* *文件名*。如果没有指定 *文件名*，则会恢复所有已更改的文件。

```
tux > sudo snapper -v undochange 350..351
     create:0 modify:13 delete:7
     undoing change...
     deleting /usr/share/doc/packages/mikachan-fonts
     deleting /usr/share/doc/packages/mikachan-fonts/COPYING
     deleting /usr/share/doc/packages/mikachan-fonts/dl.html
     deleting /usr/share/fonts/truetype/みかちゃん-p.ttf
     deleting /usr/share/fonts/truetype/みかちゃん-pb.ttf
     deleting /usr/share/fonts/truetype/みかちゃん-ps.ttf
     deleting /usr/share/fonts/truetype/みかちゃん.ttf
     modifying /usr/share/fonts/truetype/fonts.dir
     modifying /usr/share/fonts/truetype/fonts.scale
     modifying /var/cache/fontconfig/7ef2298fde41cc6eeb7af42e48b7d293-
x86_64.cache-4
     modifying /var/lib/rpm/Basenames
     modifying /var/lib/rpm/Dirnames
     modifying /var/lib/rpm/Group
     modifying /var/lib/rpm/Installtid
     modifying /var/lib/rpm/Name
     modifying /var/lib/rpm/Packages
     modifying /var/lib/rpm/Providename
     modifying /var/lib/rpm/Requirename
     modifying /var/lib/rpm/Sha1header
     modifying /var/lib/rpm/Sigmd5
     undoing change done
```

警告：还原用户添加

建议不要使用 Snapper 通过撤销更改的方式来还原用户添加。因为快照中排除了一些目录，文件系统中将保留属于用户的文件。如果使用已删除用户的 ID 创建新用户，该用户将继承原用户的文件。因此，强烈建议您使用 YaST 的用户和组管理工具来删除用户。

7.2.2 使用 Snapper 恢复文件

除了安装和管理快照之外，Snapper 还会创建时间线快照。您可以使用这些备份快照还原意外删除的文件或文件的前一个版本。您也可以使用 Snapper 的差异功能来确定在某个时间点所做的修改。

还原文件功能对于默认情况下不会生成快照的那些子卷或分区上的数据特别有用。例如，要从主目录还原文件，可以为自动生成时间线快照的 `/home` 创建单独的 Snapper 配置。有关指导，请参见第 7.5 节 “创建并修改 Snapper 配置”。

警告：恢复文件与回滚的比较

从根文件系统（由 Snapper 的根配置定义）生成的快照可用于执行系统回滚。进行此类回滚时，建议先从快照引导，然后再执行回滚。有关详细信息，请参见第 7.3 节 “通过从快照引导来执行系统回滚”。

从根文件系统快照还原所有文件（如下文中所述）也可以达到回滚的目的，但不建议这样做。您可以还原个别文件，例如还原 `/etc` 目录中的某个配置文件，但不能从快照中还原一系列文件。

此限制仅针对根文件系统的快照！

过程 7.3 使用 YAST SNAPPER 模块恢复文件

1. 从 YaST 中的其他部分或通过输入 yast2 snapper 来启动 `Snapper` 模块。
2. 选择要从中选择快照的当前配置。
3. 选择要用于恢复文件的时间线快照，并选择显示更改。时间线快照的类型为单一，描述为时间线。
4. 单击文件名从文本框中选择文件。系统会显示快照版本和当前系统之间的差异。勾选复选框来选中要恢复的文件。请对要恢复的所有文件执行此操作。
5. 单击恢复选定，然后单击是以确认该操作。

过程 7.4 使用 `snapper` 命令恢复文件

1. 运行以下命令来显示特定配置的一系列时间线快照：

```
tux > sudo snapper -c CONFIG list -t single | grep timeline
```

CONFIG 需要替换为现有的 Snapper 配置。使用 snapper list-configs 显示列表。

2. 运行以下命令来显示指定快照中发生更改的一系列文件：

```
tux > sudo snapper -c CONFIG status SNAPSHOT_ID..0
```

将 SNAPSHOT_ID 替换为您要用于恢复文件的快照的 ID。

3. （可选）运行以下命令列出当前文件版本与快照中文件版本的差异：

```
tux > sudo snapper -c CONFIG diff SNAPSHOT_ID..0 FILE NAME
```

如果不指定 <FILE NAME>，则会显示所有文件的差异。

4. 要恢复一个或多个文件，请运行

```
tux > sudo snapper -c CONFIG -v
 undochange SNAPSHOT_ID..0 FILENAME1 FILENAME2
```

如果没有指定文件名，则会恢复所有已更改的文件。

7.3 通过从快照引导来执行系统回滚

SUSE Linux Enterprise Server 上包含的 GRUB 2 版本可以从 Btrfs 快照进行引导。与 Snapper 的回滚功能相结合，就能恢复配置错误的系统。只有针对默认 Snapper 配置（ 根 ）创建的快照才可引导。

重要：受支持的配置

从 SUSE Linux Enterprise Server 12 SP5 开始，只有在根分区的默认子卷配置未更改过的情况下，才支持系统回滚。

引导快照时，快照中包含的文件系统部分会装载为只读状态；从快照中排除的所有其他文件系统和部分会加载为读写状态，并且可以修改。

重要：撤消更改与回滚的比较

通过快照来恢复数据时，必须知道，Snapper 可以处理两种完全不同的情形：

撤销更改

在如第 7.2 节 “使用 Snapper 撤销更改”中所述撤销更改时，系统会对两个快照进行比较，并还原两个快照之间的更改。通过这种方式可以明确指定不还原所选的文件。

回滚

在如下文所述进行回滚时，系统会重设置为生成快照时的状态。

要从可引导快照进行回滚，必须满足以下要求。执行默认安装时，系统会进行相应的设置。

从可引导快照进行回滚的要求

- 根文件系统必须是 Btrfs。不支持从 LVM 卷快照引导。
- 根文件系统必须在单个设备、单个分区和单个子卷上。`/srv` 等快照中排除的目录（参见第 7.1.2 节 “快照中排除的目录”了解完整列表）可以位于单独的分区。
- 系统应能够借助安装的引导加载程序进行引导。

要从可引导快照进行回滚，请执行以下操作：

1. 引导系统。在引导菜单中选择可引导快照，然后选择要引导的快照。快照会按日期顺序从近到远一一列出。
2. 登录系统。仔细检查是否一切运行正常。请注意，您无法对快照包含的任何目录进行写操作。写入其他目录的数据无论您下一步选择什么操作都不会丢失。
3. 根据您是否要执行回滚，选择下一步操作：
   a. 如果您不想对当前状态的系统执行回滚，请重引导进入当前的系统状态。然后，您便可选择另一个快照，或是启动救援系统。
   b. 要执行回滚，请运行

```
tux > sudo snapper rollback
```

然后重引导。在引导屏幕上，选择默认的引导项以重引导至恢复后的系统。系统即会创建回滚前文件系统状态的快照。根的默认子卷将替换为全新的读写快照。有关细节，请参见第 7.3.1 节 “回滚后的快照”。

通过 `-d` 选项添加快照的说明非常实用。例如：

```
New file system root since rollback on DATE TIME
```

提示：回滚到特定的安装状态

如果安装期间未禁用快照，将在初始系统安装结束时创建初始可引导快照。您随时可以通过引导此快照返回到该状态。该快照可通过 `after installation` 说明识别。

开始对服务包或新的主要版本进行系统升级时，也会创建可引导快照（前提是未禁用快照）。

7.3.1 回滚后的快照

在执行回滚之前，将会创建正在运行的文件系统的快照。快照说明会引用在回滚中恢复的快照的 ID。

对于通过回滚创建的快照，其 `Cleanup` 属性的值会设为 `number`。因此，回滚快照会在达到设置的快照数后自动删除。有关细节，请参见第 7.7 节 “自动清理快照”。如果快照包含重要数据，请在系统去除快照之前从快照中提取数据。

7.3.1.1 回滚快照示例

例如，在全新安装之后，系统上存在以下可用的快照：

```
root # snapper --iso list
Type   | # |     | Cleanup | Description           | Userdata
-------+---+ ... +---------+-----------------------+--------------
single | 0 |     |         | current               |
single | 1 |     |         | first root filesystem |
single | 2 |     | number  | after installation    | important=yes
```

运行 `sudo snapper rollback` 之后，将会创建快照 `3`，它包含执行回滚前系统的状态。快照 `4` 是新的默认 Btrfs 子卷，因此是重引导之后的系统。

```
root # snapper --iso list
Type   | # |     | Cleanup | Description           | Userdata
-------+---+ ... +---------+-----------------------+--------------
single | 0 |     |         | current               |
single | 1 |     | number  | first root filesystem |
single | 2 |     | number  | after installation    | important=yes
single | 3 |     | number  | rollback backup of #1 | important=yes
single | 4 |     |         |                       |
```

7.3.2 访问和识别快照引导项

要从快照引导，请重引导计算机并选择从只读的快照启动引导加载程序。一个屏幕即会打开，列出所有可引导的快照。最近的快照列在最前面，最旧的快照列在最后面。使用 ↓ 和 ↑ 导航，然后按 Enter 激活选定的快照。从引导菜单激活快照不会立即重引导计算机，而是打开选定快照的引导加载程序。

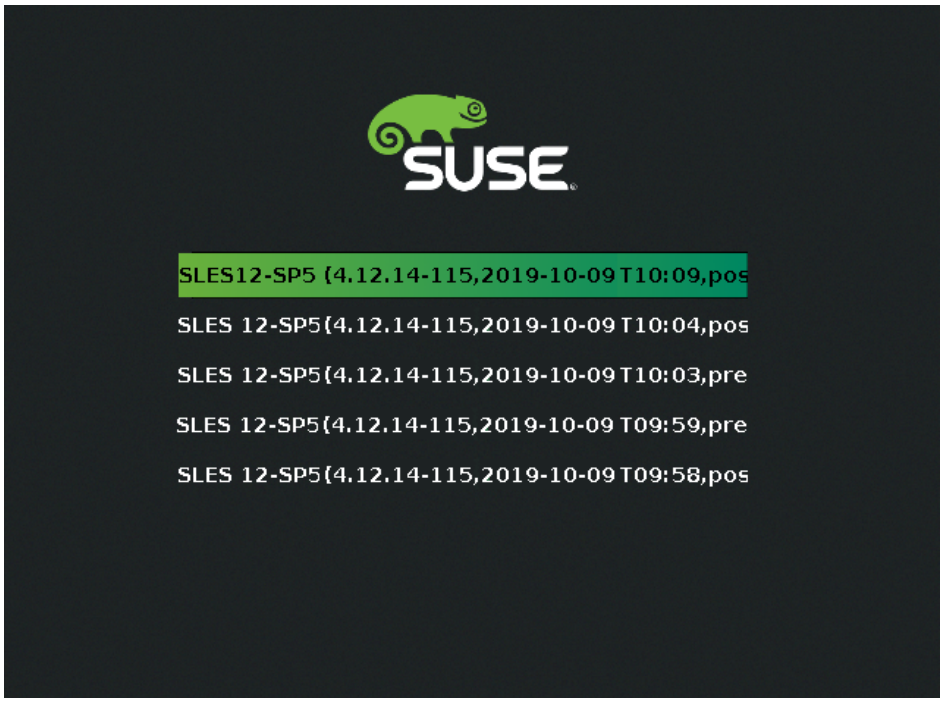


图 7.1 引导加载程序：快照

引导加载程序中的每个快照项遵循一种可方便您识别快照的命名模式：

```
[*]❶OS❷ (KERNEL❸,DATE❹TTIME❺,DESCRIPTION❻)
```

❶ 如果快照标记为 重要，该项将标有 * 号。

❷ 操作系统标签。

❹ 采用 YYYY-MM-DD 格式的日期。

❺ 采用 HH:MM 格式的时间。

❻ 此字段包含快照的说明。对于手动创建的快照，这是使用选项 --description 创建的字符串，或自定义字符串（请参见提示：为引导加载程序快照项设置自定义说明）。对于自动创建的快照，这是调用的工具，例如 zypp(zypper) 或 yast_sw_single。较长的说明可能会被截断，具体视引导屏幕的大小而定。

提示：为引导加载程序快照项设置自定义说明

可以将快照说明字段中的默认字符串替换为自定义字符串。例如，如果自动创建的说明不能提供充分的描述，或者用户提供的说明太长，这种做法将十分有用。要为快照 NUMBER 设置自定义字符串 STRING，请使用以下命令：

```
tux > sudo snapper modify --userdata "bootloader=STRING" NUMBER
```

说明的长度不应超过 25 个字符，超过此大小的任何内容都无法在引导屏幕上正常显示。

7.3.3 限制

不可能实现完整的系统回滚，即将整个系统恢复到生成快照时完全相同的状态。

7.3.3.1 快照中排除的目录

根文件系统快照并不包含所有目录。请参见第 7.1.2 节 “快照中排除的目录”了解详情和背后的原因。正因为此，这些目录中的数据并不会恢复，也就造成了以下限制。

回滚后，附加产品和第三方软件可能会无法使用

在快照中排除的子卷（如 `/opt`）上安装了数据的应用程序和附加产品，如果在快照中包含的子卷上也安装了部分应用程序数据，则回滚后，这些应用程序和附加产品将无法工作。要解决此问题，需要重新安装该应用程序或附加产品。

文件访问问题

如果某个应用程序在快照和当前系统之间更改了文件权限和/或所有权，回滚后，该应用程序可能无法访问这些文件。请在回滚后重设置受影响的文件的权限和/或所有权。

数据格式不兼容

如果服务或应用程序在快照和当前系统之间建立了新的数据格式，回滚后，该应用程序可能无法读取受影响的数据文件。

混合了代码和数据的子卷

诸如 `/srv` 之类的子卷可能同时包含代码和数据。回滚可能会导致代码失效。例如，降级 PHP 的版本可能会导致 Web 服务器的 PHP 脚本被破坏。

用户数据

如果回滚操作从系统中删除了用户，这些用户在快照中未包含的目录上所拥有的数据并不会删除。如果使用相同的用户 ID 创建新用户，该用户便会继承原用户的文件。请使用 `find` 之类的工具找到并删除孤立的文件。

7.3.3.2 不回滚引导装载程序数据

无法回滚引导装载程序，因为引导装载程序的所有“阶段”必须相互匹配。这在 `/boot` 回滚时并不能保证。

7.4 在用户主目录中启用 Snapper

您可以为用户的 `/home` 目录启用快照，以便为以下多个用例提供支持：

- 个人用户可以管理自己的快照和回滚。
- 系统用户，例如要跟踪配置文件、文档等数据副本的数据库、系统和网络管理员。
- Samba 与主目录和 Btrfs 后端进行共享。

每个用户的目录都是 /home 的 Btrfs 子卷。您可以手动设置快照（请参见第 7.4.3 节 “手动启用主目录中的快照”），但更便捷的方式是使用 pam_snapper。pam_snapper 包用于安装 pam_snapper.so 模块和帮助程序脚本，这些模块和脚本可自动创建用户和配置 Snapper。

pam_snapper 提供与 useradd 命令、可插入身份验证模块 (PAM) 及 Snapper 的集成。默认情况下，它会在用户登录和注销时创建快照；当某些用户在很长一段时间内都处于登录状态时，它还会创建基于时间的快照。您可以使用常规 Snapper 命令和配置文件更改默认值。

7.4.1 安装 pam_snapper 和创建用户

最简单的方法是使用格式化为 Btrfs 的新 /home 目录且在没有现有用户的情况下开始。安装 pam_snapper：

```
root # zypper in pam_snapper
```

将下面一行添加到 /etc/pam.d/common-session 中：

```
session optional pam_snapper.so
```

使用 /usr/lib/pam_snapper/pam_snapper_useradd.sh 脚本创建新用户和主目录。该脚本默认会执行试运行。请编辑脚本，将 DRYRUN=1 更改为 DRYRUN=0。现在，您便可以创建新用户：

```
root # /usr/lib/pam_snapper/pam_snapper_useradd.sh \
username group passwd=password
Create subvolume '/home/username'
useradd: warning: the home directory already exists.
Not copying any file from skel directory into it.
```

用户首次登录时，/etc/skel 中的文件将会复制到其主目录。列出您的 Snapper 配置，以校验是否已创建用户的配置：

```
root # snapper list --all
Config: home_username, subvolume: /home/username
Type   | # | Pre # | Date | User | Cleanup | Description | Userdata
-------+---+-------+------+------+---------+-------------+---------
```

```
single | 0 |        |       | root |          | current      |
```

一段时间过后，此输出中将会填充快照列表，用户可以使用标准 Snapper 命令对其进行管理。

7.4.2 去除用户

使用 `/usr/lib/pam_snapper/pam_snapper_userdel.sh` 脚本去除用户。该脚本默认会执行试运行，请编辑脚本，将 `DRYRUN=1` 更改为 `DRYRUN=0`。此操作将去除用户、用户的主子卷、Snapper 配置，并会删除所有快照。

```
root # /usr/lib/pam_snapper/pam_snapper_userdel.sh username
```

7.4.3 手动启用主目录中的快照

使用 Snapper 手动设置用户主目录的步骤如下。`/home` 必须已格式化为 Btrfs，并且目前尚未创建用户。

```
root # btrfs subvol create /home/username
root # snapper -c home_username create-config /home/username
root # sed -i -e "s/ALLOW_USERS=\"\"/ALLOW_USERS=\"username\"/g" \
/etc/snapper/configs/home_username
root # yast users add username=username home=/home/username password=password
root # chown username.group /home/username
root # chmod 755 /home/username/.snapshots
```

7.5 创建并修改 Snapper 配置

每一个分区或 `Btrfs` 子卷都有一个专用的配置文件用于定义 Snapper 的行为方式。这些配置文件位于 `/etc/snapper/configs/` 下。

如果根文件系统足够大（大约有 12 GB），安装时将自动对根文件系统 `/` 启用快照。相应的默认配置命名为 `root`。该配置可创建和管理 YaST 及 Zypper 快照。有关默认值列表，请参见第 7.5.1.1 节 “配置数据”。

注意：启用快照所需的最小根文件系统大小

如第 7.1 节 “默认设置”中所述，要启用快照，根文件系统中需要有额外的可用空间。所需空间取决于所安装的包数量以及快照中包括的卷更改量，另外还取决于快照频率和存档的快照数。

要在安装期间自动启用快照，需要满足最小根文件系统大小。此大小约为 12 GB。将来，这个值可能会发生变化，具体视基础系统的体系结构和大小而定。它取决于安装媒体内 `/control.xml` 文件中以下标记的值：

```
<root_base_size>
<btrfs_increase_percentage>
```

该值通过下面的公式计算得出：`ROOT_BASE_SIZE` * (1 + `BTRFS_INCREASE_PERCENTAGE`/100)

请记住，此值是最小大小。请考虑分给根文件系统更多空间。一般而言，两倍于未启用快照时将使用的大小即可。

您可以为使用 `Btrfs` 格式化的其他分区或 `Btrfs` 分区上的现有子卷创建自己的配置。在以下示例中，我们将设置 Snapper 配置，以便对驻留在单独的、以 `Btrfs` 格式化且安装点为 `/srv/www` 的分区的 Web 服务器数据进行备份。

创建配置后，您可以直接使用 `snapper`，也可以使用 YaST Snapper 模块，从这些快照恢复文件。在 YaST 中，您需要选择您的当前配置，同时还需要使用全局开关 `-c` 指定 snapper 的配置（例如 snapper `-c myconfig` list）。

要创建新的 Snapper 配置，请运行 `snapper create-config`：

```
tux > sudo snapper -c www-data❶ create-config /srv/www❷
```

❶ 配置文件的名称。

❷ 要生成快照的分区或 `Btrfs` 子卷的装入点。

此命令将使用合理的默认值（取自 `/etc/snapper/config-templates/default`）创建新的配置文件 `/etc/snapper/configs/www-data`。有关如何调整这些值的指导，请参考第 7.5.1 节 “管理现有配置”。

提示：配置的默认值

新配置的默认值取自 `/etc/snapper/config-templates/default`。要使用自己的一组默认值，请在相同的目录中创建此文件的副本然后按照需要进行调整。要使用此功能，请在 create-config 命令中指定 `-t` 选项：

```
tux > sudo snapper -c www-data create-config -t MY_DEFAULTS /srv/www
```

7.5.1 管理现有配置

`snapper` 有多个子命令可用于管理现有的配置。您可以列出、显示这些配置，也可以对它们进行删除和修改：

列出配置

使用 `snapper list-configs` 命令可以显示所有现有的配置：

```
tux > sudo snapper list-configs
Config | Subvolume
-------+----------
root   | /
usr    | /usr
local  | /local
```

显示配置

使用 `snapper -c CONFIG` get-config 子命令可以显示指定的配置。`CONFIG` 应替换为执行 `snapper list-configs` 命令后所显示的某个配置名称。请参见第 7.5.1.1 节 “配置数据”以了解有关配置选项的更多信息。

要显示默认配置，请运行

```
tux > sudo snapper -c root get-config
```

修改配置

使用 `snapper -c CONFIG set-config OPTION=VALUE` 子命令可以修改指定配置中的选项。`CONFIG` 应替换为执行 `snapper list-configs` 命令后所显示的某个配置名称。`OPTION` 和 `VALUE` 的可能值可参见第 7.5.1.1 节 “配置数据”。

删除配置

使用 `snapper -c CONFIG` delete-config 子命令可以删除配置。`CONFIG` 应替换为执行 `snapper list-configs` 命令后所显示的某个配置名称。

7.5.1.1 配置数据

每个配置都包含一系列选项，这些选项可以通过命令行进行修改。下面的列表提供了每个选项的细节。要更改某个值，请运行 `snapper -c CONFIG set-config "KEY=VALUE"`。

`ALLOW_GROUPS`、`ALLOW_USERS`

授予普通用户使用快照的权限。有关详细信息，请参见第 7.5.1.2 节 “以普通用户身份使用 Snapper”。

默认值是 `""`。

`BACKGROUND_COMPARISON`

定义在创建前后快照后是否应在后台对它们进行比较。

默认值为 `"yes"`。

`EMPTY_*`

为前后快照相同的快照对定义清理算法。有关详细信息，请参见第 7.7.3 节 “清理没有差异的快照对”。

`FSTYPE`

分区的文件系统类型。不更改。

默认值为 `"btrfs"`。

`NUMBER_*`

为安装快照与管理快照定义清理算法。有关详细信息，请参见第 7.7.1 节 “清理编号快照”。

`QGROUP` / `SPACE_LIMIT`

将定额支持添加到清理算法。有关详细信息，请参见第 7.7.5 节 “添加磁盘定额支持”。

`SUBVOLUME`

分区或子卷生成快照的安装点。不更改。

默认值是 `"/"`.

SYNC_ACL

如果普通用户要使用 Snapper（请参见第 7.5.1.2 节 “以普通用户身份使用 Snapper”），他们必须能访问 .snapshot 目录，并且能读取其中的文件。

如果 SYNC_ACL 设置为 yes，Snapper 会通过 ACL 自动允许 ALLOW_USERS 和 ALLOW_GROUPS 项指定的用户和组访问这些目录及其中的文件。

默认值为 "no"。

TIMELINE_CREATE

如果设置为 yes，便会每小时创建一个快照。有效值：yes、no.

默认值为 "no"。

TIMELINE_CLEANUP / TIMELINE_LIMIT_*

为时间线快照定义清理算法。有关详细信息，请参见第 7.7.2 节 “清理时间线快照”。

7.5.1.2 以普通用户身份使用 Snapper

默认情况下，Snapper 只能由 root 用户使用。但在特定情况下，某些组或用户也需要创建快照或通过还原至快照来撤销更改：

- 想要为 /srv/www 生成快照的网站管理员
- 想要为自己的主目录生成快照的用户

此类情况下，可以创建为用户和（或）组授予权限的 Snapper 配置。指定的用户必须能连接并访问相应的 .snapshots 目录。要实现这一点，最简单的方法是将 SYNC_ACL 选项设置为 yes。

过程 7.5 让普通用户可以使用 SNAPPER

请注意，此过程中的所有步骤都需要由 root 用户运行。

1. 如果不存在，则请为用户可以使用 Snapper 的分区或子卷创建 Snapper 配置。有关指导，请参见第 7.5 节 “创建并修改 Snapper 配置”。示例：

```
tux > sudo snapper --config web_data create /srv/www
```

2. 在 /etc/snapper/configs/CONFIG 下创建配置文件，其中“CONFIG”是您在上一步中使用 -c/--config 指定的值（例如 /etc/snapper/configs/web_data）。按照需要进行调整；有关详细信息，请参见第 7.5.1 节 “管理现有配置”。

3. 为 ALLOW_USERS 和（或）ALLOW_GROUPS 设置值，以分别为用户和（或）组授予权限。多个条目需要使用 Space 分隔。例如，要为用户 www_admin 授予权限，可运行：

```
tux > sudo snapper -c web_data set-config "ALLOW_USERS=www_admin"
 SYNC_ACL="yes"
```

4. 此时，指定的用户和（或）组便可以使用指定的 Snapper 配置。您可以使用 list 命令对其进行测试，例如：

```
www_admin:~ > snapper -c web_data list
```

7.6 手动创建和管理快照

Snapper 的功能并不仅限于根据配置自动创建和管理快照；您还可以使用命令行工具或 YaST 模块手动创建快照对（“前快照和后快照”）或单一快照。

所有 Snapper 操作皆针对现有配置执行（有关详细信息，请参见第 7.5 节 “创建并修改 Snapper 配置”）。您可以只为存在配置的分区或卷生成快照。默认情况下使用系统配置（root）。如果想要为自己的配置创建或管理快照，则需要进行明确选择。使用 YaST 中的当前配置下拉框，或在命令行上指定 -c（即 snapper -c 我的配置 命令）。

7.6.1 快照元数据

每一张快照均由快照本身以及一些元数据组成。创建快照时，您还需要指定元数据。修改快照就意味着更改其元数据——您无法修改其内容。使用 snapper list 可显示现有快照及其元数据：

`snapper --config home list`

列出配置 home 的快照。要列出默认配置 (root) 的快照，请使用 snapper -c root list 或 snapper list。

`snapper list -a`

列出所有现有配置的快照。

`snapper list -t pre-post`

列出默认（root）配置的所有前快照和后快照对。

`snapper list -t single`

列出默认 (`root`) 配置的所有 单一 类型的快照。

每一张快照可以使用以下元数据：

- 类型：快照类型，有关详细信息，请参见第 7.6.1.1 节 “快照类型”。不能更改此数据。
- 编号：快照的唯一编号。不能更改此数据。
- 前编号：指定相应前快照的编号。仅适用于后类型。不能更改此数据。
- 说明：快照的说明。
- 用户数据：扩展的说明。您可使用逗号分隔的“键=值”列表格式指定自定义数据：`reason=testing, project=foo`。此字段也可用于将快照标记为重要 (`important=yes`) 以及列出创建快照的用户 (user=tux)。
- 清理算法：快照的清理算法。有关详细信息，请参见第 7.7 节 “自动清理快照”。

7.6.1.1 快照类型

Snapper 能够分清三种不同类型的快照：前快照、后快照以及单一快照。从物理上讲，这三种快照没有什么不同，但 Snapper 会针对不同类型采用不同的处理方式。

前

修改前的文件系统快照。每一张 前 快照都有一个对应的 后 快照。目的之一就是为了自动创建 YaST/Zypper 快照。

后

修改后的文件系统快照。每一张 后 快照都有一个对应的 前 快照。目的之一就是为了自动创建 YaST/Zypper 快照。

单一

独立的快照。目的之一就是为了自动创建每小时快照。此为创建快照时的默认类型。

7.6.1.2 清理算法

Snapper 提供有三种清理旧快照的算法。这些算法在日常的 cron 作业中执行。您可以定义要在 Snapper 配置中保留的不同类型的快照数（有关细节，请参见第 7.5.1 节 “管理现有配置”）。

数量
: 当达到某一快照计数时将删除旧快照。

时间线
: 将删除超过一定时限的旧快照，但保留若干个每小时、每天、每月和每年快照。

无差异前后快照对
: 将删除无差异的前后快照对。

7.6.2 创建快照

通过运行 `snapper create` 或单击 YaST 的 Snapper 模块中的创建来创建快照。以下示例解释了如何从命令行创建快照。使用 YaST 界面会比较简单。

提示：快照说明

为了便于日后确定快照的用途，您应始终指定有意义的说明。甚至可以通过用户数据选项指定更多信息。

`snapper create --description "2014 年第二周快照"`
: 创建默认（`root`）配置的独立快照（单一类型）并附加说明。因为没有指定清理算法，将不会自动删除快照。

`snapper --config home create --description "~tux 中清理"`
: 为名为 `home` 的自定义配置创建独立快照（单一类型）并附加说明。因为没有指定清理算法，将不会自动删除快照。

`snapper --config home create --description "每日数据备份" --cleanup-algorithm timeline`>
: 为名为 `home` 的自定义配置创建独立快照（单一类型）并附加说明。一旦符合为配置中的时间线清理算法指定的条件，便会自动删除文件。

`snapper create --type pre--print-number--description "Apache 配置清理之前"--userdata "important=yes"`

创建 前 类型的快照并打印快照编号。创建用于保存“之前”和“之后”状态的快照对所需的首个命令。该快照标记为重要。

`snapper create --type post--pre-number 30--description "Apache 配置清理之后"--userdata "important=yes"`

创建 后 类型的快照且其对应的 前 快照编号为 30。创建用于保存“之前”和“之后”状态的快照对所需的第二个命令。该快照标记为重要。

`snapper create --command COMMAND--description "命令前后"`

运行 命令 前后自动创建快照对。此选项仅在于命令行上使用 snapper 时可用。

7.6.3 修改快照元数据

Snapper 允许您修改说明、清理算法及快照的用户数据。其他元数据均无法更改。以下示例解释了如何从命令行修改快照。使用 YaST 界面会比较简单。

要在命令行上修改快照，您需要知道其编号。使用 `snapper list` 可显示所有快照及其编号。

YaST 的 Snapper 模块已列出所有快照。从列表中选择一个快照，然后单击修改。

`snapper modify --cleanup-algorithm "timeline"` 10

修改默认 (`root`) 配置的第 10 张快照的元数据。清理算法设置为 `timeline`。

`snapper --config home modify --description "每日备份" -cleanup-algorithm "timeline"120`

修改名为 `home` 的自定义配置的第 120 张快照的元数据。将设置新的说明并取消设置清理算法。

7.6.4 删除快照

要使用 YaST 的 Snapper 模块删除快照，请从列表中选择快照，然后单击删除。

要使用命令行工具删除快照，需要知道其编号。运行 `snapper list` 命令获取快照编号。要删除快照，请运行 `snapper delete` 编号。

不允许删除当前的默认子卷快照。

使用 Snapper 删除快照时，在后台运行的 Btrfs 进程将会回收释放的空间。因此，可用空间的可见性与可用性会延迟。如果您希望在删除快照后立即可以使用释放的空间，请结合选项 `--sync` 使用 delete 命令。

提示：删除快照对

删除 前 快照时，您应始终删除与其对应的 后 快照（反之亦然）。

`snapper delete 65`

删除默认（`root`）配置的第 65 张快照。

`snapper -c home delete 89 90`

删除名为 `home` 的自定义配置的第 89 张和第 90 张快照。

`snapper delete --sync 23`

删除默认（`root`）配置的快照 23，并使释放的空间立即可用。

提示：删除未参照的快照

有时，虽然 Btrfs 快照存在，但却缺少包含 Snapper 元数据的 XML 文件。这种情况表示快照对 Snapper 不可见，需要手动将其删除：

```
btrfs subvolume delete /.snapshots/SNAPSHOTNUMBER/snapshot
rm -rf /.snapshots/SNAPSHOTNUMBER
```

提示：旧快照占用的磁盘空间更多

如果您要删除快照以释放硬盘上的空间，请务必先删除旧快照。快照生成的时间越长，其占用的空间就越大。

也可以通过日常的 cron 作业自动删除快照。有关详细信息，请参见第 7.6.1.2 节 “清理算法”。

7.7 自动清理快照

一段时间过后，快照大小将变大，占用的磁盘空间不断增加。为了防止磁盘上的空间耗尽，Snapper 提供了用于自动删除旧快照的算法。这些算法根据时间线快照和编号快照（管理快照与安装快照对）而有所不同。您可以指定要为每种类型保留的快照数。

除此之外，您可以选择指定一个磁盘空间定额，用于定义快照可占用的最大磁盘空间大小。系统还可以自动删除前快照与后快照没有任何不同的快照对。

清理算法始终绑定到单个 Snapper 配置，因此您需要为每个配置指定算法。要防止自动删除特定的快照，请参见问：。

默认设置（`root`）配置为清理编号快照以及空的前快照与后快照对。默认设置中会启用定额支持，并且快照必须在根分区上至少留出 20% 的可用磁盘空间。时间线快照默认处于禁用状态，因此，时间线清理算法也处于禁用状态。

注意：改进的清理算法

之前实施的清理算法仅可确保快照使用的磁盘空间未超过指定数量的磁盘空间（默认值为 50%）。在某些情况下，这无法防止系统耗尽磁盘空间。自 SUSE Linux Enterprise Server 12 SP1 起，Snapper 的清理算法进行了改进，在任何情况下都可保留 20% 的可用磁盘空间。

7.7.1 清理编号快照

编号快照（管理快照与安装快照对）的清理由 Snapper 配置的以下参数控制。

`NUMBER_CLEANUP`

启用或禁用安装快照与管理快照对的清理。如果启用该参数，则当快照总数超过 `NUMBER_LIMIT` 和/或 `NUMBER_LIMIT_IMPORTANT` 指定的数字以及 `NUMBER_MIN_AGE` 指定的时限时，将删除快照对。有效值：`yes`（启用）、`no`（禁用）。

默认值为 `"yes"`。

用于更改或设置值的示例命令：

```
tux > sudo snapper -c CONFIG set-config "NUMBER_CLEANUP=no"
```

NUMBER_LIMIT / NUMBER_LIMIT_IMPORTANT

定义要保留多少个普通和/或重要安装快照与管理快照对。所保留的会是最新的那些快照。如果 NUMBER_CLEANUP 设置为 "no"，则会忽略此参数。

NUMBER_LIMIT 的默认值为 "2-10"，NUMBER_LIMIT_IMPORTANT 的默认值为 "4-10"。

用于更改或设置值的示例命令：

```
tux > sudo snapper -c CONFIG set-config "NUMBER_LIMIT=10"
```

重要：范围值与常量值的比较

如果启用定额支持（请参见第 7.7.5 节 “添加磁盘定额支持”），则需要将限制指定为最小值-最大值范围，例如 2-10。如果禁用定额支持，则需要提供常量值，例如 10，否则清理将会失败并返回错误。

NUMBER_MIN_AGE

定义快照在自动删除前必须保留的最小时限（以秒为单位）。保留时间小于此处指定值的快照不会删除，不管这样的快照有多少。

默认值为 "1800"。

用于更改或设置值的示例命令：

```
tux > sudo snapper -c CONFIG set-config "NUMBER_MIN_AGE=864000"
```

注意：限制和时限

NUMBER_LIMIT、NUMBER_LIMIT_IMPORTANT 和 NUMBER_MIN_AGE 始终都会评估。只有同时符合全部条件才会删除快照。

如果您希望不考虑时限而始终保留 NUMBER_LIMIT* 所定义数量的快照，可将 NUMBER_MIN_AGE 设置为 0。

下面的示例显示了保留最近 10 个重要和 10 个普通快照（不论保留期限）的配置：

```
NUMBER_CLEANUP=yes
NUMBER_LIMIT_IMPORTANT=10
NUMBER_LIMIT=10
NUMBER_MIN_AGE=0
```

另外，如果不想保留超过一定时限的快照，可将 NUMBER_LIMIT* 设置为 0，并用 NUMBER_MIN_AGE 指定时限。

下面的示例显示了只保留十天以内的快照的配置：

```
NUMBER_CLEANUP=yes
NUMBER_LIMIT_IMPORTANT=0
NUMBER_LIMIT=0
NUMBER_MIN_AGE=864000
```

7.7.2 清理时间线快照

时间线快照的清理由 Snapper 配置的以下参数控制。

TIMELINE_CLEANUP

启用或禁用时间线快照的清理。如果启用该参数，则当快照总数超过 TIMELINE_LIMIT_* 指定的数字以及 TIMELINE_MIN_AGE 指定的时限时，将删除快照。有效值：yes、no.

默认值为 "yes"。

用于更改或设置值的示例命令：

```
tux > sudo snapper -c CONFIG set-config "TIMELINE_CLEANUP=yes"
```

TIMELINE_LIMIT_DAILY、TIMELINE_LIMIT_HOURLY、TIMELINE_LIMIT_MONTHLY、TIMELINE_LIMIT_WEEKLY、TIMELINE_LIMIT_YEARLY

按小时、天、月、周和年保留的快照数量。

每项的默认值为 "10"，但 TIMELINE_LIMIT_WEEKLY 除外，该参数默认设置为 "0"。

TIMELINE_MIN_AGE

定义快照在自动删除前必须保留的最小时限（以秒为单位）。

默认值为 "1800"。

例 7.1 时间线配置示例

```
TIMELINE_CLEANUP="yes"
```

```
TIMELINE_CREATE="yes"
TIMELINE_LIMIT_DAILY="7"
TIMELINE_LIMIT_HOURLY="24"
TIMELINE_LIMIT_MONTHLY="12"
TIMELINE_LIMIT_WEEKLY="4"
TIMELINE_LIMIT_YEARLY="2"
TIMELINE_MIN_AGE="1800"
```

此示例配置能够实现按小时生成将自动清理的快照。`TIMELINE_MIN_AGE` 和 `TIMELINE_LIMIT_*` 始终会一起评估。在本示例中，快照删除前的最小保留时限设置为 30 分钟（1800 秒）。因为我们会每小时创建一次快照，所以确保了只会保留最近的快照。如果 `TIMELINE_LIMIT_DAILY` 设置为非零值，则表示还会保留当天的首张快照。

快照保留

- 每小时：最近创建的 24 个快照。
- 每天：保留最近 7 天内每天创建的首个快照。
- 每月：保留最近 12 个月内每月的最后一天创建的首个快照。
- 每周：保留最近 4 周内每周最后一天创建的首个快照。
- 每年：保留最近 2 年内每年最后一天创建的首个快照。

7.7.3 清理没有差异的快照对

如第 7.1.1 节 “快照类型”中所述，每当您运行 YaST 模块或执行 Zypper 时，将在启动时创建一个前快照，在退出时创建一个后快照。如果您未进行过任何更改，则前快照与后快照没有差异。在 Snapper 配置中设置以下参数可自动删除此类“空”快照对：

`EMPTY_PRE_POST_CLEANUP`

如果设置为 `yes`，系统会删除前后快照相同的快照对。

默认值为 `"yes"`。

`EMPTY_PRE_POST_MIN_AGE`

定义前后快照相同的快照对在自动删除之前必须保留的最短时限（以秒为单位）。

默认值为 `"1800"`。

7.7.4 清理手动创建的快照

Snapper 未针对手动创建的快照提供自定义清理算法。但是，您可以向手动创建的快照指派 number 或 timeline 清理算法。如此，该快照将加入您所指定的算法的“清理队列”。可以在创建快照时或通过修改现有快照来指定清理算法：

`snapper create --description "Test" --cleanup-algorithm number`

为默认 (root) 配置创建独立快照（单一类型）并指派 `number` 清理算法。

`snapper modify --cleanup-algorithm "timeline" 25`

使用数字 25 修改快照，并指派 `timeline` 清理算法。

7.7.5 添加磁盘定额支持

除了上述 number 和/或 timeline 清理算法外，Snapper 还支持定额。您可以定义允许快照占用的可用空间百分比。此百分比值始终适用于相应 Snapper 配置中定义的 Btrfs 子卷。

如果在安装期间启用了 Snapper，则会自动启用定额支持。如果您是在安装后的某个时间手动启用 Snapper 的，则可以通过运行 `snapper setup-quota` 来启用定额支持。这需要您具有有效的配置（有关详细信息，请参见第 7.5 节 “创建并修改 Snapper 配置”）。

定额支持由 Snapper 配置的以下参数控制。

`QGROUP`

Snapper 使用的 Btrfs 定额组。如果未设置，请运行 `snapper setup-quota`。如果已设置，则仅在您熟悉 `man 8 btrfs-qgroup` 的情况下才可对其进行更改。此值是使用 `snapper setup-quota` 设置的，不应更改。

`SPACE_LIMIT`

允许快照使用的空间限制，以 1 (100%) 的分数表示。有效值范围为 0 到 1（0.1 = 10%，0.2 = 20%...）。

需遵守以下限制和指导原则：

- 只能在已激活现有 number 和/或 timeline 清理算法的前提下才能激活定额。如果未激活任何清理算法，则无法应用定额限制。
- 启用定额支持后，Snapper 将根据需要执行两轮清理。第一轮清理将应用针对编号快照和时间线快照指定的规则。仅当完成这一轮清理后超出定额时，在第二轮清理中才会应用定额特定的规则。
- 即使启用了定额支持，Snapper 也始终会保留 `NUMBER_LIMIT*` 和 `TIMELINE_LIMIT*` 值指定的快照数，而不管是否超出了定额。因此，建议为 `NUMBER_LIMIT*` 和 `TIMELINE_LIMIT*` 指定范围值 (`MIN-MAX`)，以确保可以应用定额。
  例如，如果设置了 `NUMBER_LIMIT=5-20`，Snapper 将执行第一轮清理，并将普通的编号快照数量减至 20 个。如果这 20 个快照超出定额，Snapper 将在第二轮清理中删除最旧的快照，直到符合定额限制。系统始终会至少保留 5 个快照，不管这些快照占用了多少空间。

7.8 常见问题

问： 为什么 Snapper 从不显示 `/var/log`、`/tmp` 以及其他目录中的更改？

答： 因为我们将部分目录排除在了快照之外。详细的列表和具体原因请参见第 7.1.2 节 “快照中排除的目录”。为了将路径从快照中排除，我们为该路径创建了子卷。

问： 快照要占用多少磁盘空间？如何释放磁盘空间？

答： 目前，`Btrfs` 工具还不支持显示分配给快照的磁盘空间大小。不过，如果启用了定额，便可确定在删除所有快照后将会释放多少空间：

1. 获取定额组 ID（在下面的示例中为 `1/0`）：

```
tux > sudo snapper -c root get-config | grep QGROUP
QGROUP                  | 1/0
```

2. 重新扫描子卷定额：

```
tux > sudo btrfs quota rescan -w /
```

3. 显示定额组（在下面的示例中为 `1/0`）的数据：

```
tux > sudo btrfs qgroup show / | grep "1/0"
1/0           4.80GiB    108.82MiB
```

第三列显示删除所有快照后将释放的空间（108.82MiB）。

为了释放包含快照的 Btrfs 分区的空间，您需要删除不需要的快照，而不是文件。旧快照比新快照占用的磁盘空间更多。有关详细信息，请参见第 7.1.3.4 节 “控制快照存档”。

升级服务包时，由于会更改大量数据（包更新），将导致快照占用大量系统子卷的磁盘空间。因此对于不再需要的快照，建议手动删除。有关详细信息，请参见第 7.6.4 节 “删除快照”。

问： 我可以从引导加载程序引导快照吗？

答： 可以。有关细节，请参考第 7.3 节 “通过从快照引导来执行系统回滚”。

问： 如何永久保留快照？

答： 目前，Snapper 尚无防止手动删除快照的功能。不过，您可以防止清理算法自动删除快照。除非您使用 --cleanup-algorithm 指定了清理算法，否则不会为手动创建的快照（请参见第 7.6.2 节 “创建快照”）指派清理算法。自动创建的快照始终会被指派 number 或 timeline 算法。要从一个或多个快照去除此类指派，请执行以下操作：

1. 列出所有可用快照：

```
tux > sudo snapper list -a
```

2. 记住您要防止删除的快照数。

3. 运行以下命令并将数字占位符替换为您记住的数字：

```
tux > sudo snapper modify --cleanup-algorithm "" #1 #2 #n
```

4. 再次运行 snapper list -a 检查结果。在 Cleanup 列中，您修改的快照所对应的项现在应该为空。

问： 何处能获得有关 Snapper 的详细信息？

答： 请访问 Snapper 的主页，网址为：http://snapper.io/。

8 使用 VNC 远程访问

利用虚拟网络计算（Virtual Network Computing，VNC）可以通过图形桌面控制远程计算机（与远程外壳访问相对）。VNC 是独立于平台的并允许您从任何操作系统访问远程计算机。

SUSE Linux Enterprise Server 支持两种不同种类的 VNC 会话：自客户端启动起在 VNC 连接期间“在线”的一次性会话和始终“在线”直到被明确终止的永久会话。

注意：会话类型

一台计算机可在不同端口上同时提供两种会话，但当会话打开后不能从一种类型转换为另一种类型。

8.1 `vncviewer` 客户端

要连接到服务器提供的 VNC 服务，需要使用客户端。SUSE Linux Enterprise Server 中的默认客户端是 `tigervnc` 包提供的 `vncviewer`。

8.1.1 使用 vncviewer CLI 进行连接

要启动 VNC 查看器并发起与服务器的会话，请使用以下命令：

```
tux > vncviewer jupiter.example.com:1
```

若不使用 VNC 显示器编号，您也可以指定带两个冒号的端口号：

```
tux > vncviewer jupiter.example.com::5901
```

注意：显示号和端口号

您在 VNC 客户端中指定的实际显示号或端口号必须与在目标计算机上通过 `vncserver` 命令提取的显示号或端口号相同。有关更多信息，请参见第 8.4 节 “持续 VNC 会话”。

8.1.2 使用 vncviewer GUI 进行连接

在不指定 `--listen` 或要连接的主机的情况下运行 `vncviewer` 会显示一个窗口，要求您输入连接细节。按第 8.1.1 节 “使用 vncviewer CLI 进行连接”中所述在 VNC 服务器字段中输入主机，然后单击连接。

图 8.1 VNCVIEWER

8.1.3 未加密连接通知

VNC 协议支持不同类型的加密连接，请不要将这些连接与口令身份验证相混淆。如果某个连接未使用 TLS，VNC 查看器的窗口标题中可能会出现“（连接未加密！）”文本。

8.2 Remmina：远程桌面客户端

Remmina 是功能丰富的新式远程桌面客户端。它支持多种访问方法，例如 VNC、SSH、RDP 或 Spice。

8.2.1 安装

要使用 Remmina，请检查系统上是否安装了 `remmina` 包，如未安装，请加以安装。记得还要安装适用于 Remmina 的 VNC 插件：

```
root # zypper in remmina remmina-plugin-vnc
```

8.2.2 主窗口

通过输入 `remmina` 命令运行 Remmina。

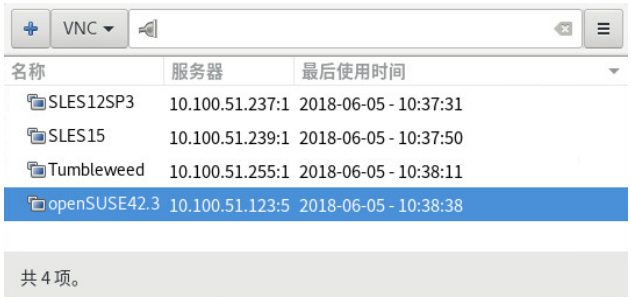


图 8.2 REMMINA 的主窗口

该应用程序主窗口显示储存的远程会话列表。在这里，您可以添加和保存新远程会话、快速启动新会话而不保存、启动以前保存的会话，或设置 Remmina 的全局首选项。

8.2.3 添加远程会话

要添加和保存新远程会话，请单击主窗口左上方的 。远程桌面首选项窗口即会打开。

图 8.3 远程桌面首选项

在用于指定新添加的远程会话配置文件的字段中填写信息。最重要的技术包括：

名称
: 配置文件的名称，将列于主窗口中。

协议
: 连接到远程会话时要使用的协议，例如 VNC。

服务器
: 远程服务器的 IP 或 DNS 地址和显示号。

用户名、口令
: 要用于进行远程身份验证的身份凭证。保留为空表示不进行身份验证。

色深、质量
: 根据连接速度和质量选择最佳选项。

选择高级选项卡可输入更具体的设置。

提示：禁用加密

如果客户端与远程服务器之间的通讯不加密，请激活禁用加密，否则连接会失败。

选择 SSH 选项卡可显示高级 SSH 隧道通讯进程和身份验证选项。

单击保存进行确认。新设置的配置文件将列在主窗口中。

8.2.4 启动远程会话

您可以启动以前保存的会话，也可以快速启动一个远程会话而不保存连接细节。

8.2.4.1 快速启动远程会话

要快速启动远程会话而不添加并保存连接细节，请使用主窗口顶部的下拉框和文本字段。

图 8.4 快速启动

从下拉框中选择通讯协议（例如 VNC），然后输入 VNC 服务器的 DNS 或 IP 地址，后跟一个冒号和显示号，然后按 Enter 确认。

8.2.4.2 打开保存的远程会话

要打开特定的远程会话，请从会话列表中双击该会话。

8.2.4.3 远程会话窗口

远程会话会在新窗口的标签中打开。每个标签托管一个会话。窗口左侧的工具栏可用来管理窗口/会话，例如切换全屏模式、调整窗口大小以适应会话的显示大小、将特定按键发送到会话、对会话进行屏幕截图，或设置图像质量。

图 8.5 正查看 SLES 15 远程会话的 REMMINA

8.2.5 编辑、复制和删除保存的会话

要编辑保存的某个远程会话，请在 Remmina 的主窗口中右键单击其名称，然后选择编辑。有关相关字段的说明，请参见第 8.2.3 节 “添加远程会话”。

要复制保存的某个远程会话，请在 Remmina 的主窗口中右键单击其名称，然后选择复制。在远程桌面首选项窗口中，更改配置文件的名称，（可选）调整相关选项，然后单击保存确认。

要删除保存的某个远程会话，请在 Remmina 的主窗口中右键单击其名称，然后选择删除。在下一个对话框中，单击是确认。

8.2.6 从命令行运行远程会话

如果您需要从命令行或使用批处理文件打开远程会话，而不先打开应用程序主窗口，请使用以下语法：

```
tux > remmina -c profile_name.remmina
```

Remmina 的配置文件储存在您主目录下的 `.local/share/remmina/` 目录中。要确定哪个配置文件属于您要打开的会话，请运行 Remmina，在主窗口中单击会话名称，然后在窗口底部的状态行中查看配置文件的路径。

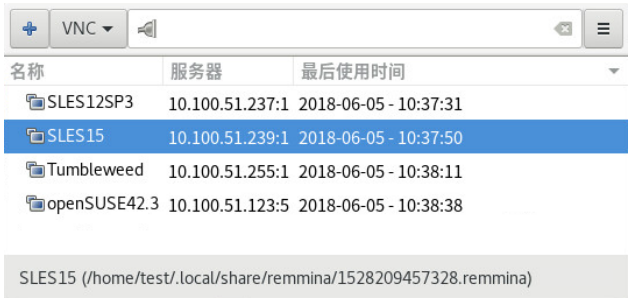


图 8.6 查看配置文件的路径

如果 Remmina 未在运行，您可以将配置文件重命名为更合理的文件名，例如 `sle15.remmina`。您甚至可以将配置文件复制到自定义目录，并从该目录中使用 `remmina -c` 命令来运行它。

8.3 一次性 VNC 会话

一次性会话由远程客户端启动。它在服务器上启动图形登录屏幕。这样您可以选择启动会话的用户，并且如果登录管理器支持，还可以选择桌面环境。终止与此类 VNC 会话的客户端连接时，此会话中启动的所有应用程序也将终止。一次性 VNC 会话不能共享，但可以在一台主机上同时存在多个会话。

过程 8.1 启用一次性 VNC 会话

1. 启动 YaST › 网络服务 › 远程管理 (VNC)。
2. 选中允许远程管理（不含会话管理）。
3. 如果您打算在 Web 浏览器窗口中访问 VNC 会话，请激活允许使用 Web 浏览器访问。
4. 如果需要，还可以选中打开防火墙中的端口（例如，当网络接口配置为在外部区域中时）。如果有多个网络接口，请通过防火墙细节将打开防火墙端口限制为特定的接口。
5. 单击下一步确认您的设置。
6. 如果不是所有需要的包都可使用，则需要批准安装缺少的包。

提示：重启动显示管理器

YaST 对显示管理器设置进行了更改。您需要注销当前图形会话并重启动显示管理器，以使更改生效。

图 8.7 远程管理

8.3.1 可用配置

SUSE Linux Enterprise Server 上的默认配置对会话使用 1024x768 像素（颜色深度 16 位）。会话在端口 5901（对于“普通”VNC 查看器，等同于 VNC 显示器 1）和端口 5801 上可用（对于 Web 浏览器）。

其他配置可在不同端口上使用，请参见第 8.3.3 节 “配置一次性 VNC 会话”。

VNC 显示器编号和 X 显示器编号是独立于一次性会话的。VNC 显示器编号手动指派给服务器支持的每个配置（如上例中的 :1）。只要 VNC 会话启动时带任一配置，就会自动获取可用 X 显示器编号。

默认情况下，VNC 客户端与服务器将尝试通过安装后生成的自我签名 SSL 证书安全通讯。您可以使用默认的证书，也可以将它替换为您自己的证书。使用自我签名证书时，需在首次连接之前确认其签名。

8.3.2 启动一次性 VNC 会话

要连接一次性 VNC 会话，必须安装 VNC 查看器，另请参见第 8.1 节 “`vncviewer` 客户端”。

8.3.3 配置一次性 VNC 会话

如果不需要或想修改默认配置，则可以跳过此部分。

一次性 VNC 会话通过 `systemd` 套接字 `xvnc.socket` 启动。默认情况下提供六个配置块：三个用于 VNC 查看器（`vnc1` 到 `vnc3`），另外三个用于 Java 小程序（`vnchttpd1` 到 `vnchttpd3`）。默认情况下，只有 `vnc1` 和 `vnchttpd1` 是活动的。

要在引导时激活 VNC 服务器套接字，请运行以下命令：

```
sudo systemctl enable xvnc.socket
```

要立即启动套接字，请运行：

```
sudo systemctl start xvnc.socket
```

`Xvnc` 服务器可通过 `server_args` 选项配置。有关选项列表，请参见 `Xvnc --help`。

当添加自定义配置时，请确保它们未使用已由其他配置、其他服务或同一主机上的现有永久 VNC 会话使用的端口。

通过输入以下命令激活配置更改：

```
tux > sudo systemctl reload xvnc.socket
```

重要：防火墙和 VNC 端口

按照过程 8.1 “启用一次性 VNC 会话” 中的描述激活远程管理时，端口 `5801` 和 `5901` 将在防火墙中打开。如果用于 VNC 会话的网络接口受防火墙保护，则为 VNC 会话激活更多端口时，需要手动打开各个端口。有关指导，请参见《Security Guide》，第 16 章 “Masquerading and Firewalls”。

8.4 持续 VNC 会话

可以从多个客户端同时访问持续会话。为了便于演示，我们选择一个较为理想的配置，一个客户端具有完全访问权限，所有其他客户端只具有查看访问权限。另一个用例是教员可能需要访问学员桌面的培训。

提示：连接持续 VNC 会话

要连接持续 VNC 会话，必须安装 VNC 查看器。有关更多详细信息，请参见第 8.1 节 “vncviewer 客户端”。

持续 VNC 会话分为以下两类：

- 使用 vncserver 启动的 VNC 会话
- 使用 vncmanager 启动的 VNC 会话

8.4.1 使用 vncserver 启动的 VNC 会话

此类型的持续 VNC 会话在服务器上启动。该会话和其上启动的所有应用程序运行时不考虑客户端连接，直到会话被终止。永久会话访问受到两种可用口令类型的保护：

- 授予完全访问权限的普通口令或
- 可选仅查看口令，授予非交互（仅查看）访问权限。

一个会话可一次具有两种类型的多个客户端连接。

过程 8.2 使用 vncserver 启动持续 VNC 会话

1. 打开外壳，确保以拥有 VNC 会话的用户身份登录。
2. 如果用于 VNC 会话的网络接口受防火墙保护，则需要手动打开防火墙中您的会话所使用的端口。如果启动多个会话，还可以选择打开一个端口范围。有关如何配置防火墙的细节，请参见《Security Guide》, 第 16 章 “Masquerading and Firewalls”。vncserver 对显示器 :1 使用端口 5901，对显示器 :2 使用端口 5902，依次类推。对于永久会话，VNC 显示器和 X 显示器通常具有相同编号。
3. 要其他具有 1024x769 像素和颜色深度为 16 的会话，请输入以下命令：

   ```
   vncserver -alwaysshared -geometry 1024x768 -depth 16
   ```

   vncserver 命令会在您未指定时选取一个未使用的显示编号，并打印输出它的选择。有关更多选项，请参见 man 1 vncserver。

当您首次运行 `vncserver` 时，它会要求您输入一个口令，以获取会话的完全访问权限。如果需要，还可以提供口令用于会话的仅查看访问。

这里提供的口令还用作同一用户将来启动会话的口令。这些口令可以用 `vncpasswd` 命令更改。

重要：安全考虑因素

确保使用长度够长的高强度口令（八个或更多字符）。不要共享这些口令。

要终止会话，请从 VNC 查看器中关闭运行于 VNC 会话内的桌面环境，像您关闭普通本地 X 会话那样关闭它。

如果希望手动终止会话，请在 VNC 服务器上打开外壳并确保您已作为拥有要终止的 VNC 会话的用户登录。运行以下命令来终止在显示器 `:1` 上运行的会话：`vncserver -kill :1`

8.4.1.1 配置持续 VNC 会话

通过编辑 `$HOME/.vnc/xstartup` 可以配置持续 VNC 会话。默认情况下，此外壳脚本会启动它启动时所处的同一个 GUI/窗口管理器。在 SUSE Linux Enterprise Server 中，此 GUI/窗口管理器为 GNOME 或 IceWM。如果要使用您选择的窗口管理器启动会话，请设置变量 `WINDOWMANAGER`：

```
WINDOWMANAGER=gnome vncserver -geometry 1024x768
WINDOWMANAGER=icewm vncserver -geometry 1024x768
```

注意：每个用户一种配置

持续 VNC 会话在单个按用户配置中进行配置。由同一个用户启动的多个会话都使用相同的启动文件和口令文件。

8.4.2 使用 vncmanager 启动的 VNC 会话

过程 8.3 启用持续 VNC 会话

1. 启动 YaST › 网络服务 › 远程管理 (VNC)。

2. 激活允许远程管理（含会话管理）。
3. 如果您打算在 Web 浏览器窗口中访问 VNC 会话，请激活允许使用 Web 浏览器访问。
4. 如果需要，还可以选中打开防火墙中的端口（例如，当网络接口配置为在外部区域中时）。如果有多个网络接口，请通过防火墙细节将打开防火墙端口限制为特定的接口。
5. 单击下一步确认您的设置。
6. 如果不是所有需要的包都可使用，则需要批准安装缺少的包。

提示：重启动显示管理器

YaST 对显示管理器设置进行了更改。您需要注销当前图形会话并重启动显示管理器，以使更改生效。

8.4.2.1 配置持续 VNC 会话

按过程 8.3 “启用持续 VNC 会话”中所述启用 VNC 会话管理后，便可以使用您喜欢的 VNC 查看器（例如 `vncviewer` 或 Remmina）正常连接到远程会话。此时将显示登录屏幕。登录后，您桌面环境的系统托盘中将出现“VNC”图标。单击该图标可打开 VNC 会话窗口。如果该图标未出现，或者您的桌面环境不支持图标放在系统托盘中，请手动运行 `vncmanager-controller`。

图 8.8 VNC 会话设置

有几个设置会影响 VNC 会话的行为：

非持续，私用

这相当于一次性会话。其他用户将看不到此会话，它在您断开连接后即会终止。有关更多信息，请参考第 8.3 节 “一次性 VNC 会话”。

持续，可见

其他用户可以看见此类会话，它在您断开连接后仍保持运行。

会话名称

您可以在此处指定持续会话的名称，以便在重新连接时可以轻松识别它。

不需要口令

任何人不必使用用户身份凭证登录即可访问会话。

需要用户登录

需要使用有效的用户名和口令登录后才能访问会话。该选项会在允许的用户文本框中列出有效的用户名。

一次允许一个客户端

防止多个用户同时加入该会话。

一次允许多个客户端

允许多个用户同时加入持续会话。适合用于远程演示或培训。

单击确定进行确认。

8.4.2.2 加入持续 VNC 会话

按第 8.4.2.1 节 “配置持续 VNC 会话”中所述设置持续 VNC 会话后，可通过 VNC 查看器加入它。当 VNC 客户端连接到服务器后，系统将提示您选择是要创建新会话还是加入现有会话：

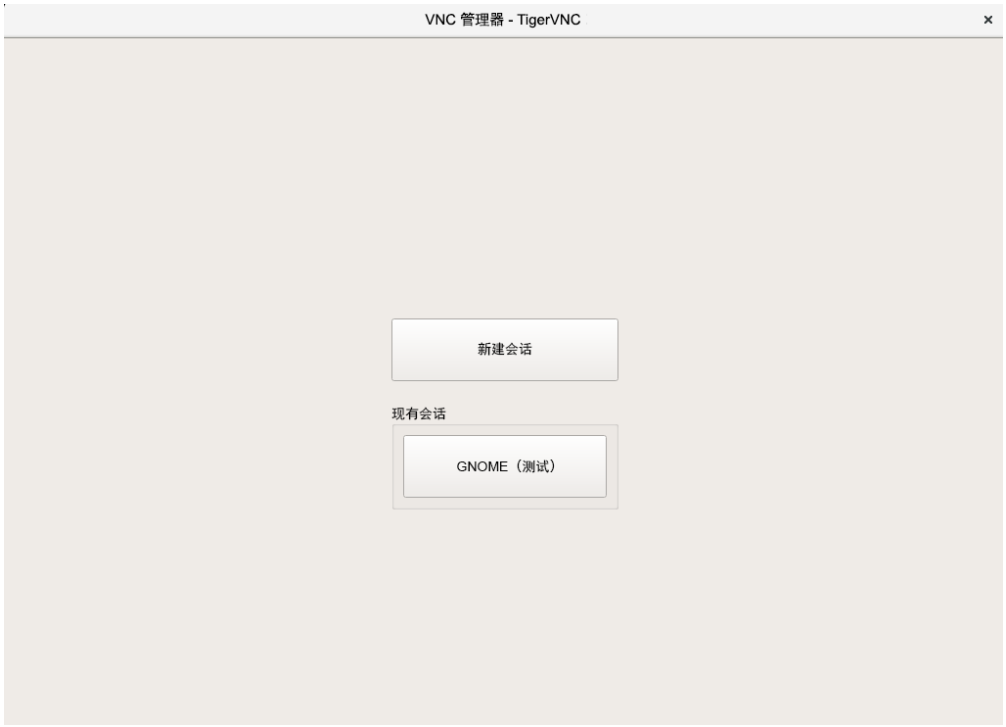


图 8.9 加入持续 VNC 会话

当您单击现有会话的名称后，系统可能要求您输入登录身份凭证，具体取决于持续会话设置。

8.5 加密 VNC 通讯

如果 VNC 服务器设置正确，则 VNC 服务器与客户端之间的所有通讯都会被加密。身份验证在会话开始时进行；实际的数据传输在身份验证后开始。

无论是一次性还是持续 VNC 会话，安全性选项都通过 `server_args` 行中 `/usr/bin/Xvnc` 命令的 `-securitytypes` 参数配置。`-securitytypes` 参数会选择身份验证方法和加密。它的选项如下：

身份验证

None、TLSNone、X509None
: 无身份验证。

VncAuth、TLSVnc、X509Vnc
: 身份验证使用自定义口令。

Plain、TLSPlain、X509Plain
: 身份验证使用 PAM 来验证用户的口令。

加密

None、VncAuth、Plain
: 不加密。

TLSNone、TLSVnc、TLSPlain
: 匿名 TLS 加密。对所有内容加密，但不校验远程主机。因此，您可以防护被动攻击者，但不能防御中间人攻击者。

X509None、X509Vnc、X509Plain
: 使用证书进行 TLS 加密。如果使用自我签名证书，则在第一次连接时，系统将要求您校验证书。在以后的连接中，仅当证书有变动时，系统才会向您发出警告。因此，在第一次连接时，您可以防御中间人攻击之外的所有其他攻击（类似于使用典型的 SSH）。如果使用由证书颁发机构签名且与计算机名称匹配的证书，您的安全将得到全面保障（类似于使用典型的 HTTPS）。

提示：证书和密钥的路径

如果使用基于 X509 的加密，需要通过 `-X509Cert` 和 `-X509Key` 选项指定 X509 证书和密钥的路径。

如果您选择多种安全性类型（用逗号分隔），将会使用客户端和服务器都支持且允许的第一种安全性。如此，您便可在服务器上配置随机加密。如果您需要支持不支持加密的 VNC 客户端，此功能将十分有用。

在客户端上，您也可以指定允许的安全性类型，以防在您连接到已知启用了加密的服务器时遭到降级攻击（虽然在该情况下，我们的 vncviewer 会发出“连接未加密！”讯息来警告您）。

9 使用 RSync 复制文件

当今时代，用户通常都会有数台计算机：家用计算机和办公计算机、笔记本电脑、智能手机或平板电脑。因而，在多个设备之间保持文件和文档同步的任务就变得越发重要。

警告：数据丢失风险

您在开始使用同步工具之前，应该先熟悉其特性和功能。请务必备份您的重要文件。

9.1 概念概述

对于要通过慢速网络连接同步大量数据的情况，Rsync 提供了可靠的方法来只传输文件中的更改。此方法不仅适用于文本文件，还适用于二进制文件。为了检测文件之间的差异，Rsync 将文件分为多个块，并计算它们的校验和。

检测更改对计算能力有一定的要求。因此，请确保两端的计算机均具有足够的资源，包括 RAM。

当需要定期传输大量只包含微小更改的数据时，Rsync 特别有用。进行备份时就常常用到该工具。Rsync 也非常适合用来镜像试验服务器，此类服务器将 Web 服务器的完整目录树储存到 DMZ 内的某台 Web 服务器中。

Rsync 并不是同步工具，虽然它的名字看上去有些像。Rsync 工具一次只能在一个方向复制数据。它不会也不能反向复制数据。如果您需要既能同步源又能同步目标的双向工具，请使用 Csync。

9.2 基本语法

Rsync 是一个命令行工具，基本语法如下：

```
rsync [OPTION] SOURCE [SOURCE]... DEST
```

您可以在任何本地或远程计算机上使用 Rsync，前提是您拥有相应的访问权限和写入权限。可以有多个 SOURCE 项。SOURCE 和 DEST 占位符可以是路径和/或 URL。

下面介绍一些最常用的 Rsync 选项：

`-v`
: 输出较详细的文本

`-a`
: 存档模式；以递归方式复制文件并保留时间戳、用户/组所有权、文件权限和符号链接

`-z`
: 压缩传输的数据

注意：尾部斜杠计数

使用 Rsync 时，要特别注意尾部斜杠。目录后面的尾部斜杠表示目录的内容。没有尾部斜杠表示目录本身。

9.3 在本地复制文件和目录

下面的说明假设当前用户拥有 /var/backup 目录的写入许可权限。要将单个文件从计算机上的一个目录复制到另一个路径，请使用以下命令：

```
tux > rsync -avz backup.tar.xz /var/backup/
```

文件 backup.tar.xz 会复制到 /var/backup/，绝对路径是 /var/backup/backup.tar.xz。

请勿忘记在 /var/backup/ 目录后面加上尾部斜杠！如果不插入斜杠，文件 backup.tar.xz 会复制到 /var/backup（文件）中，而不是 /var/backup/ 目录中！

复制目录与复制单个文件相似。下面的示例将目录 tux/ 及其内容复制到 /var/backup/ 目录中：

```
tux > rsync -avz tux /var/backup/
```

在绝对路径 /var/backup/tux/ 中可找到副本。

9.4 远程复制文件和目录

两台计算机上都需要有 Rsync 工具。要从远程目录复制文件或将文件复制到远程目录，需要提供 IP 地址或域名。如果本地计算机和远程计算机上当前的用户名相同，则可以不指定用户名。

要使用相同的用户（在本地和远程主机上）将文件 `file.tar.xz` 从本地主机复制到远程主机 `192.168.1.1`，请使用以下命令：

```
tux > rsync -avz file.tar.xz  tux@192.168.1.1:
```

根据您的个人偏好，也可以使用下面的命令，它们的作用相同：

```
tux > rsync -avz file.tar.xz 192.168.1.1:~
tux > rsync -avz file.tar.xz 192.168.1.1:/home/tux
```

在使用标准配置的所有情况下，系统会提示您输入远程用户的通行口令。此命令会将 `file.tar.xz` 复制到用户 `tux` 的主目录（通常为 `/home/tux`）。

远程复制目录与在本地复制目录相似。下面的示例将目录 `tux/` 及其内容复制到 `192.168.1.1` 主机上的远程目录 `/var/backup/`：

```
tux > rsync -avz tux 192.168.1.1:/var/backup/
```

假设您在主机 `192.168.1.1` 上拥有写入许可权限，便可在绝对路径 `/var/backup/tux` 中找到副本。

9.5 配置和使用 Rsync 服务器

Rsync 可作为在用于传入连接的默认端口 873 上列出的守护程序（`rsyncd`）运行。此守护程序可以接收“复制目标”。

下面的说明介绍如何在 `jupiter` 上创建具有备份目标的 Rsync 服务器。此目标可用于储存您的备份。要创建 Rsync 服务器，请执行以下操作：

过程 9.1 设置 RSYNC 服务器

1. 在 jupiter 上，创建用于储存您所有备份文件的目录。在此示例中，我们使用 `/var/backup`：

```
root # mkdir /var/backup
```

2. 指定所有权。在此示例中，该目录为 用户 组中的用户 `tux` 所拥有：

```
root # chown tux.users /var/backup
```

3. 配置 rsyncd 守护程序。
   我们将配置文件分割成一个主文件，和一些用于存放您的备份目标的“模块”。如此，以后便可更轻松地添加其他目标。全局值可以储存在 `/etc/rsyncd.d/*.inc` 文件中，而模块放置在 `etc/rsyncd.d/*.conf` 文件中：

   a. 创建目录 `/etc/rsyncd.d/`：

   ```
   root # mkdir /etc/rsyncd.d/
   ```

   b. 在主配置文件 `/etc/rsyncd.conf` 中，添加以下几行：

   ```
   # rsyncd.conf main configuration file
   log file = /var/log/rsync.log
   pid file = /var/lock/rsync.lock

   &merge /etc/rsyncd.d ❶
   &include /etc/rsyncd.d ❷
   ```

   ❶ 将 `/etc/rsyncd.d/*.inc` 文件中的全局值合并到主配置文件中。

   ❷ 从 `/etc/rsyncd.d/*.conf` 文件中装载任何模块（或目标）。这些文件不应该包含对全局值的任何参照。

   c. 在文件 `/etc/rsyncd.d/backup.conf` 中通过以下几行创建您的模块（您的备份目标）：

   ```
   # backup.conf: backup module
   [backup] ❶
      uid = tux ❷
      gid = users ❷
      path = /var/backup ❸
      auth users = tux ❹
   ```

```
   secrets file = /etc/rsyncd.secrets ❺
   comment = Our backup target
```

❶ 备份目标。可以使用您喜欢的任何名称。但最好根据目标的用途来命名，并使用在 `*.conf` 文件中所用的相同名称。

❷ 指定在进行文件传输时所用的用户名或组名。

❸ 定义用于储存备份的路径（从步骤 1 中）。

❹ 指定允许的用户的逗号分隔列表。列表以最简单的方式包含允许连接到此模块的用户名。在我们的示例中，只允许用户 `tux`。

❺ 指定包含用户名和明文口令的行所在文件的路径。

d. 创建包含以下内容的 `/etc/rsyncd.secrets` 文件，并替换 `PASSPHRASE`：

```
# user:passwd
tux:PASSPHRASE
```

e. 确保该文件只有 `root` 用户可读：

```
root # chmod 0600 /etc/rsyncd.secrets
```

4. 通过以下命令启动并启用 rsyncd 守护程序：

```
root # systemctl enable rsyncd
root # systemctl start rsyncd
```

5. 测试是否可访问 Rsync 服务器：

```
tux > rsync jupiter::
```

您应该会看到类似如下的响应：

```
backup          Our backup target
```

若非如此，请检查您的配置文件、防火墙和网络设置。

上述步骤创建了 Rsync 服务器，现在可以使用它来储存备份。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。如果您要对传输进行调试，那么它非常有用。

要列出备份目标的内容，请使用以下命令：

```
rsync -avz jupiter::backup
```

此命令会列出服务器上 `/var/backup` 目录中存在的所有文件。此请求还记录在日志文件 `/var/log/rsyncd.log` 中。要开始实际传输，请提供源目录。使用 `.` 表示当前目录。例如，下面的命令会将当前目录复制到 Rsync 备份服务器：

```
rsync -avz . jupiter::backup
```

默认情况下，Rsync 不会在运行时删除文件和目录。要允许删除，必须另外指定选项 `##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。

9.6 更多信息

CSync

双向文件同步程序，请参见 https://www.csync.org/。

RSnapshot

创建增量备份，请参见 http://rsnapshot.org。

Unison

与 CSync 类似的文件同步程序，但具有图形界面，请参见 http://www.seas.upenn.edu/~bcpierce/unison/。

Rear

一个灾难恢复框架，请参见 https://documentation.suse.com/sle-ha-12/ 上 SUSE Linux Enterprise High Availability Extension 的《管理指南》。

II 引导 Linux 系统

10 引导过程简介 126

11 UEFI（统一可扩展固件接口） 134

12 引导加载程序 GRUB 2 143

13 systemd 守护程序 163

10 引导过程简介

引导 Linux 系统涉及不同组件和任务。固件和硬件初始化过程（取决于计算机的体系结构）完成后，系统将通过引导加载程序 GRUB 2 启动内核。在此之后，引导进程完全由操作系统控制，并由 `systemd` 处理。`systemd` 会提供一组“目标”，用于引导与日常使用、维护或紧急情况相关的配置。

10.1 术语

本章使用的术语可能存在歧义。为了理解本章中术语的用法，请阅读以下定义：

`init`

有两个不同的进程通常命名为 “init”：

- 用于装入根文件系统的 `initramfs` 进程
- 从实际根文件系统执行且用于启动其他所有进程的操作系统进程

在这两种情况下，`systemd` 程序都会处理此任务。首先会从 `initramfs` 执行此进程，以装入根文件系统。装入成功后，将从根文件系统以初始进程的形式重新执行此进程。为了避免混淆这两个 `systemd` 进程，我们将第一个进程称为 init on initramfs，将第二个进程称为 systemd。

`initrd` / `initramfs`

`initrd`（初始 RAM 磁盘）是一个映像文件，内含内核所装载的并且作为临时根文件系统从 `/dev/ram` 装入的根文件系统映像。装入此文件系统需要使用文件系统驱动程序。从内核 2.6.13 开始，initrd 已由 `initramfs`（初始 RAM 文件系统）取代，后者无需文件系统驱动程序即可装入。SUSE Linux Enterprise Server 只使用 `initramfs`。但是，由于 `initramfs` 作为 `/boot/initrd` 储存，因此通常将其称为“initrd”。本章只使用名称 `initramfs`。

10.2 Linux 引导进程

Linux 引导进程包括多个阶段，每个阶段由一个不同组件来代表：

1. 第 10.2.1 节 “初始化和引导加载程序阶段”
2. 第 10.2.2 节 “内核阶段”
3. 第 10.2.3 节 “init on initramfs 阶段”
4. 第 10.2.4 节 “systemd 阶段”

10.2.1 初始化和引导加载程序阶段

在初始化阶段，将设置计算机的硬件并准备好设备。此过程根据硬件体系结构的不同有很大的差别。

SUSE Linux Enterprise Server 在所有体系结构中都使用引导加载程序 GRUB 2。根据体系结构和固件，启动 GRUB 2 引导加载程序的过程可能包括多个步骤。引导加载程序的用途是装载内核以及基于 RAM 的初始文件系统 (initramfs)。有关 GRUB 2 的详细信息，请参见第 12 章 “引导加载程序 GRUB 2”。

10.2.1.1 AArch64 及 Intel 64/AMD64 上的初始化和引导加载程序阶段

在打开计算机之后，BIOS 或 UEFI 将初始化屏幕和键盘并测试主内存。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 值装载有关当前日期、时间和最重要的外设的信息。识别引导媒体及其几何尺寸之后，系统控制权将从 BIOS/UEFI 转到引导加载程序。

在装配传统 BIOS 的计算机上，只能装载引导磁盘的第一个 512 字节物理数据扇区（主引导记录，MBR）中的代码。只有极少量的 GRUB 2 代码能够装入 MBR。引导加载程序的唯一作用就是从 MBR 与第一个分区（MBR 分区表）之间的间隙处，或是从 BIOS 引导分区（GPT 分区表）装载包含文件系统驱动程序的 GRUB 2 核心映像。此映像包含文件系统驱动程序，因此能够访问根文件系统中的 `/boot`。`/boot` 包含 GRUB 2 核心的附加模块以及内核和 initramfs 映像。获取此分区的访问权限后，GRUB 2 会将内核和 initramfs 映像装载到内存中，并将控制权交接到内核。

从包含已加密分区 `/boot` 的加密文件系统引导 BIOS 系统时，需要输入解密口令两次。GRUB 2 使用第一次输入的口令来解密 `/boot`，`systemd` 使用第二次输入的口令来装载加密的卷。

在装配 UEFI 的计算机上，引导过程比装配传统 BIOS 的计算机要简单得多。固件能够读取包含 GPT 分区表的磁盘的 FAT 格式化系统分区。此 EFI 系统分区（在运行的系统中装载为 `/boot/efi`）可提供足够的空间用于托管由固件直接装载和执行的完备 GRUB 2。

如果 BIOS/UEFI 支持网络引导，则也可以配置提供引导加载程序的引导服务器。然后，可以通过 PXE 引导系统。BIOS/UEFI 充当引导加载程序。它会从引导服务器获取引导映像，然后启动系统。这完全不依赖本地硬盘。

10.2.1.2 IBM Z 上的初始化和引导加载程序阶段

在 IBM Z 上，必须通过名为 `zipl`（z initial program load，z 初始程序装载）的引导加载程序来初始化引导过程。尽管 `zipl` 支持读取不同的文件系统，但它不支持 SLE 默认文件系统 (Btrfs) 或者从快照引导。因此，SUSE Linux Enterprise Server 使用两阶段的引导过程来确保在引导时提供完整的 Btrfs 支持：

1. `zipl` 从 ext2 格式化分区 `/boot/zipl` 引导。此分区包含一个极简的内核，以及一个装载到内存中的 initramfs。initramfs 包含 Btrfs 驱动程序（及其他组件）和引导加载程序 GRUB 2。内核是使用参数 `initgrub`（告知要启动 GRUB 2）启动的。
2. 内核会装入根文件系统，使 `/boot` 可访问。现在，将从 initramfs 启动 GRUB 2。GRUB 2 从 `/boot/grub2/grub.cfg` 读取其配置，并从 `/boot` 装载最终的内核和 initramfs。现在，将通过 Kexec 装载新内核。

10.2.2 内核阶段

引导加载程序转交系统控制权后，所有体系结构中的引导过程都是相同的。引导加载程序会将内核和基于 RAM 的初始文件系统 (`initramfs`) 都装载到内存中，而内核将接管控制权。

内核设立内存管理并检测 CPU 类型及其功能后，将初始化硬件，并从内存中装入使用 `initramfs` 装载的临时根文件系统。

10.2.2.1 initramfs 文件

initramfs（初始 RAM 文件系统）是一个小型 cpio 存档，可由内核装载到 RAM 磁盘中。该文件位于 /boot/initrd 中。可以使用名为 dracut 的工具创建该文件，有关细节，请参见 man 8 dracut。

initramfs 提供了一个极简的 Linux 环境，可用于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 或 UEFI 例程载入内存，而且除了需要足够的内存外没有特定的硬件要求。initramfs 存档必须始终提供一个名为 init 的可执行文件，该文件执行根文件系统上的 systemd 守护程序，使引导进程得以继续。

在能够装入 root 文件系统并启动操作系统之前，内核需要相应的驱动程序来访问 root 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。根文件系统所需的模块由 init on initramfs 装载。装载模块后，udev 将为 initramfs 提供所需的设备。在引导过程的后面，更改 root 文件系统之后需要重新生成设备。可以使用 systemd 单元 systemd-udev-trigger.service 来实现此目的。

10.2.2.1.1 重新生成 initramfs

由于 initramfs 包含多个驱动程序，因此，每当其中某个驱动程序有新版本发布时，都需要更新 initramfs。在安装包含驱动程序更新的包时可以自动完成这种更新。YaST 或 zypper 通过显示用于生成 initramfs 的命令的输出来告知此状况。但在某些情况下，您需要手动重新生成 initramfs：

- 由于更换硬件而需添加驱动程序
- 将系统目录移到 RAID 或 LVM
- 将磁盘添加到包含根文件系统的 LVM 组或 Btrfs RAID
- 更改内核变量

由于更换硬件而需添加驱动程序

如果需要更换硬件（例如硬盘），并且引导时此硬件需要内核中的不同驱动程序，则您必须更新 initramfs 文件。

打开或创建 /etc/dracut.conf.d/10-DRIVER.conf 并添加以下行（请注意前导空格）：

```
force_drivers+=" DRIVER1"
```

用驱动程序的模块名称替换 `DRIVER1`。如果您需要添加多个驱动程序，请将其全部列出并以空格分隔：

```
force_drivers+=" DRIVER1 DRIVER2"
```

继续过程 10.1 “生成 initramfs”。

将系统目录移到 RAID 或 LVM

每当您要将正在运行的系统中的交换文件或系统目录（例如 `/usr`）移到 RAID 或逻辑卷时，都需要创建一个包含软件 RAID 或 LVM 驱动程序支持的 `initramfs`。

为此，请在 `/etc/fstab` 中创建相关的项，并装入新项（例如，使用 `mount -a` 和/或 `swapon -a`）。

继续过程 10.1 “生成 initramfs”。

将磁盘添加到包含根文件系统的 LVM 组或 Btrfs RAID

每当您要在包含根文件系统的逻辑卷组或者 Btrfs RAID 中添加（或去除）磁盘时，都需要创建一个支持扩容的卷的 `initramfs`。请按照过程 10.1 “生成 initramfs”中的指导操作。

继续过程 10.1 “生成 initramfs”。

更改内核变量

如果您在 `sysctl` 界面中通过编辑相关文件（`/etc/sysctl.conf` 或 `/etc/sysctl.d/*.conf`）更改了内核变量的值，系统下次重引导时，这项更改将会丢失。即使您在运行时使用 `sysctl --system` 装载这些值，更改也不会保存到 initramfs 文件中。 您需要根据过程 10.1 “生成 initramfs”中所述更新该文件。

过程 10.1 生成 INITRAMFS

请注意，以下过程中的所有命令都需要以 `root` 用户身份执行。

1. 运行以下命令生成新的 `initramfs` 文件

```
dracut MY_INITRAMFS
```

请将 `MY_INITRAMFS` 替换为所选的文件名。新的 `initramfs` 将创建为 `/boot/MY_INITRAMFS`。

或者运行 `dracut -f`。这会重写当前使用的现有文件。

2. （如果在上一步中运行了 `dracut -f`，请跳过此步骤。）为上一步中创建的 `initramfs` 文件创建链接：

```
(cd /boot && ln -sf MY_INITRAMFS initrd)
```

3. 在 IBM Z 体系结构中，还需额外运行 `grub2-install`。

10.2.3 init on initramfs 阶段

内核从 `initramfs` 装入的临时根文件系统包含可执行文件 `systemd`（下文称为 `init` on `initramfs`，另请参见第 10.1 节 “术语”）。此程序执行装入正确根文件系统所需的全部操作。它为所需的文件系统提供内核功能，并为使用 `udev` 的大量储存控制器提供设备驱动程序。

`initramfs` 上的 `init` 的主要用途是准备真实 root 文件系统的装入和访问。根据您的系统配置的不同，`initramfs` 上的 `init` 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组件（最重要的组件是硬盘）。要访问最终的 root 文件系统，内核需要装载适当的文件系统驱动程序。

提供块特殊文件

内核根据装载的模块生成设备事件。`udev` 会处理这些事件并在 RAM 文件系统的 `/dev` 中生成所需的特殊块文件。没有这些特殊文件，文件系统和其他设备将不可访问。

管理 RAID 和 LVM 设置

如果将系统配置为在 RAID 或 LVM 下保存根文件文件系统，则 `initramfs` 上的 `init` 将设置 LVM 或 RAID 以支持以后对根文件系统的访问。

管理网络配置

如果将系统配置为使用通过网络装入的 root 文件系统（通过 NFS 装入），则 `init` 必须确保装载了正确的网络驱动程序，并确保将其设置为支持访问 root 文件系统。

如果文件系统驻留在一个联网的块设备（如 iSCSI 或 SAN）上，则与储存服务器的连接也由 `initramfs` 上的 `init` 设置。SUSE Linux Enterprise Server 支持在主要目标不可用的情况下从次要 iSCSI 目标引导。有关 iSCSI 引导目标配置的更多细节，请参见《储存管理指南》，第 14 章 “经由 IP 网络的大容量储存：iSCSI”，第 14.3.1 节 “使用 YaST 执行 iSCSI 发起端配置”。

注意：处理装入错误

如果根文件系统无法从引导环境中装入，则必须先对其进行检查和修复，才能继续引导。如果文件系统为 Ext3 和 Ext4，文件系统检查程序将会自动启动。如果是 XFS 和 Btrfs 文件系统，则不会自动开始修复过程，而是向用户显示有关可用于修复文件系统的选项的信息。成功修复文件系统后，退出引导环境将会使系统重试装入根文件系统。如果装入成功，将正常继续引导。

10.2.3.1 安装过程中的 init on initramfs 阶段

如果在安装过程的初始引导阶段调用 `init` on `initramfs`，它要执行的任务将与上述任务不同。请注意，安装系统不会从 `initramfs` 启动 `systemd`，这些任务由 `linuxrc` 执行。

查找安装媒体

当您启动安装进程时，计算机会装载一个安装内核以及一个包含 YaST 安装程序的特殊 `init`。YaST 安装程序正在 RAM 文件系统中运行，它需要知道安装媒体的位置，才能访问安装媒体以安装操作系统。

启动硬件识别并装载适当的内核模块

如第 10.2.2.1 节 “initramfs 文件”中所述，引导过程从最少的一组驱动程序（可在大多数硬件配置中使用）开始。在 AArch、POWER 和 Intel 64/AMD64 计算机上，`linuxrc` 会启动初始硬件扫描过程，以确定适合您的硬件配置的驱动程序集。在 IBM Z 上，需要提供驱动程序列表及其参数，例如，通过 linuxrc 或 parmfile 提供。这些驱动程序用于生成引导系统所需的自定义 `initramfs`。如果引导时不需要这些模块，但冷插拔时需要，您可以使用 `systemd` 装载这些模块。有关详细信息，请参见第 13.6.4 节 “装载内核模块”。

装载安装系统

系统在正确识别硬件后会装载相应的驱动程序。`udev` 程序会创建特殊的设备文件，`linuxrc` 使用 YaST 安装程序启动安装系统。

启动 YaST

最后，`linuxrc` 启动 YaST，后者则启动包安装和系统配置。

10.2.4 systemd 阶段

找到“实际的”根文件系统后，对其进行错误检查并装入。如果装入成功，系统会清理 `initramfs` 并执行根文件系统上的 `systemd` 守护程序。`systemd` 是 Linux 的系统和服务管理器。它是作为 PID 1 启动的父进程，充当用于启动和维护用户空间服务的 init 系统。有关详细信息，请参见第 13 章 “systemd 守护程序”。

11 UEFI（统一可扩展固件接口）

UEFI（统一可扩展固件接口）是用于系统硬件自带的固件、系统所有的硬件组件以及操作系统之间的接口。

UEFI 在 PC 系统上的应用范围越来越广，因此正在逐渐替代传统的 PC-BIOS。例如，UEFI 能够很好地支持 64 位系统，提供安全的引导（“安全引导”，要求固件为 2.3.1c 或以上版本）。安全引导是其最为重要的特性之一。最后，借助 UEFI，标准固件将能够用于所有 x86 平台。

除此之外，UEFI 还具有以下优点：

- 从带有 GUID 分区表 (GPT) 的大磁盘（超过 2 TiB）引导。
- 独立于 CPU 的架构和驱动程序。
- 带有网络功能的灵活的预操作系统环境。
- 通过 PC-BIOS 式仿真支持引导老式操作系统的 CSM（兼容支持模块）。

有关详细信息，请参见 http://en.wikipedia.org/wiki/Unified_Extensible_Firmware_Interface。以下小节仅针对部分功能如何在 SUSE Linux Enterprise Server 中实现而列举一些提示，并不代表 UEFI 的整体概述。

11.1 安全引导

在 UEFI 领域中，要想保障引导程序的安全，需要建立一个信任链。“平台”是此信任链的根；在 SUSE Linux Enterprise Server 环境中，可将主板和板载固件视为“平台”。换句话说，它就是硬件供应商，信任链从硬件供应商流向组件制造商、OS 供应商等。

系统通过公共密钥加密法表示信任。硬件供应商将所谓的“平台密钥 (PK)”放入固件中，代表可信根。操作系统供应商与其他方将其密钥与“平台密钥”签署在一起，以此记录他们之间的信任关系。

最后，除非这些“可信的”密钥之一（即 OS 引导加载程序、位于一些 PCI Express 卡的闪存上或磁盘上的一些驱动程序，或者更新的固件本身）签署了代码，否则要求固件不得执行该代码，从而建立安全保障。

要使用安全引导，您需要用固件信任的密钥对您的 OS 加载程序签名，并且需要 OS 加载程序校验其加载的内核是否可信。

可以将密钥交换密钥 (KEK) 添加到 UEFI 密钥数据库中。这样，其他证书只要签署了 PK 的私用部分，即可由您使用。

11.1.1 在 SUSE Linux Enterprise Server 上实施

默认情况下安装微软的密钥交换密钥 (KEK)。

注意：需要 GUID 分区表 (GPT)

UEFI/x86_64 安装中默认会启用安全引导功能。您可在引导加载程序设置对话框的引导代码选项选项卡中找到启用安全引导支持选项。该选项支持在固件中的安全引导已激活时引导，同时也支持在安全引导已停用时引导。

图 11.1 安全引导支持

“安全引导”特性要求 GUID 分区表 (GPT) 使用主引导记录 (MBR) 替代旧的分区。如果安装期间 YaST 检测到 EFI 模式，则会设法创建 GPT 分区。UEFI 预期会在 FAT 格式的 EFI 系统分区(ESP) 上查找到 EFI 程序。

支持 UEFI 安全引导基本上要求具有固件认可作为可信密钥的数字签名的引导加载程序。该密钥需要先天为固件所信任（无需任何手动干预）。

有两种办法可以实现。一种是与硬件供应商合作，让其签署 SUSE 密钥，然后 SUSE 会使用该 SUSE 密钥签署引导加载程序；另一种是通过微软的 Windows 徽标认证计划使引导加载程序获得认证，并使微软认可 SUSE 签名密钥（也就是让加载程序使用他们的 KEK 签名）。至此，SUSE 使引导加载程序获得了 UEFI 签名服务（在此情况下为 Microsoft）的签名。

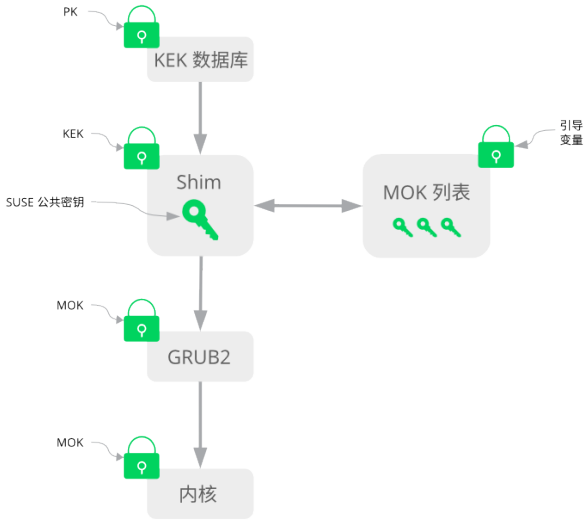


图 11.2 UEFI：安全引导流程

SUSE 在实施层使用默认将会安装的 `shim` 加载程序。这是一种可以避免法律纠纷的智能解决方案，能够大幅简化认证和签名步骤。`shim` 加载程序的任务是装载 GRUB 2 等引导加载程序并对其进行校验，之后，此引导加载程序将装载仅由一个 SUSE 密钥签名的内核。SUSE 从全新安装的 SLE11 SP3 开始提供此功能，并要求启用 UEFI 安全引导。

可信用户分为以下两类：

- 首先是持有密钥的用户。平台密钥 (PK) 几乎允许所有操作。密钥交换密钥 (KEK) 的许可范围与 PK 一致，但不能更改 PK。
- 其次是能够以物理方式访问机器的任何用户。具有物理访问权限的用户可以重引导机器并对 UEFI 进行配置。

UEFI 提供下列两类变量以满足这些用户的需求：

- 第一类变量即所谓的“已验证的变量”，它们可从引导过程（所谓的“引导服务环境”）和正在运行的操作系统中更新。仅当对变量新值签名的密钥是用于对变量旧值签名的相同密钥时，才能进行更新。而且，您只能向这类变量追加或将其更改为序列号更高的数值。
- 第二类变量即所谓的“仅供引导服务使用的变量”。引导进程中运行的任何代码都可以获取这些变量。在结束引导进程并准备启动 OS 的时间间隔内，引导加载程序必须调用 `ExitBootServices` 呼叫。此后将无法获取这些变量，OS 也无法接触到这些变量。

各类 UEFI 密钥列表属于第一类，因为该类变量除了允许联机更新外，还允许添加密钥、驱动程序、固件指纹以及将其列入黑名单。第二类变量即“仅供引导服务使用的变量”。该类变量有助于以安全且支持开源的方式实施安全引导，因此符合 GPLv3 要求。

SUSE 首先启动 `shim`，它是一个小而简单的 EFI 引导加载程序，由 SUSE 和 Microsoft 签名。这样一来，`shim` 即可加载并执行。

`shim` 随后继续验证其想要加载的引导加载程序是否可信。默认情况下，`shim` 会使用其主体中所嵌入的独立的 SUSE 证书。此外，`shim` 还允许“登记”其他密钥，用于覆盖默认的 SUSE 密钥。下文将这些密钥称为“机器拥有者密钥”或缩写为 MOK。

接下来，引导加载程序会验证内核，然后加以引导。该内核将在模块上执行同样的操作。

11.1.2 MOK（机器拥有者密钥）

如果用户（“机器拥有者”）想要更换引导进程的任何组件，则会用到“机器拥有者密钥 (MOK)”。他们可以借助 `mokutils` 工具对组件签名及管理 MOK。

当加载 `shim` 时，登记进程便会开始重引导计算机并中断引导进程（例如按下某个键）。`shim` 随后转入登记模式，允许用户使用引导分区上的文件的密钥替换默认的 SUSE 密钥。如果用户选择这样做，则 `shim` 将计算该文件的哈希，并将计算结果置入“仅供引导服务”的变量中。这样，`shim` 可以检测到文件除引导服务之外出现的任何更改，从而避免篡改用户核准的 MOK 列表。

上述各步都在引导时发生，此时仅执行已经校验的代码。因此，只有控制台上所示的一位用户可以使用机器拥有者的一组密钥。它不可能是远程访问 OS 的恶意程序或骇客，因为骇客或恶意程序只能更改文件，但无法更改存储在“仅供引导服务使用”的变量中的哈希。

引导加载程序经 `shim` 装载并校验后，如果需要校验内核以免校验代码重复，则会回调 `shim`。为此，`Shim` 将使用同一份 MOK 列表并通知引导加载程序能否加载内核。

这样，您就可以安装自己的内核或引导加载程序。您只需要安装一组新的密钥，并在首次重引导期间以物理方式呈现，从而予以授权。由于 MOK 是个列表而不是单个 MOK，因此您可以让 `shim` 信任来自多个供应商的密钥，以允许从引导加载程序进行双重或多重引导。

11.1.3 引导自定义内核

以下内容基于 http://en.opensuse.org/openSUSE:UEFI#Booting_a_custom_kernel。

安全引导不会阻止您使用自行编译的内核。您必须使用自己的证书在该内核上签名，并让固件或 MOK 得以识别该证书。

1. 创建一个自定义的 X.509 密钥以及用于签名的证书：

   ```
   openssl req -new -x509 -newkey rsa:2048 -keyout key.asc \
     -out cert.pem -nodes -days 666 -subj "/CN=$USER/"
   ```

   有关创建证书的详细信息，请参见 http://en.opensuse.org/openSUSE:UEFI_Image_File_Sign_Tools#Create_Your_Own_Certificate。

2. 将密钥和证书打包成 PKCS#12 结构：

   ```
   openssl pkcs12 -export -inkey key.asc -in cert.pem \
     -name kernel_cert -out cert.p12
   ```

3. 生成用于 `pesign` 的 NSS 数据库：

   ```
   certutil -d . -N
   ```

4. 将 PKCS#12 中包含的密钥和证书导入 NSS 数据库：

   ```
   pk12util -d . -i cert.p12
   ```

5. 使用 `pesign` 将新签名“赋予”内核：

   ```
   pesign -n . -c kernel_cert -i arch/x86/boot/bzImage \
     -o vmlinuz.signed -s
   ```

6. 列出内核映像上的签名：

```
pesign -n . -S -i vmlinuz.signed
```

此时，您可以照常在 `/boot` 中安装内核。由于内核现有一个自定义的签名，因此需要将用于签名的证书导入 UEFI 固件或 MOK 中。

7. 将证书转为 DER 格式，以供导入固件或 MOK：

```
openssl x509 -in cert.pem -outform der -out cert.der
```

8. 将证书复制到 ESP 以简化访问：

```
sudo cp cert.der /boot/efi/
```

9. 使用 `mokutil` 自动启动 MOK 列表。

- a. 将证书导入到 MOK 中：

```
mokutil --root-pw --import cert.der
```

`--root-pw` 选项可让 `root` 用户直接使用。

b. 检查准备注册的证书列表：

```
mokutil --list-new
```

c. 重引导系统；`shim` 应该会起动 MokManager。您需要输入 `root` 口令以确认将证书导入到 MOK 列表中。

d. 检查新导入的密钥以前是否注册过：

```
mokutil --list-enrolled
```

- a. 此外，若要手动启动 MOK，也可以采用这一过程：
重引导

b. 在 GRUB 2 菜单中，按 `c` 键。

c. 类型：

```
chainloader $efibootdir/MokManager.efi
boot
```

d. 选择从磁盘登记密钥。

e. 导航至 `cert.der` 文件并按 Enter。

f. 按照指导登记密钥。正常情况下应按“0”，然后按“y”予以确认。
除此之外，固件菜单也可能提供了多种向“签名数据库”中添加新密钥的方式。

11.1.4 使用非内置驱动程序

在启用安全引导的情况下，不支持在安装过程中添加非内置驱动程序（即，不是 SUSE Linux Enterprise Server 自带的驱动程序）。用于 SolidDriver/PLDP 的签名密钥默认不受信任。

您可以通过两种不同的方式，在启用安全引导的情况下于安装期间安装第三方驱动程序。在这两种情况下，都要：

- 在安装前，通过固件或系统管理工具将所需密钥添加到固件数据库中。此选项取决于您当前使用的具体硬件。请咨询您的硬件供应商了解详细信息。
- 使用 https://drivers.suse.com/ 上或硬件供应商提供的可引导驱动程序 ISO，在首次引导时将所需密钥登记到 MOK 列表中。

要使用可引导驱动程序 ISO 将驱动程序密钥登记到 MOK 列表中，请执行以下步骤：

1. 将上文所述 ISO 映像刻录到空 CD/DVD 媒体中。
2. 使用新的 CD/DVD 媒体开始安装，并准备好标准的安装媒体或网络安装服务器的 URL。
   如果您要进行网络安装，请在引导命令行上使用 `install=` 选项输入网络安装源的 URL。
   如果您是从光学媒体安装，安装程序会先从驱动程序包引导，然后要求插入产品的第一张安装光盘。
3. 安装时将会使用包含经过更新的驱动程序的 initrd。

有关详细信息，请参见 https://drivers.suse.com/doc/Usage/Secure_Boot_Certificate.html。

11.1.5 功能和限制

以安全引导模式引导时，可以使用以下功能：

- 安装到 UEFI 默认的引导加载程序位置，这是为了保留或恢复 EFI 引导项而采用的机制。
- 通过 UEFI 重引导。
- 如果没有其他可回退到的旧版 BIOS，Xen 超级管理程序将使用 UEFI 引导。
- 支持 UEFI IPv6 PXE 引导。
- 支持 UEFI 视频模式，内核可以从 UEFI 检索视频模式，以使用相同的参数配置 KMS 模式。
- UEFI 支持从 USB 设备引导。

以安全引导模式引导时，存在以下限制：

- 为确保他人无法轻易绕过安全引导，系统在安全引导下运行时会禁用部分内核特性。
- 引导加载程序、内核以及内核模块必须经过签名。
- Kexec 和 Kdump 处于禁用状态。
- 休眠（挂起到磁盘）处于禁用状态。
- 无法访问 `/dev/kmem` 和 `/dev/mem`，连 root 用户也不例外。
- 无法访问 I/O 端口，连 root 用户也不例外。所有 X11 图形驱动程序必须使用内核驱动程序。
- 无法通过 sysfs 访问 PCI BAR。
- 无法使用 ACPI 中的 `custom_method`。
- 无法使用 asus-wmi 模块的 debugfs。
- `acpi_rsdp` 参数对内核没有任何影响。

11.2 更多信息

- http://www.uefi.org — UEFI 主页，其中列出了最新的 UEFI 规范。
- 由 Olaf Kirch 与 Vojtěch Pavlík 撰写的博文（上述章节内容主要取材于这些博文）：
  - http://www.suse.com/blogs/uefi-secure-boot-plan/
  - http://www.suse.com/blogs/uefi-secure-boot-overview/
  - http://www.suse.com/blogs/uefi-secure-boot-details/
- http://en.opensuse.org/openSUSE:UEFI — UEFI 与 openSUSE。

12 引导加载程序 GRUB 2

本章介绍如何配置 SUSE® Linux Enterprise Server 中使用的引导加载程序 GRUB 2。GRUB 是传统 GRUB 引导加载程序（现在称作“GRUB Legacy”）的后继产品。从 SUSE® Linux Enterprise Server 版本 12 开始，就已使用 GRUB 2 作为默认的引导加载程序。产品中提供了一个 YaST 模块来配置最重要的设置。第 10 章 “引导过程简介” 中将引导过程作为一个整体进行了介绍。有关 UEFI 计算机的安全引导支持的细节，请参见第 11 章 “UEFI（统一可扩展固件接口）”。

12.1 GRUB Legacy 与 GRUB 2 之间的主要差异

- 配置储存在不同的文件中。
- 支持更多的文件系统（例如 Btrfs）。
- 可以直接读取 LVM 或 RAID 设备上储存的文件。
- 用户界面可翻译，并可以改变主题。
- 包含一个用于装载模块的机制，以支持更多功能，例如文件系统等。
- 自动搜索和生成其他内核与操作系统（例如 Windows）的引导项。
- 包含一个类似于 Bash 的精简控制台。

12.2 配置文件结构

GRUB 2 的配置基于以下文件：

`/boot/grub2/grub.cfg`

此文件包含 GRUB 2 菜单项的配置。它替代了 GRUB Legacy 中的 `menu.lst`。不要编辑 `grub.cfg` — 它是由命令 `grub2-mkconfig -o /boot/grub2/grub.cfg` 自动生成的。

`/boot/grub2/custom.cfg`

此可选文件在引导时由 `grub.cfg` 直接检索，可用于向引导菜单添加自定义项。从 SUSE Linux Enterprise Server 开始，使用 `grub-once` 时也将分析这些项目。

`/etc/default/grub`

此文件控制 GRUB 2 的用户设置，通常包含背景和主题等其他环境设置。

`/etc/grub.d/` 下的脚本

在执行 `grub2-mkconfig -o /boot/grub2/grub.cfg` 命令期间，将读取此目录中的脚本。主配置文件 `/boot/grub/grub.cfg` 中集成了这些脚本的说明。

`/etc/sysconfig/bootloader`

此配置文件保存一些基本设置，例如引导加载程序类型，或者是否要启用 UEFI 安全引导支持。

`/boot/grub2/x86_64-efi`、`/boot/grub2/power-ieee1275`、`/boot/grub2/s390x`

这些配置文件包含特定于体系结构的选项。

可以通过多种方式控制 GRUB 2。可以在图形菜单（启动屏幕）中选择现有配置的引导项。配置从文件 `/boot/grub2/grub.cfg` 装载，而该文件是基于其他配置文件编译的（参见下文）。所有 GRUB 2 配置文件都被视为系统文件，编辑这些配置文件需要拥有 `root` 特权。

注意：激活配置更改

手动编辑 GRUB 2 配置文件后，需要运行 `grub2-mkconfig -o /boot/grub2/grub.cfg` 以激活更改。但使用 YaST 更改配置时就不需要如此，因为 YaST 会自动运行此命令。

12.2.1 文件 /boot/grub2/grub.cfg

带有引导菜单的图形启动屏幕内容由 GRUB 2 配置文件 `/boot/grub2/grub.cfg` 控制，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信息。

系统每次引导时，GRUB 2 会直接从文件系统装载菜单文件。因此，在更改配置文件后不需要重新安装 GRUB 2。安装或去除内核后，系统会自动重构建 `grub.cfg`。

`grub.cfg` 是运行 `grub2-mkconfig -o /boot/grub2/grub.cfg` 命令时，基于文件 `/etc/default/grub` 以及 `/etc/grub.d/` 目录中的脚本编译的。因此，切勿手动编辑该文件，而应该编辑相关的源文件，或者根据第 12.3 节 “使用 YaST 配置引导加载程序”中所述，使用 YaST 引导加载程序模块来修改配置。

12.2.2 文件 /etc/default/grub

此文件包含 GRUB 2 的其他常规选项，例如，显示菜单的时间，或者要引导的默认操作系统。要列出所有可用选项，请查看以下命令的输出：

```
grep "export GRUB_DEFAULT" -A50 /usr/sbin/grub2-mkconfig | grep GRUB_
```

除了已定义的变量外，用户还可以引入自己的变量，以后在 `/etc/grub.d` 目录中的脚本内使用。

编辑 `/etc/default/grub` 后，请使用 `grub2-mkconfig -o /boot/grub2/grub.cfg` 更新主要配置文件。

注意：范围

此文件中设置的所有选项是会影响所有引导项的常规选项。通过 GRUB_*_XEN_* 配置选项可以设置 Xen 内核或 Xen 超级管理程序的特定选项。有关细节，请参见下文。

`GRUB_DEFAULT`

设置默认会引导的引导菜单项。它的值可以是数字值、菜单项的完整名称或“saved”。

`GRUB_DEFAULT=2` 引导第三个（从零开始计数）引导菜单项。

`GRUB_DEFAULT="2>0"` 引导第三个顶级菜单项的第一个子菜单项。

`GRUB_DEFAULT="Example boot menu entry"` 引导标题为“Example boot menu entry”的菜单项。

`GRUB_DEFAULT=saved` 引导 `grub2-once` 或 `grub2-set-default` 命令指定的项。`grub2-reboot` 只设置下一次重引导的默认引导项，而 `grub2-set-default` 设置发生更改之前的默认引导项。`grub2-editenv list` 列出下一个引导项。

GRUB_HIDDEN_TIMEOUT

等待用户按某个键的指定秒数。在此期间，除非用户按下某个键，否则不显示菜单。如果用户在指定的时间内未按任何键，控制权将转交给 GRUB_TIMEOUT。GRUB_HIDDEN_TIMEOUT=0 首先会检查是否已按下 Shift，如果是，则显示引导菜单，否则会立即引导默认的菜单项。如果 GRUB 2 只识别了一个可引导操作系统，则默认行为就是如此。

GRUB_HIDDEN_TIMEOUT_QUIET

如果指定 false，则在激活了 GRUB_HIDDEN_TIMEOUT 功能时，会在一个空白屏幕上显示倒数计时器。

GRUB_TIMEOUT

在自动引导默认引导项之前，显示引导菜单的期限（以秒为单位）。如果按下某个键，则会取消超时，GRUB 2 将等待您手动做出选择。如果指定 GRUB_TIMEOUT=-1，则在您手动选择引导项之前，会一直显示菜单。

GRUB_CMDLINE_LINUX

此行中的项将添加到正常和恢复模式的引导项的末尾。使用它可以向引导项添加内核参数。

GRUB_CMDLINE_LINUX_DEFAULT

与 GRUB_CMDLINE_LINUX 一样，但只能在正常模式下追加项。

GRUB_CMDLINE_LINUX_RECOVERY

与 GRUB_CMDLINE_LINUX 一样，但只能在恢复模式下追加项。

GRUB_CMDLINE_LINUX_XEN_REPLACE

此项将完全替代所有 Xen 引导项的 GRUB_CMDLINE_LINUX 参数。

GRUB_CMDLINE_LINUX_XEN_REPLACE_DEFAULT

与 GRUB_CMDLINE_LINUX_XEN_REPLACE 一样，但只会替代 GRUB_CMDLINE_LINUX_DEFAULT 的参数。

GRUB_CMDLINE_XEN

此项只为 Xen guest 内核指定内核参数 — 工作原理与 GRUB_CMDLINE_LINUX 相同。

GRUB_CMDLINE_XEN_DEFAULT

与 GRUB_CMDLINE_XEN 一样 — 工作原理与 GRUB_CMDLINE_LINUX_DEFAULT 相同。

`GRUB_TERMINAL`

启用和指定输入/输出终端设备。可以是 `console`（PC BIOS 和 EFI 控制台）、`serial`（串行终端）、`ofconsole`（开放式固件控制台）或默认值 `gfxterm`（图形模式输出）。用引号括住所需的选项可以启用多个设备，例如 `GRUB_TERMINAL="console serial"`。

`GRUB_GFXMODE`

`gfxterm` 图形终端使用的分辨率。请注意，您只能使用您的图形卡 (VBE) 支持的模式。默认值为“auto”，即尝试选择首选的分辨率。在 GRUB 2 命令行中键入 `videoinfo` 可以显示 GRUB 2 适用的屏幕分辨率。GRUB 2 引导菜单屏幕显示后，键入 C 即可访问命令行。

您还可以指定颜色深度，方法是将其值追加到分辨率设置的后面，例如 `GRUB_GFXMODE=1280x1024x24`。

`GRUB_BACKGROUND`

设置 `gfxterm` 图形终端的背景图像。该图像必须是 GRUB 2 在引导时可读的文件，并且必须以 `.png`、`.tga`、`.jpg` 或 `.jpeg` 后缀结尾。必要时，系统会缩放该图像以适合屏幕。

`GRUB_DISABLE_OS_PROBER`

如果将此选项设置为 `true`，将禁用自动搜索其他操作系统的功能。系统只会检测 `/boot/` 中的内核映像，以及 `/etc/grub.d/` 中您自己的脚本内的选项。

`SUSE_BTRFS_SNAPSHOT_BOOTING`

如果将此选项设置为 `true`，GRUB 2 可直接引导至 Snapper 快照。有关详细信息，请参见第 7.3 节 “通过从快照引导来执行系统回滚”。

有关选项的完整列表，请参见 GNU GRUB 手册 (http://www.gnu.org/software/grub/manual/grub.html#Simple-configuration)。有关可用参数的完整列表，请参见 http://en.opensuse.org/Linuxrc。

12.2.3 /etc/grub.d 中的脚本

在执行 `grub2-mkconfig -o /boot/grub2/grub.cfg` 命令期间，将读取此目录中的脚本。脚本中的指令将整合到 `/boot/grub2/grub.cfg` 中。`grub.cfg` 中菜单项的顺序由此目录中文件的运行顺序来决定。具有前导编号的文件先执行，从最小的编号开始。`00_header`

在 10_linux 之前运行，而后者又在 40_custom 之前运行。如果存在采用字母名称的文件，这些文件将在采用编号命名的文件后面执行。在执行 grub2-mkconfig 期间，只有可执行文件才能在 grub.cfg 中生成输出。默认情况下， /etc/grub.d 目录中的所有文件都是可执行文件。

提示：将自定义内容永久保存在 grub.cfg 中

由于每次运行 grub2-mkconfig 时都会重新编译 /boot/grub2/grub.cfg，因此所有自定义内容都会丢失。如果要将您的行直接插入到 /boot/grub2/grub.cfg 中，并且希望在运行 grub2-mkconfig 之后它们不会丢失，请将这些行插入到下面两行之间 the

```
### BEGIN /etc/grub.d/90_persistent ###
```

and

```
### END /etc/grub.d/90_persistent ###
```

lines. 90_persistent 脚本可确保此类内容会保留下来。

下面列出了最重要的脚本：

00_header

设置环境变量，例如系统文件位置、显示设置、主题和以前保存的项。它还可以导入 /etc/default/grub 中储存的首选项。通常，不需要对此文件进行更改。

10_linux

识别根设备上的 Linux 内核，并创建相关的菜单项。这包括关联的恢复模式选项（如果已启用）。主菜单页中只显示最新内核，其他内核包含在子菜单中。

30_os-prober

此脚本使用 OS-prober 来搜索 Linux 和其他操作系统，并将结果放入 GRUB 2 菜单。其中的某些部分可以识别其他特定操作系统，例如 Windows 或 macOS。

40_custom

使用此文件可以方便地在 grub.cfg 中包含自定义引导项。切勿更改开头的 exec tail -n +3 $0 部分。

处理顺序根据前导编号确定，编号最小的脚本最先执行。如果多个脚本的前导编号相同，则根据整个名称的字母顺序来决定执行顺序。

提示：/boot/grub2/custom.cfg

如果您创建了 `/boot/grub2/custom.cfg` 并在其中填充了内容，则引导时系统会自动将它包含到 `/boot/grub40/grub.cfg` 中紧接在 40_custom 后面的位置。

12.2.4 BIOS 驱动器与 Linux 设备之间的映射

在 GRUB Legacy 中，`device.map` 配置文件用于基于 BIOS 驱动器号派生 Linux 设备名称。不一定总能正确猜测出 BIOS 驱动器与 Linux 设备之间的映射。例如，如果 IDE 和 SCSI 驱动器的引导顺序在 BIOS 配置中被交换，则 GRUB Legacy 就会发生顺序错误。

GRUB 2 在生成 `grub.cfg` 时使用设备 ID 字符串 (UUID) 或文件系统标签，因此避免了此问题。GRUB 2 实用程序会即时创建一个临时设备映射，这通常足以满足需要，在单磁盘系统中尤其如此。

但是，如果您需要覆盖 GRUB 2 的自动设备映射机制，请创建自定义映射文件 `/boot/grub2/device.map`。以下示例将更改映射，使 `DISK 3` 成为引导磁盘。请注意，GRUB 2 分区编号以 `1` 开始，而不是像 GRUB Legacy 中那样以 `0` 开始。

```
(hd1)  /dev/disk-by-id/DISK3 ID
(hd2)  /dev/disk-by-id/DISK1 ID
(hd3)  /dev/disk-by-id/DISK2 ID
```

12.2.5 在引导过程中编辑菜单项

当系统由于配置错误而不再能够引导时，如果能够直接编辑菜单项，就会很有帮助。使用菜单编辑器还可以在不更改系统配置的情况下测试新设置。

1. 在图形引导菜单中，使用方向键选择要编辑的项。
2. 按 E 打开基于文本的编辑器。
3. 使用方向键移到您要编辑的行。

图 12.1 GRUB 2 引导编辑器

现在，您可以采取以下两种做法：

a. 将用空格分隔的参数添加到以 `linux` 或 `linuxefi` 开头的行的末尾，以编辑内核参数。http://en.opensuse.org/Linuxrc 上提供了完整的参数列表。

b. 或者编辑常规选项，以更改内核版本等。按 →| 键会显示所有可能的补全建议。

4. 按 F10 使用您所做的更改引导系统，或者按 Esc 放弃您的编辑，并返回到 GRUB 2 菜单。

通过这种方式进行的更改仅会应用到当前引导过程，而不会永久保存。

重要：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。请参见图 41.2 “美式键盘布局”。

注意：安装媒体中的引导加载程序

在使用传统 BIOS 的系统上，安装媒体中的引导加载程序仍是 GRUB Legacy。要添加引导选项，请选择一个项，然后开始键入。在安装引导项中添加的选项将永久保存在安装的系统中。

注意：在 IBM Z 上编辑 GRUB 2 菜单项

IBM Z 上的光标移动方式和编辑命令有所不同，有关细节，请参见第 12.4 节 “IBM Z 上终端使用方式的差异”。

12.2.6 设置引导口令

即使在操作系统引导之前，GRUB 2 也支持对文件系统的访问。没有 root 权限的用户可以访问 Linux 系统中的文件，而在引导系统后，他们将无权访问这些文件。要阻止此类访问或防止用户引导某些菜单项，请设置引导口令。

重要：引导需要口令

如果设置了引导口令，则每次引导时都需要输入该口令，这意味着系统不会自动引导。

按如下步骤设置引导口令。或者使用 YaST（使用口令保护引导加载程序）。

1. 使用 `grub2-mkpasswd-pbkdf2` 来加密口令：

   ```
   tux > sudo grub2-mkpasswd-pbkdf2
   Password: ****
   Reenter password: ****
   PBKDF2 hash of your password is
    grub.pbkdf2.sha512.10000.9CA4611006FE96BC77A...
   ```

2. 将生成的字符串连同 `set superusers` 命令一起粘贴到文件 `/etc/grub.d/40_custom` 中。

   ```
   set superusers="root"
   ```

```
password_pbkdf2 root grub.pbkdf2.sha512.10000.9CA4611006FE96BC77A...
```

3. 要将更改导入到主要配置文件中，请运行：

```
tux > sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

在重引导后，当您尝试引导某个菜单项时，系统会提示您输入用户名和口令。输入 `root` 以及您在执行 `grub2-mkpasswd-pbkdf2` 命令期间键入的口令。如果身份凭证正确，系统将引导选定的引导项。

有关详细信息，请参见https://www.gnu.org/software/grub/manual/grub.html#Security↗。

12.3 使用 YaST 配置引导加载程序

在 SUSE Linux Enterprise Server 系统中，配置引导加载程序一般选项最简单的方式就是使用 YaST 模块。在 YaST 控制中心，选择系统 › 引导加载程序。该模块会显示您系统的当前引导加载程序配置，并允许您进行更改。

使用引导代码选项选项卡可查看和更改类型、位置和高级加载程序设置的相关设置。您可以选择要在标准模式还是 EFI 模式下使用 GRUB 2。

图 12.2 引导代码选项

重要：EFI 系统要求使用 GRUB2-EFI

如果您使用的是 EFI 系统，则只能安装 GRUB2-EFI，否则您的系统不再能够引导。

重要：重新安装引导加载程序

要重新安装引导加载程序，请务必在 YaST 中更改设置，然后再将其改回来。例如，要重新安装 GRUB2-EFI，请先选择 GRUB2，然后立即将其切换回 GRUB2-EFI。

否则，可能只会重新安装引导加载程序的一部分。

注意：自定义引导加载程序

要使用此处未列出的引导加载程序，请选择不要安装任何引导加载程序。在选择该选项之前，请仔细阅读您的引导加载程序文档。

12.3.1 引导加载程序位置和引导代码选项

引导加载程序的默认位置为主引导记录 (MBR) 或 / 分区的引导扇区，具体视分区设置而定。要修改引导加载程序的位置，请遵循以下步骤：

过程 12.1 更改引导加载程序位置

1. 选择引导代码选项选项卡，然后为引导加载程序位置选择以下某个选项：

   从主引导记录引导

   选择此选项将在包含 /boot 目录的磁盘 MBR 中安装引导加载程序。通常，这将是装入到 / 的磁盘，但如果 /boot 装入到其他磁盘上的独立分区中，则将会使用该磁盘的 MBR。

   从根分区引导

   这将在 / 分区的引导扇区安装引导加载程序。

   自定义引导分区

   手动使用此选项来指定引导加载程序的位置。

2. 单击确定以应用您的更改。

图 12.3 代码选项

引导代码选项选项卡包含以下额外的选项：

在分区表中设置活动标志，以引导分区

激活包含 `/boot` 目录的分区（PReP 分区）。此选项用于具有旧 BIOS 的系统和/或旧式操作系统，因为它们可能无法从非活动的分区引导。您可以放心地启用此选项。

将通用引导代码写入 MBR 中

如果 MBR 包含自定义的“非 GRUB”代码，此选项会用不受操作系统限制的通用代码替换该代码。如果您停用此选项，系统可能变得无法引导。

启用可信引导支持

启动支持可信计算功能（可信平台模块 (TPM)）的 TrustedGRUB2。有关详细信息，请参见 https://github.com/Sirrix-AG/TrustedGRUB2 。

12.3.2 调整磁盘顺序

如果您的计算机有多个硬盘，您可以指定磁盘的引导顺序。如果从 MBR 引导，将在列表中的第一个磁盘中安装 GRUB 2。默认在该磁盘中安装 SUSE Linux Enterprise Server。列表的其余部分是有关 GRUB 2 的设备映射程序的提示（请参见第 12.2.4 节 “BIOS 驱动器与 Linux 设备之间的映射”）。

警告：无法引导的系统

通常情况下，默认值几乎对所有部署都有效。如果您错误地更改了磁盘的引导顺序，系统下次重引导时可能无法引导。例如，如果列表中的第一个磁盘不在 BIOS 引导序列中，并且列表中的其他磁盘有空 MBR，系统将无法引导。

过程 12.2 设置磁盘顺序

1. 打开引导代码选项选项卡。
2. 单击编辑磁盘引导顺序。
3. 如果列出了多个磁盘，请选择一个，然后单击向上或向下来对显示的磁盘重新排序。
4. 单击确定两次以保存更改。

12.3.3 配置高级选项

高级引导选项可以通过引导加载程序选项选项卡来配置。

12.3.3.1 引导加载程序选项选项卡

图 12.4 引导加载程序选项

引导加载程序超时

通过键入新值或者用鼠标单击相应的方向键来更改超时（以秒为单位）的值。

探测外来操作系统

如果选择该选项，引导加载程序将会搜索其他系统（例如 Windows）或其他 Linux 安装。

引导时隐藏菜单

隐藏引导菜单并引导默认项。

调整默认引导项

从“默认引导部分”列表中选择所需的项。请注意，引导项名称中的“>”符号用于分隔引导部分及其子部分。

使用口令保护引导加载程序

使用一个附加的口令保护引导加载程序和系统。有关详细信息，请参见 第 12.2.6 节 “设置引导口令”。

12.3.3.2 内核参数选项卡

图 12.5 内核参数

控制台分辨率

控制台分辨率选项指定引导过程中的默认屏幕分辨率。

内核命令行参数

可选内核参数添加在默认参数的末尾。有关所有可用参数的列表，请参见 http://en.opensuse.org/Linuxrc。

CPU 缓解措施

SUSE 针对为防御 CPU 边信道攻击而部署的所有软件缓解措施发布了一个或多个内核引导命令行参数。其中一些参数可能会导致性能损失。请根据您的设置选择以下其中一个选项，在安全性和性能之间取得平衡：

自动： 启用您的 CPU 型号所需的全部缓解措施，但不防范跨 CPU 线程的攻击。此设置可能会在一定程度上影响性能，具体视工作负载而定。

自动 + 禁用 SMT： 提供整套可用的安全缓解措施。启用您的 CPU 型号所需的全部缓解措施。此外，它还会禁用同步多线程 (SMT)，以免受到跨多个 CPU 线程发起的边信道攻击。此设置可能会进一步影响性能，具体取决于工作负载。

关： 禁用所有缓解措施。CPU 可能会遭到边信道攻击，具体取决于 CPU 型号。此设置不会对性能造成任何影响。

手动： 不设置任何缓解措施级别。请使用内核命令行选项手动指定 CPU 缓解措施。

使用图形控制台

如果选中该选项，引导菜单会显示在图形启动屏幕中，而不是以文本模式显示。此时，您可以通过控制台分辨率列表设置引导屏幕的分辨率，并使用控制台主题文件选择器指定图形主题定义文件。

使用串行控制台

如果您的计算机是通过串行控制台控制的，则可激活此选项并指定要使用的 COM 端口及其运行速度。请参见 `info grub` 或 http://www.gnu.org/software/grub/manual/grub.html#Serial-terminal

12.4 IBM Z 上终端使用方式的差异

在 3215 和 3270 终端上，光标的移动方式以及在 GRUB 2 中发出编辑命令的方式存在一些差异和限制。

12.4.1 限制

交互能力

交互性存在严重的限制。键入时通常不能获得直观的反馈。要查看光标所在的位置，请键入下划线 (_)。

注意：3270 与 3215 的对比

与 3215 终端相比，3270 终端在显示和刷新屏幕方面要好得多。

光标的移动

无法进行“传统的”光标移动操作。 Alt 、 Meta 、 Ctrl 和光标键不起作用。要移动光标，请使用第 12.4.2 节 “组合键”中列出的组合键。

加字符

插入符 (^) 用作控制字符。要键入文本 ^ 后再键入一个字母，请键入 ^、 ^ 和字母。

输入

Enter 键不起作用，请改用 ^-J。

12.4.2 组合键

常用的替代键	^-J	确认（“Enter”）
	^-L	中止，返回前一“状态”
	^-I	Tab 键补全（在编辑模式与外壳模式下）
菜单模式下可用的键：	^-A	移到第一项
	^-E	移到最后一项
	^-P	移到上一项
	^-N	移到下一项
	^-G	上移一页
	^-C	下移一页
	^-F	引导选定的项或进入子菜单（与 ^-J 的作用相同）
	E	编辑选定的项
	C	进入 GRUB-Shell
编辑模式下可用的键：	^-P	上移一行

	^ + N	下移一行
	^ + B	向左移一个字符
	^ + F	向右移一个字符
	^ + A	移到行首
	^ + E	移到行尾
	^ + H	退格
	^ + D	删除
	^ + K	删除光标起当前行
	^ + Y	复制
	^ + O	插入空行
	^ + L	刷新屏幕
	^ + X	引导项
	^ + C	进入 GRUB-Shell
命令行模式下可用的键：	^ + P	上一个命令
	^ + N	历史中的下一个命令
	^ + A	移到行首
	^ + E	移到行尾
	^ + B	向左移一个字符
	^ + F	向右移一个字符
	^ + 高	退格

	^ + D	删除
	^ + K	删除光标起当前行
	^ + U	删除行
	^ + Y	复制

12.5 有用的 GRUB 2 命令

`grub2-mkconfig`

基于 `/etc/default/grub` 以及 `/etc/grub.d/` 中的脚本生成新的 `/boot/grub2/grub.cfg`。

例 12.1 GRUB2-MKCONFIG 用法

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

提示：语法检查

运行不带任何参数的 `grub2-mkconfig` 会将配置打印到 STDOUT 中供用户查看。在写入 `/boot/grub2/grub.cfg` 后使用 `grub2-script-check` 可以检查其语法。

重要：`grub2-mkconfig` 无法修复 UEFI 安全引导表

如果要使用 UEFI 安全引导且系统无法再正确连接 GRUB 2，您可能需要另外重新安装 Shim 并重新生成 UEFI 引导表。要执行此操作，请使用：

```
root # shim-install --config-file=/boot/grub2/grub.cfg
```

`grub2-mkrescue`

为安装的 GRUB 2 配置创建一个可引导的救援映像。

例 12.2 GRUB2-MKRESCUE 用法

```
grub2-mkrescue -o save_path/name.iso iso
```

`grub2-script-check`

检查给定文件中有无语法错误。

例 12.3 GRUB2-SCRIPT-CHECK 用法

```
grub2-script-check /boot/grub2/grub.cfg
```

`grub2-once`

仅设置下次引导的默认引导项。要获取可用引导项的列表，请使用 `--list` 选项。

例 12.4 GRUB2-ONCE 用法

```
grub2-once number_of_the_boot_entry
```

提示：`grub2-once` 帮助

不使用任何选项调用该程序可以获取所有可用选项的完整列表。

12.6 更多信息

http://www.gnu.org/software/grub/ 上提供了有关 GRUB 2 的大量信息。还请参见 `grub` 信息页面。您也可以在技术信息搜索 (“) 中搜索关键字”GRUB 2http://www.suse.com/support，以获取有关特殊问题的信息。

13 systemd 守护程序

systemd 程序是进程 ID 为 1 的进程。它负责以所需的方式初始化系统。systemd 由内核直接启动，信号 9（该信号通常会终止进程）对它不起作用。所有其他程序都由 systemd 直接启动，或由它的其中一个子进程启动。

从 SUSE Linux Enterprise Server 12 起，systemd 取代了常用的 System V init 守护程序。systemd 与 System V init 完全兼容（通过支持 init 脚本实现）。systemd 的其中一个主要优点是可通过激进方式并行启动多项服务来大幅加快引导速度。另外，systemd 只在确实有需要时才会启动服务。守护程序并不是在系统引导时无条件地启动，而是在第一次需要时启动。systemd 还支持内核控制组 (cgroups)，以拍摄系统状态快照以及恢复系统状态等等。有关详细信息，请参见http://www.freedesktop.org/wiki/Software/systemd/。

13.1 systemd 概念

本节将详细介绍有关 systemd 的概念。

13.1.1 systemd 是什么

systemd 是适用于 Linux 的系统和会话管理器，它与 System V 及 LSB init 脚本兼容。主要功能包括：

- 提供激进式并行化功能
- 使用套接字及 D-Bus 激活来启动服务
- 提供按需启动守护程序功能
- 使用 Linux cgroups 跟踪进程
- 支持拍摄系统状态快照及恢复系统状态
- 维护安装点和自动安装点
- 实施精细的、基于事务依赖项的服务控制逻辑

13.1.2 单元文件

单元配置文件包含有关以下项目的信息：服务、套接字、设备、安装点、自动安装点、交换文件或分区、启动目标、监控的文件系统路径、受 systemd 控制和监管的计时器、临时系统状态快照、资源管理部分或一组外部创建的进程。“单元文件”是 systemd 用于描述下列各项的通用术语：

- 服务： 进程相关信息（例如运行守护程序）；文件以 .service 结尾
- 目标： 用于将单元分组以及在启动期间用作同步点；文件以 .target 结尾
- 套接字： IPC 或网络套接字或文件系统 FIFO 相关信息，适用于基于套接字的激活（如 `inetd`）；文件以 .socket 结尾
- 路径： 用于触发其他单元（例如，在文件更改时运行服务）；文件以 .path 结尾
- 计时器： 受控制计时器相关信息，适用于基于计时器的激活；文件以 .timer 结尾
- 装入点： 通常由 fstab 生成器自动生成；文件以 .mount 结尾
- 自动安装点： 文件系统自动安装点相关信息；文件以 .automount 结尾
- 交换： 内存分页的交换设备或文件相关信息；文件以 .swap 结尾
- 设备： sysfs/udev(7) 设备树中公开的设备相关信息；文件以 .device 结尾
- 范围/部分： 有关分层管理一组进程的资源的概念；文件以 .scope/.slice 结尾

有关 systemd.unit 的更多信息，请参见 http://www.freedesktop.org/software/systemd/man/systemd.unit.html ↗

13.2 基本用途

System V init 系统使用若干命令来处理服务：init 脚本、`insserv`、`telinit` 及其他。systemd 使服务管理变得简单，要运行大部分服务处理任务，只需要记住一条命令：`systemctl`。它使用“命令加子命令”表示法，与 `git` 或 `zypper` 相似：

```
systemctl GENERAL OPTIONS SUBCOMMAND SUBCOMMAND OPTIONS
```

有关完整的手册，请参见 `man 1 systemctl`。

提示：终端输出和 Bash 补全

如果输出传递到某个终端（而不是某个管道或文件），systemd 命令默认会将长输出发送到分页器。使用 `--no-pager` 选项可关闭分页模式。

systemd 还支持 bash 补全，允许您输入子命令的头几个字母，然后按 →| 自动补全子命令。此功能只能在 `bash` 外壳中使用，并且需要安装 `bash-completion` 包。

13.2.1 管理正在运行的系统中的服务

用于管理服务的子命令与使用 System V init 管理服务的子命令相同（`start`、`stop` 等）。服务管理命令的一般语法如下所示：

systemd

```
systemctl reload|restart|start|status|stop|... MY_SERVICE(S)
```

System V init

```
rcMY_SERVICE(S) reload|restart|start|status|stop|...
```

systemd 允许您一次管理多个服务。不用像 System V init 要逐个执行 init 脚本，而是执行如下命令：

```
systemctl start MY_1ST_SERVICE MY_2ND_SERVICE
```

要列出系统上所有可用的服务，请运行：

```
systemctl list-unit-files --type=service
```

下表列出了 systemd 和 System V init 最重要的服务管理命令：

表 13.1 服务管理命令

任务	systemd 命令	System V init 命令
启动：	`start`	`start`

任务	systemd 命令	System V init 命令
停止：	`stop`	`stop`
重启动： 关闭服务，然后再启动这些服务。如果某项服务尚未运行，它将会启动。	`restart`	`restart`
有条件地重启动： 如果服务当前正在运行，则重启动它们。对未在运行的服务不执行任何操作。	`try-restart`	`try-restart`
重新装载： 让服务在不中断操作的情况下重新装载它们的配置文件。使用案例：让 Apache 重新装载修改过的 `httpd.conf` 配置文件。请注意，并非所有服务都支持重新装载。	`reload`	`reload`
重新装载或重启动： 如果支持重新装载则重新装载服务，否则重启动服务。如果某项服务尚未运行，它将会启动。	`reload-or-restart`	`n/a`
有条件地重新装载或重启动： 如果支持重新装载则重新装载服务，否则，如果服务当前正在运行，则重启动服务。对未在运行的服务不执行任何操作。	`reload-or-try-restart`	`n/a`
获得详细的状态信息： 列出服务状态的相关信息。 `systemd` 命令会显示说明、可执行文件、状态、cgroup 及服务发出的最新讯息等细节（请参见第 13.6.9 节 “调试服务”）。使用 System V init 显示的详细程度因服务而异。	`status`	`status`
获得简要的状态信息： 显示服务是否处于活动状态。	`is-active`	`status`

13.2.2 永久启用/禁用服务

上一节中提到的服务管理命令可让您操控当前会话的服务。systemd 还允许您永久启用或禁用服务，让它们在用户要求时自动启动或永远不可用。您可以使用 YaST 或在命令行上运行命令来实现此目的。

13.2.2.1 在命令行上启用/禁用服务

下表列出了 systemd 和 System V init 用于启用和禁用服务的命令：

重要：启动服务

在命令行上启用服务时，服务不会自动启动。它会安排在下一次系统启动或运行级别/目标发生更改时启动。要在启用服务之后立即启动它，请显式运行 `systemctl start MY_SERVICE` 或 `rc MY_SERVICE start`。

表 13.2 用于启用和禁用服务的命令

任务	`systemd` 命令	System V init 命令
启用：	`systemctl enable MY_SERVICE(S)`	`insserv MY_SERVICE(S)`，`chkconfig -a MY_SERVICE(S)`
禁用：	`systemctl disable MY_SERVICE(S).service`	`insserv -r MY_SERVICE(S)`、`chkconfig -d MY_SERVICE(S)`
检查： 显示某项服务是否启用。	`systemctl is-enabled MY_SERVICE`	`chkconfig MY_SERVICE`
重新启用： 与重启动服务相似，此命令先禁用服务，	`systemctl reenable MY_SERVICE`	无

任务	`systemd` 命令	System V init 命令
然后再启用该服务。可用来使用默认值重新启用服务。		
屏蔽： “禁用”某项服务之后，仍然可以手动启动它。要彻底禁用服务，您需要屏蔽它。须谨慎使用该功能。	`systemctl mask MY_SERVICE`	无
取消屏蔽： 屏蔽某项服务之后，只有在将其取消屏蔽之后才能再次使用它。	`systemctl unmask MY_SERVICE`	无

13.3 系统启动和目标管理

启动和关闭系统的整个过程由 systemd 负责维护。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。

13.3.1 目标与运行级别的比较

使用 System V init 时，系统引导到所谓的“运行级别”。运行级别定义系统如何启动以及正在运行的系统中有哪些服务可用。运行级别是有编号的；最知名的运行级别是 `0`（关闭系统）、`3`（联网的多用户模式）和 `5`（联网并使用显示管理器的多用户模式）。

systemd 使用所谓的“目标单元”引入新的概念。不过，它仍然与运行级别概念完全兼容。目标单元用名称而不是编号来标识，并具有特定的作用。例如，目标 `local-fs.target` 和 `swap.target` 用于装入本地文件系统和交换空间。

目标 `graphical.target` 提供联网并使用显示管理器的多用户系统，作用与运行级别 5 相当。复杂目标（如 `graphical.target`）通过将一部分其他目标组合起来充当“元”目标。systemd 通过组合现有目标简化了创建自定义目标的工作，从而提供了极大的灵活性。

下面的列表显示了最重要的 systemd 目标单元。有关完整列表，请参见 `man 7 systemd.special`。

选定的 SYSTEMD 目标单元

`default.target`
: 默认引导的目标。这并不是“实际”目标，而是指向另一个目标（如 `graphic.target`）的符号链接，可通过 YaST 永久更改（请参见第 13.4 节 “使用 YaST 管理 服务”）。要为某个会话更改它，请在引导提示处使用内核参数 `systemd.unit=MY_TARGET.target`。

`emergency.target`
: 在控制台上启动紧急外壳。请仅在引导提示处按以下格式使用它：`systemd.unit=emergency.target`。

`graphical.target`
: 启动联网的具有多用户支持和显示管理器的系统。

`halt.target`
: 关闭系统。

`mail-transfer-agent.target`
: 启动发送和接收邮件所需的所有服务。

`multi-user.target`
: 启动联网的多用户系统。

`reboot.target`
: 重新引导系统。

`rescue.target`
: 启动不联网的单用户系统。

为了与 System V init 运行级别系统保持兼容，systemd 提供了名为 `runlevelX.target` 的特殊目标，与编号为 `X` 的相应运行级别相对应。

如果您想知道当前的目标，请使用命令：`systemctl get-default`

表 13.3 SYSTEM V 运行级别和 systemd 目标单元

System V 运行级别	systemd 目标	用途
0	runlevel0.target、halt.target、poweroff.target	系统关闭
1、S	runlevel1.target、rescue.target	单用户模式
2	runlevel2.target、multi-user.target	无远程联网的本地多用户模式
3	runlevel3.target、multi-user.target	完整的联网多用户模式
4	runlevel4.target	未使用/用户定义
5	runlevel5.target、graphical.target	联网并使用显示管理器的完整多用户模式
6	runlevel6.target、reboot.target	系统重引导

重要：systemd 会忽略 /etc/inittab

System V init 系统中的运行级别在 /etc/inittab 中配置。systemd 不使用此配置。有关如何创建您自己的可引导目标的指导，请参考第 13.5.3 节 “创建自定义目标”。

13.3.1.1 用于更改目标的命令

可使用下列命令来操作目标单元：

任务	systemd 命令	System V init 命令
更改当前目标/运行级别	`systemctl isolate` `MY_TARGET`.target	`telinit` `X`
更改为默认目标/运行级别	`systemctl default`	无
获得当前目标/运行级别	`systemctl list-units --type=target` 使用 systemd 时，一般会有多个活动目标。该命令可列出当前处于活动状态的所有目标。	`who -r` 或者 运行级别
永久更改默认运行级别	使用服务管理器或运行以下命令： `ln -sf /usr/lib/systemd/system/` `MY_TARGET`.target /etc/systemd/system/default.target	使用服务管理器或更改以下行 `id:` `X`:initdefault: （`/etc/inittab` 中）
更改当前引导进程的默认运行级别	在引导提示处输入以下选项 `systemd.unit=` `MY_TARGET`.target	在引导提示处输入所需的运行级别编号。
显示目标/运行级别的依赖项	`systemctl show -p "Requires"` `MY_TARGET`.target `systemctl show -p "Wants"` `MY_TARGET`.target “Requires” 会列出硬性依赖项（必须解决的依赖项），而 “Wants” 会列出软性依赖项（情况允许时解决的依赖项）。	无

13.3.2 调试系统的启动

systemd 提供了分析系统启动过程的方法。您可以查看所有服务及其状态的列表（而不必分析 `/varlog/`）。systemd 还允许您扫描启动过程，以了解每项服务启动用了多长时间。

13.3.2.1 查看服务的启动情况

要查看系统引导后所启动服务的完整列表，请输入命令 `systemctl`。该命令会列出所有活动服务，如下所示（已精简）。要获得特定服务的详细信息，请使用 `systemctl status MY_SERVICE`。

例 13.1 列出活动服务

```
root # systemctl
UNIT                          LOAD   ACTIVE SUB       JOB DESCRIPTION
[...]
iscsi.service                 loaded active exited    Login and scanning of iSC+
kmod-static-nodes.service     loaded active exited    Create list of required s+
libvirtd.service              loaded active running   Virtualization daemon
nscd.service                  loaded active running   Name Service Cache Daemon
ntpd.service                  loaded active running   NTP Server Daemon
polkit.service                loaded active running   Authorization Manager
postfix.service               loaded active running   Postfix Mail Transport Ag+
rc-local.service              loaded active exited    /etc/init.d/boot.local Co+
rsyslog.service               loaded active running   System Logging Service
[...]
LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

161 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
```

要想只列出无法启动的服务，请使用 `--failed` 选项。

例 13.2 列出失败的服务

```
root # systemctl --failed
```

```
UNIT                    LOAD   ACTIVE SUB    JOB DESCRIPTION
apache2.service         loaded failed failed     apache
NetworkManager.service loaded failed failed     Network Manager
plymouth-start.service loaded failed failed     Show Plymouth Boot Screen

[...]
```

13.3.2.2 调试启动时间

为了调试系统启动时间，systemd 提供了 `systemd-analyze` 命令。该命令会显示总启动时间及按启动时间排序的服务列表，还可以生成 SVG 图，以显示各服务相对于其他服务的启动时间。

列出系统启动时间

```
root # systemd-analyze
Startup finished in 2666ms (kernel) + 21961ms (userspace) = 24628ms
```

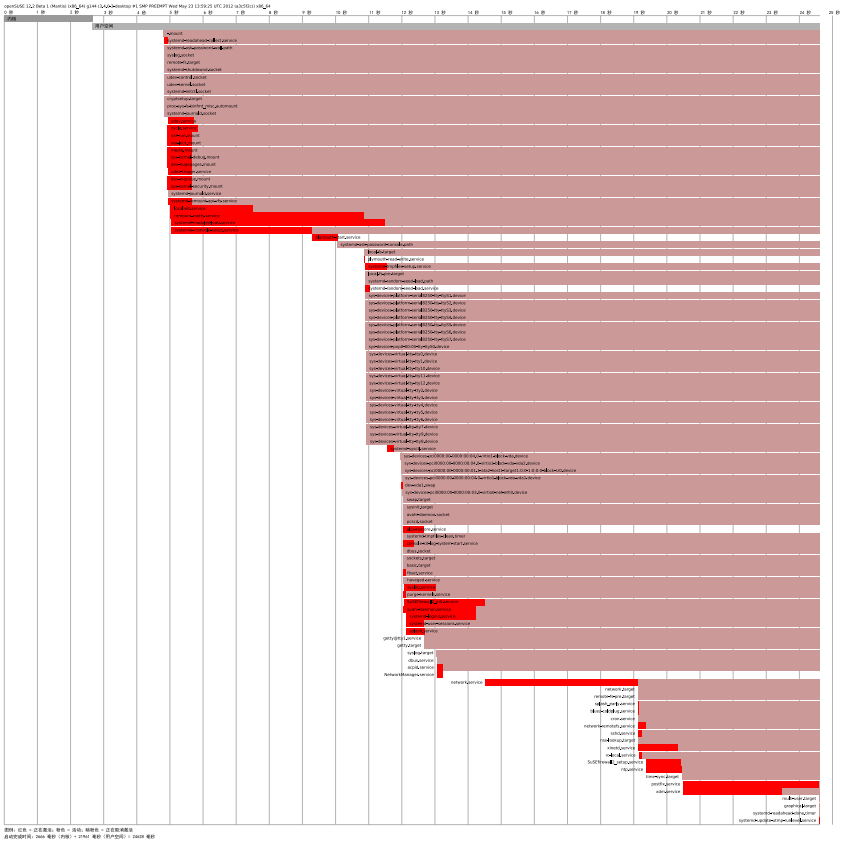
列出服务启动时间

```
root # systemd-analyze blame
  6472ms systemd-modules-load.service
  5833ms remount-rootfs.service
  4597ms network.service
  4254ms systemd-vconsole-setup.service
  4096ms postfix.service
  2998ms xdm.service
  2483ms localnet.service
  2470ms SuSEfirewall2_init.service
  2189ms avahi-daemon.service
  2120ms systemd-logind.service
  1210ms xinetd.service
  1080ms ntp.service
[...]
    75ms fbset.service
    72ms purge-kernels.service
    47ms dev-vda1.swap
    38ms bluez-coldplug.service
```

```
    35ms splash_early.service
```

服务启动时间图

```
root # systemd-analyze plot > jupiter.example.com-startup.svg
```



13.3.2.3 查看完整的启动过程

上面提到的命令可让您查看已启动的服务以及启动各服务所需的时间。如果您需要知道更多细节，可以在引导提示处输入下列参数，让 `systemd` 详细记录整个启动过程：

```
systemd.log_level=debug systemd.log_target=kmsg
```

现在，`systemd` 会将日志讯息写入内核环缓冲区。使用 `dmesg` 查看该缓冲区：

```
dmesg -T | less
```

13.3.3 System V 兼容性

systemd 与 System V 兼容，因此，您仍可以使用现有的 System V init 脚本。但是，至少有一个已知问题会导致 System V init 脚本默认不能与 systemd 配合使用：在 init 脚本中通过 `su` 或 `sudo` 以其他用户身份启动服务会导致脚本失败，生成“访问被拒绝”错误。

使用 `su` 或 `sudo` 更改用户时，会启动 PAM 会话。完成 init 脚本后会终止此会话。因此，init 脚本启动的服务也会终止。要解决此问题，请执行以下步骤：

1. 创建与 init 脚本同名、扩展名为 `.service` 的服务文件封装程序：

```
[Unit]
Description=DESCRIPTION
After=network.target

[Service]
User=USER
Type=forking (1)
PIDFile=PATH TO PID FILE (1)
ExecStart=PATH TO INIT SCRIPT start
ExecStop=PATH TO INIT SCRIPT stop
ExecStopPost=/usr/bin/rm -f PATH TO PID FILE (1)

[Install]
WantedBy=multi-user.target (2)
```

   将 `UPPERCASE LETTERS` 中写入的所有值替换为适当的值。

   (1) 可选 — 仅当 init 脚本启动守护程序时才使用。

   (2) `multi-user.target` 在引导进入 `graphical.target` 时也会启动 init 脚本。如果只应在引导进入显示管理器时才将它启动，请在此处使用 `graphical.target`。

2. 使用 `systemctl start 应用程序` 启动守护程序。

13.4 使用 YaST 管理 服务

基本服务管理也可以通过 YaST 服务管理器模块实现。该模块支持启动、停止、启用和禁用服务。它还可让您显示服务的状态以及更改默认目标。要启动 YaST 模块，请选择 YaST › 系统 › 服务管理器。

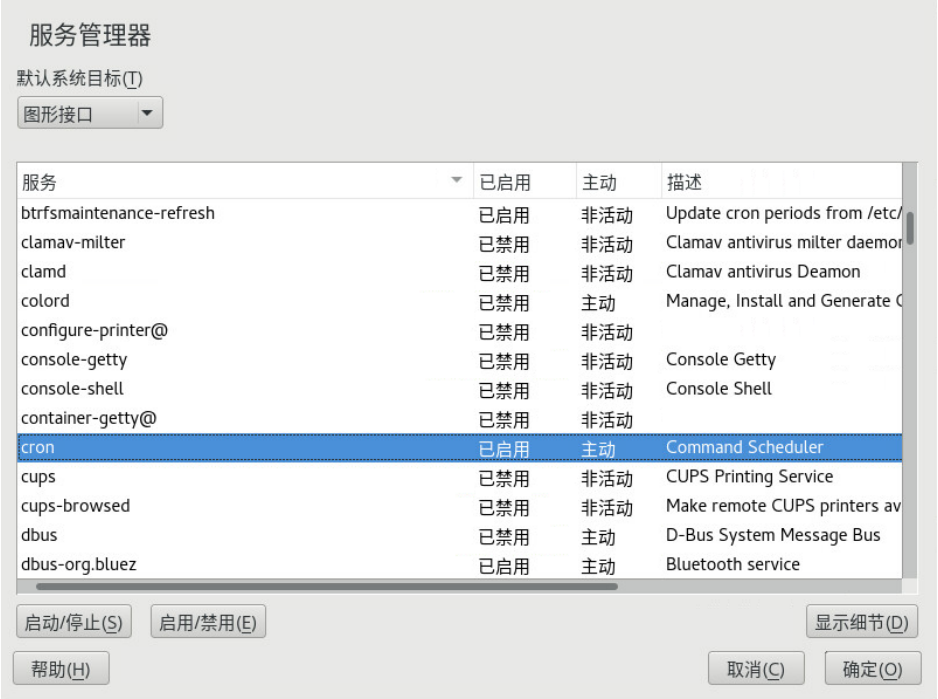


图 13.1 服务管理器

更改默认系统目标

要更改系统引导到的目标，请从默认系统目标下拉框中选择某个目标。最常用的目标是图形界面（启动图形登录屏幕）和多用户（以命令行模式启动系统）。

启动或停止服务

从表中选择一个服务。活动列显示它当前是（活动）否（非活动）正在运行。通过选择启动/停止可切换其状态。

为当前正在运行的会话启动或停止服务会更改其状态。要更改服务在整个重引导过程中的状态，您需要启用或禁用服务。

启用或禁用服务

从表中选择一个服务。已启用列显示它当前已启用还是已禁用。通过选择启用/禁用可更改其状态。

通过启用或禁用服务，可配置服务在引导期间是否启动（已启用或已禁用）。此设置不会影响当前的会话。要更改服务在当前会话中的状态，您需要将其启动或停止。

查看状态讯息

要查看某个服务的状态讯息，请从列表中选择该服务，然后选择显示细节。您看到的输出与 `systemctl -l status MY_SERVICE` 命令生成的输出完全相同。

警告：有问题的运行级别设置可能会对您的系统造成损害

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。

13.5 systemd 自定义

以下几节介绍了 `systemd` 自定义的一些示例。

警告：避免自定义被覆盖

一律在 `/etc/systemd/` 中进行 systemd 自定义设置，切勿在 `/usr/lib/systemd/` 中进行。否则，您的更改将在 systemd 下次更新时被覆盖。

13.5.1 自定义服务文件

systemd 服务文件位于 `/usr/lib/systemd/system` 中。如果您要自定义服务文件，请执行下列步骤：

1. 将要修改的文件从 `/usr/lib/systemd/system` 复制到 `/etc/systemd/system`。将文件名保持不变。
2. 根据需要修改 `/etc/systemd/system` 中的副本。
3. 如需配置更改的概述，请使用 `systemd-delta` 命令。它会比较并识别覆盖了其他配置文件的配置文件。有关细节，请参考 `systemd-delta` 手册页。

`/etc/systemd` 中修改过的文件优先于 `/usr/lib/systemd/system` 中的原始文件，前提是它们的文件名相同。

13.5.2 创建“插入式”文件

如果您只想在配置文件中添加几行或修改文件的一小部分，可以使用所谓的“插入式”文件。使用插入式文件，您无需编辑或覆盖单元文件本身，即可扩展单元文件的配置。

例如，要更改位于 `/usr/lib/systemd/system/FOOBAR.SERVICE` 中 `FOOBAR` 服务的一个值，请按以下步骤操作：

1. 创建名为 `/etc/systemd/system/MY_SERVICE.service.d/` 的目录。
   注意 `.d` 后缀。该目录必须命名为与要用插入式文件修补的服务相似的名称。
2. 在该目录中，创建 `WHATEVERMODIFICATION.conf` 文件。
   确保文件中仅包含要修改的值所在的那一行。
3. 将更改保存到 文件中。它将作为原始文件的扩展。

13.5.3 创建自定义目标

System V init SUSE 系统上未使用运行级别 4，以便允许管理员创建自己的运行级别配置。systemd 允许您创建任意数目的自定义目标。建议您从采用 `graphical.target` 等现有目标开始。

1. 将配置文件 `/usr/lib/systemd/system/graphical.target` 复制到 `/etc/systemd/system/MY_TARGET.target`，并根据需要调整该文件。
2. 上一步中复制的配置文件已涵盖目标的必要（“硬性”）依赖项。如果还要涵盖需要的（“软性”）依赖项，请创建目录 `/etc/systemd/system/MY_TARGET.target.wants`。
3. 对每个需要的服务，创建从 `/usr/lib/systemd/system` 链到 `/etc/systemd/system/我的目标.target.wants` 的符号链接。
4. 设置好目标后，重新装载 systemd 配置以让新目标可用。

   ```
   systemctl daemon-reload
   ```

13.6 高级用途

以下几节介绍了适合系统管理员的高级主题。有关更高级的 systemd 文档，请参考 Lennart Pöttering 撰写的适合管理员的 systemd 相关系列，网址为 http://0pointer.de/blog/projects↗。

13.6.1 清理临时目录

`systemd` 支持定期清理临时目录。将会自动迁移并激活前一系统版本中的配置。`tmpfiles.d`（负责管理临时文件）将从 `/etc/tmpfiles.d/*.conf`、`/run/tmpfiles.d/*.conf` 和 `/usr/lib/tmpfiles.d/*.conf` 文件中读取其配置。`/etc/tmpfiles.d/*.conf` 中的配置将覆盖其他两个目录中的相关配置（`/usr/lib/tmpfiles.d/*.conf` 是包用于储存其配置文件的位置）。

配置格式为每个路径一行，该行包含操作与路径、（可选）模式、所有权、期限和自变量字段，具体取决于操作。以下示例将取消链接 X11 锁定文件：

```
Type Path               Mode UID  GID  Age Argument
r    /tmp/.X[0-9]*-lock
```

要获取 tmpfile 计时器的状态：

```
systemctl status systemd-tmpfiles-clean.timer
systemd-tmpfiles-clean.timer - Daily Cleanup of Temporary Directories
 Loaded: loaded (/usr/lib/systemd/system/systemd-tmpfiles-clean.timer; static)
 Active: active (waiting) since Tue 2014-09-09 15:30:36 CEST; 1 weeks 6 days ago
   Docs: man:tmpfiles.d(5)
         man:systemd-tmpfiles(8)

Sep 09 15:30:36 jupiter systemd[1]: Starting Daily Cleanup of Temporary
 Directories.
Sep 09 15:30:36 jupiter systemd[1]: Started Daily Cleanup of Temporary
 Directories.
```

有关处理临时文件的详细信息，请参见 `man 5 tmpfiles.d`。

13.6.2 系统日志

第 13.6.9 节 “调试服务” 介绍如何查看给定服务的日志讯息。但可显示的日志讯息并不仅限于服务日志。您还可以访问和查询 `systemd` 写入的完整日志，即所谓的“日记”。使用 `journalctl` 命令可显示完整的日志讯息，从最早的项开始。有关诸如应用过滤器或更改输出格式的选项，请参考 `man 1 journalctl`。

13.6.3 快照

您可以使用 `isolate` 子命令将 `systemd` 的当前状态保存到指定快照中，并在日后还原到该状态。此功能在测试服务或自定义目标时非常有用，因为它可让您随时回到定义的状态。快照仅在当前会话中可用，重引导时将自动删除。快照名称必须以 `.snapshot` 结尾。

创建快照

```
systemctl snapshot MY_SNAPSHOT.snapshot
```

删除快照

```
systemctl delete MY_SNAPSHOT.snapshot
```

查看快照

```
systemctl show MY_SNAPSHOT.snapshot
```

激活快照

```
systemctl isolate MY_SNAPSHOT.snapshot
```

13.6.4 装载内核模块

通过使用 `systemd` 以及 `/etc/modules-load.d` 中的配置文件，可以在引导时自动装载内核模块。该文件应该命名为 `MODULE.conf` 并包含以下内容：

```
# load module MODULE at boot time
```

```
MODULE
```

如果某个包安装了用于装载内核模块的配置文件，该文件将安装到 `/usr/lib/modules-load.d`。如果存在两个同名的配置文件，将优先使用 `/etc/modules-load.d` 中的那个配置文件。

有关详细信息，请参见 `modules-load.d(5)` 手册页。

13.6.5 装载服务之前执行必要操作

使用 System V init 时，需要在装载服务之前执行的操作必须在 `/etc/init.d/before.local` 中指定。systemd 不再支持此过程。如果您需要在启动服务之前执行操作，请执行以下步骤：

装载内核模块

在 `/etc/modules-load.d` 目录中创建一个 drop-in 文件（有关语法，请参见 `man modules-load.d`）

创建文件或目录、清理目录、更改所有权

在 `/etc/tmpfiles.d` 中创建一个 drop-in 文件（有关语法，请参见 `man tmpfiles.d`）

其它任务

基于下面的模板创建一个系统服务文件，例如 `/etc/systemd/system/before.service`：

```
[Unit]
Before=NAME OF THE SERVICE YOU WANT THIS SERVICE TO BE STARTED BEFORE
[Service]
Type=oneshot
RemainAfterExit=true
ExecStart=YOUR_COMMAND
# beware, executable is run directly, not through a shell, check the man
 pages
# systemd.service and systemd.unit for full syntax
[Install]
# target in which to start the service
WantedBy=multi-user.target
```

```
#WantedBy=graphical.target
```

创建服务文件后，应运行以下命令（以 `root` 身份）：

```
systemctl daemon-reload
systemctl enable before
```

每次修改服务文件时，都需要运行：

```
systemctl daemon-reload
```

13.6.6 内核控制组 (cgroups)

在传统 System V init 系统上，并不总是能够明确地将某个进程指派给生成它的服务。一些服务（例如 Apache）会生成大量第三方进程（例如 CGI 或 Java 进程），这些进程本身又会生成更多进程。这使得明确指派变得非常困难，甚至无法做到。另外，服务可能不会正常终止，导致部分子服务仍保持运行状态。

systemd 通过将每个服务放入它自己的 cgroup 中，解决了这个问题。cgroups 是一项内核功能，允许将进程及其所有子进程聚合到分层组织的组中。systemd 根据每个 cgroup 的服务名称命名各 cgroup。因为非特权进程不允许“离开”它的 cgroup，这提供了一种行之有效的方式，可通过服务名称来标记该服务生成的所有进程。

要列出属于某个服务的所有进程，请使用命令 `systemd-cgls`。结果将如下所示（已精简）：

例 13.3 列出属于某个服务的所有进程

```
root # systemd-cgls --no-pager
├─1 /usr/lib/systemd/systemd --switched-root --system --deserialize 20
├─user.slice
│ └─user-1000.slice
│   ├─session-102.scope
│   │ ├─12426 gdm-session-worker [pam/gdm-password]
│   │ ├─15831 gdm-session-worker [pam/gdm-password]
│   │ ├─15839 gdm-session-worker [pam/gdm-password]
│   │ ├─15858 /usr/lib/gnome-terminal-server

[...]
```

```
└─system.slice
  ├─systemd-hostnamed.service
  │ └─17616 /usr/lib/systemd/systemd-hostnamed
  ├─cron.service
  │ └─1689 /usr/sbin/cron -n
  ├─ntpd.service
  │ └─1328 /usr/sbin/ntpd -p /var/run/ntp/ntpd.pid -g -u ntp:ntp -c /etc/
ntp.conf
  ├─postfix.service
  │ ├─ 1676 /usr/lib/postfix/master -w
  │ ├─ 1679 qmgr -l -t fifo -u
  │ └─15590 pickup -l -t fifo -u
  ├─sshd.service
  │ └─1436 /usr/sbin/sshd -D

[...]
```

有关 cgroups 的更多信息，请参见《System Analysis and Tuning Guide》, 第 9 章 “Kernel Control Groups”。

13.6.7 终止服务（发送信号）

如第 13.6.6 节 “内核控制组 (cgroups)”中所述，在 System V init 系统中，并不总是能够将某个进程指派给其父服务。这导致终止服务及其所有子进程变得很困难。尚未终止的子进程将一直保持为僵停状态。

systemd 将每个服务限定在某个 cgroup 中的概念使您可以明确识别一个服务的所有子进程，从而向这些进程中的每一个发送信号。您可使用 `systemctl kill` 向服务发送信号。有关可用信号的列表，请参考 `man 7 signals`。

向服务发送 `SIGTERM`

`SIGTERM` 是发送的默认信号。

```
systemctl kill MY_SERVICE
```

向服务发送 `SIGNAL`

可使用 `-s` 选项指定应该发送的信号。

```
systemctl kill -s SIGNAL MY_SERVICE
```

选择进程

默认情况下，`kill` 命令会向指定 cgroup 的所有进程发送信号。您可以将发送范围限制为 `control` 或 `main` 进程。后一个选项可用于通过发送 `SIGHUP` 强制服务重新装载其配置的情况：

```
systemctl kill -s SIGHUP --kill-who=main MY_SERVICE
```

13.6.8 有关 D-Bus 服务的重要说明

D-Bus 服务是 systemd 客户端与作为 pid 1 运行的 systemd 管理器之间进行通讯的讯息总线。虽然 `dbus` 是个独立的守护程序，但它也是 init 基础架构的组成部分。

在正在运行的系统中终止或重启动 `dbus` 的效果类似于尝试终止或重启动 pid 1。此操作将中断 systemd 客户端与服务器间的通讯，并使大部分 systemd 功能不可用。

因此，不建议也不支持终止或重启动 `dbus`。

更新 `dbus` 或与 `dbus` 有关的包需要重引导。如果不确定是否需要重引导，请运行 `sudo zypper ps -s`。如果 `dbus` 显示在所列服务之中，则表明您需要重引导系统。

请记住，即使自动更新配置为跳过需要重引导的包，`dbus` 仍会更新。

13.6.9 调试服务

默认情况下，systemd 的输出不会太详细。如果服务启动成功，则不会产生任何输出。如果服务启动失败，则会显示简短的错误讯息。不过，`systemctl status` 提供了调试服务的启动和操作的途径。

systemd 附带了自己的日志记录机制（“日志”）来记录系统讯息。这可让您一并显示服务讯息与状态讯息。`status` 命令的工作方式与 `tail` 相似，也可以采用不同的格式显示日志讯息，是一个功能强大的调试工具。

显示服务启动失败讯息

每当服务启动失败时，使用 `systemctl status MY_SERVICE` 可获得详细的错误讯息：

```
root # systemctl start apache2
Job failed. See system journal and 'systemctl status' for details.
root # systemctl status apache2
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; disabled)
   Active: failed (Result: exit-code) since Mon, 04 Jun 2012 16:52:26
 +0200; 29s ago
   Process: 3088 ExecStart=/usr/sbin/start_apache2 -D SYSTEMD -k start
 (code=exited, status=1/FAILURE)
   CGroup: name=systemd:/system/apache2.service

Jun 04 16:52:26 g144 start_apache2[3088]: httpd2-prefork: Syntax error on
 line
205 of /etc/apache2/httpd.conf: Syntax error on li...alHost>
```

显示最后 `N` 条服务讯息

`status` 子命令的默认行为是显示服务发出的最后 10 条讯息。若要更改要显示的讯息数目，请使用 `--lines=N` 参数：

```
systemctl status ntp
systemctl --lines=20 status ntp
```

以附加模式显示服务讯息

要显示服务讯息的“实时流”，请使用 `--follow` 选项，效果与 `tail -f` 相似：

```
systemctl --follow status ntp
```

讯息输出格式

`--output=模式` 参数可让您更改服务讯息的输出格式。最重要的可用模式包括：

`short`

默认格式。显示日志讯息，以及用户能看懂的时戳。

`verbose`

所有字段的完整输出。

`cat`

精简输出，不含时戳。

13.7 更多信息

有关 systemd 的更多信息，请参考下列联机资源：

主页

http://www.freedesktop.org/wiki/Software/systemd

systemd for Administrators（面向管理员的 systemd）

作者：Lennart Pöttering，一位 systemd 作家，他撰写了一系列博客（写本章时为 2013 年）。它们可在 http://0pointer.de/blog/projects 找到。

III 系统

14 64 位系统环境中的 32 位和 64 位应用程序 188

15 `journalctl`：查询 systemd 日记 190

16 基本联网知识 198

17 打印机操作 266

18 X Window 系统 280

19 使用 FUSE 访问文件系统 294

20 管理内核模块 296

21 使用 udev 进行动态内核设备管理 300

22 使用 kGraft 在线增补 Linux 内核 312

23 特别的系统功能组件 318

24 持久内存 329

14 64 位系统环境中的 32 位和 64 位应用程序

SUSE® Linux Enterprise Server 可用于多种 64 位平台。但是这并不表示内含的所有应用程序都已移植到 64 位平台上。SUSE Linux Enterprise Server 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 Enterprise Server 平台上实现这种支持。

用于 64 位平台 IBM POWER、IBM Z 和 AMD64/Intel 64 的 SUSE Linux Enterprise Server 可让现有的 32 位应用程序“无需额外配置” 即可在 64 位环境中运行。相应的 32 位平台为：用于 POWER 的 ppc 和用于 AMD64/Intel 64 的 x86。。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。当前的 POWER 系统以 32 位模式运行大多数应用程序，但您可以运行 64 位应用程序。

注意：不支持构建 32 位应用程序

SUSE Linux Enterprise Server 不支持编译 32 位应用程序，只提供 32 位二进制文件的运行时支持。

14.1 运行时支持

重要：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。

此规则的一个例外是 PAM（可插入身份验证模块）。SUSE Linux Enterprise Server 在身份验证过程中使用 PAM 作为在用户和应用程序之间充当媒介的层。在另外还运行 32 位应用程序的 64 位操作系统上，始终需要安装两个版本的 PAM 模块。

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。通常预计会在 `/lib` 和 `/usr/lib` 下找到的 64 位对象文件，现在可以在 `/lib64` 和 `/usr/lib64` 下找到。这意味着 `/lib` 和 `/usr/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 32 位 `/lib` 目录的子目录包含的数据内容不依赖于字大小，则不移动这些目录。此方案符合 LSB（Linux 标准库）和 FHS（文件系统层次标准）。

14.2 内核规范

AMD64/Intel 64、IBM POWER 和 IBM Z 适用的 64 位内核提供 64 位和 32 位两种内核 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`）必须在非 POWER 平台上编译为 64 位程序才能正常运行。在 IBM Z 上，并非所有 ioctl 都可在 32 位内核 ABI 中使用。

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。

提示：内核可装载模块

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

15 journalctl：查询 systemd 日记

systemd 在取代 SUSE Linux Enterprise 12 中的传统 init 脚本时（参见第 13 章 “systemd 守护程序”），引入了自身的称为日记的日志记录系统。由于所有系统事件都将写入到日记中，因此，用户不再需要运行基于 syslog 的服务。

日记本身是 systemd 管理的系统服务，全名为 systemd-journald.service。它会根据从内核、用户进程、标准输入和系统服务错误收到的日志记录信息，维护结构化的索引日记，并以此方式来收集和储存日志记录数据。systemd-journald 服务默认处于启用状态。

```
# systemctl status systemd-journald
systemd-journald.service - Journal Service
   Loaded: loaded (/usr/lib/systemd/system/systemd-journald.service; static)
   Active: active (running) since Mon 2014-05-26 08:36:59 EDT; 3 days ago
     Docs: man:systemd-journald.service(8)
           man:journald.conf(5)
 Main PID: 413 (systemd-journal)
   Status: "Processing requests..."
   CGroup: /system.slice/systemd-journald.service
           └─413 /usr/lib/systemd/systemd-journald
[...]
```

15.1 将日记设为永久

默认情况下，日记在 /run/log/journal/ 中储存日志数据。由于 /run/ 目录具有易失本性，因此，在重引导时会丢失日志数据。要永久保存日志数据，/var/log/journal/ 目录必须存在且具有正确的所有权和权限，如此，systemd-journald 服务便可在其中储存其数据。systemd 将为您创建该目录，如果您执行以下操作，它将会切换到永久日志记录：

1. 以 root 身份打开 /etc/systemd/journald.conf 进行编辑。

   ```
   # vi /etc/systemd/journald.conf
   ```

2. 将包含 Storage= 的行取消注释，并将它更改为

   ```
   [...]
   ```

```
[Journal]
Storage=persistent
#Compress=yes
[...]
```

3. 保存该文件，然后重启动 systemd-journald：

```
systemctl restart systemd-journald
```

15.2 `journalctl` 的有用开关

本节介绍了一些可用来增强 `journalctl` 默认行为的常见有用选项。`journalctl` 手册页 `man 1 journalctl` 中介绍了所有开关。

提示：与特定可执行文件相关的讯息

要显示与特定可执行文件相关的所有日记讯息，请指定该可执行文件的完整路径：

```
journalctl /usr/lib/systemd/systemd
```

-f

仅显示最近的日记讯息，另外，在将新的日志项添加到日记时会列显这些新项。

-e

列显讯息并跳转到日记末尾，以便在页导航中显示最新的项。

-r

以倒序列显日记讯息，让最新的项列在最前面。

-k

仅显示内核讯息。这等效于字段匹配 `_TRANSPORT=kernel`（参见第 15.3.3 节 “根据字段过滤”）。

-u

仅显示指定 `systemd` 单元的讯息。这等效于字段匹配 `_SYSTEMD_UNIT=UNIT`（参见第 15.3.3 节 “根据字段过滤”）。

```
# journalctl -u apache2
[...]
Jun 03 10:07:11 pinkiepie systemd[1]: Starting The Apache Webserver...
Jun 03 10:07:12 pinkiepie systemd[1]: Started The Apache Webserver.
```

15.3 过滤日记输出

如果不结合任何开关调用 `journalctl`，它将显示日记的整个内容，最旧的项列在最前面。您可按特定的开关和字段过滤输出。

15.3.1 根据引导编号过滤

`journalctl` 可以根据特定的系统引导过滤讯息。要列出所有可用引导，请运行

```
# journalctl --list-boots
-1 097ed2cd99124a2391d2cffab1b566f0 Mon 2014-05-26 08:36:56 EDT—Fri 2014-05-30
 05:33:44 EDT
 0 156019a44a774a0bb0148a92df4af81b Fri 2014-05-30 05:34:09 EDT—Fri 2014-05-30
 06:15:01 EDT
```

第一列列出引导偏移：`0` 表示当前引导，`-1` 表示上一次引导，`-2` 表示再上一次引导，以此类推。第二列包含引导 ID，其后是特定引导的限制时间戳。

显示当前引导中的所有讯息：

```
# journalctl -b
```

如果需要查看来自前一引导的日记讯息，请加一个偏移参数。下面的示例将输出前一引导的讯息：

```
# journalctl -b -1
```

另一种方法是根据引导 ID 列出引导讯息。要实现此目的，请使用 _BOOT_ID 字段：

```
# journalctl _BOOT_ID=156019a44a774a0bb0148a92df4af81b
```

15.3.2 根据时间间隔过滤

您可通过指定开始日期和/或结束日期来过滤 `journalctl` 的输出。日期规范应采用类似于“2014-06-30 9:17:16”的格式。如果省略时间部分，则会假设为午夜。如果省略秒，则会假设为“:00”。如果省略日期部分，则会假设为当日。您也可以不采用数字表示，而是指定关键字“yesterday”、“today”或“tomorrow”。它们表示当日前一天、当日或者当日后一天的午夜。如果指定“now”，则表示当前时间。您还可以指定以 `-` 或 `+` 为前缀的相对时间，分别表示当前时间之前或之后的特定时间。

仅显示从现在开始生成的新讯息，并持续更新输出：

```
# journalctl --since "now" -f
```

显示从昨天午夜到 3:20AM 的所有讯息：

```
# journalctl --since "today" --until "3:20"
```

15.3.3 根据字段过滤

您可以按特定的字段过滤日记输出。要匹配的字段语法为 `FIELD_NAME=MATCHED_VALUE`，例如 `_SYSTEMD_UNIT=httpd.service`。您可以在单个查询中指定多个匹配条件，以进一步过滤输出讯息。有关默认字段的列表，请参见 `man 7 systemd.journal-fields`。

显示特定进程 ID 生成的讯息：

```
# journalctl _PID=1039
```

显示属于特定用户 ID 的讯息：

```
# journalctl _UID=1000
```

显示来自内核环缓冲区的讯息（与 `dmesg` 的生成结果相同）：

```
# journalctl _TRANSPORT=kernel
```

显示来自服务的标准输出或错误输出的讯息：

```
# journalctl _TRANSPORT=stdout
```

仅显示指定服务生成的讯息：

```
# journalctl _SYSTEMD_UNIT=avahi-daemon.service
```

如果指定了两个不同的字段，则仅显示同时与两个表达式匹配的项：

```
# journalctl _SYSTEMD_UNIT=avahi-daemon.service _PID=1488
```

如果两个匹配条件引用了相同的字段，则显示与两个表达式中任意一个匹配的所有项：

```
# journalctl _SYSTEMD_UNIT=avahi-daemon.service _SYSTEMD_UNIT=dbus.service
```

您可以使用“+”分隔符将两个表达式组合成一个逻辑“OR”。以下示例将显示来自进程 ID 为 1480 的 Avahi 服务进程的所有讯息，以及来自 D-Bus 服务的所有讯息：

```
# journalctl _SYSTEMD_UNIT=avahi-daemon.service _PID=1480 +
 _SYSTEMD_UNIT=dbus.service
```

15.4 调查 systemd 错误

本节将介绍一个简单的示例，演示如何找出并修复在 `apache2` 启动期间 `systemd` 报告的错误。

1. 尝试启动 apache2 服务：

   ```
   # systemctl start apache2
   Job for apache2.service failed. See 'systemctl status apache2' and
    'journalctl -xn' for details.
   ```

2. 我们来看看该服务的状态如何：

   ```
   # systemctl status apache2
   apache2.service - The Apache Webserver
      Loaded: loaded (/usr/lib/systemd/system/apache2.service; disabled)
      Active: failed (Result: exit-code) since Tue 2014-06-03 11:08:13 CEST;
    7min ago
     Process: 11026 ExecStop=/usr/sbin/start_apache2 -D SYSTEMD -DFOREGROUND \
   ```

```
           -k graceful-stop (code=exited, status=1/FAILURE)
```

导致错误的进程 ID 为 11026。

3. 显示与进程 ID 11026 相关的详细讯息：

```
# journalctl -o verbose _PID=11026
[...]
MESSAGE=AH00526: Syntax error on line 6 of /etc/apache2/default-
server.conf:
[...]
MESSAGE=Invalid command 'DocumenttRoot', perhaps misspelled or defined by a
 module
[...]
```

4. 修复 `/etc/apache2/default-server.conf` 中的拼写错误，启动 apache2 服务，然后列显其状态：

```
# systemctl start apache2 && systemctl status apache2
apache2.service - The Apache Webserver
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; disabled)
   Active: active (running) since Tue 2014-06-03 11:26:24 CEST; 4ms ago
  Process: 11026 ExecStop=/usr/sbin/start_apache2 -D SYSTEMD -DFOREGROUND
           -k graceful-stop (code=exited, status=1/FAILURE)
 Main PID: 11263 (httpd2-prefork)
   Status: "Processing requests..."
   CGroup: /system.slice/apache2.service
           ├─11263 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf -D
 [...]
           ├─11280 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf -D
 [...]
           ├─11281 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf -D
 [...]
           ├─11282 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf -D
 [...]
           ├─11283 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf -D
 [...]
           └─11285 /usr/sbin/httpd2-prefork -f /etc/apache2/httpd.conf -D
 [...]
```

15.5 Journald 配置

通过修改 `/etc/systemd/journald.conf` 可调整 systemd-journald 服务的行为。本节只介绍了基本的选项设置。有关完整的文件描述，请参见 `man 5 journald.conf`。请注意，要使更改生效，需要使用以下命令重启动日记

```
# systemctl restart systemd-journald
```

15.5.1 更改日记大小限制

如果将日记日志数据保存到永久位置（参见第 15.1 节 “将日记设为永久”），这些数据最多会占用 `/var/log/journal` 所在文件系统空间的 10%。例如，如果 `/var/log/journal` 位于一个 30 GB `/var` 的分区中，则日记最多会占用 3 GB 磁盘空间。要更改此限制，请更改（并取消注释）`SystemMaxUse` 选项：

```
SystemMaxUse=50M
```

15.5.2 将日记转发到 /dev/ttyX

您可以将日记转发到终端设备，以便在首选的终端屏幕（例如 `/dev/tty12`）上了解相关的系统讯息。将以下 journald 选项更改为

```
ForwardToConsole=yes
TTYPath=/dev/tty12
```

15.5.3 将日记转发到 Syslog 工具

Journald 与传统的 syslog 实施（例如 `rsyslog`）向后兼容。请确保满足以下条件：

- 已安装 rsyslog。

  ```
  # rpm -q rsyslog
  ```

```
rsyslog-7.4.8-2.16.x86_64
```

- 已启用 rsyslog 服务。

```
# systemctl is-enabled rsyslog
enabled
```

- 已在 `/etc/systemd/journald.conf` 中启用转发到 syslog 设置。

```
ForwardToSyslog=yes
```

15.6 使用 YaST 过滤 systemd 日记

过滤 systemd 日记的简单方法（无需处理 journalctl 语法）是使用 YaST 日记模块。使用 `sudo zypper in yast2-journal` 安装该模块后，请在 YaST 中选择 系统 › Systemd 日记 启动该模块。也可以在命令行中输入 `sudo yast2 journal` 来启动该模块。

图 15.1 YAST SYSTEMD 日记

模块将在表中显示日志项。使用顶部的搜索框可以搜索包含特定字符的项，这类似于使用 `grep`。要按日期和时间、单位、文件或优先级过滤项，请单击更改过滤器，然后设置相应的选项。

16 基本联网知识

Linux 提供集成进各类网络结构中所需的联网工具和功能。可以通过 YaST 配置使用网络卡进行的网络访问。也可以手动进行配置。在本章中，仅描述基础机制和相关网络配置文件。

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。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全球网络，也称作“因特网”。

RFC 指注释请求 (Request for Comments)。RFC 由一些文档组成，用来描述各种因特网协议和操作系统及其应用程序的实施过程。RFC 文档用来描述如何设置因特网协议。有关 RFC 的更多信息，请参见 http://www.ietf.org/rfc.html。

TCP/IP 系列协议中的若干协议

TCP
: 传送控制协议：面向连接的安全协议。要传输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 确定在传输过程中是否有任何数据丢失或发生混乱。只要涉及到数据序列就会实施 TCP。

UDP
: 用户数据报协议：无连接、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，也可能丢失数据。UDP 适用于面向记录的应用程序。它的等待时间比 TCP 稍短。

ICMP
: 因特网控制消息协议：这不是面向最终用户的协议，而是用来发出错误报告的特殊控制协议，能够控制参与 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 16.1 “TCP/IP 的简化层次模型” 中所示，数据交换在不同的层中进行。实际的网络层是通过 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 层由底层硬件相关协议（例如以太网）提供支持。

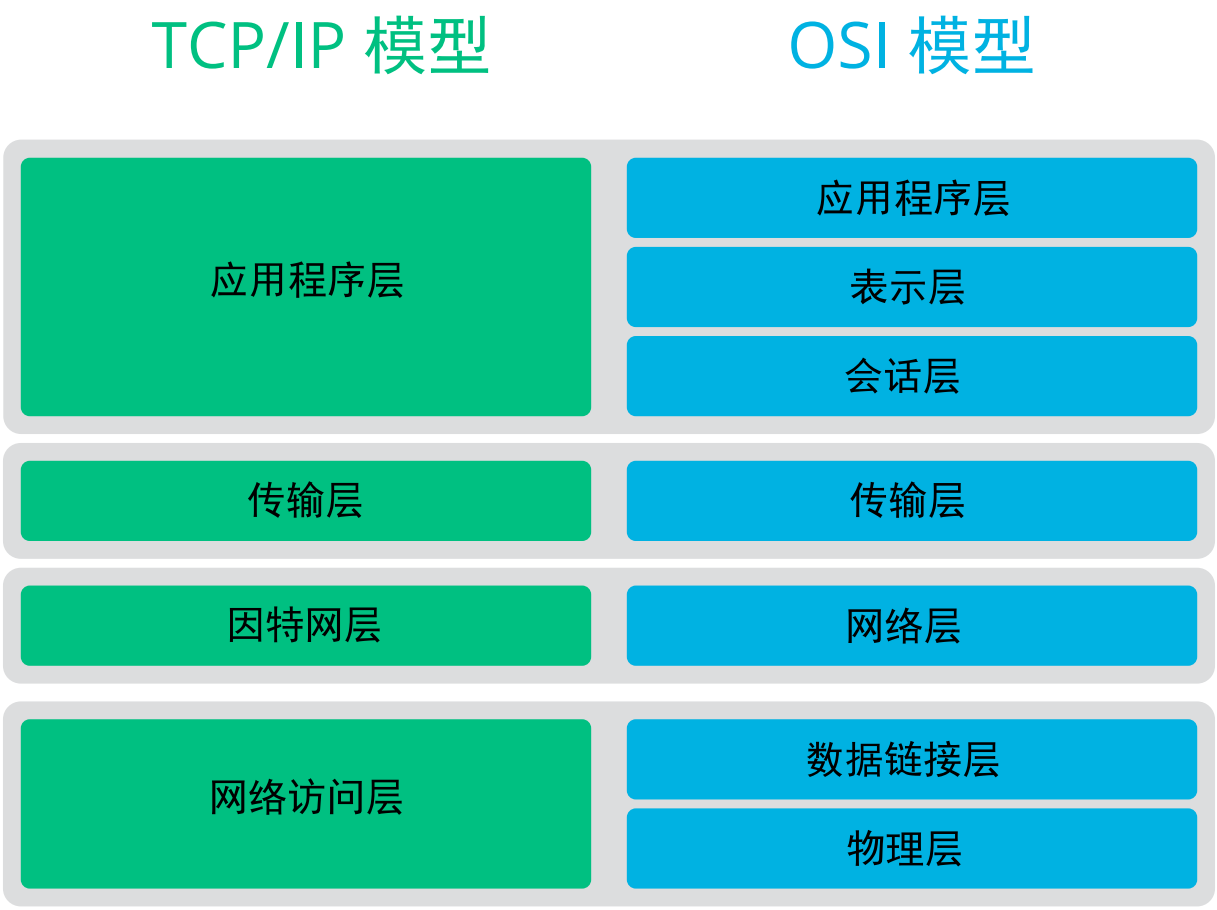


图 16.1 TCP/IP 的简化层次模型

该图为每一层都提供了一到两个示例。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描述中。数据链路层和物理层表示所用的物理网络（如以太网）。

几乎所有硬件协议都在面向数据包的基础上发挥作用。要传送的数据收集在包中（一次无法发送所有数据）。TCP/IP 包最大约为 64 KB。包通常要小得多，因为可能受到网络硬件的限制。以太网上的数据包最大约为 1500 个字节。通过以太网发送数据时，TCP/IP 包的大小不能超过这个限额。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 16.2 “TCP/IP 以太网包”中演示了一个通过以太网电缆传送的 TCP/IP 数据包示例。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。

图 16.2 TCP/IP 以太网包

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（只有物理层除外）。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删除协议报头。最后，传输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。对于应用程序来说，无论数据是通过 100 Mbit/s（兆位/秒）的 FDDI 网络传送还是通过 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紧要。

16.1 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，请参见第 16.2 节 “IPv6 — 下一代的因特网”。

16.1.1 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 16.1 “编写 IP 地址”的第二行所示的格式书写。

例 16.1 编写 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。它在全球只能使用一次。这条规则也有例外，但这些例外与下文无关。

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。但是，此系统经证实太过死板，已经废止。现已改为使用无类别路由 -（CIDR，无类别域间路由）。

16.1.2 网络掩码和路由

网络掩码用于定义子网的地址范围。如果两台主机位于同一子网中，它们可直接相互访问。如果它们位于不同子网中，则需要用于处理此子网的所有通讯的网关地址才能相互访问。要检查两个 IP 地址是否位于同一个子网中，只需分别将两个地址与网络掩码进行“AND”操作。如果结果相同，则两个 IP 地址在同一个本地网络中。如果结果不同，则仅能通过网关连接远程 IP 地址和远程接口。

要了解网络掩码如何工作，可查看例 16.2 “将 IP 地址链接到网络掩码”。网络掩码有 32 位，它确定 IP 地址有多少属于网络。对于所有为 1 的位，将它们在 IP 地址中的相应位标记为属于网络。对于所有值为 0 的位，标记其属于子网内。这意味着值为 1 的位越多，子网就越小。因为网络掩码总是由多个连续的 1 位组成，所以也可通过计算网络掩码中的位数来确定。在例 16.2 “将 IP 地址链接到网络掩码”中，第一个 24 位的网络也可写作 192.168.0.0/24。

例 16.2 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
```

再举个例子：通过同一以太网电缆相连的所有计算机通常都位于同一子网中，可直接访问。即使用交换机或网桥物理分隔该子网，这些主机仍然可以直接访问。

仅在网关是为目标网络配的时，才能访问本地子网外部的 IP 地址。通常情况下，只有一个网关处理所有外部流量。然而，也可能为不同的子网配置多个网关。

如果配置了网关，所有的外部 IP 包将发送到相应的网关。此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）。

特定地址

基础网络地址

这是网络掩码和该网络中的任意地址，如例 16.2 “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

这可以解释为：“访问此子网中的所有主机”。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。因此，以上示例会生成 192.168.0.255。该地址无法指派给任何主机。

本地主机

地址 `127.0.0.1` 指派给每台主机的“回路设备”。可以使用此地址以及通过 IPv4 定义的完整 `127.0.0.0/8` 回写网络中的所有地址为您自己的计算机设置一个连接。对于 IPv6，仅存在一个回写地址 (`::1`)。

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。这些地址无法与因特网上的其他地址建立任何连接，因为它们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 16.1 “专用 IP 地址域” 中。

表 16.1 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`

16.2 IPv6 — 下一代的因特网

重要：IBM Z：IPv6 支持

IBM Z 硬件的 CTC 和 IUCV 网络连接不支持 IPv6。

由于万维网 (WWW) 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，IPv4 地址只有 32 位。而且还有不少 IP 地址丢失，它们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：广播和基础网络地址。

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 网络中设置主机，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。以下各节进一步描述了 IPv6 带来的改进和优点，以及如何从旧协议过渡到新协议。

16.2.1 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 16.2.2 节 “地址类型和结构”。

下面列出了新协议的其他优点：

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动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）。

不过，如果路由器已连接到交换机，则路由器应发送带标志的定期通告，告知网络中的主机彼此应如何交互。有关更多信息，请参见 RFC 2462 和 `radvd.conf(5)` 手册页以及 RFC 3315。

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。这使得用户能方便地访问多个网络，可媲美手机公司提供的国际漫游服务。当您出国时，进入相应区域后手机会自动登录国外服务，因此无论您身在何处，别人都可以用同一个号码联系到您，您也可以像在国内一样拨打电话。

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。

向后兼容性

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共存，还应能够共存于一个系统中，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），另一方面还要使用多个隧道。请参见第 16.2.3 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 使服务器能够通过多路广播对主机寻址（即将多个主机作为组的一部分寻址），因而提供了更精细的方法。这种方法与通过广播对所有主机寻址，或通过单路广播单独对每个主机寻址均不同。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）。

16.2.2 地址类型和结构

如上所述，当前的 IP 协议存在两个重要限制：IP 地址日益短缺，并且配置网络、维护路由选择表的任务变得越来越复杂繁重。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），第二个问题也得到缓解。

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。它们之间用冒号 (:) 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 :: 。中演示了这种简写表示法，其中的三行全部表示同一地址。例 16.3 “示例 IPv6 地址”

例 16.3 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 16.4 “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。与 IPv4 一样，IP 地址使用 AND 结合网络掩码值，以确定主机位于同一子网中还是其他网络中。

例 16.4 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。各种 IPv6 前缀中列出了其中的一部分。

各种 IPV6 前缀

`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。目前，有以下地址空间：`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的网络之内。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。

站点拓朴结构
: 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 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（此时尚无法通过其他方法确定地址），这类地址可被主机用作其源地址。

::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 16.2.3 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。不能具有此类源地址或目标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，中间部分为零字节。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 16 位字段。其余部分也会填入零字节。

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。其中一个网络可以用 MAC 和已知前缀进行全自动配置，这样当启用 IPv6（使用链路本地地址）后，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朴结构和公共拓朴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。

16.2.3 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。但这仍然没有解决启用了 IPv6 的主机如何与 IPv4 主机通讯，以及应如何通过当前的网络（绝大部分都基于 IPv4）传输 IPv6 数据包的问题。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 16.2.2 节 “地址类型和结构”）。

IPv6 主机多少孤立于（全球）IPv4 网络，它可通过隧道通讯：IPv6 包封装为 IPv4 包，以便在 IPv4 网络中移动。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），以及隧道接收端远程主机的 IPv4 地址。根据主机管理员间的协议，可以手动配置基本的隧道。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，这种方法的可扩展性不够好，而且不易推行，因为 IP 多路广播在因特网上尚未普及。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描述。

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 中对此进行了描述。

16.2.4 配置 IPv6

要配置 IPv6，通常无需在各个工作站上执行任何更改。默认情况下启用 IPv6。要在已安装系统上禁用或启用 IPv6，请使用 YaST 网络设置模块。在全局选项选项卡上，根据需要选择或取消选择启用 IPv6 选项。要暂时启用直至下次重引导，请以 `root` 身份输入 `modprobe -i ipv6`。装载 IPv6 模块后无法将其卸载。

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。一般不在工作站上管理路由选择表。工作站可以使用路由器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/quagga 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，请参见 `ifcfg-tunnel` 的手册页 (`man ifcfg-tunnel`)。

16.2.5 更多信息

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，请参见以下联机文档和书目：

http://www.ipv6.org/

学习 IPv6 知识的起点。

http://www.ipv6day.org

启动您自己的 IPv6 网络所需的所有信息。

http://www.ipv6-to-standard.org/

已启用 IPv6 的产品列表。

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。

RFC 2460

有关 IPv6 的基础 RFC。

IPv6 Essentials

Silvia Hagen 所著的 IPv6 Essentials(ISBN 0-596-00125-8) 中描述了该主题的所有重要方面。

16.3 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。这些名称构成了具有层次结构的系统，各个名称组成部分之间用句点分隔。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 `jupiter.example.com`。完整名称，即完全限定的域名 (FQDN)，由主机名和域名 (`example.com`) 组成。后者还包含顶级域或 TLD (`com`)。

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 root 名称服务器。这些 root 名称服务器管理顶级域，并由网络信息中心 (NIC) 运行。每个 root 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，即邮件交换器 (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。使用 YaST 可轻松指定此类名称服务器。有关如何在 SUSE® Linux Enterprise Server 中配置针对名称服务器的访问，请参见第 16.4.1.4 节 “配置主机名和 DNS”。有关如何设置您自己的名称服务器，请参见第 26 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给定域的服务器。

注意：MDNS 和 .local 域名

`.local` 顶级域由解析程序视为 link-local 域。DNS 请求作为多路广播 DNS 请求（而不是常规 DNS 请求）发送。如果已在名称服务器配置中使用 `.local` 域，必须在 `/etc/host.conf` 中关闭此选项。有关更多信息，请参见 `host.conf` 手册页。

如果要在安装期间关闭 MDNS，请使用 `nomdns=1` 作为引导参数。

有关多路广播 DNS 的详细信息，请参见 http://www.multicastdns.org。

16.4 使用 YaST 配置网络连接

Linux 上有多个支持的联网类型。其中多数使用不同的设备名，配置文件分布在文件系统上的多个位置。关于手动网络配置方面的详细概述，请参见第 16.5 节 “手动配置网络连接”。

已建立链接的所有网络接口（已连接网络电缆）将自动进行配置。可随时在已安装系统中配置额外的硬件。以下章节将介绍 SUSE Linux Enterprise Server 支持的所有网络连接类型的网络配置。

提示：IBM Z：可热插拔网卡

IBM Z 平台支持可热插拔网卡，但不支持通过 DHCP 自动将它们进行网络集成（与在 PC 上一样）。检测到网卡后需要手动配置接口。

16.4.1 使用 YaST 配置网卡

要在 YaST 中配置以太网卡或 Wi-Fi/蓝牙卡，请选择系统 › 网络设置。启动模块后，YaST 将显示网络设置对话框，其中包括四个选项卡：全局选项、概述、主机名/DNS 和路由选择。

通过全局选项选项卡可设置常规联网选项，例如网路设置方法、IPv6 和常规 DHCP 选项。有关详细信息，请参见第 16.4.1.1 节 “配置全局联网选项”。

概述选项卡包含关于已安装网络接口和配置的信息。会列出已正确检测到的所有网卡及其名称。您可在此对话框中手动配置新卡、删除或更改其配置。要手动配置未自动检测到的网卡，请参见第 16.4.1.3 节 “配置未检测到的网卡”。如果要更改已配置卡的配置，请参见第 16.4.1.2 节 “更改网卡的配置”。

通过主机名/DNS 选项卡可设置计算机的主机名和要使用的服务器名称。有关详细信息，请参见第 16.4.1.4 节 “配置主机名和 DNS”。

路由选择选项卡用于配置路由选择。有关更多信息，请参见第 16.4.1.5 节 “配置路由选择”。

图 16.3 配置网络设置

16.4.1.1 配置全局联网选项

通过 YaST 网络设置模块的全局选项选项卡，可设置重要的全局联网选项，如使用 NetworkManager、IPv6 和 DHCP 客户端选项。这些设置适用于所有网络接口。

注意：NetworkManager 由 Workstation Extension 提供

现在，NetworkManager 由 Workstation Extension 提供。要安装 NetworkManager，请激活 Workstation Extension 储存库，然后选择 NetworkManager 包。

在网络设置方法中，选择管理网络连接的方法。如果希望 NetworkManager 桌面小程序管理所有接口的连接，请选择 NetworkManager 服务。NetworkManager 非常适用于在多个有线和无线网络之间切换。如果您运行的不是桌面环境，或者您的计算机是 Xen 服务器、虚拟系统或者会在网络中提供 DHCP 或 DNS 等网络服务，请使用 Wicked 服务方法。如果使用 NetworkManager，则应使用 `nm-applet` 配置网络选项，并且网络设置模块的概述、主机名/DNS和路由选择选项卡会被禁用。有关 NetworkManager 的详细信息，请参见 SUSE Linux Enterprise Desktop 文档。

在 IPv6 协议设置中，选择是否使用 IPv6 协议。可将 IPv6 与 IPv4 一起使用。默认情况下，会启用 IPv6。但是在不使用 IPv6 协议的网络中，如果禁用 IPv6 协议，响应时间会更快。要禁用 IPv6，请停用启用 IPv6。如果禁用了 IPv6，内核将不再自动装载 IPv6 模块。重启动后会应用此设置。

在 DHCP 客户端选项中，配置 DHCP 客户端的选项。在单个网络上，每个 DHCP 客户端的 DHCP 客户端标识符必须不同。如果保留为空，会默认为网络接口的硬件地址。但是，如果正在运行若干使用相同网络接口（即相同硬件地址）的虚拟机，则在此处指定唯一的自由格式标识符。

要发送的主机名指定当 DHCP 客户端将消息发送到 DHCP 服务器时，主机名选项字段使用的字符串。某些 DHCP 服务器会根据此主机名（动态 DNS）来更新名称服务器区域（正向和反向记录）。此外，有些 DHCP 服务器要求要发送的主机名选项字段包含来自客户端的 DHCP 消息中的特定字符串。如果保留 `AUTO`，将发送当前的主机名（即 `/etc/HOSTNAME` 中定义的主机名）。将选项字段保留空白则不会发送任何主机名。

如果您不希望根据 DHCP 中的信息更改默认路由，请停用通过 DHCP 更改默认路由。

16.4.1.2 更改网卡的配置

要更改网卡的配置，请在 YaST 网络设置 › 概述中已检测到的网卡列表中选择一个网卡，然后单击编辑。将显示网卡设置对话框，可在其中使用常规、地址和硬件选项卡调整网卡配置。

16.4.1.2.1 配置 IP 地址

您可在网卡设置对话框的地址选项卡中设置网卡的 IP 地址或 IP 地址的确定方法。同时支持 IPv4 和 IPv6 地址。网卡可设置为无 IP 地址（对于绑定设备很有用）、静态指派的 IP 地址（IPv4 或 IPv6）或通过 DHCP和/或 Zeroconf 指派的动态地址。

如果使用动态地址，则选择是使用仅 DHCP 版本 4（用于 DHCPv4）、DHCP 版本 6（用于 DHCPv6）还是 DHCP 版本 4 和 6。

如果可能，安装期间的首个带链接的可用网卡将会通过 DHCP 自动配置为使用自动 IP 地址。

注意：IBM Z 和 DHCP

在 IBM Z 平台上，只有具有 MAC 地址的网卡才支持基于 DHCP 的地址配置。属于这种情况的只有 OSA 和 OSA Express 网卡。

如果使用的是 DSL 线路，但 ISP（因特网服务提供商）没有指派静态 IP，此时还应使用 DHCP。如果决定使用 DHCP，请打开 YaST 网卡配置模块的网络设置对话框，在全局选项选项卡的 DHCP 客户端选项中配置细节。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信，则需要用 DHCP 客户端标识符来区分。

DHCP 比较适合客户端配置，但不太适合服务器配置。要设置静态 IP 地址，请如下继续操作：

1. 在 YaST 网卡配置模块的概述选项卡中，于已检测到的网卡列表中选择一个网卡，然后单击编辑。
2. 在地址选项卡中，选择静态指派的 IP 地址。
3. 输入 IP 地址。IPv4 和 IPv6 地址均可使用。在子网掩码中输入子网掩码。如果使用 IPv6 地址，则对于前缀长度使用 `/64` 格式的子网掩码。
   或者，您可以为此地址输入一个完全限定的主机名，该主机名将写入到 `/etc/hosts` 配置文件。

4. 单击下一步。

5. 要激活配置，请单击确定。

注意：接口激活和链路检测

在激活网络接口期间，`wicked` 会检查载波，并且只有在检测到链路之后，才会应用 IP 配置。如果不管链路状态为何，您都需要应用配置（例如，您要测试侦听某个地址的服务），则可以在 `/etc/sysconfig/network/ifcfg` 内的接口配置文件中添加变量 `LINK_REQUIRED=no` 来跳过链路检测。

另外，您可以使用变量 `LINK_READY_WAIT=5` 来指定等待链路的超时值（以秒为单位）。

有关 `ifcfg-*` 配置文件的详细信息，请参见第 16.5.2.5 节 “/etc/sysconfig/network/ifcfg-*” 和 `man 5 ifcfg`。

如果使用静态地址，则不会自动配置名称服务器和默认网关。要配置名称服务器，请按照第 16.4.1.4 节 “配置主机名和 DNS” 中所述进行。要配置网关，请按照第 16.4.1.5 节 “配置路由选择” 中所述进行。

16.4.1.2.2 配置多个地址

一个网络设备可以有多个 IP 地址。

注意：别名是兼容功能

这些所谓的别名或标签只能分别用于 IPv4。对于 IPv6 则忽略它们。使用 `iproute2` 网络接口可以有一个或多个地址。

要使用 YaST 设置网卡的其他地址，请执行以下步骤：

1. 在 YaST 网络设置对话框的概述选项卡中，于已检测到的网卡列表中选择一个网卡，然后单击编辑。

2. 在 地址 › 附加地址 选项卡中，单击添加。

3. 输入 IPv4 地址标签、IP 地址和网络掩码。不要在别名中包含接口名称。
4. 要激活该配置，请确认设置。

16.4.1.2.3 更改设备名称和 Udev 规则

可更改网卡在使用时的设备名称。还可确定 udev 是通过网卡的硬件 (MAC) 地址还是通过总线 ID 来标识网卡。后者更适合大型服务器，因为便于热插拔网卡。要使用 YaST 设置这些选项，请执行以下步骤：

1. 在 YaST 网络设置对话框的概述选项卡中，于已检测到的网卡列表中选择一个网卡，然后单击编辑。
2. 转到硬件选项卡。当前设备名称显示在 Udev 规则中。单击更改。
3. 选择 udev 应通过网卡的 MAC 地址还是总线 ID 来识别网卡。网卡的当前 MAC 地址和总线 ID 显示在对话框中。
4. 要更改设备名称，请选中更改设备名称选项并编辑名称。
5. 要激活该配置，请确认设置。

16.4.1.2.4 更改网卡内核驱动程序

对于某些网卡，可能会提供某些内核驱动程序。如果网卡已配置，YaST 允许您从可用的合适驱动程序列表中选择一个要使用的内核驱动程序。还可为内核驱动程序指定选项。要使用 YaST 设置这些选项，请执行以下步骤：

1. 在 YaST 网络设置模块的概述选项卡中，于已检测到的网卡列表中选择一个网卡，然后单击编辑。
2. 转到硬件选项卡。
3. 在模块名称中选择要使用的内核驱动程序。在选项中以 `=` =VALUE 格式为所选驱动程序输入任何选项。如果使用多个选项，应用空格分隔这些选项。
4. 要激活该配置，请确认设置。

16.4.1.2.5 激活网络设备

如果使用结合 `wicked` 的方法，便可以将设备配置为在引导期间、连接电缆时或检测到网卡时启动、以手动方式启动或永不启动设备。要更改设备启动，请如下继续操作：

1. 在 YaST 的系统 › 网络设置中，于已检测到的网卡列表中选择一个网卡，然后单击编辑。
2. 在常规选项卡中，从设备激活选择所希望的项。
   选择在引导时可在系统引导时启动设备。使用在电缆连接时将对任何现有物理连接监视接口。使用在热插拔时，可在接口可用时对其进行设置。这与在引导时选项很相似，唯一区别是如果引导时接口不存在，将不会发生错误。选择手动可通过 `ifup` 手动控制接口。选择从不将不启动设备。通过 NFSroot 与在引导时相似，区别是使用 `systemctl stop wicked.service` 命令不会关闭接口；如果 `wicked` 处于活动状态，则 `network` 服务还会处理 `wicked` 服务。如果您使用 NFS 或 iSCSI 根文件系统，则选择此选项。
3. 要激活该配置，请确认设置。

提示：NFS 用作根文件系统

在通过网络以 NFS 共享形式装入根分区的（无磁盘）系统中，配置可供访问 NFS 共享的网络设备时需保持谨慎。

关闭或重引导系统时，默认的处理顺序是关闭网络连接，然后卸载根分区。对于 NFS 根分区，这种顺序会产生问题，因为在尚未激活与 NFS 共享的网络连接的情况下，根分区无法完全卸载。为防止系统停用相关的网络设备，请按第 16.4.1.2.5 节 “激活网络设备”中所述打开网络设备配置选项卡，然后在设备激活窗格中选择通过 NFSroot。

16.4.1.2.6 设置最大传输单位大小

您可为接口设置最大传输单位 (MTU)。MTU 是指允许的最大包大小（以字节为单位）。更高的 MTU 可带来更高的带宽效率。但是较大的包有时可能会堵塞较慢的接口，从而增加后续包的延迟。

1. 在 YaST 的系统 › 网络设置中，于已检测到的网卡列表中选择一个网卡，然后单击编辑。
2. 在常规选项卡中，从设置 MTU 列表中选择所需项。

3. 要激活该配置，请确认设置。

16.4.1.2.7 PCIe 多功能设备

系统支持对以下技术提供支持的多功能设备：LAN、iSCSI 和 FCoE。YaST FCoE 客户端 (`yast2 fcoe-client`) 会在额外的列中显示私用标志，以允许用户选择用于 FCoE 的设备。YaST 网络模块 (`yast2 lan`) 会针对网络配置排除“仅储存设备”。

关于 FCoE 的详细信息，请参见《储存管理指南》，第 15 章 “以太网光纤通道储存：FCoE”，第 15.3 节 “使用 YaST 管理 FCoE 服务”。

16.4.1.2.8 IP-over-InfiniBand (IPoIB) 的 Infiniband 配置

1. 在 YaST 的系统 › 网络设置中选择 InfiniBand 设备，然后单击编辑。
2. 在常规选项卡中，选择一种 IP-over-InfiniBand (IPoIB) 模式：已连接（默认）或数据报。
3. 要激活该配置，请确认设置。

有关 InfiniBand 的更多信息，请参见 `/usr/src/linux/Documentation/infiniband/ipoib.txt`。

16.4.1.2.9 配置防火墙

您不必按《Security Guide》，第 16 章 “Masquerading and Firewalls”，第 16.4.1 节 “Configuring the Firewall with YaST”中所述输入详细的防火墙设置，只需在设置设备的过程中决定设备的基本防火墙配置。按如下所示继续：

1. 打开 YaST 的系统 › 网络设置模块。在概述选项卡中，从已检测到的网卡列表中选择一个网卡，然后单击编辑。
2. 进入网络设置对话框的常规选项卡。
3. 确定要将接口指派到的防火墙区域。下列选项可用：

   防火墙已禁用

   此选项只有在禁用防火墙和防火墙未在运行时才可用。仅当计算机属于受外部防火墙保护的大型网络时才使用此选项。

自动指派区域

此选项只有在启用防火墙后才可用。防火墙正在运行且接口自动指派给防火墙区域。包含关键字 `any` 的区域或外部区域将用于此类接口。

内部区域（未保护）

防火墙正在运行，但不会强制执行任何规则来保护此接口。当计算机属于受外部防火墙保护的大型网络时才使用此选项。当计算机具有多个网络接口时，此选项还可用于连接到内部网络的接口。

隔离区域

隔离区域是位于内部网络和（恶意）因特网之前的附加防线。可从内部网络和因特网访问指派到此区域的主机，但指派到此区域的主机无法访问内部网络。

外部区域

防火墙在此接口上运行，并且全面保护其抵御其他假定有害的网络流量。这是默认选项。

4. 要激活该配置，请确认设置。

16.4.1.3 配置未检测到的网卡

如果未正确检测到某个网卡，该卡将不会包含在已检测到的网卡列表中。如果确定系统包含网卡的驱动程序，则可以手动对其进行配置。还可以配置特殊网络设备类型，例如网桥、绑定、TUN 或 TAP。要配置未检测到的网卡（或特殊设备），请如下操作：

1. 在 YaST 的系统 › 网络设置 › 概述对话框中，单击添加。

2. 在硬件对话框中，从可用选项中设置接口的设备类型和配置名称。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。或者，如果需要，您可定义要用于网卡的内核模块名称及其选项。

   在 Ethtool 选项中，您可以为接口设置 `ifup` 使用的 `ethtool` 选项。有关可用选项的信息，请参见 `ethtool` 手册页。

   如果选项字符串以 `-` 开头（例如，`-K INTERFACE_NAME rx on`），则会用当前接口名称替换字符串中的第二个词。否则（例如，`autoneg off speed 10`）`ifup` 会在开头添加 `-s INTERFACE_NAME`。

3. 单击下一步。
4. 在常规、地址和硬件选项卡中，配置所有所需的选项，如接口的 IP 地址、设备激活或防火墙区域。有关配置选项的更多信息，请参见第 16.4.1.2 节 “更改网卡的配置”。
5. 如果选择无线作为接口的设备类型，则在下一个对话框中配置无线连接。
6. 要激活新的网络配置，请确认设置。

16.4.1.4 配置主机名和 DNS

如果您在安装期间未更改网络配置，并且已有以太网卡可用，则系统会自动为您的计算机生成主机名并激活 DHCP。这同样适用于主机连接到网络环境所需的名称服务信息。如果网络地址设置使用了 DHCP，则会向域名服务器列表自动填充相应数据。如果希望使用静态设置，则手动设置这些值。

要更改计算机名称并调整名称服务器搜索列表，则如下继续操作：

1. 转到 YaST 的系统 › 模块中的网络设置主机名/DNS 选项卡。
2. 输入主机名，如果需要，也输入域名。如果此计算机是邮件服务器，则该域特别重要。请注意，主机名是全局性的，将应用到所有已设置的网络接口。
   如果使用 DHCP 获取 IP 地址，则计算机的主机名将由 DHCP 自动设置。如果连接到不同网络，您应禁用此行为，因为其他网络可能会指派不同的主机名，而在运行时更改主机名可能会导致混淆图形桌面。要禁止使用 DHCP 获取 IP 地址，请停用通过 DHCP 更改主机名。
   指派主机名给回写 IP 会将您的主机名与 `/etc/hosts` 中的 `127.0.0.2`（回写）IP 地址关联。如果您想让主机名即使在没有活动的网络情况下也可随时解析，则可使用该选项。
3. 在修改 DNS 配置中，请选择修改 DNS 配置（名称服务器、搜索列表以及 `/etc/resolv.conf` 文件的内容）的方式。
   如果选择了使用默认策略选项，则配置由 `netconfig` 脚本处理，该脚本合并了静态定义的数据（通过 YaST 或在配置文件中）与动态获取的数据（来自 DHCP 客户端或 NetworkManager）。此默认策略可满足通常情况。
   如果选择了仅手动选项，则不允许 `netconfig` 修改 `/etc/resolv.conf` 文件。但是，此文件可手动编辑。

如果已选择自定义策略选项，则应指定用于定义合并策略的自定义策略规则字符串。该字符串包含了接口名称的逗号分隔列表，可考虑作为设置的有效源。除完整接口名外，也可使用基本通配符来匹配多个接口。例如，`eth* ppp?` 首先以所有 eth 为目标，然后是 ppp0 到 ppp9 的所有接口。有两个特殊策略值表示如何应用 `/etc/sysconfig/network/config` 文件中定义的静态设置：

`STATIC`
: 静态设置需要与动态设置合并到一起。

`STATIC_FALLBACK`
: 仅当动态配置不可用时，才使用静态设置。

有关更多信息，请参见 `netconfig`(8) 的手册页 (`man 8 netconfig`)。

4. 输入名称服务器并填写域搜索列表。名称服务器必须由 IP 地址指定（如 192.168.1.116），而非由主机名指定。域搜索选项卡中指定的名称是用于解析主机名（未指定域）的域名。如果使用多个域搜索，则使用逗号或空格分隔域。

5. 要激活该配置，请确认设置。

也可以使用 YaST 从命令行编辑主机名。YaST 所做更改会立即生效（手动编辑 `/etc/HOSTNAME` 文件时则不是这样）。要更改主机名，请使用以下命令：

```
yast dns edit hostname=HOSTNAME
```

要更改名称服务器，请使用以下命令：

```
yast dns edit nameserver1=192.168.1.116
yast dns edit nameserver2=192.168.1.117
yast dns edit nameserver3=192.168.1.118
```

16.4.1.5 配置路由选择

要使计算机能够与其他计算机和其他网络进行通信，必须提供路由选择信息以使网络流量使用正确的路径。如果使用 DHCP，则将自动提供此信息。如果使用静态设置，则必须手动添加此数据。

1. 在 YaST 中，转到网络设置 › 路由选择。
2. 输入默认网关（如果需要是 IPv4 和 IPv6）的 IP 地址。默认网关与每个可能的目标匹配，但是如果存在与所需地址匹配的路由表项，则会使用此项，而不是通过默认网关使用默认路由。
3. 可在路由选择表中输入多个项。输入目标网络 IP 地址、网关 IP 地址和网络掩码。选择将流量路由到定义的网络要经过的设备（减号代表任何设备）。要忽略这些值中的任何值，请使用减号 `-`。要在表中输入默认网关，请在目标字段中使用 默认。

   注意：路由优先级

   如果使用更多的默认路由，则可以指定用于确定具有更高优先级的路由的度量选项。要指定度量选项，请在选项中输入 `- metric NUMBER`。默认情况下使用具有最高度量的路由。如果网络设备已断开连接，则删除其路由并使用下一个路由。但是，当前内核不在静态路由中使用度量，只有 `multipathd` 等路由守护程序使用度量。
4. 如果系统是路由器，请根据需要在网络设置中启用 IPv4 转发和 IPv6 转发。
5. 要激活该配置，请确认设置。

16.4.2 IBM Z：配置网络设备

SUSE Linux Enterprise Server for IBM Z 支持多种类型的网络接口。可使用 YaST 来对所有这些接口进行配置。

16.4.2.1 qeth-hsi 设备

要将 `qeth-hsi` (Hipersockets) 接口添加到安装的系统中，请启动 YaST 中的系统 › 网络设置模块。选择标记为 Hipersocket 的设备之一以用作 READ 设备地址，然后单击编辑。输入设备编号供读、写和控制通道（例如设备编号格式：`0.0.0800`）。然后单击“下一步”。在网络地址设置对话框中，为新接口指定 IP 地址和网络掩码，然后单击下一步和确定退出网络配置。

16.4.2.2 qeth-ethernet 设备

要将 `qeth-ethernet`（IBM OSA Express 以太网卡）接口添加到安装的系统中，请启动 YaST 中的系统 › 网络设置模块。选择标有 IBM OSA 快速以太网卡的任一设备以用作“读”设备地址并单击编辑。输入设备编号供读、写和控制通道（例如设备编号格式：`0.0.0700`）。输入所需端口名称、端口号（如果适用）和其他的一些选项（请参见 http://www.ibm.com/developerworks/linux/linux390/documentation_suse.html 上的《Linux for IBM Z: Device Drivers, Features, and Commands》（适用于 IBM Z 的 Linux：设备驱动程序、功能和命令）手册)、您的 IP 地址及相应的网络掩码。单击下一步和确定退出网络配置。

16.4.2.3 ctc 设备

要将 `ctc`（IBM 并行 CTC 适配器）接口添加到安装的系统中，请启动 YaST 中的系统 › 网络设置模块。选择标记为 IBM 并行 CTC 适配器的设备之一，用作读取通道，然后单击配置。选择适合您设备的 设备设置（通常为兼容性方式）。指定您的 IP 地址和远程合作伙伴的 IP 地址。如果需要，可使用高级 › 详细设置调整 MTU 的大小。单击下一步和确定退出网络配置。

警告：不再支持 CTC

不建议使用此接口。未来版本的 SUSE Linux Enterprise Server 将不支持此接口。

16.4.2.4 lcs 设备

要将 `lcs`（IBM OSA-2 适配器）接口添加到安装的系统中，请启动 YaST 中的系统 › 网络设置模块。选择标记为 IBM OSA-2 适配器的设备之一，然后单击配置。输入所需端口号、其他的一些选项（请参见 http://www.ibm.com/developerworks/linux/linux390/documentation_suse.html 上的《Linux for IBM Z: Device Drivers, Features, and Commands》（适用于 IBM Z 的 Linux：设备驱动程序、功能和命令）手册）、您的 IP 地址及相应的网络掩码。单击下一步和确定退出网络配置。

16.4.2.5 IUCV 设备

要将 `iucv` (IUCV) 接口添加到安装的系统中，请启动 YaST 中的系统 › 网络设置模块。选择标记为 IUCV 的设备并单击编辑。YaST 会提示您输入 IUCV 合作伙伴（同级）的名称。输入该名称（此项区分大小写），然后选择下一步。指定您的合作伙伴的 IP 地址和远程 IP 地址。如果需要，在常规选项卡上设置 MTU 大小。单击下一步和确定退出网络配置。

警告：不再支持 IUCV

不建议使用此接口。未来版本的 SUSE Linux Enterprise Server 将不支持此接口。

16.5 手动配置网络连接

应该将手动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。

16.5.1 `wicked` 网络配置

名为 `wicked` 的工具和库提供了一个用于配置网络的新框架。

传统网络接口管理面临的其中一项挑战是，各种不同的网络管理层混杂在一个脚本中，最多在两个不同的脚本中。这些脚本彼此之间如何交互没有明确的定义。这会导致出现无法预测的问题、模糊的约束和约定等情况。针对各种不同的情况部署多个特殊入侵层增大了维护负担。所用的地址配置协议是通过 dhcpcd 等守护程序实现的，而这些守护程序与基础架构中其他组件的交互很不通畅。为了持续识别接口，引入了新潮的接口命名模式，而这需要繁重的 udev 支持。

wicked 的理念是通过多种方式剖析问题。它没有采用任何全新的技术，而是尝试将不同项目中的观点集中起来，以建立一个更好的整体解决方案。

实现此目的的方法之一是使用客户端/服务器模型。wicked 可借此为地址配置等任务定义能够很好地集成到整个框架中的标准化工具。例如，使用特定的地址配置时，管理员可能要求应该通过 DHCP 或 IPv4 zeroconf 配置接口。在这种情况下，地址配置服务只会从它的服务器获得租用，并传递到安装了所请求地址和路由的 wicked 服务器进程。

剖析问题的另一种方法是强制实施分层机制。对于任何类型的网络接口，都可以定义一个 dbus 服务，用于配置网络接口的设备层 — VLAN、桥接、绑定或超虚拟化设备。地址配置等常用功能通过在这些设备特定的服务基础上分层的联合服务实现，您不必专门实施这些功能。

wicked 框架使用各种 dbus 服务来实现这两个方面的功能，这些服务将会根据其类型挂接到网络接口。本文提供了 wicked 中的当前对象层次的简要概述。

每个网络接口以 `/org/opensuse/Network/Interfaces` 的子对象表示。该子对象的名称由其 ifindex 指定。例如，ifindex 通常为 1 的回写接口是 `/org/opensuse/Network/Interfaces/1`，注册的第一个以太网接口是 `/org/opensuse/Network/Interfaces/2`。

每个网络接口都有一个关联的“类”，该类用于选择该接口支持的 dbus 接口。默认情况下，每个网络接口的类为 `netif`，wickedd 将自动挂接与此类兼容的所有接口。 在当前实施中，这些兼容的接口包括：

org.opensuse.Network.Interface
: 一般网络接口功能，例如，打开或关闭链路、指派 MTU 等

org.opensuse.Network.Addrconf.ipv4.dhcp,
org.opensuse.Network.Addrconf.ipv6.dhcp,
org.opensuse.Network.Addrconf.ipv4.auto
: 适用于 DHCP、IPv4 zeroconf 等的地址配置服务

除此之外，网络接口可能还需要或者提供特殊的配置机制。例如，对于某个以太网设备，您应该能够控制链路速度、校验和卸载等。为了实现此目的，以太网设备都有一个名为 `netif-ethernet` 的自己的类，该类属于 `netif` 的子类。因此，指派给以太网接口的 dbus 接口具有上面列出的所有服务以及 `org.opensuse.Network.Ethernet` 服务，后者只适用于属于 `netif-ethernet` 类的对象。

同样，桥接、VLAN、绑定或 infiniband 等接口类型也存在适用类。

您要如何与某个首先需要创建的接口（例如 VLAN，它实际上是位于以太网设备上的虚拟网络接口）交互呢？为此，wicked 定义了出厂接口，例如 `org.opensuse.Network.VLAN.Factory`。这种出厂接口只提供单一功能，就是让您创建所请求类型的接口。这些出厂接口将挂接到 `/org/opensuse/Network/Interfaces` 列表节点。

16.5.1.1 wicked 体系结构和功能

如图 16.4 “wicked 体系结构”中所述，wicked 服务由几个部分组成。

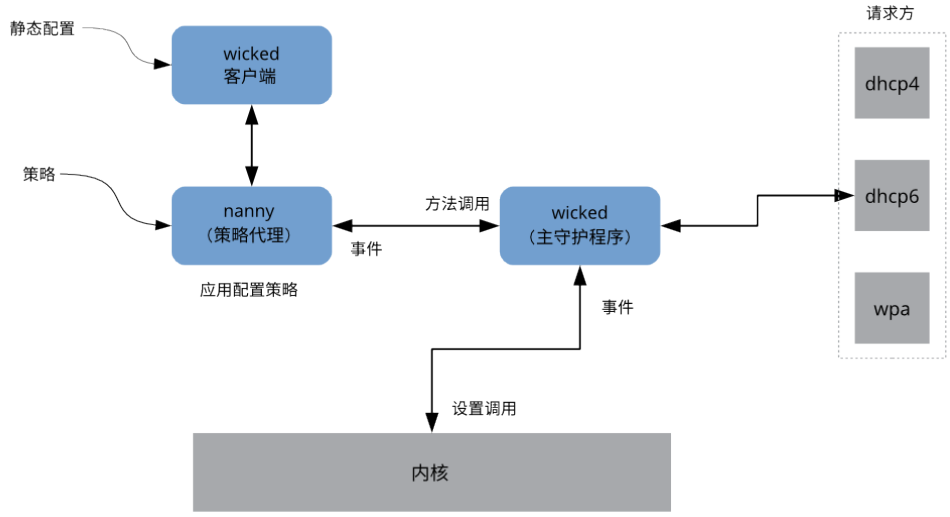


图 16.4 wicked 体系结构

wicked 目前支持以下功能：

- 使用配置文件后端来分析 SUSE 样式的 /etc/sysconfig/network 文件。
- 使用内部配置后端以 XML 格式表示网络接口配置。
- 打开和关闭“常规”网络接口，例如以太网或 InfiniBand，VLAN、网桥、绑定、tun、tap、虚设设备、macvlan、macvtap、hsi、qeth、iucv 和无线（当前限制为一个 wpa-psk/eap 网络）设备。
- 内置 DHCPv4 客户端和内置 DHCPv6 客户端。
- 默认启动的 nanny 守护程序有助于在设备可用（接口热插入）时自动激活配置的接口，以及在检测到链路（载波）时设置 IP 配置。有关详细信息，请参见 第 16.5.1.3 节 “Nanny”。
- wicked 过去是以与 systemd 集成的 DBus 服务组形式来实现的。因此，一般的 systemctl 命令将会适用于 wicked。

16.5.1.2 使用 wicked

在 SUSE Linux Enterprise 上，默认会运行 `wicked`。如果您要检查当前启用了哪个组件以及该组件是否正在运行，请调用：

```
systemctl status network
```

如果已启用 `wicked`，您将看到类似于下面的行：

```
wicked.service - wicked managed network interfaces
    Loaded: loaded (/usr/lib/systemd/system/wicked.service; enabled)
    ...
```

如果运行的是其他组件（例如 NetworkManager）并且您想切换到 `wicked`，请先停止正在运行的组件，然后启用 `wicked`：

```
systemctl is-active network && \
systemctl stop      network
systemctl enable --force wicked
```

如此会启用 wicked 服务、创建从 `network.service` 到 `wicked.service` 的别名链路，并在下次引导时启动网络。

启动服务器进程：

```
systemctl start wickedd
```

这将会启动 `wickedd`（主服务器）和关联的请求方：

```
/usr/lib/wicked/bin/wickedd-auto4 --systemd --foreground
/usr/lib/wicked/bin/wickedd-dhcp4 --systemd --foreground
/usr/lib/wicked/bin/wickedd-dhcp6 --systemd --foreground
/usr/sbin/wickedd --systemd --foreground
/usr/sbin/wickedd-nanny --systemd --foreground
```

然后激活网络：

```
systemctl start wicked
```

或者使用 `network.service` 别名：

```
systemctl start network
```

这些命令使用 `/etc/wicked/client.xml` 中定义的默认配置源或系统配置源。

要启用调试，请在 `/etc/sysconfig/network/config` 中设置 `WICKED_DEBUG`，例如：

```
WICKED_DEBUG="all"
```

或者，要省略一些信息：

```
WICKED_DEBUG="all,-dbus,-objectmodel,-xpath,-xml"
```

使用客户端实用程序显示所有接口的接口信息，或者显示以 `IFNAME` 指定的接口的接口信息：

```
wicked show all
wicked show IFNAME
```

XML 格式的输出：

```
wicked show-xml all
wicked show-xml IFNAME
```

打开一个接口：

```
wicked ifup eth0
wicked ifup wlan0
...
```

由于未指定配置源，wicked 客户端将检查 `/etc/wicked/client.xml` 中为它定义的默认配置源：

1. `firmware:` iSCSI 引导固件表 (iBFT)
2. `compat:` `ifcfg` 文件 — 为兼容性而实施

将会应用 `wicked` 从指定接口的这些源中获取的任何设置。预期的重要性顺序为 `firmware`、`compat` - 将来此顺序可能会发生变化。

有关更多信息，请参见 `wicked` 手册页。

16.5.1.3 Nanny

Nanny 是一个事件与策略驱动的守护程序，负责热插拔设备等异步或被动性方案。因此，Nanny 守护程序可帮助启动或者重启动延迟的设备或临时消失的设备。Nanny 将监视设备和链路变化，并集成当前策略集定义的新设备。即使 `ifup` 已经因指定的超时约束而退出，Nanny 也会继续设置。

默认情况下，Nanny 守护程序在系统上处于活动状态。您可在 `/etc/wicked/common.xml` 配置文件中启用它：

```
<config>
  ...
  <use-nanny>true</use-nanny>
</config>
```

如果使用此设置，ifup 和 ifreload 会将包含有效配置的策略应用到 Nanny 守护程序；然后，Nanny 将配置 `wickedd`，从而确保支持热插拔。它将在后台等待事件或更改（例如，打开新的设备或载体）。

16.5.1.4 打开多个接口

对于绑定和网桥，有效的做法是在一个文件 (ifcfg-bondX) 中定义整个设备拓扑，并一次性将它激活。然后，当您指定（网桥或绑定的）顶级接口名称时，wicked 可以激活整个配置：

```
wicked ifup br0
```

此命令会按适当的顺序自动设置网桥及其依赖项，而无需分别列出依赖项（端口等）。

要在一个命令中激活多个接口：

```
wicked ifup bond0 br0 br1 br2
```

要激活所有接口：

```
wicked ifup all
```

16.5.1.5 通过 Wicked 使用隧道

如果您需要将隧道与 Wicked 结合使用，可以使用专门针对此用途的 `TUNNEL_DEVICE`。它可让您指定一个可选的设备名称，以将隧道绑定至该设备。隧道式包将只能通过此设备路由。

有关详细信息，请参见 `man 5 ifcfg-tunnel`。

16.5.1.6 处理增量变化

有了 `wicked`，当您要重新配置某个接口时，并不需要真正将它关闭（除非内核有此要求）。例如，要将另一个 IP 地址或路由添加到静态配置的网络接口，请将该 IP 地址添加到接口定义，然后再次执行“ifup”操作。服务器会尽量做到只更新那些已更改的设置。这适用于链路级选项，例如设备 MTU 或 MAC 地址；也适用于网络级设置，例如地址、路由，甚至地址配置模式（例如，从静态配置转为 DHCP 时）。

当然，对于合并了多个实体设备（例如桥接或绑定设备）的虚拟接口，事情会变得有些棘手。对于绑定设备，当设备运行时，您无法更改某些参数，否则会导致出错。

但是，您仍可以添加或去除绑定设备或桥接的子设备，或者选择绑定设备的主接口。

16.5.1.7 Wicked 扩展：地址配置

`wicked` 设计为使用外壳脚本扩展。这些扩展可在 `config.xml` 文件中定义。

目前支持多个种类的扩展：

- 链路配置：这些脚本负责根据客户端提供的配置来设置设备的链路层，以及负责将链路层再次拆解。
- 地址配置：这些脚本负责管理设备的地址配置。通常，地址配置和 DHCP 由 `wicked` 自身管理，但是，可借助扩展来实现。
- 防火墙扩展：这些脚本可以应用防火墙规则。

通常，扩展中包含一个启动命令和一个停止命令、一个可选的“pid 文件”，以及传递给脚本的一组环境变量。

为了演示此扩展的工作原理，请查看 `etc/server.xml` 中定义的防火墙扩展：

```
<dbus-service interface="org.opensuse.Network.Firewall">
```

```
 <action name="firewallUp"   command="/etc/wicked/extensions/firewall up"/>
 <action name="firewallDown" command="/etc/wicked/extensions/firewall down"/>

 <!-- default environment for all calls to this extension script -->
 <putenv name="WICKED_OBJECT_PATH" value="$object-path"/>
 <putenv name="WICKED_INTERFACE_NAME" value="$property:name"/>
 <putenv name="WICKED_INTERFACE_INDEX" value="$property:index"/>
</dbus-service>
```

该扩展附加到 `<dbus-service>` 标记，并定义针对此接口的操作应执行的命令。此外，声明可以定义并初始化传递给操作的环境变量。

16.5.1.8 Wicked 扩展：配置文件

您也可以使用脚本来扩展配置文件的处理。例如，`extensions/resolver` 脚本根据 `server.xml` 中配置的行为来最终处理租用中的 DNS 更新：

```
<system-updater name="resolver">
 <action name="backup" command="/etc/wicked/extensions/resolver backup"/>
 <action name="restore" command="/etc/wicked/extensions/resolver restore"/>
 <action name="install" command="/etc/wicked/extensions/resolver install"/>
 <action name="remove" command="/etc/wicked/extensions/resolver remove"/>
</system-updater>
```

当 `wickedd` 中收到更新时，系统更新程序例程将分析租用，并调用解析程序脚本中的适当命令（`backup`、`install` 等）。此后便可以使用 `/sbin/netconfig` 或者通过手动写入 `/etc/resolv.conf`（作为回退）来配置 DNS 设置。

16.5.2 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。

16.5.2.1 /etc/wicked/common.xml

/etc/wicked/common.xml 文件包含所有应用程序应使用的通用定义。该文件源自/包含在此目录中的其他配置文件中。尽管您可以使用此文件允许在所有 wicked 组件间进行调试，但建议使用 /etc/wicked/local.xml 文件来实现此目的。应用维护更新后，您所做的更改可能会丢失，因为 /etc/wicked/common.xml 可能会被覆盖。/etc/wicked/common.xml 文件包含默认安装中的 /etc/wicked/local.xml，因此您通常不需要修改 /etc/wicked/common.xml。

如果要通过将 <use-nanny> 设置为 false 来禁用 nanny，请重启动 wickedd.service，然后运行以下命令以应用所有配置和策略：

```
wicked ifup all
```

注意：配置文件

如果 wickedd、wicked 或 nanny 程序自身的配置文件不存在，则会尝试读取 /etc/wicked/common.xml。

16.5.2.2 /etc/wicked/server.xml

wickedd 服务器进程会在启动时读取文件 /etc/wicked/server.xml。该文件将扩展储存到 /etc/wicked/common.xml 中。除此之外，此文件可配置解析程序的处理方式，以及从 addrconf 请求方（例如 DHCP）接收信息的方式。

建议您将需要对此文件进行的所有更改都添加到单独的文件 /etc/wicked/server-local.xml（/etc/wicked/server.xml）。使用单独的文件可避免在维护更新期间覆盖您的更改。

16.5.2.3 /etc/wicked/client.xml

/etc/wicked/client.xml 供 wicked 命令使用。该文件指定发现 ibft 管理的设备时所用脚本的位置，并可配置网络接口配置的位置。

建议您将需要对此文件进行的所有更改都添加到单独的文件 `/etc/wicked/client-local.xml`（`/etc/wicked/server.xml` 会包含该文件的内容）。使用单独的文件可避免在维护更新期间覆盖您的更改。

16.5.2.4 `/etc/wicked/nanny.xml`

`/etc/wicked/nanny.xml` 配置链接层的类型。建议您将特定配置添加到单独的文件 `/etc/wicked/nanny-local.xml` 中，以免在维护更新期间丢失更改。

16.5.2.5 `/etc/sysconfig/network/ifcfg-*`

这些文件包含网络接口的传统配置。在 SUSE Linux Enterprise 11 中，这是除 iBFT 固件以外唯一支持的格式。

注意：`wicked` 和 `ifcfg-*` 文件

如果您指定 `compat:` 前缀，`wicked` 将读取这些文件。根据 `/etc/wicked/client.xml` 中 SUSE Linux Enterprise Server 的默认配置，`wicked` 将尝试先读取这些文件，然后再读取 `/etc/wicked/ifconfig` 中的 XML 配置文件。

通常，提供 `--ifconfig` 开关仅用于测试。如果指定该开关，则不会应用 `/etc/wicked/ifconfig` 中定义的默认配置源。

`ifcfg-*` 文件包含启动模式和 IP 地址等信息。可能的参数在 `ifup` 的手册页中有所介绍。此外，如果一个常规设置只能用于一个接口，则文件 `dhcp` 和 `wireless` 中的大多数变量可用于 `ifcfg-*` 文件。但是，`/etc/sysconfig/network/config` 中的大多数变量是全局变量，不能在 ifcfg-files 中被覆盖。例如，`NETCONFIG_*` 变量是全局变量。

要配置 `macvlan` 和 `macvtab` 接口，请参见 `ifcfg-macvlan` 和 `ifcfg-macvtap` 手册页。例如，对于 macvlan 接口，请提供使用以下设置的 `ifcfg-macvlan0`：

```
STARTMODE='auto'
MACVLAN_DEVICE='eth0'
#MACVLAN_MODE='vepa'
#LLADDR=02:03:04:05:06:aa
```

有关 ifcfg.template 的信息，请参见第 16.5.2.6 节 “/etc/sysconfig/network/config、/etc/sysconfig/network/dhcp 和 /etc/sysconfig/network/wireless”。

System z IBM Z 不支持 USB。接口文件的名称和网络别名包含专用于 IBM Z 的元素，例如 qeth。

16.5.2.6 /etc/sysconfig/network/config、/etc/sysconfig/network/dhcp 和 /etc/sysconfig/network/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。所有三个配置文件中的变量均已注释掉。/etc/sysconfig/network/config 中的一些变量也可用于 ifcfg-* 文件，在这些文件中它们具有更高优先级。/etc/sysconfig/network/ifcfg.template 文件列出可以按接口指定的变量。但是，/etc/sysconfig/network/config 中的大多数变量是全局变量，不能在 ifcfg-files 中被覆盖。例如，NETWORKMANAGER 或 NETCONFIG_* 变量是全局变量。

注意：使用 DHCPv6

在 SUSE Linux Enterprise 11 中，即使是在未正确配置 IPv6 路由器广播 (RA) 的网络中，DHCPv6 一向也能正常工作。从 SUSE Linux Enterprise 12 开始，DHCPv6 将适当地要求网络中至少有一个路由器发出 RA，用于指示此网络由 DHCPv6 管理。

对于无法在其中正确配置路由器的网络，用户可以通过在 ifcfg 文件中指定 DHCLIENT6_MODE='managed'，使用 ifcfg 选项来覆盖此行为。您也可以在安装系统中使用引导参数来启用这种解决方法：

```
ifcfg=eth0=dhcp6,DHCLIENT6_MODE=managed
```

16.5.2.7 /etc/sysconfig/network/routes 和 /etc/sysconfig/network/ifroute-*

TCP/IP 包的静态路由是 `/etc/sysconfig/network/routes` 和 `/etc/sysconfig/network/ifroute-*` 文件确定的。可以在 `/etc/sysconfig/network/routes` 中指定各种系统任务所需的所有静态路由：主机的路由、主机通过网关的路由以及网络的路由。对于需要个别路由的每个接口，定义另一个配置文件：`/etc/sysconfig/network/ifroute-*`。将通配符 (`*`) 替换为接口名称。路由选择配置文件中的项如下所示：

```
# Destination     Gateway           Netmask           Interface  Options
```

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。应该以 CIDR 表示法（地址加上关联的路由前缀长度）输入网络名称，例如 10.10.0.0/16（对于 IPv4 路由）或 fc00::/7（对于 IPv6 路由）。关键字 `default` 表示该路由是与网关位于相同地址系列中的默认网关。对于没有网关的设备，请使用显式 0.0.0.0/0 或 ::/0 目标。

第二列包含默认网关或通过其可访问主机或网络的网关。

第三列已弃用；它用于包含目标的 IPv4 网络掩码。对于 IPv6 路由、默认路由，或者如果在第一列中使用了前缀长度（CIDR 表示法），请在此处输入破折号 (`-`)。

第四列包含接口名称。如果使用破折号 (`-`) 将它保留空白，可能会导致 `/etc/sysconfig/network/routes` 出现意外的行为。有关更多信息，请参见 `routes` 手册页。

第五列（可选）可用于指定特殊选项。有关详细信息，请参见 `routes` 手册页。

例 16.5 通用网络接口和部分静态路由

```
# --- IPv4 routes in CIDR prefix notation:
# Destination     [Gateway]         -                 Interface
127.0.0.0/8       -                 -                 lo
204.127.235.0/24  -                 -                 eth0
default           204.127.235.41    -                 eth0
207.68.156.51/32  207.68.145.45     -                 eth1
192.168.0.0/16    207.68.156.51     -                 eth1

# --- IPv4 routes in deprecated netmask notation"
# Destination     [Dummy/Gateway]   Netmask           Interface
```

```
#
127.0.0.0          0.0.0.0           255.255.255.0      lo
204.127.235.0      0.0.0.0           255.255.255.0      eth0
default            204.127.235.41    0.0.0.0            eth0
207.68.156.51      207.68.145.45     255.255.255.255    eth1
192.168.0.0        207.68.156.51     255.255.0.0        eth1

# --- IPv6 routes are always using CIDR notation:
# Destination      [Gateway]                 -          Interface
2001:DB8:100::/64 -                          -          eth0
2001:DB8:100::/32 fe80::216:3eff:fe6d:c042 -            eth0
```

16.5.2.8 /etc/resolv.conf

主机所属的域在 `/etc/resolv.conf` 中指定（关键字 `search`）。使用 `search` 选项最多可以指定六个域，总共 256 个字符。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。使用 `nameserver` 选项最多可以指定 3 个名称服务器，每行指定一个。注释以井号或分号（`#` 或 `;`）开头。有关示例，请参见例 16.6 “/etc/resolv.conf”。

但是，`/etc/resolv.conf` 不应手动编辑。而是由 `netconfig` 脚本生成。要定义静态 DNS 配置而不使用 YaST，请手动编辑 `/etc/sysconfig/network/config` 文件中的相应变量：

`NETCONFIG_DNS_STATIC_SEARCHLIST`
: 用于主机名查找的 DNS 域名列表

`NETCONFIG_DNS_STATIC_SERVERS`
: 用于主机名查找的名称服务器 IP 地址列表

`NETCONFIG_DNS_FORWARDER`
: 需要配置的 DNS 转发器名称，例如 `bind` 或 `resolver`

`NETCONFIG_DNS_RESOLVER_OPTIONS`
: 将写入到 `/etc/resolv.conf` 的任意选项，例如：

```
debug attempts:1 timeout:10
```

有关更多信息，请参见 `resolv.conf` 手册页。

`NETCONFIG_DNS_RESOLVER_SORTLIST`

最多包含 10 项的列表，例如：

```
130.155.160.0/255.255.240.0 130.155.0.0
```

有关更多信息，请参见 `resolv.conf` 手册页。

要使用 netconfig 禁用 DNS 配置，请设置 `NETCONFIG_DNS_POLICY=''`。有关 `netconfig` 的更多信息，请参见 `netconfig(8)` 手册页 (`man 8 netconfig`)。

例 16.6 `/etc/resolv.conf`

```
# Our domain
search example.com
#
# We use dns.example.com (192.168.1.116) as nameserver
nameserver 192.168.1.116
```

16.5.2.9 /sbin/netconfig

`netconfig` 是一个用于管理附加网络配置设置的模块化工具。它合并了静态定义的设置和自动配置机制根据预定义策略以 DHCP 或 PPP 形式提供的设置。通过调用负责修改配置文件和重启动服务或相似操作的 netconfig 模块将所需更改应用于系统。

`netconfig` 识别三种主要操作。`netconfig modify` 和 `netconfig remove` 命令由诸如 DHCP 或 PPP 的守护程序用于在 netconfig 中提供设置或从中删除设置。仅 `netconfig update` 命令可用于用户：

`modify`

`netconfig modify` 命令修改特定于当前接口和服务的动态设置并更新网络配置。Netconfig 会从标准输入或从使用 `--lease-file FILENAME` 选项指定的文件中读取设置，并将其储存在内部，直到系统重引导（或者执行下一个修改或删除操作）为止。已存在的相同接口和服务组合设置将会重写。该接口由 `-i INTERFACE_NAME` 参数指定。该服务由 `-s SERVICE_NAME` 参数指定。

`remove`

`netconfig remove` 命令为指定接口和服务组合删除由修改操作提供的动态设置并更新网络配置。该接口由 `-i INTERFACE_NAME` 参数指定。该服务由 `-s SERVICE_NAME` 参数指定。

`update`

`netconfig update` 命令使用当前设置更新网络配置。当策略或静态配置更改时，这非常有用。如果想要只更新指定服务（`dns`、`nis` 或 `ntp`），请使用 `-m MODULE_TYPE` 参数。

netconfig 策略和静态配置设置可手动定义或者使用 YaST 在 `/etc/sysconfig/network/config` 文件中定义。自动配置工具（例如 DHCP 或 PPP）提供的动态配置设置由这些工具通过 `netconfig modify` 和 `netconfig remove` 操作直接递送。启用 NetworkManager 时，netconfig（在策略模式 `auto` 中）仅使用 NetworkManager 设置，忽略使用传统 ifup 方法配置的任何其他接口的设置。如果 NetworkManager 未提供任何设置，将使用静态设置作为后备设置。不支持混合使用 NetworkManager 和 `wicked` 方法。

有关 `netconfig` 的更多信息，请参见 `man 8 netconfig`。

16.5.2.10 /etc/hosts

在此文件中，如例 16.7 “/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，则将与其建立 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数据，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。

例 16.7 /etc/hosts

```
127.0.0.1 localhost
192.168.2.100 jupiter.example.com jupiter
192.168.2.101 venus.example.com venus
```

16.5.2.11 /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 16.8 “/etc/networks”。

例 16.8 /etc/networks

```
loopback      127.0.0.0
localnet      192.168.0.0
```

16.5.2.12 /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，请参见 /etc/nsswitch.conf 中的设置。每个参数都必须始终在单独的一行中输入。注释以 # 符号开头。表 16.2 “/etc/host.conf 的参数” 显示了可用的参数。例 16.9 “/etc/host.conf” 中显示了 /etc/host.conf 的示例。

表 16.2 /ETC/HOST.CONF 的参数

order hosts、bind	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）：
	hosts：搜索 /etc/hosts 文件
	bind：访问名称服务器
	nis：使用 NIS
multi on/off	定义 /etc/hosts 中输入的主机是否可以具有多个 IP 地址。
nospoof on spoofalert on/off	这些参数影响名称服务器 spoofing，但对网络配置没有任何影响。
trim domainname	在主机名解析后，指定的域名将与主机名分开（前提是主机名包含域名）。此选项仅当本地域名在 /etc/hosts 文件中时才有用，但仍应通过附带的域名进行识别。

例 16.9 /etc/host.conf

```
# We have named running
```

```
order hosts bind
# Allow multiple address
multi on
```

16.5.2.13 /etc/nsswitch.conf

GNU C Library 2.0 的引入与 名称服务转换 (NNS) 的引入是同时进行的。有关详细信息，请参见 `nsswitch.conf(5)` 手册页和《GNU C 库参考手册》。

查询的顺序是在文件 `/etc/nsswitch.conf` 中定义的。例 16.10 “/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 26 章 “域名系统”）将请求发送到 `/etc/hosts`（`files`）。

例 16.10 /etc/nsswitch.conf

```
passwd:     compat
group:      compat

hosts:      files dns
networks:   files dns

services:   db files
protocols:  db files
rpc:        files
ethers:     files
netmasks:   files
netgroup:   files nis
publickey:  files

bootparams: files
automount:  files nis
aliases:    files nis
shadow:     compat
```

表 16.3 “通过 /etc/nsswitch.conf 可用的数据库” 中列出了 NSS 上可用的“数据库”。

表 16.4 “NSS“数据库”的配置选项” 中列出了 NSS 数据库的配置选项。

表 16.3 通过 /ETC/NSSWITCH.CONF 可用的数据库

aliases	sendmail 实施的邮件别名；请参见 man 5 aliases。
ethers	以太网地址。
netmasks	网络及其子网掩码的列表。只有在使用子网划分时才需要。
group	getgrent 使用的用户组。另请参见 group 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有效主机和用户列表，请参见 netgroup(5) 手册页。
networks	getnetent 使用的网络名称和地址。
publickey	NFS 和 NIS+ 使用的 Secure_RPC 的公钥和密钥。
password	getpwent 使用的用户口令；请参见 passwd(5) 手册页。
protocols	网络协议，由 getprotoent 使用；请参见 protocols(5) 手册页。
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	用户阴影口令，由 getspnam 使用；请参见 shadow(5) 手册页。

表 16.4 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《Security Guide》，第 3 章 “Using NIS”
dns	仅可用作 hosts 和 networks 的扩展名
compat	仅可用作 passwd、shadow 和 group 的扩展名

16.5.2.14 /etc/nscd.conf

此文件用于配置 nscd（名称服务缓存守护程序）。请参见 nscd(8) 和 nscd.conf(5) 手册页。默认情况下，passwd、groups 和 hosts 的系统项由 nscd 进行缓存。这对 NIS 和 LDAP 等目录服务的性能而言非常重要，否则，每次访问名称、组或主机都需要网络连接。

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。使用以下命令重启动 nscd，缩短这段等待时间：

```
systemctl restart nscd
```

16.5.2.15 /etc/HOSTNAME

/etc/HOSTNAME 包含完全限定的主机名 (FQHN)。完全限定的主机名是附有域名的主机名。此文件只能包含一行（在此行中设置主机名）。计算机引导时会读取此文件。

16.5.3 测试配置

向配置文件写配置之前，可对其进行测试。要设置测试配置，请使用 ip 命令。要测试连接，请使用 ping 命令。

命令 `ip` 会直接更改网络配置，而不会将其保存到配置文件中。如果未在正确的配置文件中输入配置，重引导时将丢失已更改的网络配置。

注意：`ifconfig` 和 `route` 已过时

`ifconfig` 和 `route` 工具已过时。请改为使用 `ip`。例如，`ifconfig` 会将接口名限制为 9 个字符。

16.5.3.1 使用 `ip` 配置网络接口

`ip` 是用来显示和配置网络设备、路由选择、策略路由选择以及隧道的工具。

`ip` 是非常复杂的工具。它的常用语法为 `ip OPTIONS OBJECT COMMAND`。可使用以下对象：

link
: 此对象表示网络设备。

address
: 此对象表示设备的 IP 地址。

neighbor
: 此对象表示 ARP 或 NDISC 超速缓存项。

route
: 此对象表示路由选择表项。

rule
: 此对象表示路由选择策略数据库中的规则。

maddress
: 此对象表示多路广播地址。

mroute
: 此对象表示多路广播路由缓存项。

tunnel
: 此对象表示 IP 上的隧道。

如果未提供命令，则将使用默认命令（通常为 `list`）。

使用 `ip link set DEVICE_NAME` 命令更改设备的状态。例如，要停用设备 eth0，请输入 `ip link set eth0 down`。要重激活它，可使用 `ip link set eth0 up`。

激活设备后，可对设备进行配置。要设置 IP 地址，可使用 `ip addr add IP_ADDRESS + dev DEVICE_NAME`。例如，要将接口 eth0 的地址设置为带标准广播（选项 `brd`）的 192.168.12.154/30，则输入 `ip addr add 192.168.12.154/30 brd + dev eth0`。

要拥有活动连接，还必须配置默认网关。要设置系统的网关，请输入 `ip route add gateway_ip_address`。要将一个 IP 地址转换为另一个 IP 地址，请使用 `nat`：`ip route add nat ip_address via other_ip_address`。

要显示所有设备，可使用 `ip link ls`。要只显示正在运行的接口，可使用 `ip link ls up`。要打印设备的接口统计信息，可输入 `ip -s link ls device_name`。要查看设备的地址，请输入 `ip addr`。在 `ip addr` 的输出中，还可找到有关设备 MAC 地址的信息。要显示所有路由，可使用 `ip route show`。

有关使用 `ip` 的更多信息，请输入 `ip help` 或参见 `ip(8)` 手册页。`help` 选项还可用于所有 `ip` 子命令。例如，如果需要有关 `ip addr` 的帮助，请输入 `ip addr help`。可在 `/usr/share/doc/packages/iproute2/ip-cref.pdf` 中找到 `ip` 手册。

16.5.3.2 使用 ping 测试连接

`ping` 命令是用于测试 TCP/IP 连接是否有效的标准工具。它使用 ICMP 协议来将小数据包和 ECHO_REQUEST 数据报文发送到目标主机，并请求即时答复。如果成功，`ping` 将显示表示这一结果的消息。这表示网络链路正在运作。

`ping` 不仅能测试两台计算机之间的连接功能：它还能提供关于连接质量的一些基本信息。在例 16.11 “命令 ping 的输出” 中，可查看 `ping` 输出示例。倒数第二行包含有关已传输的包数、丢失的包和 `ping` 的总运行时间的信息。

因此，您可以使用主机名或 IP 地址（例如 `ping example.com` 或 `ping 192.168.3.100`）。程序会一直发送包，直到您按 Ctrl–C。

如果只需要检查连接功能，则可使用 `-c` 选项来限制包数。例如，要将 ping 限制为三个包，请输入 `ping -c 3 example.com`。

例 16.11 命令 PING 的输出

```
ping -c 3 example.com
```

```
PING example.com (192.168.3.100) 56(84) bytes of data.
64 bytes from example.com (192.168.3.100): icmp_seq=1 ttl=49 time=188 ms
64 bytes from example.com (192.168.3.100): icmp_seq=2 ttl=49 time=184 ms
64 bytes from example.com (192.168.3.100): icmp_seq=3 ttl=49 time=183 ms
--- example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 183.417/185.447/188.259/2.052 ms
```

两个包之间的默认时间间隔为一秒。ping 提供了选项 `-i` 来更改间隔。例如，要将 ping 间隔增加到十秒，请输入 `ping -i 10 example.com`。

在带有多个网络设备的系统中，有时通过特定接口地址发送 ping 将会非常有用。要执行此操作，可将 `-I` 选项结合选定设备名称一起使用，例如 `ping -I wlan1 example.com`。

有关使用 ping 的更多选项和信息，请输入 `ping -h` 或查看 `ping (8)` 手册页。

提示：Ping IPv6 地址

对于 IPv6 地址，请使用 `ping6` 命令。请注意，要 ping 本地链路地址，必须用 `-I` 指定接口。如果通过 `eth1` 可获取地址，则以下命令有效：

```
ping6 -I eth1 fe80::117:21ff:feda:a425
```

16.5.4 单元文件和启动脚本

除了上面介绍的配置文件之外，还存在一些负责在引导计算机时装载网络服务的 systemd 单元文件和多个脚本。系统切换到 `multi-user.target` 目标后，即会启动这些文件和脚本。网络程序的一些单元文件和启动脚本中介绍了其中的部分单元文件和脚本。有关 `systemd` 的更多信息，请参见第 13 章 “systemd 守护程序”；有关 `systemd` 目标的更多信息，请参见 `systemd.special` 的手册页 (`man systemd.special`)。

网络程序的一些单元文件和启动脚本

`network.target`

`network.target` 是网络的 systemd 目标，但其具体含义取决于系统管理员提供的设置。

有关详细信息，请参见 http://www.freedesktop.org/wiki/Software/systemd/NetworkTarget/。

`multi-user.target`

`multi-user.target` 是包含所有必需网络服务的多用户系统的 systemd 目标。

`xinetd`

启动 xinetd。xinetd 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 vsftpd。

`rpcbind`

启动用于将 RPC 程序号转换为通用地址的 rpcbind 实用程序。它是 RPC 服务所必需的，如 NFS 服务器。

`ypserv`

启动 NIS 服务器。

`ypbind`

启动 NIS 客户端。

`/etc/init.d/nfsserver`

启动 NFS 服务器。

`/etc/init.d/postfix`

控制 postfix 进程。

16.6 路由器基本设置

路由器是一种联网设备，可收发往来于多个网络的数据（网络包）。路由器常用于将本地网络连接到远程网络（因特网）或连接多个本地网段。通过 SUSE Linux Enterprise Server，您可以构建一个具备 NAT（网络地址转换）或高级防火墙等功能的路由器。

下面是将 SUSE Linux Enterprise Server 转变为路由器的基本步骤。

1. 例如，在 `/etc/sysctl.d/50-router.conf` 中启用转发

```
net.ipv4.conf.all.forwarding = 1
net.ipv6.conf.all.forwarding = 1
```

然后提供接口的静态 IPv4 和 IPv6 IP 设置。启用转发会禁用多种机制，例如，IPv6 不再接受 IPv6 RA（路由器广告），这也会阻止创建默认路由。

2. 在许多情况下，例如，当您可以通过多个接口连接同一个网络，或者通常使用的是 VPN（已位于“常规多宿主主机”上）时，必须禁用 IPv4 反向路径过滤（此功能当前不适用于 IPv6）：

```
net.ipv4.conf.all.rp_filter = 0
```

您也可以改为通过防火墙设置进行过滤。

3. 要从外部、上行或 ISP 接口上的路由器接受 IPv6 RA 并重新创建默认（或者更具特定性）的 IPv6 路由，请设置：

```
net.ipv6.conf.${ifname}.accept_ra = 2
net.ipv6.conf.${ifname}.autoconf = 0
```

（注意：在以点分隔的 sysfs 路径中，“`eth0.42`”需写成 `eth0/42`。）

有关更多路由器行为和转发依赖项的信息，请参见 https://www.kernel.org/doc/Documentation/networking/ip-sysctl.txt。

要在内部 (DMZ) 接口上提供 IPv6 并将您自己通告为 IPv6 路由器，同时为客户端“自动配置网络”，请安装 `radvd` 并在 `/etc/radvd.conf` 中进行配置，例如：

```
interface eth0
{
    IgnoreIfMissing on;         # do not fail if interface missed

    AdvSendAdvert on;           # enable sending RAs
    AdvManagedFlag on;          # IPv6 addresses managed via DHCPv6
    AdvOtherConfigFlag on;      # DNS, NTP... only via DHCPv6

    AdvDefaultLifetime 3600;    # client default route lifetime of 1 hour

    prefix 2001:db8:0:1::/64    # (/64 is default and required for autoconf)
    {
        AdvAutonomous off;         # Disable address autoconf (DHCPv6 only)
```

```
        AdvValidLifetime 3600;     # prefix (autoconf addr) is valid 1 h
        AdvPreferredLifetime 1800; # prefix (autoconf addr) is prefered 1/2 h
    }
}
```

最后配置防火墙。在 SuSEfirewall2 中，您需要设置 `FW_ROUTE="yes"`（否则它将再次重设置转发 sysctl），并根据需要定义 `FW_DEV_INT`、`FW_DEV_EXT`（和 `FW_DEV_DMZ`）区域变量中的接口，另外可能还需要定义 `FW_MASQUERADE="yes"` 和 `FW_MASQ_DEV`。

16.7 设置绑定设备

对于某些系统，需要实施高于典型以太网设备的标准数据安全性或可用性要求的网络连接。在这些情况下，可以将多个以太网设备聚合到单个绑定设备。

绑定设备的配置通过绑定模块选项来完成。其行为主要受绑定设备模式的影响。默认情况下是 `active-backup`，即如果活动从属设备发生故障，则另一个从属设备将变成活动从属设备。可以使用以下绑定模式：

0 (balance-rr)
: 数据包依次通过第一个到最后一个可用接口传输。提供容错和负载平衡。

1 (active-backup)
: 只有一个网络接口处于活动状态。如果它发生故障，另一个接口将变成活动状态。此设置是 SUSE Linux Enterprise Server 的默认设置。提供容错。

2 (balance-xor)
: 根据以下策略在所有可用接口间拆分通讯：`[(source MAC address XOR'd with destination MAC address XOR packet type ID) modulo slave count]` 需要交换机的支持。提供容错和负载平衡。

3 (broadcast)
: 在所有接口上广播所有通讯。需要交换机的支持。提供容错。

4 (802.3ad)
: 将接口聚合成共享相同速度和双工设置的组。需要接口驱动程序中的 `ethtool` 支持，以及支持 IEEE 802.3ad 动态链路聚合并进行了相应配置的交换机。提供容错和负载平衡。

5 (balance-tlb)

自适应传输负载平衡。需要接口驱动程序中的 `ethtool` 支持，但不需要交换机支持。提供容错和负载平衡。

6 (balance-alb)

自适应负载平衡。需要接口驱动程序中的 `ethtool` 支持，但不需要交换机支持。提供容错和负载平衡。

有关各种模式的详细说明，请参见https://www.kernel.org/doc/Documentation/networking/bonding.txt。

提示：绑定和 Xen

绑定设备只对于有多个真实网卡可用的计算机有效。这意味着在大多数配置中，您仅应在 Dom0 中使用绑定配置。换言之，只有当您将多个网卡指派给一个 VM Guest 系统时，在 VM Guest 中设置绑定才有效。

注意： IBM POWER：ibmveth 不支持绑定模式 5 和 6 (balance-tlb/balance-alb)

tlb/alb 绑定配置与 Power 固件之间存在冲突。简单而言，处于 tlb/alb 模式的绑定驱动程序会发送同时包含源和目标 MAC 地址（列作虚拟以太网 MAC 地址）的以太网回写包。Power 固件并不支持这些包。因此，ibmveth 不支持绑定模式 5 和 6。

要配置绑定设备，请使用以下过程：

1. 运行 YaST › 系统 › 网络设置。
2. 使用添加并将设备类型更改为绑定。单击下一步继续。

3. 选择如何为绑定设备指派 IP 地址。有三种方法可供选择：

   - 无 IP 地址
   - 动态地址（使用 DHCP 或 Zeroconf）
   - 静态指派的 IP 地址

   请使用最适合您环境的方法。

4. 在绑定从属选项卡中，通过激活相关复选框选择应加入到绑定中的以太网设备。

5. 编辑绑定驱动程序选项并选择绑定模式。

6. 确保将参数 `miimon=100` 添加到绑定驱动程序选项。如果没有此参数，则不会定期检查数据完整性。

7. 单击下一步，然后单击确定退出 YaST 以创建设备。

16.7.1 绑定从属的热插拔

在特定网络环境（如高可用性）下，有几种情况需要替换绑定从属接口。原因可能在于网络设备持续故障。解决方案是设置绑定从属的热插拔。

按常规配置绑定（按照 `man 5 ifcfg-bonding`），例如：

```
ifcfg-bond0
          STARTMODE='auto' # or 'onboot'
          BOOTPROTO='static'
          IPADDR='192.168.0.1/24'
          BONDING_MASTER='yes'
          BONDING_SLAVE_0='eth0'
          BONDING_SLAVE_1='eth1'
          BONDING_MODULE_OPTS='mode=active-backup miimon=100'
```

使用 `STARTMODE=hotplug` 和 `BOOTPROTO=none` 指定从属：

```
ifcfg-eth0
          STARTMODE='hotplug'
          BOOTPROTO='none'

ifcfg-eth1
          STARTMODE='hotplug'
          BOOTPROTO='none'
```

`BOOTPROTO=none` 使用 `ethtool` 选项（如果提供），但不会在 `ifup eth0` 上设置链路，因为从属接口由绑定主接口控制。

`STARTMODE=hotplug` 会使从属接口在可用时自动加入绑定。

需要更改 `/etc/udev/rules.d/70-persistent-net.rules` 中的 `udev` 规则，以便按总线 ID（udev `KERNELS` 关键字等同于 `hwinfo --netcard` 中的“SysFS BusID”）而不是 MAC 地址匹配设备。这样允许更换有缺陷的硬件（位于同一插槽但 MAC 不同的网卡），并避免在绑定更改其所有从属设备的 MAC 地址时出现混淆。

例如：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
KERNELS=="0000:00:19.0", ATTR{dev_id}=="0x0", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth0"
```

在引导时，systemd `network.service` 不会等待热插拔从属，但会等待绑定准备就绪，而这需要至少有一个从属可用。当从系统中去除一个从属接口时（从 NIC 驱动程序拆开联结、执行 NIC 驱动程序的 `rmmod` 命令或 PCI 热插拔去除为 true），内核会自动从绑定中将其去除。当

向系统添加新网卡时（替换插槽中的硬件），`udev` 会使用基于总线的网卡设备名称规则将其重命名为从属接口的名称，并为其调用 `ifup` 命令。`ifup` 命令会自动调用以将新网卡加入绑定。

16.8 设置小组设备以进行网络协作

“链路聚合”属于通用术语，指组合（或聚合）网络连接以提供逻辑层。有时，您还会看到“通道聚合”、“以太网绑定”、“端口汇聚”等术语，这些同义词都是指同一个概念。

这个概念通常被称为“绑定”，最初是集成到 Linux 内核中的（请参见第 16.7 节 “设置绑定设备”了解最初的实现方式）。网络协作一词用来指代这个概念的新实现方式。

绑定和网络协作之间的主要差别在于，协作提供一组小型内核模块，由它们负责提供用于 teamd 实例的接口。其他一切都在用户空间中处理。这一点与最初的绑定实现方式不同，后者是将自己的所有功能都专门包含在内核中。如需两者的比较，请参见表 16.5 “绑定与组合的功能比较”。

表 16.5 绑定与组合的功能比较

特性	绑定	组合
广播、循环 TX 策略	是	是
活动备份 TX 策略	是	是
LACP (802.3ad) 支持	是	是
基于哈希的 TX 策略	是	是
用户可以设置哈希函数	否	是
TX 负载平衡支持 (TLB)	是	是
针对 LACP 的 TX 负载平衡支持	否	是
Ethtool 链接监视	是	是
ARP 链接监视	是	是
NS/NA (IPV6) 链接监视	否	是

特性	绑定	组合
针对 TX/RX 路径的 RCU 锁定	否	是
端口优先级和粘性	否	是
单独的按端口链接监视设置	否	是
多链接监视设置	有限制	是
VLAN 支持	是	是
多设备堆叠	是	是

源: http://libteam.org/files/teamdev.pp.pdf

绑定和网络协作这两种实现方式可以并行使用。可将网络协作作为现有绑定实现方式的备选。它不会取代绑定。

网络协作可用于不同使用情况。稍后将会介绍其中两种最重要的使用情况，它们涉及：

- 不同网络设备之间的负载平衡。
- 从一个网络设备到另一个网络设备的故障转移（当其中一个设备出现故障时）。

目前没有用于支持创建协作设备的 YaST 模块。您需要手动配置网络协作。以下是适用于所有网络协作配置的一般过程：

过程 16.1 一般过程

1. 确保已安装所有必需的包。安装包 `libteam-tools`、`libteamdctl0` 和 `python-libteam`。
2. 在 `/etc/sysconfig/network/` 下创建一个配置文件，通常为 `ifcfg-team0`。如果您需要多个网络协作设备，请为它们指定依次递增的编号。
   该配置文件包含若干变量，手册页中对这些变量做了说明（请参见 `man ifcfg` 和 `man ifcfg-team`）。系统内的 `/etc/sysconfig/network/ifcfg.template` 文件中提供了示例配置。
3. 去除将用于协作设备的接口的配置文件（通常为 `ifcfg-eth0` 和 `ifcfg-eth1`）。
   建议您先备份这两个文件，然后再将其去除。Wicked 将会使用协作的必要参数重新创建配置文件。

4. （可选）检查 Wicked 的配置文件中是否已包含所有内容：

```
wicked show-config
```

5. 启动网络协作设备 `team0`：

```
wicked ifup all team0
```

如果您需要其他调试信息，请在 `all` 子命令后面使用 `--debug all` 选项。

6. 检查网络协作设备的状态。通过执行以下命令可以完成该操作：

- 从 Wicked 获取 teamd 实例的状态：

```
wicked ifstatus --verbose team0
```

- 获取整个实例的状态：

```
teamdctl team0 state
```

- 获取 teamd 实例的 systemd 状态：

```
systemctl status teamd@team0
```

以上各命令将根据您的需要分别显示稍有不同的视图。

7. 如果您之后需要对 `ifcfg-team0` 文件中的内容进行更改，请使用以下命令重新装载其配置：

```
wicked ifreload team0
```

请勿使用 `systemctl` 来启动或停止协作设备！而是使用如上所示的 `wicked` 命令。

要彻底去除组合设备，请执行以下过程：

过程 16.2 去除组合设备

1. 停止网络组合设备 `team0`：

```
wicked ifdown team0
```

2. 将文件 `/etc/sysconfig/network/ifcfg-team0` 重命名为 `/etc/sysconfig/network/.ifcfg-team0`。在文件名前面插入一个点，以使 wicked“看不到”它。如果您确实不再需要该配置，也可以去除该文件。
3. 重新装载配置：

```
wicked ifreload all
```

16.8.1 使用案例：使用网络协作实现负载平衡

负载平衡用于提高带宽。使用下面的配置文件可创建具有负载平衡功能的网络协作设备。继续过程 16.1 “一般过程”以设置设备。使用 `teamdctl` 检查输出。

例 16.12 通过网络协作进行负载平衡的配置

```
STARTMODE=auto ❶
BOOTPROTO=static ❷
IPADDRESS="192.168.1.1/24" ❷
IPADDR6="fd00:deca:fbad:50::1/64" ❷

TEAM_RUNNER="loadbalance" ❸
TEAM_LB_TX_HASH="ipv4,ipv6,eth,vlan"
TEAM_LB_TX_BALANCER_NAME="basic"
TEAM_LB_TX_BALANCER_INTERVAL="100"

TEAM_PORT_DEVICE_0="eth0" ❹
TEAM_PORT_DEVICE_1="eth1" ❹

TEAM_LW_NAME="ethtool" ❺
TEAM_LW_ETHTOOL_DELAY_UP="10" ❻
TEAM_LW_ETHTOOL_DELAY_DOWN="10" ❻
```

❶ 控制协作设备的启动。`auto` 这个值表示接口将在网络服务可用时设置，并且会在每次重引导时自动启动。
如果您需要自己来控制设备（并阻止其自动启动），请将 `STARTMODE` 设置为 `manual`。

❷ 设置静态 IP 地址（此处对于 IPv4 指定 `192.168.1.1`，对于 IPv6 指定 `fd00:deca:fbad:50::1`）。

如果网络协作设备应该使用动态 IP 地址，请设置 `BOOTPROTO="dhcp"` 并去除（或注释掉）带有以下内容的行： `IPADDRESS` 和 `IPADDR6`。

❸ 将 `TEAM_RUNNER` 设置为 `loadbalance`，以激活负载平衡模式。

❹ 指定应聚合以创建网络协作设备的一个或多个设备。

❺ 定义链路监视器，以监视从属设备的状态。只有当设备已启动并可访问时，默认值 `ethtool` 才会执行检查。因此，检查速度将会足够快。但是，它不会检查设备实际上是否可以发送或接收包。

如果您需要确保连接的可信度更高，请使用 `arp_ping` 选项。这样会将 ping 发送给一个任意主机（在 `TEAM_LW_ARP_PING_TARGET_HOST` 变量中进行配置）。仅当收到答复时，网络协作设备才会被视为已启动。

❻ 定义链路启动（或关闭）与运行程序收到通知之间的延迟（以毫秒为单位）。

16.8.2 使用案例：使用网络协作实现故障转移

故障转移用于确保关键网络协作设备的高可用性，方法是纳入并行的备用网络设备。备用网络设备时刻都在运行，并在主设备出现故障时接替主设备。

使用以下配置文件可创建具有故障转移功能的网络协作设备。继续过程 16.1 “一般过程”以设置设备。使用 `teamdctl` 检查输出。

例 16.13 DHCP 网络协作设备的配置

```
STARTMODE=auto ❶
BOOTPROTO=static ❷
IPADDR="192.168.1.2/24" ❷
IPADDR6="fd00:deca:fbad:50::2/64" ❷

TEAM_RUNNER=activebackup ❸
TEAM_PORT_DEVICE_0="eth0" ❹
TEAM_PORT_DEVICE_1="eth1" ❹

TEAM_LW_NAME=ethtool ❺
TEAM_LW_ETHTOOL_DELAY_UP="10" ❻
TEAM_LW_ETHTOOL_DELAY_DOWN="10" ❻
```

❶ 控制协作设备的启动。 `auto` 这个值表示接口将在网络服务可用时设置，并且会在每次重引导时自动启动。

如果您需要自己来控制设备（并阻止其自动启动），请将 `STARTMODE` 设置为 `manual`。

❷ 设置静态 IP 地址（此处对于 IPv4 指定 `192.168.1.2`，对于 IPv6 指定 `fd00:deca:fbad::50::2`）。

如果网络协作设备应该使用动态 IP 地址，请设置 `BOOTPROTO="dhcp"` 并去除（或注释掉）带有以下内容的行： `IPADDRESS` 和 `IPADDR6`。

❸ 将 `TEAM_RUNNER` 设置为 `activebackup` 以激活故障转移模式。

❹ 指定应聚合以创建网络协作设备的一个或多个设备。

❺ 定义链路监视器，以监视从属设备的状态。只有当设备已启动并可访问时，默认值 `ethtool` 才会执行检查。因此，检查速度将会足够快。但是，它不会检查设备实际上是否可以发送或接收包。

如果您需要确保连接的可信度更高，请使用 `arp_ping` 选项。这样会将 ping 发送给一个任意主机（在 `TEAM_LW_ARP_PING_TARGET_HOST` 变量中进行配置）。仅当收到答复时，网络协作设备才会被视为已启动。

❻ 定义链路启动（或关闭）与运行程序收到通知之间的延迟（以毫秒为单位）。

16.8.3 用例：组合设备上的 VLAN

VLAN 是虚拟局域网 (Virtual Local Area Network) 的缩写。它允许在单个物理以太网上运行多个逻辑（虚拟）以太网。它以逻辑方式将网络分为不同的广播域，以便数据包仅在为同一 VLAN 指定的端口之间交换。

下面的用例会在组合设备的基础上创建两个静态 VLAN：

- `vlan0`，绑定到 IP 地址 `192.168.10.1`
- `vlan1`，绑定到 IP 地址 `192.168.20.1`

按如下所示继续：

1. 在交换机上启用 VLAN 标记。如果您要针对组合设备使用负载平衡，则交换机需要支持链接聚合控制协议 (LACP) (802.3ad)。有关细节，请查阅硬件手册。

2. 确定是否要针对组合设备使用负载平衡或故障转移。按第 16.8.1 节 “使用案例：使用网络协作实现负载平衡”或第 16.8.2 节 “使用案例：使用网络协作实现故障转移”中所述设置组合设备。

3. 在 `/etc/sysconfig/network` 中，创建包含以下内容的 `ifcfg-vlan0` 文件：

```
STARTMODE="auto"
BOOTPROTO="static" ❶
IPADDR='192.168.10.1/24' ❷
ETHERDEVICE="team0" ❸
VLAN_ID="0" ❹
VLAN='yes'
```

   ❶ 定义固定的 IP 地址（在 `IPADDR` 中指定）。

   ❷ 定义 IP 地址，这里包含其网络掩码。

   ❸ 包含要用于 VLAN 接口的实际接口，这里是我们的组合设备（`team0`）。

   ❹ 为 VLAN 指定唯一的 ID。文件名和 `VLAN_ID` 最好与名称 `ifcfg-vlanVLAN_ID` 对应。在我们的示例中，`VLAN_ID` 为 `0`，因而文件名为 `ifcfg-vlan0`。

4. 将 `/etc/sysconfig/network/ifcfg-vlan0` 文件复制到 `/etc/sysconfig/network/ifcfg-vlan1`，并更改以下值：

   - `IPADDR`，从 `192.168.10.1/24` 更改为 `192.168.20.1/24`。
   - `VLAN_ID`，从 `0` 更改为 `1`。

5. 启动两个 VLAN：

```
root # wicked ifup vlan0 vlan1
```

6. 检查 `ifconfig` 的输出：

```
root # ifconfig -a
[...]
vlan0     Link encap:Ethernet  HWaddr 08:00:27:DC:43:98
          inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fedc:4398/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

```
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:816 (816.0 b)

vlan1     Link encap:Ethernet  HWaddr 08:00:27:DC:43:98
          inet addr:192.168.20.1 Bcast:192.168.20.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fedc:4398/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:816 (816.0 b)
```

16.9 采用 Open vSwitch 的软件定义网络

软件定义网络 (SDN) 指的是将控制流量发送来源的系统（控制面）与将流量转发到选定目标的底层系统（数据面，也称为转发面）分离开来。这意味着先前由单一且通常不灵活的交换机执行的功能，现在可在交换机（数据面）与它的控制器（控制面）之间分离。在此模式下，控制器可以编程且具有极高的灵活性，并且能够快速适应多变的网络条件。

Open vSwitch 是一款可实施与 OpenFlow 协议兼容的分布式虚拟多层交换机的软件。OpenFlow 允许控制器应用程序修改交换机的配置。OpenFlow 构建于 TCP 协议之上，并且广泛实施于各种硬件和软件中。如此，单个控制器便可驱动多个大不相同的交换机。

16.9.1 Open vSwitch 的优点

采用 Open vSwitch 的软件定义网络具备多项优点，尤其在与虚拟机配合使用时：

- 可轻松识别联网状态。
- 网络及其实时状态可从一个主机移到另一个主机。
- 网络动态可跟踪，并且可允许使用外部软件对其进行响应。
- 您可以在网络包中应用标记并操作这些标记，以识别网络包往来的计算机并维护其他联网环境。标记规则可以配置和迁移。

- Open vSwitch 实施的是 GRE 协议（通用路由封装）。例如，这可让您将专用 VM 网络相互连接起来。
- Open vSwitch 可单独使用，但设计它是为了与联网硬件相集成并且能够控制硬件交换机。

16.9.2 安装 Open vSwitch

1. 安装 Open vSwitch 和补充包：

```
root # zypper install openvswitch openvswitch-switch
```

   如果您计划将 Open vSwitch 与 KVM 超级管理程序配合使用，请另外安装 `tunctl`。如果您计划将 Open vSwitch 与 Xen 超级管理程序配合使用，请另外安装 `openvswitch-kmp-xen`。

2. 启用 Open vSwitch 服务：

```
root # systemctl enable openvswitch
```

3. 重启动计算机或使用 `systemctl` 立即启动 Open vSwitch 服务：

```
root # systemctl start openvswitch
```

4. 要检查 Open vSwitch 是否已正确激活，请使用：

```
root # systemctl status openvswitch
```

16.9.3 Open vSwitch 守护程序和实用程序概述

Open vSwitch 包含多个组件，其中有内核模块和各种用户空间组件。内核模块用于加速数据路径，但最精简的 Open vSwitch 安装并不需要该模块。

16.9.3.1 守护程序

Open vSwitch 的中心可执行文件是它的两个守护程序。当您启动 `openvswitch` 服务时，便会间接启动它们。

Open vSwitch 的主守护程序 (`ovs-vswitchd`) 提供交换机的实施。Open vSwitch 的数据库守护程序 (`ovsdb-server`) 为储存 Open vSwitch 配置和状态的数据库提供服务。

16.9.3.2 实用程序

Open vSwitch 还自带多个协助您使用该服务的实用程序。下面的列表并不全面，只是介绍了一些重要的命令。

`ovsdb-tool`
: 创建、升级、压缩和查询 Open vSwitch 的数据库。处理 Open vSwitch 数据库相关的事务。

`ovs-appctl`
: 配置运行中的 `ovs-vswitchd` 或 `ovsdb-server` 守护程序。

`ovs-dpctl`、`ovs-dpctl-top`
: 创建、修改、可视化及删除数据路径。使用此工具可能会干扰也负责执行数据路径管理的 `ovs-vswitchd`。因此，它通常仅作诊断之用。
`ovs-dpctl-top` 可创建类似于 `top` 的数据路径可视化。

`ovs-ofctl`
: 管理任何遵循 OpenFlow 协议的交换机。`ovs-ofctl` 并非仅可用于与 Open vSwitch 交互。

`ovs-vsctl`
: 提供配置数据库的高级别接口。它可用于查询和修改该数据库。实际上，它会显示 `ovs-vswitchd` 的状态并可对其进行配置。

16.9.4 使用 Open vSwitch 创建网桥

下面的配置示例使用 SUSE Linux Enterprise Server 上默认使用的 Wicked 网络服务。要了解 Wicked 的更多信息，请参见第 16.5 节 “手动配置网络连接”。

如果您已安装并启动 Open vSwitch，请执行如下操作：

1. 要配置供虚拟机使用的网桥，请创建包含以下内容的文件：

```
STARTMODE='auto' ❶
BOOTPROTO='dhcp' ❷
OVS_BRIDGE='yes' ❸
OVS_BRIDGE_PORT_DEVICE_1='eth0' ❹
```

❶ 网络服务启动后自动设置网桥。

❷ 用于配置 IP 地址的协议。

❸ 将配置标记为 Open vSwitch 网桥。

❹ 选择应加入网桥的一个或多个设备。要添加更多设备，请在文件中另外为每个设备追加相应的行：

```
OVS_BRIDGE_PORT_DEVICE_SUFFIX='DEVICE'
```

`SUFFIX` 可以是任何字母数字字符串。不过，为了避免覆盖先前的定义，请确保每个设备的 `SUFFIX` 都是唯一的。

将文件保存到 `/etc/sysconfig/network` 目录中并命名为 `ifcfg-br0`。您也可以不使用 `br0`，而是使用任何您喜欢的名称。但是，文件名必须以 `ifcfg-` 开头。
要了解更多选项，请参见 `ifcfg` 的手册页 (`man 5 ifcfg`) 以及 `ifcfg-ovs-bridge` 的手册页 (`man 5 ifcfg-ovs-bridge`)。

2. 现在，启动网桥：

```
root # wicked ifup br0
```

Wicked 完成后，应该会输出网桥的名称，旁边会显示状态 `up`。

16.9.5 将 Open vSwitch 直接与 KVM 配合使用

如第 16.9.4 节 “使用 Open vSwitch 创建网桥”中所述创建网桥后，您便可使用 Open vSwitch 管理通过 KVM/QEMU 创建的虚拟机的网络访问。

1. 为了能够最充分地利用 Wicked 的功能，请对之前配置的网桥进行进一步的更改。打开先前创建的 `/etc/sysconfig/network/ifcfg-br0`，为其他端口设备追加一行：

```
OVS_BRIDGE_PORT_DEVICE_2='tap0'
```

此外，请将 BOOTPROTO 设置为 none。文件现在应如下所示：

```
STARTMODE='auto'
BOOTPROTO='none'
OVS_BRIDGE='yes'
OVS_BRIDGE_PORT_DEVICE_1='eth0'
OVS_BRIDGE_PORT_DEVICE_2='tap0'
```

新的端口设备 tap0 将在下一步中配置。

2. 现在，为 tap0 设备添加配置文件：

```
STARTMODE='auto'
BOOTPROTO='none'
TUNNEL='tap'
```

将文件保存到 /etc/sysconfig/network 目录中并命名为 ifcfg-tap0。

提示：允许其他用户访问 Tap 设备

若要能够从以非 root 身份的用户启动的虚拟机使用此 Tap 设备，请追加：

```
TUNNEL_SET_OWNER=USER_NAME
```

要为整个组授予访问权，请追加：

```
TUNNEL_SET_GROUP=GROUP_NAME
```

3. 最后，打开定义为第一个 OVS_BRIDGE_PORT_DEVICE 的设备的配置。如果其名称未更改过，则应为 eth0。因此，打开 /etc/sysconfig/network/ifcfg-eth0 并确保已设置以下选项：

```
STARTMODE='auto'
BOOTPROTO='none'
```

如果文件尚不存在，请创建该文件。

4. 使用 Wicked 重启动网桥接口：

```
root # wicked ifreload br0
```

这也会触发重新装载新定义的网桥端口设备。

5. 例如，要启动虚拟机，请使用：

```
root # qemu-kvm \
-drive file=/PATH/TO/DISK-IMAGE❶ \
-m 512 -net nic,vlan=0,macaddr=00:11:22:EE:EE:EE \
-net tap,ifname=tap0,script=no,downscript=no❷
```

❶ 要启动的 QEMU 磁盘映像路径。

❷ 使用之前创建的 Tap 设备（`tap0`）。

有关 KVM/QEMU 用法的更多信息，请参见《Virtualization Guide》。

16.9.6 将 Open vSwitch 与 `libvirt` 搭配使用

如前文第 16.9.4 节 “使用 Open vSwitch 创建网桥”中所述创建网桥后，您可以将该网桥添加到通过 `libvirt` 管理的现有虚拟机。由于 `libvirt` 已对 Open vSwitch 网桥提供一定程度的支持，因此您可以使用第 16.9.4 节 “使用 Open vSwitch 创建网桥”中创建的网桥，而无需进一步更改网络配置。

1. 为所需的虚拟机打开域 XML 文件：

```
root # virsh edit VM_NAME
```

以所需虚拟机的名称替换 `VM_NAME`。这样将会打开默认的文本编辑器。

2. 查找以 `<interface type="...">` 开头并以 `</interface>` 结尾的部分，找到文档的网络部分。
以如下所示的网络部分替换现有部分：

```
<interface type='bridge'>
```

```
  <source bridge='br0'/>
  <virtualport type='openvswitch'/>
</interface>
```

重要：`virsh iface-*` 和虚拟机管理器与 Open vSwitch 的兼容性

目前，在使用 `virsh iface-*` 工具和虚拟机管理器的情况下，Open vSwitch 与 `libvirt` 还不兼容。如果使用以上任一工具，您的配置可能会损坏。

3. 您现在便可照常启动或重启动虚拟机。

有关 `libvirt` 用法的更多信息，请参见《Virtualization Guide》。

16.9.7 更多信息

http://openvswitch.org/support/

Open vSwitch 项目网站的文档部分

https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf

开放网络基金会发布的关于软件定义网络和 OpenFlow 协议的白皮书

17 打印机操作

SUSE® Linux Enterprise Server 支持使用多种类型的打印机进行打印，其中包括远程网络打印机。可以手动或使用 YaST 配置打印机。有关配置描述，请参见《部署指南》, 第 11 章 “使用 YaST 设置硬件组件”, 第 11.3 节 “设置打印机”。启动和管理打印任务时既可以使用图形实用程序，也可以使用命令行实用程序。如果打印机未能按预期正常工作，请参见第 17.8 节 “查错”。

CUPS（通用 Unix 打印系统）是 SUSE Linux Enterprise Server 中的标准打印系统。

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。购买打印机时，确保打印机配有支持的接口（USB、以太网或 Wi-Fi）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任务。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。

PDF 正在逐渐取代 PostScript，成为标准打印作业格式。可直接打印 PDF（而不仅仅是 PostScript）的 PostScript+PDF 打印机已经面世。传统的 PostScript 打印机需要在打印工作流程中将 PDF 转换为 PostScript。

标准打印机（PCL 和 ESC/P 等语言）

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。此处理阶段称为解释。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，可以生成令人满意的打印效果。Linux 可能无法使用某些特殊打印机功能。除了 HP 和 Epson 之外，当前尚没有其他打印机制造商开发 Linux 驱动程序，并通过开放源代码许可证将这些驱动程序提供给 Linux 发行套件供应商。

专有打印机（也称作 GDI 打印机）

这些打印机不支持任何常见的打印机语言。这些打印机使用自己的无文档记录打印机语言，该语言在发布新版本时可能发生变化。通常只有 Windows 驱动程序供这些打印机使用。有关更多信息，请参见第 17.8.1 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://www.linuxfoundation.org/OpenPrinting/

包含打印机数据库的 OpenPrinting 主页。数据库显示最新的 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，当前标为“完全支持”的打印机在最新的 SUSE Linux Enterprise Server 版本发布后，不一定还具有此状态。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。

http://pages.cs.wisc.edu/~ghost/

Ghostscript 网页

`/usr/share/doc/packages/ghostscript/catalog.devices`

内置 Ghostscript 驱动程序列表。

17.1 CUPS 工作流程

用户创建一个打印任务。打印作业由要打印的数据和有关假脱机程序的信息组成。其中包括打印机的名称或打印队列的名称，还有可能包括有关过滤器（例如特定于打印机的选项）的信息。

每台打印机至少有一个专用打印队列。假脱机程序储存着队列中的打印任务，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。

过滤器将转换正在打印的应用程序生成的数据（通常为 PostScript 或 PDF，也可能为 ASCII、JPEG 等）特定于打印机的数据（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，则过滤器系统将数据转换为打印机专用的数据。这样做需要一个适合您的打印机的 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。

17.2 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 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。有关打印机连接的更多信息，请阅读 http://en.opensuse.org/SDB:CUPS_in_a_Nutshell 上的文章 CUPS in a Nutshell（CUPS 概述）。

System z CUPS 不支持 z/VM 提供的可与 IBM Z 大型机进行本地连接的打印机和类似设备。在这些平台上，只能通过网络进行打印。必须根据打印机制造商的描述安装网络打印机的电缆。

警告：更改处于运行状态系统中的电缆连接

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下 USB 设备。为防止损坏系统或打印机，请在更改任何非 USB 连接前先关闭系统。

17.3 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。

要配置 PostScript 打印机，最佳的方法是获得一个合适的 PPD 文件。`manufacturer-PPDs` 和 `OpenPrintingPPDs-postscript` 包中提供了许多 PPD 文件。请参见第 17.7.3 节 “多种包中的 PPD 文件”和第 17.8.2 节 “没有合适的 PPD 文件可用于 PostScript 打印机”。

新的 PPD 文件可以储存在目录 `/usr/share/cups/model/` 中，或如《部署指南》，第 11 章 “使用 YaST 设置硬件组件”，第 11.3.1.1 节 “使用 YaST 添加驱动程序”中所述使用 YaST 添加到打印系统中。随后，可以在打印机设置过程中选择 PPD 文件。

如果打印机制造商希望您安装整个软件包，请务必小心。这种安装类型可能导致 SUSE Linux Enterprise Server 提供的支持失效。另外，打印命令可能会以不同的方式工作，并且系统可能不再能够对其他制造商的设备寻址。出于此原因，不建议安装制造商软件。

17.4 网络打印机

网络打印机可以支持多种协议，有些甚至支持并发打印不同协议。尽管大部分支持的协议都已标准化，但某些制造商可能修改了标准。他们仅提供适用于少数操作系统的驱动程序。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以找出起作用的配置。

CUPS 支持 `socket`、`LPD`、`IPP` 和 `smb` 协议。

套接字

套接字是指将纯文本打印数据直接发送到 TCP 套接字的连接。一些常用的套接字端口号包括 `9100` 或 `35`。设备 URI（统一资源标识符）的语法为 socket://`IP.OF.THE.PRINTER`:`PORT`，例如 `socket://192.168.2.202:9100/`。

LPD（行式打印机守护程序）

LPD 协议如 RFC 1179 中所述。使用此协议，打印队列 ID 等作业相关数据将先于实际打印数据发送。因此，配置 LPD 协议时必须指定打印队列。各打印机制造商的实施非常灵活，可以接受为打印队列指定任何名称。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。LPD 服务的端口号是 `515`。设备 URI 示例：`lpd://192.168.2.202/LPT1`。

IPP（因特网打印协议）

IPP 是一个基于 HTTP 协议的相对较新的协议 (1999)。使用 IPP，所传送的与任务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。要正确配置 IPP，必须提供打印队列的名称。IPP 的端口号是 `631`。设备 URI 示例：`ipp://192.168.2.202/ps` 和 `ipp://192.168.2.202/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`。设备 URI 示例：`smb://user:password@workgroup/smb.example.com/printer`、`smb://user:password@smb.example.com/printer` 和 `smb://smb.example.com/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`（随 `nmap` 包提供）来确定协议。`nmap` 检查主机端口是否打开。例如：

```
nmap -p 35,137-139,515,631,9100-10000 IP.OF.THE.PRINTER
```

17.5 使用命令行工具配置 CUPS

CUPS 可使用 `lpinfo`、`lpadmin` 和 `lpoptions` 之类的命令行工具配置。您需要一个包含一个后端（例如 USB）和多个参数的设备 URI。要确定系统上的有效设备 URI，请使用命令 `lpinfo -v | grep ":/"`：

```
# lpinfo -v | grep ":/"
direct usb://ACME/FunPrinter%20XL
network socket://192.168.2.253
```

使用 `lpadmin`，CUPS 服务器管理员可以添加、删除或管理打印队列。要添加打印队列，请使用以下语法：

```
lpadmin -p QUEUE -v DEVICE-URI -P PPD-FILE -E
```

设备（`-v`）便会用作 `QUEUE`（`-p`），并使用指定的 PPD 文件（`-P`）。这意味着如果要手动配置打印机，则必须了解 PPD 文件和设备 URI。

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 usb://ACME/FunPrinter%20XL -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://192.168.2.202:9100/ -P \
/usr/share/cups/model/Postscript-level1.ppd.gz -E
```

有关 `lpadmin` 的更多选项，请参考 `lpadmin(8)` 的手册页。

在系统安装期间，某些选项被设置为默认值。可以为每个打印任务修改这些选项（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
   ```

   激活的默认选项通过加星号前缀 (*) 进行标识。

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
   ```

普通用户运行 `lpoptions` 时，设置将写到 `~/.cups/lpoptions`。但是，根 设置将写到 `/etc/cups/lpoptions`。

17.6 从命令行打印

要从命令行打印，请输入 `lp -d QUEUENAME FILENAME`，并用相应的名称替换 `QUEUENAME` 和 `FILENAME`。

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`。

17.7 SUSE Linux Enterprise Server 中的特殊功能

某些 CUPS 功能已针对 SUSE Linux Enterprise Server 做出调整。这里将介绍一些最重要的更改。

17.7.1 CUPS 和防火墙

执行默认 SUSE Linux Enterprise Server 安装后，SuSEFirewall2 将处于活动状态，且网络接口配置为位于 外部区域 中，这会阻止传入通讯。《Security Guide》, 第 16 章 “Masquerading and Firewalls”, 第 16.4 节 “SuSEFirewall2”中和 http://en.opensuse.org/SDB:CUPS_and_SANE_Firewall_settings 上提供了有关 SuSEFirewall2 配置的更多信息。

17.7.1.1 CUPS 客户端

通常 CUPS 客户端在使用防火墙的可信网络环境中的常规工作站上运行。在这种情况下，建议将网络接口配置为在 内部区域 中，这样可以从网络内部访问工作站。

17.7.1.2 CUPS 服务器

如果 CUPS 服务器在受防火墙保护的可信网络环境中，则应将网络接口配置为在防火墙的 内部区域 中。建议不要在不可信网络环境中安装 CUPS 服务器，除非您确定该服务器受到特殊防火墙规则和 CUPS 配置中的安全设置的保护。

17.7.2 浏览网络打印机

CUPS 服务器会定期在网络上公告共享打印机的可用性及状态信息。客户端可以访问此信息，以便在打印对话框中显示可用打印机列表。此过程称为“浏览”。

CUPS 服务器通过传统的 CUPS 浏览协议或 Bonjour/DND-SD 在网络上公告它们的打印队列。为了能够浏览网络打印队列，通过 CUPS 服务器打印的所有客户端上都需要运行 `cups-browsed` 服务。默认情况下，`cups-browsed` 不会启动。要为活动会话启动它，请使用 `sudo systemctl start cups-browsed`。要确保它在系统引导后自动启动，请在所有客户端上使用 `sudo systemctl enable cups-browsed` 启用它。

如果在启动 `cups-browsed` 之后浏览不起作用，则表明 CUPS 服务器可能是通过 Bonjour/DND-SD 公告网络打印队列的。在此情况下，您需要另外安装 `avahi` 包，并在所有客户端上使用 `sudo systemctl start avahi-daemon` 启动关联的服务。

17.7.3 多种包中的 PPD 文件

YaST 打印机配置使用 `/usr/share/cups/model` 中安装的 PPD 文件为 CUPS 设置队列。为查找适用于打印机型号的 PPD 文件，YaST 将对照硬件检测过程中确定的供应商和型号比较所有 PPD 文件中的供应商和型号。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。

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。例如，如果您有 PostScript 打印机，可直接将 PPD 文件复制到 `/usr/share/cups/model`（如果这些文件尚不存在于 `manufacturer-PPDs` 或 `OpenPrintingPPDs-postscript` 包中），以实现打印机最佳配置。

其他 PPD 文件由下列包提供：

- gutenprint：Gutenprint 驱动程序及其匹配的 PPD
- splix：SpliX 驱动程序及其匹配的 PPD
- OpenPrintingPPDs-ghostscript：Ghostscript 内置驱动程序的 PPD
- OpenPrintingPPDs-hpijs：适用于非 HP 打印机的 HPIJS 驱动程序的 PPD

17.8 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。讨论的主题有 GDI 打印机、PPD 文件和端口配置。另外还讨论常见网络打印机问题、打印件问题以及队列处理。

17.8.1 打印机没有标准打印机语言支持

这些打印机不支持任何常见的打印机语言，只能使用专门的专有控制系列来进行寻址。因此这些打印机只能用于制造商提供了驱动程序的操作系统版本。GDI 是 Microsoft* 为图形设备开发的编程接口。通常制造商只提供 Windows 的驱动程序，而因为 Windows 驱动程序使用 GDI 界面，所以这些打印机也称作 GDI 打印机。问题实际并不是出在编程接口上，而是因这些打印机只能通过相应打印机型号的专用打印机语言来寻址所造成。

某些 GDI 打印机可切换成以 GDI 方式或一种标准打印机语言进行操作。请参见打印机手册以了解这是否可行。有些型号需要有专门的 Windows 软件来进行切换（注：Windows 打印机驱动程序在通过 Windows 进行打印时可能总是将打印机切换回 GDI 模式）。对于其他 GDI 打印机，还有针对标准打印机语言的扩展模块。

某些制造商为他们的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。

与其花时间尝试使专有 Linux 驱动程序运行，购买支持标准打印机语言（最好是 PostScript）的打印机可能更经济高效。这可以一次性彻底解决驱动程序问题，您不再需要安装并配置特殊驱动程序软件，以及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。

17.8.2 没有合适的 PPD 文件可用于 PostScript 打印机

如果 `manufacturer-PPDs` 或 `OpenPrintingPPDs-postscript` 包不包含适用于 PostScript 打印机的 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 文件的许可证协议条款。然后使用 `cupstestppd` 实用程序来确认 PPD 文件是否与“Adobe PostScript 打印机描述文件格式规范 V4.3” 相符合，如果实用程序返回 “FAIL，”则描述 PPD 文件中的错误很严重，可能导致重大问题。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。

17.8.3 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。如果相应的 lpd 处于活动状态并且主机接受查询，请以 root 身份运行以下命令，以查询远程 HOST 上 QUEUE 的状态报告：

```
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，响应应该描述为什么在主机的队列上不能进行打印。如果您接收到类似例 17.1 “来自 lpd 的错误消息” 中的响应，则问题是由远程 lpd 引起的。

例 17.1 来自 lpd 的错误消息

```
lpd: your host does not have line printer access
lpd: queue does not exist
printer: spooling disabled
printer: printing disabled
```

检查远程 cupsd

CUPS 网络服务器可以在 UDP 端口 631 上广播其队列，默认每 30 秒广播一次。因此，以下命令可用于测试网络中是否存在广播 CUPS 网络服务器。执行此命令之前，务必停止本地 CUPS 守护程序。

```
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 17.2 “来自 CUPS 网络服务器的广播” 所示。

例 17.2 来自 CUPS 网络服务器的广播

```
ipp://192.168.2.202:631/printers/queue
```

System z 请注意，IBM Z 以太网设备默认不接收广播。

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
```

如果不能与 cupsd 建立连接，则可能是 cupsd 未处于活动状态，或者存在基本网络问题。如果相应的 cupsd 处于活动状态并且主机接受查询，lpstat -h HOST -l -t 会返回 HOST 上所有队列的状态报告（可能非常长）。

下面的命令可用于测试 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

有时，当在打印服务器计算机中运行的假脱机程序需要处理多个打印作业时会产生问题。这是打印服务器计算机中的假脱机程序导致的，目前尚无解决此问题的方法。变通方法是，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 17.4 节 “网络打印机”。

这样，打印服务器计算机仅充当各种不同数据传送方式之间（TCP/IP 网络和本地打印机连接）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。如果打印机连接到打印服务器计算机并且已启动，则通常可以在打开打印服务器计算机电源一段时间后使用 nmap 包中的 nmap 实用程序确定此 TCP 端口。例如，nmap IP-address 可能会返回打印服务器计算机的以下输出：

```
Port       State       Service
23/tcp     open        telnet
80/tcp     open        http
515/tcp    open        printer
631/tcp    open        cups
9100/tcp   open        jetdirect
```

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，请参见 nmap 的手册页。

输入如下命令

```
echo -en "\rHello\r\f" | netcat -w 1 IP-address port
cat file | netcat -w 1 IP-address port
```

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。

17.8.4 打印件有问题但没有错误消息

对于打印系统，打印任务完成的标志是 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特定于打印机的数据），则打印系统不会对此进行通知。如果打印机无法打印特定于打印机的数据，请选择一个更适合该打印机的 PPD 文件。

17.8.5 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。后端用于确定在将数据传送报告为不可行前应执行的失败尝试次数。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 `cupsenable` 命令重新启用打印。

17.8.6 CUPS 浏览：删除打印任务

如果 CUPS 网络服务器通过浏览向客户端主机广播其队列并且客户端主机上合适的本地 `cupsd` 处于活动状态，则客户端 `cupsd` 接受来自应用程序的打印任务并将它们转发到服务器上的 `cupsd`。当服务器上的 `cupsd` 接受打印任务后，会为该任务指派一个新的任务号。因此，客户端主机上的任务号与服务器上的任务号不同。因为打印作业通常都会立即转发出去，所以不能用客户端主机上的作业号将其删除，原因是当打印作业已转发到服务器 `cupsd` 后，客户端 `cupsd` 会将打印作业视为已完成。

要删除服务器上的打印作业，请使用 `lpstat -h cups.example.com -o` 之类的命令来确定服务器上的作业编号。此情况假设服务器尚未完成该打印作业（即尚未完全将它发送到打印机）。按如下方式使用获得的作业编号来删除服务器上的打印作业：

```
cancel -h cups.example.com QUEUE-JOBNUMBER
```

17.8.7 有问题的打印任务和数据传送错误

如果在打印过程中关闭打印机或计算机，则打印任务将保留在队列中。再次打开计算机（或打印机）后，打印将继续。必须使用 `cancel` 从队列中删除有问题的打印任务。

如果打印作业损坏，或主机与打印机之间的通讯出现错误，打印机将无法正确处理数据，并会打印出很多张有乱码的纸。要修复该问题，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. 打印任务可能仍在队列中，因为只有在将任务完全发送到打印机后才会将它们删除。使用 `lpstat -o` 或 `lpstat -h cups.example.com -o` 检查哪个队列当前正在打印。使用 `cancel QUEUE-JOBNUMBER` 或 `cancel -h cups.example.com QUEUE-JOBNUMBER` 删除打印作业。
3. 即使已将打印任务从队列中删除，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。
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。

17.8.8 调试 CUPS

使用以下通用过程确定 CUPS 中的问题：

1. 在 `/etc/cups/cupsd.conf` 中设置 `LogLevel debug`。
2. 停止 `cupsd`。
3. 删除 `/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*` 中的消息以确定问题的原因。

17.8.9 更多信息

有关在 SUSE Linux 上打印的详细信息，请参见 openSUSE 支持数据库，网址为 http://en.opensuse.org/Portal:Printing。SUSE 知识库 (http://www.suse.com/support/) 中提供了对许多特定问题的解决方案。通过对 CUPS 的文本搜索找到相关文章。

18 X Window 系统

X Window 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。本章提供 X 配置的基本信息，以及在 SUSE® Linux Enterprise Server 中使用字体的背景信息。

X Window 系统一般不需要进行任何配置。X 启动期间会动态检测硬件。因此，`xorg.conf` 已被弃用。如果您仍然需要指定自定义选项来更改 X 的行为方式，还是可以通过修改 `/etc/X11/xorg.conf.d/` 下的配置文件来实现。

提示：IBM Z：配置图形用户界面

IBM Z 没有 X.Org 所支持的输入或输出设备。因此，本节中所述的所有配置过程都不适用。有关 IBM Z 的更多信息，请参见《部署指南》, 第 4 章 “在 IBM Z 上安装”。

18.1 安装和配置字体

Linux 中的字体可分为两大类：

轮廓或矢量字体

包含作为字形组成相关绘图指导的数学描述。因此，每个字形都可以缩放为任意大小而无损质量。在可以使用此类字体（或字形）之前，需要将数学描述转换为光栅（网格）。此过程称为字体光栅化。字体微调（嵌入在字体中）可改进和优化特定大小的渲染效果。光栅化和微调通过 FreeType 库实现。

Linux 下的常用格式为 PostScript Type 1 和 Type 2、TrueType 及 OpenType。

位图或光栅字体

包含一个为特定字号设计的像素阵列。位图字体渲染速度极快，而且非常简单。然而，与矢量字体相比，位图字体无法在不损质量的情况下进行缩放。因此，这些字体通常以不同的大小发布。现在，Linux 控制台中仍然使用位图字体，有时终端中也会使用这些字体。

在 Linux 下，便携式编译格式 (PCF) 或字形位图分布格式 (BDF) 是最常用的格式。

这些字体的外观主要会受两个方面的影响：

- 选择合适的字体系列，
- 采用某种算法渲染字体，达到接收者眼睛最舒服的效果。

最后一点只与矢量字体相关。虽然上面两点都需要根据个人情况而定，但仍有一些默认值需要创建。

Linux 字体渲染系统由具有不同关系的几个库组成。基本字体渲染库是 FreeType (http://www.freetype.org/)，它会将支持的格式的字体字形转换为优化的位图字形。渲染过程由算法及其参数（可能受专利问题影响）控制。

使用 FreeType 的每个程序或库都应该参考 Fontconfig (http://www.fontconfig.org/) 库。此库会从用户及系统那里收集字体配置。用户修改其 Fontconfig 设置后，此更改将导致发生 Fontconfig 感知的应用。

Arabic、Han 或 Phags-Pa 等脚本所需的更复杂的 OpenType 成型以及其他更高级别的文本处理使用 Harfbuzz (http://www.harfbuzz.org/) 或 Pango (http://www.pango.org/) 进行。

18.1.1 显示安装的字体

要获得系统上安装了哪些字体的概观，请运行 `rpm` 或 `fc-list` 命令。这两个命令都可为您提供不错的答案，但有可能会因系统和用户配置不同而返回不同的列表。

`rpm`

调用 `rpm` 可查看系统上安装了哪些包含字体的软件包：

```
rpm -qa '*fonts*'
```

每个字体包都应该满足此表达式。不过，命令可能会返回误报，例如 `fonts-config`（它即不是字体，也不包含字体）。

`fc-list`

调用 `fc-list` 可获得哪些字体系列可以访问、系统上或主目录中是否已安装这些字体的概观：

```
fc-list ':' family
```

注意：命令 `fc-list`

命令 `fc-list` 是 Fontconfig 库的封装程序。从 Fontconfig（更确切地说，从它的超速缓存）可以查询许多有趣的信息。有关更多细节，请参见 `man 1 fc-list`。

18.1.2 查看字体

如果想了解已安装字体系列的外观，请使用命令 `ftview`（`ft2demos` 包）或访问 http://fontinfo.opensuse.org/。例如，要以 14 号字显示 FreeMono 字体，请按如下所示使用 `ftview`：

```
ftview 14 /usr/share/fonts/truetype/FreeMono.ttf
```

如果需要更多信息，请访问 http://fontinfo.opensuse.org/ 了解支持哪些样式（标准、粗体、斜体等）和语言。

18.1.3 查询字体

要查询指定了某种模式时使用哪种字体，请使用 `fc-match` 命令。

例如，如果您的模式包含已安装字体，`fc-match` 会返回文件名、字体系列和样式：

```
tux > fc-match 'Liberation Serif'
LiberationSerif-Regular.ttf: "Liberation Serif" "Regular"
```

如果所需字体在您的系统上不存在，Fontconfig 的匹配规则将会生效，并尝试找到最接近的可用字体。这意味着用另一种字体来替代了您要求的字体。

```
tux > fc-match 'Foo Family'
DejaVuSans.ttf: "DejaVu Sans" "Book"
```

Fontconfig 支持别名：用另一个系列名称替代某个名称。通用名称就是一个典型的例子，例如“sans-serif”、“serif” 和 “monospace”。这些别名可由实际的系列名称，甚至是系列名称的首选设置列表替代：

```
tux > for font in serif sans mono; do fc-match "$font" ; done
```

```
DejaVuSerif.ttf: "DejaVu Serif" "Book"
DejaVuSans.ttf: "DejaVu Sans" "Book"
DejaVuSansMono.ttf: "DejaVu Sans Mono" "Book"
```

结果可能因系统而异，具体视当前安装的字体而定。

注意：根据 Fontconfig 而定的相似度规则

Fontconfig 总是根据具体请求返回最相似的实际系列（如果至少安装了一个系列）。“相似度”根据 Fontconfig 的内部度量以及用户或管理员的 Fontconfig 设置而定。

18.1.4 安装字体

安装新字体的方法主要有以下几种：

1. 将 `*.ttf` 或 `*.otf` 等字体文件手动安装到知道的一个字体目录。如果字体要用于整个系统，请使用标准目录 `/usr/share/fonts`。如果要安装在主目录中，请使用 `~/.config/fonts`。
   如果不想使用标准目录，Fontconfig 允许您选择其他目录。使用 `<dir>` 元素告知 Fontconfig 所用目录，有关细节，请参见第 18.1.5.2 节 “深入了解 Fontconfig XML”。

2. 使用 `zypper` 安装字体。大量字体已通过包提供，随附在 SUSE 分发包中或包含在 M17N:fonts (http://download.opensuse.org/repositories/M17N:/fonts/) 储存库中。请使用以下命令将储存库添加到您的列表中。例如，要为 SLE 12 添加储存库：

   ```
   sudo zypper ar
        http://download.opensuse.org/repositories/M17N:/fonts/SLE_12_SP5/
   ```

   要搜索您的 `FONT_FAMILY_NAME`，请使用以下命令：

   ```
   sudo zypper se 'FONT_FAMILY_NAME*fonts'
   ```

18.1.5 配置字体的外观

根据渲染媒体和字号的不同，结果可能并不令人满意。例如，现在的显示器分辨率一般为 100dpi，这导致像素太大，字形显得粗陋难看。

有几种算法可用来处理低分辨率，例如消除锯齿（灰度平滑）、微调（适合网格）或子像素渲染（在一个方向将分辨率增至三倍）。这些算法还可能因字体格式而异。

重要：子像素渲染的专利问题

SUSE 发行套件中不使用子像素渲染。虽然 FreeType2 支持此算法，但有几项将于 2019 年底到期的专利涉及到了此算法。因此，除非系统中有 FreeType2 库并且该库中已编译子像素渲染，否则在 Fontconfig 中设置子像素渲染选项没有任何效果。

通过 Fontconfig，可单独为每种字体选择渲染算法，也可为一组字体选择渲染算法。

18.1.5.1 通过 sysconfig 配置字体

SUSE Linux Enterprise Server 在 Fontconfig 上提供一个 `sysconfig` 层。要尝试进行字体配置，这是一个不错的着手点。要更改默认设置，请编辑配置文件 `/etc/sysconfig/fonts-config`。（或使用 YaST sysconfig 模块）。编辑该文件之后，请运行 `fonts-config`。

```
sudo /usr/sbin/fonts-config
```

重启动该应用程序以显示成效。请牢记以下要点：

- 一些应用程序不需要重启动。例如，Firefox 会不时重新读取 Fontconfig 配置。新创建或重新装载的标签可在稍后获得新的字体配置。
- 每次发生包安装或去除操作之后，系统会自动调用 `fonts-config` 脚本（如未调用，则表示字体软件包有错误）。
- 使用 `fonts-config` 命令行选项可以暂时覆盖每个 sysconfig 变量。有关细节，请参见 `fonts-config --help`。

有几个 sysconfig 变量可以更改。请参见 `man 1 fonts-config` 或 YaST sysconfig 模块的帮助页。以下是一些变量示例：

渲染算法的使用

考虑使用 FORCE_HINTSTYLE、 FORCE_AUTOHINT、 FORCE_BW、 FORCE_BW_MONOSPACE、 USE_EMBEDDED_BITMAPS 和 EMBEDDED_BITMAP_LANGAGES

通用别名的首选项列表

使用 PREFER_SANS_FAMILIES、 PREFER_SERIF_FAMILIES、 PREFER_MONO_FAMILIES 和 SEARCH_METRIC_COMPATIBLE

下面的列表提供了一些配置示例，按从“最清晰”字体（对比度较高）到“最漂亮”（较平滑）的顺序显示。

位图字体

通过 PREFER_*_FAMILIES 变量可指定首选位图字体。请按照帮助部分的示例使用这些变量。请注意，这些字体渲染为黑白色，不进行平滑处理，并且位图字体只有几种字号。考虑使用

```
SEARCH_METRIC_COMPATIBLE="no"
```

来禁用基于度量兼容性的系列名称替代。

渲染为黑白色的可缩放字体

渲染时未消除锯齿的可缩放字体与位图字体的显示效果相似，同时又可保持字体可缩放性。请使用经过精细微调的字体，如 Liberation 系列。遗憾的是，系统中经过精细微调的字体并不充足。设置下面的变量可强制采用此方法：

```
FORCE_BW="yes"
```

渲染为黑白色的等宽字体

仅采用不消除锯齿的方式渲染等宽字体，否则，请使用默认设置：

```
FORCE_BW_MONOSPACE="yes"
```

默认设置

渲染所有字体时都消除锯齿。经过精细微调的字体将通过字节码解释器 (BCI) 渲染，其余字体将使用自动微调器 (hintstyle=hintslight) 渲染。让所有相关的 sysconfig 变量保持默认设置。

CFF 字体

以 CFF 格式使用字体。FreeType2 中有了当前的改进后，该字体应该也会比默认的 TrueType 字体更清晰。请按照 `PREFER_*_FAMILIES` 的示例尝试一下。可以使用以下选项将字体调得更黑更粗：

```
SEARCH_METRIC_COMPATIBLE="no"
```

因为它们默认是通过 `hintstyle=hintslight` 来渲染的。还可以考虑使用：

```
SEARCH_METRIC_COMPATIBLE="no"
```

专用自动微调器

即使对于精细微调的字体，也可以使用 FreeType2 的自动微调器。这可能会导致字形变得更粗，有时还会变得更模糊、对比度更低。设置下面的变量可激活此功能：

```
FORCE_AUTOHINTER="yes"
```

使用 `FORCE_HINTSTYLE` 可控制微调级别。

18.1.5.2 深入了解 Fontconfig XML

Fontconfig 的配置格式是可扩展标记语言 (XML)。下面的几个示例不是完整参考，只是简要概述。细节及其他启示可在 `man 5 fonts-conf` 或 `/etc/fonts/conf.d/` 中找到。

中心 Fontconfig 配置文件是 `/etc/fonts/fonts.conf`，它及其他作品包括整个 `/etc/fonts/conf.d/` 目录。要自定义 Fontconfig，可在两个位置插入您的更改：

FONTCONFIG 配置文件

1. 系统范围的更改： 编辑文件 `/etc/fonts/local.conf`（默认情况下，它包含空的 `fontconfig` 元素）。
2. 用户特定的更改： 编辑文件 `~/.config/fontconfig/fonts.conf`。将 Fontconfig 配置文件放在 `~/.config/fontconfig/conf.d/` 目录中。

用户特定的更改会覆盖任何系统范围的设置。

注意：弃用的用户配置文件

文件 `~/.fonts.conf` 标记为已弃用，不应该再使用。请改为使用 `~/.config/fontconfig/fonts.conf`。

每个配置文件都需要有一个 `fontconfig` 元素。因此，最精简的文件应如下所示：

```
<?xml version="1.0"?>
   <!DOCTYPE fontconfig SYSTEM "fonts.dtd">
   <fontconfig>
   <!-- Insert your changes here -->
   </fontconfig>
```

如果默认目录不够用，请插入 `dir` 元素及相应的目录：

```
<dir>/usr/share/fonts2</dir>
```

Fontconfig 会以递归方式搜索字体。

使用下面的 Fontconfig 片段可以选择字体渲染算法（请参见例 18.1 “指定渲染算法”）：

例 18.1 指定渲染算法

```
<match target="font">
 <test name="family">
  <string>FAMILY_NAME</string>
 </test>
 <edit name="antialias" mode="assign">
  <bool>true</bool>
 </edit>
 <edit name="hinting" mode="assign">
  <bool>true</bool>
 </edit>
 <edit name="autohint" mode="assign">
  <bool>false</bool>
 </edit>
 <edit name="hintstyle" mode="assign">
  <const>hintfull</const>
 </edit>
```

```
</match>
```

字体的各种属性都可以测试。例如，`<test>` 元素可测试字体系列（如该例中所示）、字体间隔、间距、字体格式以及其他。如果完全不使用 `<test>`，所有 `<edit>` 元素都会应用于每个字体（全局更改）。

例 18.2 别名和系列名称替代

规则 1

```
<alias>
 <family>Alegreya SC</family>
 <default>
  <family>serif</family>
 </default>
</alias>
```

规则 2

```
<alias>
 <family>serif</family>
 <prefer>
  <family>Droid Serif</family>
 </prefer>
</alias>
```

规则 3

```
<alias>
 <family>serif</family>
 <accept>
  <family>STIXGeneral</family>
 </accept>
</alias>
```

例 18.2 “别名和系列名称替代”中的规则会创建一份排定了优先级的系列列表 (PFL)。根据元素的不同，执行的操作也不同：

规则 1 中的 `<default>`

此规则会在 PFL 末尾添加一个 `serif` 系列名称。

规则 2 中的 `<prefer>`

只要 PFL 中存在 `Alegreya SC`，此规则就会在 PFL 中的第一个 `serif` 之前添加“Droid Serif”。

规则 3 中的 `<accept>`

此规则会在 PFL 中第一个 `serif` 系列名称之后添加 “STIXGeneral” 系列名称。

如果将这些片段按规则 1 - 规则 2 - 规则 3 的顺序组合起来，当用户请求 “Alegreya SC” 时，系统便会如创建如表 18.1 “基于 Fontconfig 规则生成 PFL”中所述的 PFL。

表 18.1 基于 FONTCONFIG 规则生成 PFL

顺序	当前的 PFL
请求	`Alegreya SC`
规则 1	`Alegreya SC`, `serif`
规则 2	`Alegreya SC`, `Droid Serif`, `serif`
规则 3	`Alegreya SC`, `Droid Serif`, `serif`, `STIXGeneral`

在 Fontconfig 的度量中，系列名称具有最高优先级，高于样式、大小等其他模式。Fontconfig 会检查系统上当前安装了哪个系列。如果安装了 “Alegreya SC”，Fontconfig 将会返回它。如果未安装，则会查找 “Droid Serif”，依次类推。

请小心。如果 Fontconfig 片段的顺序发生变化，Fontconfig 可能会返回不同的结果，如表 18.2 “基于更改了顺序的 Fontconfig 规则生成 PFL 的结果”中所述。

表 18.2 基于更改了顺序的 FONTCONFIG 规则生成 PFL 的结果

顺序	当前的 PFL	记事
请求	`Alegreya SC`	执行相同的请求。
规则 2	`Alegreya SC`	PFL 中没有 `serif`，未替代任何内容
规则 3	`Alegreya SC`	PFL 中没有 `serif`，未替代任何内容

顺序	当前的 PFL	记事
规则 1	`Alegreya SC`, `serif`	PFL 中存在 `Alegreya SC`，执行替代

注意：隐含意义

将 `<default>` 别名视为此组的分类或内含项（如果未安装）。如该例所示，`<default>` 应该一律优先于该组的 `<prefer>` 和 `<accept>` 别名。

`<default>` 分类不限于通用别名 serif、sans-serif 和 monospace。有关复杂示例，请参见 `/usr/share/fontconfig/conf.avail/30-metric-aliases.conf`。

例 18.3 “别名和系列名称替代”中的以下 Fontconfig 片段会创建一个 `serif` 组。如果前一种字体未安装，此组中的每个系列可替代其他系列。

例 18.3 别名和系列名称替代

```
<alias>
 <family>Alegreya SC</family>
 <default>
  <family>serif</family>
 </default>
</alias>
<alias>
 <family>Droid Serif</family>
 <default>
  <family>serif</family>
 </default>
</alias>
<alias>
 <family>STIXGeneral</family>
 <default>
  <family>serif</family>
 </default>
</alias>
<alias>
 <family>serif</family>
```

```
 <accept>
  <family>Droid Serif</family>
  <family>STIXGeneral</family>
  <family>Alegreya SC</family>
 </accept>
</alias>
```

优先级由 `<accept>` 别名中的顺序指定。类似地，可以使用更强的 `<prefer>` 别名。

例 18.4 “别名和系列名称替代”扩展了例 18.2 “别名和系列名称替代”。

例 18.4 别名和系列名称替代

规则 4

```
<alias>
 <family>serif</family>
 <accept>
  <family>Liberation Serif</family>
 </accept>
</alias>
```

规则 5

```
<alias>
 <family>serif</family>
 <prefer>
  <family>DejaVu Serif</family>
 </prefer>
</alias>
```

例 18.4 “别名和系列名称替代”中的扩展配置将导致下列 PFL 变化：

表 18.3 基于 FONTCONFIG 规则生成 PFL 的结果

顺序	当前的 PFL
请求	`Alegreya SC`
规则 1	`Alegreya SC`, `serif`
规则 2	`Alegreya SC`, `Droid Serif`, `serif`

顺序	当前的 PFL
规则 3	`Alegreya SC`, `Droid Serif`, `serif`, `STIXGeneral`
规则 4	`Alegreya SC`, `Droid Serif`, `serif`, `Liberation Serif`, `STIXGeneral`
规则 5	`Alegreya SC`, `Droid Serif`, `DejaVu Serif`, `serif`, `Liberation Serif`, `STIXGeneral`

注意：含义.

- 如果同一个通用名称存在多个 `<accept>` 声明，则最后分析的声明“胜出”。如有可能，创建系统范围的配置时，不要在用户 (`/etc/fonts/conf.d/*-user.conf`) 之后使用 `<accept>`。
- 如果同一个通用名称存在多个 `<prefer>` 声明，则最后分析的声明“胜出”。如有可能，在系统范围的配置中，不要在用户之前使用 `<prefer>`。
- 同一个通用名称的每个 `<prefer>` 声明都会覆盖 `<accept>` 声明。如果管理员不仅希望用户可使用 `<prefer>`，还想允许其使用 `<accept>`，就不应该在系统范围的配置中使用 `<prefer>`。另一方面，因为用户通常都是使用 `<prefer>`，这种做法应该不会产生任何不利影响。我们还发现存在在统范围的配置中使用 `<prefer>` 的情况。

18.2 更多信息

您可安装 `xorg-docs` 包以更深入地了解 X11。`man 5 xorg.conf` 提供了有关手动配置（如果需要）的格式的详细信息。有关 X11 开发的更多信息，请参见该项目的主页：http://www.x.org。

驱动程序位于 `xf86-video-*` 包中，例如 `xf86-video-nv`。相关手册页中详细说明了这些包附带的很多驱动程序。例如，如果使用 `nv` 驱动程序，在 `man 4 nv` 中可以找到有关此驱动程序的更多信息。

有关第三方驱动程序的信息位于 /usr/share/doc/packages/<package_name> 中。例如，x11-video-nvidiaG03 的文档在安装包之后位于 /usr/share/doc/packages/x11-video-nvidiaG03 中。

19 使用 FUSE 访问文件系统

FUSE 是用户空间中的文件系统 (file system in user space) 的缩写。这表示您可以将文件系统作为非特权用户配置和装入。通常，只有 `root` 用户才能执行此任务。FUSE 自身是一个内核模块。它与插件组合，允许您扩展 FUSE 以访问几乎所有文件系统，如远程 SSH 连接、ISO 映像等。

19.1 配置 FUSE

需要安装 `fuse` 包才能使用 FUSE。根据要使用的文件系统，您需要作为独立包提供的附加插件。

一般而言，您无需配置 FUSE。但是建议创建一个合并所有安装点的目录。例如，可以创建目录 `~/mounts` 并在该处插入不同文件系统的子目录。

19.2 装入 NTFS 分区

NTFS (新技术文件系统) 是 Windows 的默认文件系统。在一般情况下，由于非特权用户无法使用外部 FUSE 库装入 NTFS 块设备，因此下文所述的 Windows 分区装入过程需要 root 特权。

1. 切换为 `root` 用户并安装包 `ntfs-3g`。SUSE Linux Enterprise Workstation Extension 中提供了该包。
2. 创建一个要充当安装点的目录，如 `~/mounts/windows`。
3. 确定所需的 Windows 分区。使用 YaST 并启动分区程序模块查看哪些分区属于 Windows，但不要修改任何内容。或者转换为 `root` 用户并执行 `/sbin/fdisk -l`。查找分区类型为 `HPFS/NTFS` 的分区。
4. 以读写模式装入分区。使用相应的 Windows 分区替换占位符 `DEVICE`：

   ```
   ntfs-3g /dev/DEVICE MOUNT POINT
   ```

   要在只读模式下使用 Windows 分区，请追加 `-o ro`：

```
ntfs-3g /dev/DEVICE MOUNT POINT -o ro
```

`ntfs-3g` 命令使用当前用户 (UID) 和组 (GID) 装入给定设备。如果要为其他用户设置写权限，请使用命令 `id USER` 获取 UID 和 GID 值的输出。设置方式：

```
id tux
uid=1000(tux) gid=100(users) groups=100(users),16(dialout),33(video)
ntfs-3g /dev/DEVICE MOUNT POINT -o uid=1000,gid=100
```

在手册页中查找其他选项。

要卸载资源，请运行 `fusermount -u 安装点`。

19.3 更多信息

有关更多信息，请参见 FUSE 主页 http://fuse.sourceforge.net。

20 管理内核模块

虽然 Linux 属于单内核，但可通过内核模块加以扩展。这些特殊对象可以插入到内核中，并可按需去除。就实际角度而言，内核模块使添加和去除内核本身未包含的驱动程序和接口成为现实。Linux 提供了若干用于管理内核模块的命令。

20.1 使用 lsmod 和 modinfo 列出装载的模块

使用 `lsmod` 命令可查看目前装载了哪些内核模块。该命令的输出可能如下所示：

```
tux > lsmod
Module                  Size  Used by
snd_usb_audio         188416  2
snd_usbmidi_lib        36864  1 snd_usb_audio
hid_plantronics        16384  0
snd_rawmidi            36864  1 snd_usbmidi_lib
snd_seq_device         16384  1 snd_rawmidi
fuse                  106496  3
nfsv3                  45056  1
nfs_acl                16384  1 nfsv3
```

输出内容分为三列：`Module` 列列出所装载模块的名称，`Size` 列显示各模块的大小。`Used by` 列显示引用模块的进程数及其名称。请注意，此列表可能不完整。

要查看有关特定内核模块的详细信息，请使用 `modinfo MODULE_NAME` 命令。其中 `MODULE_NAME` 为所需内核模块的名称。请注意，`modinfo` 二进制文件位于用户的 PATH 环境变量中未包含的 `/sbin` 目录下。这意味着，当您以普通用户身份运行 `modinfo` 命令时，必须指定该二进制文件的完整路径：

```
$ /sbin/modinfo kvm
filename:       /lib/modules/4.12.14-94.37-default/kernel/arch/x86/kvm/kvm.ko
license:        GPL
author:         Qumranet
srcversion:     BDFD8098BEEA517CB75959B
depends:        irqbypass
intree:         Y
```

```
vermagic:       4.4.57-18.3-default SMP mod_unload modversions
signer:         openSUSE Secure Boot Signkey
sig_key:        03:32:FA:9C:BF:0D:88:BF:21:92:4B:0D:E8:2A:09:A5:4D:5D:EF:C8
sig_hashalgo:   sha256
parm:           ignore_msrs:bool
parm:           min_timer_period_us:uint
parm:           kvmclock_periodic_sync:bool
parm:           tsc_tolerance_ppm:uint
parm:           lapic_timer_advance_ns:uint
parm:           halt_poll_ns:uint
parm:           halt_poll_ns_grow:int
parm:           halt_poll_ns_shrink:int
```

20.2 添加和去除内核模块

虽然可以使用 `insmod` 和 `rmmod` 分别添加和去除内核模块，但建议使用 `modprobe` 工具来执行这些操作。`modprobe` 具有多项重要优势，包括自动解析依赖项和将内核模块列入黑名单。

如果不指定任何参数，使用 `modprobe` 命令会安装指定的内核模块。必须使用 root 特权来运行 `modprobe`：

```
tux > sudo modprobe acpi
```

要去除内核模块，请使用 `-r` 参数：

```
sudo modprobe -r acpi
```

20.2.1 引导时自动装载内核模块

您可以选择不手动装载内核模块，而是使用 `system-modules-load.service` 服务在引导过程中自动装载这些模块。要启用内核模块，请将 `.conf` 文件添加到 `/etc/modules-load.d/` 目录下。建议为配置文件指定与模块相同的名称，例如：

```
/etc/modules-load.d/rt2800usb.conf
```

配置文件中必须包含所需内核模块的名称（例如 `rt2800usb`）。

通过上述的这个技巧，无需指定任何参数即可装载内核模块。如果您需要使用特定选项装载内核模块，请将配置文件添加到 `/etc/modprobe.d/` 目录下。该文件的扩展名必须为 `.conf`。文件名必须符合以下命名约定：`priority-modulename.conf`，例如：`50-thinkfan.conf`。配置文件中必须包含内核模块名称及所需参数。您可以使用以下示例命令来创建包含内核模块名称及其参数的配置文件：

```
echo "options thinkpad_acpi fan_control=1" | sudo tee /etc/modprobe.d/
thinkfan.conf
```

注意：装载内核模块

当检测到设备或用户空间请求特定功能时，系统会自动装载大多数内核模块。因此，很少需要手动将模块添加到 `/etc/modules-load.d/`。

20.2.2 使用 modprobe 将内核模块列入黑名单

将某个内核模块列入黑名单后，引导期间便不再会装载该模块。当要禁用您怀疑可能导致系统出现问题的某个模块时，此功能十分有用。请注意，您仍可通过使用 `insmod` 或 `modprobe` 工具来手动装载列入黑名单的内核模块。

要将模块列入黑名单，请在 `/etc/modprobe.d/50-blacklist.conf` 文件中添加 `blacklist MODULE_NAME` 行。例如：

```
blacklist nouveau
```

以 root 身份运行 `mkinitrd` 命令生成新的 `initrd` 映像，然后重引导计算机。可使用以下命令执行上述步骤：

```
su
echo "blacklist nouveau" >> /etc/modprobe.d/50-blacklist.conf && mkinitrd &&
 reboot
```

如果只想临时禁用内核模块，可在引导期间即时将其列入黑名单。要实现此目的，请在引导屏幕显示时按 E 键。这样，您会进入一个可供您修改引导参数的小编辑器。找到如下所示的行：

```
linux /boot/vmlinuz...splash= silent quiet showopts
```

在该行末尾添加 `modprobe.blacklist=MODULE_NAME` 命令。例如：

```
linux /boot/vmlinuz...splash= silent quiet showopts modprobe.blacklist=nouveau
```

按 F10 或 Ctrl–X 以按照指定配置引导。

要通过 GRUB 将某个内核模块永久列入黑名单，请打开要编辑的 `/etc/default/grub` 文件，在 `GRUB_CMD_LINUX` 命令中添加 `modprobe.blacklist=MODULE_NAME` 选项。然后运行 `sudo grub2-mkconfig -o /boot/grub2/grub.cfg` 命令使更改生效。

21 使用 udev 进行动态内核设备管理

内核几乎可以添加或删除运行系统中的任何设备。设备状态的更改（无论插入还是移除设备）需要传播给用户空间。插入及识别设备后需要对其进行配置。某个设备已识别状态的任何更改都需要通知给此设备的用户。 udev 可提供所需的基础结构来动态维护 /dev 目录中的设备节点文件和符号链接。 udev 规则提供了将外部工具插入内核设备事件处理的方式。因而，您可以通过添加在内核设备处理过程中执行的特定脚本，来自定义 udev 设备处理方式，或者可以在设备处理期间请求并导入其他数据进行评估。

21.1 /dev 目录

/dev 目录中的设备节点提供对相应的内核设备的访问。使用 udev 时， /dev 目录反映内核的当前状态。每个内核设备都有相应的设备文件。如果设备从系统断开，则删除此设备节点。

/dev 目录的内容保存在临时文件系统中，所有文件都是在每个系统启动时提供的。手动创建或修改的文件在重引导时是有意不保存的。无论可使用 systemd-tmpfiles 创建的相应内核设备状态如何，静态文件和目录都始终应位于 /dev 目录中。配置文件位于 /usr/lib/tmpfiles.d/ 和 /etc/tmpfiles.d/ 中。有关详细信息，请参见 systemd-tmpfiles(8) 手册页。

21.2 内核 uevents 和 udev

必需的设备信息由 sysfs 文件系统导出。对于内核检测到并已初始化的设备，将创建一个带有该设备名称的目录。它包含带有特定于设备属性的属性文件。

每次添加或删除设备时，内核都会发送 uevent 来向 udev 通知更改。一旦启动， udev 守护程序便会读取并分析 /usr/lib/udev/rules.d/*.rules 和 /etc/udev/rules.d/*.rules 文件中的所有规则，并将它们保留在内存中。如果更改、添加或去除了规则文件，守护程序可以使用 udevadm control --reload 命令重新装载这些规则在内存中的表示。有关 udev 规则及其语法的更多细节，请参见第 21.6 节 “使用 udev 规则影响内核设备事件处理”。

每个接收到的事件都根据所提供的规则集进行匹配。这些规则可以增加或更改事件环境键、为要创建的设备节点请求特定名称、添加指向该节点的符号链接或者添加设备节点创建后运行的程序。从内核 netlink 套接字接收驱动程序核心 `uevent`。

21.3 驱动程序、内核模块和设备

设备的内核总线驱动程序探测。对于每个检测到的设备，内核都会在驱动程序核心将 uevent 发送到 `udev` 守护程序时创建内部设备结构。总线设备通过特殊格式的 ID 来标识自己，这可以识别设备的类型。通常，这些 ID 由供应商和产品 ID 以及其他特定于子系统的值组成。每个总线都有自己对于这些 ID 的方案，称为 `MODALIAS`。内核获取设备信息，由此组成一个 `MODALIAS` ID 字符串，并将该字符串与事件一起发送。对于 USB 鼠标，如下所示：

```
MODALIAS=usb:v046DpC03Ed2000dc00dsc00dp00ic03isc01ip02
```

每个设备驱动程序都带有它可以处理的设备的已知别名列表。这个列表包含在内核模块文件中。程序 depmod 读取 ID 列表并在内核的 `/lib/modules` 目录中为所有当前可用的模块创建文件 `modules.alias`。使用这种基础结构，模块的装载就如为每个带有 `MODALIAS` 键的事件调用 `modprobe` 一样简单。如果调用 `modprobe $MODALIAS`，它将组成该设备的设备别名与模块提供的别名相匹配。如果找到匹配的项，则装载该模块。所有这些操作均由 `udev` 自动触发。

21.4 引导和启动设备设置

在 `udev` 守护程序运行之前的引导进程中发生的所有设备事件都会丢失，因为处理这些事件的基础结构保存在 root 文件系统中，并且此时不可用。为了弥补此损失，内核提供了一个 `uevent` 文件，该文件位于 sysfs 文件系统每个设备的设备目录中。通过将 `add` 写入到该文件，内核将再次发送引导时丢失的相同事件。`/sys` 触发器中所有 `uevent` 文件的简单循环将再次触发所有事件来创建设备节点并执行设备设置。

例如，在引导期间出现的 USB 鼠标可能不会由早期引导逻辑初始化，因为驱动程序在那时不可用。此设备发现的事件丢失并且不能为该设备查找内核模块。您无需手动搜索连接的设备，`udev` 会在根文件系统可用后向内核请求所有设备事件，这样 USB 鼠标设备的事件就会再次运行。现在它在装入的 root 文件系统上找到内核模块，因此可以初始化 USB 鼠标。

在用户空间中，设备冷插入序列和运行时期间发现的设备之间没有明显的区别。在这两种情况下，使用相同的规则来匹配并且运行相同的配置程序。

21.5 监视正在运行的 udev 守护程序

程序 `udevadm monitor` 可以用于将驱动程序核心事件和 `udev` 事件处理的计时可视化。

```
UEVENT[1185238505.276660] add   /devices/pci0000:00/0000:00:1d.2/usb3/3-1 (usb)
UDEV  [1185238505.279198] add   /devices/pci0000:00/0000:00:1d.2/usb3/3-1 (usb)
UEVENT[1185238505.279527] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0 (usb)
UDEV  [1185238505.285573] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0 (usb)
UEVENT[1185238505.298878] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0/input/input10 (input)
UDEV  [1185238505.305026] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0/input/input10 (input)
UEVENT[1185238505.305442] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0/input/input10/mouse2 (input)
UEVENT[1185238505.306440] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0/input/input10/event4 (input)
UDEV  [1185238505.325384] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0/input/input10/event4 (input)
UDEV  [1185238505.342257] add   /devices/pci0000:00/0000:00:1d.2/
usb3/3-1/3-1:1.0/input/input10/mouse2 (input)
```

`UEVENT` 行显示内核已经通过 netlink 发送的事件。`UDEV` 行显示已经完成的 `udev` 事件处理程序。计时以微秒为单位显示。`UEVENT` 和 `UDEV` 之间的时间是 `udev` 用于处理此事件或者 `udev` 守护程序延迟执行从而同步此事件与相关以及已运行的事件的时间。例如，硬盘分区的事件总是等待主磁盘设备事件完成，因为分区事件可能依赖于主磁盘事件从硬件查询的数据。

`udevadm monitor --env` 显示完整的事件环境：

```
ACTION=add
DEVPATH=/devices/pci0000:00/0000:00:1d.2/usb3/3-1/3-1:1.0/input/input10
SUBSYSTEM=input
SEQNUM=1181
```

```
NAME="Logitech USB-PS/2 Optical Mouse"
PHYS="usb-0000:00:1d.2-1/input0"
UNIQ=""
EV=7
KEY=70000 0 0 0 0
REL=103
MODALIAS=input:b0003v046DpC03Ee0110-e0,1,2,k110,111,112,r0,1,8,amlsfw
```

udev 也将消息发送给 syslog。用于控制将哪些消息发送到系统日志的默认系统日志优先级在 udev 配置文件 /etc/udev/udev.conf 中指定。可以使用 udevadm control --log_priority= LEVEL/NUMBER 更改正在运行的守护程序的日志优先级。

21.6 使用 udev 规则影响内核设备事件处理

udev 规则可以与内核添加到事件本身的属性或者内核导出到 sysfs 的任何信息匹配。规则还可以从外部程序请求其他信息。系统会将事件与目录 /usr/lib/udev/rules.d/（适用于默认规则）和 /etc/udev/rules.d（系统特定的配置）中提供的所有规则进行匹配。

规则文件中的每一行至少包含一个键值对。有两种类型的键，匹配键和指派键。如果所有匹配键与它们的值匹配，则应用此规则并将指派键指派给特定的值。匹配规则可以指定设备节点的名称、添加指向该节点的符号链接或者运行作为事件处理一部分的特定程序。如果找不到匹配的规则，则使用默认设备节点名来创建设备节点。udev 手册页中描述了有关规则语法和提供用来与数据匹配或导入数据的键的详细信息。以下示例规则提供了 udev 规则语法的基本介绍。这些示例规则全部取自 udev 默认规则集 /usr/lib/udev/rules.d/50-udev-default.rules。

例 21.1 示例 udev 规则

```
# console
KERNEL=="console", MODE="0600", OPTIONS="last_rule"

# serial devices
KERNEL=="ttyUSB*", ATTRS{product}=="[Pp]alm*Handheld*", SYMLINK+="pilot"

# printer
SUBSYSTEM=="usb", KERNEL=="lp*", NAME="usb/%k", SYMLINK+="usb%k", GROUP="lp"
```

```
# kernel firmware loader
SUBSYSTEM=="firmware", ACTION=="add", RUN+="firmware.sh"
```

console 规则由三个键构成：一个匹配键 (KERNEL) 和两个赋值键 (MODE、OPTIONS)。KERNEL 匹配规则搜索设备列表以查找类型为 console 的所有项。只有完全匹配才有效，才能触发执行此规则。在这种情况下，MODE 键为设备节点指派特殊权限，仅为此设备的拥有者指派读写权限。OPTIONS 键将该规则标记为此类型的所有设备最后采用的规则。匹配此特殊设备类型的任何后续规则都不产生任何影响。

50-udev-default.rules 中不再提供 serial devices 规则，但该规则仍然值得考虑。该规则由两个匹配键 (KERNEL 和 ATTRS) 和一个赋值键 (SYMLINK) 构成。KERNEL 键搜索类型为 ttyUSB 的所有设备。该键使用 * 通配符匹配这些设备中的几个。第二个匹配键 ATTRS 检查任何 ttyUSB 设备的 sysfs 中的 product 属性文件是否包含特定字符串。赋值键 (SYMLINK) 将符号链接添加至该设备的 /dev/pilot 下。此键中使用的运算符 (+=) 告知 udev 进一步执行此操作，即使前面或后面的规则添加其他符号链接。 由于此规则包含两个匹配键，因此仅当两个条件都满足时，才应用。

printer 规则处理 USB 打印机，其中包含两个匹配键 (SUBSYSTEM 和 KERNEL)，并且必须同时应用这两个键，才能应用整个规则。三个赋值键处理该设备类型的命名 (NAME)、符号设备链接 (SYMLINK) 的创建，以及此设备类型的组成员资格 (GROUP)。在 KERNEL 键中使用通配符 * 将使其匹配若干 lp 打印机设备。NAME 和 SYMLINK 键中都使用了替换项，以便按内部设备名称扩展这些字符串。例如，指向第一个 lp USB 打印机的符号链接为 /dev/usblp0。

kernel firmware loader 规则用于使 udev 在运行时期间通过外部助手脚本装载其他固件。 SUBSYSTEM 匹配键搜索 firmware 子系统。ACTION 键检查是否添加了属于 firmware 子系统的任何设备。RUN+= 键触发执行 firmware.sh 脚本，以便找到应装载的固件。

所有规则具有一些共同的特征：

- 每个规则由一个或多个以逗号分隔的键值对构成。
- 键的运算由运算符确定。udev 规则支持多个运算符。
- 每个给定值必须用引号引起来。
- 规则文件的每一行代表一个规则。如果某个规则超过一行，请使用 \ 合并不同行，就像在外壳语法中一样。

- udev 规则支持与 *、? 和 [] 模式匹配的外壳式模式。
- udev 规则支持替换。

21.6.1 在 udev 规则中使用运算符

创建可以从多个运算符选择的键，具体取决于要创建的键的类型。匹配键通常用于查找匹配或明显不匹配搜索值的值。匹配键包含以下运算符之一：

==
比较等于性。如果键包含搜索模式，则匹配该模式的所有结果均有效。

!=
比较不等于性。如果键包含搜索模式，则匹配该模式的所有结果均有效。

赋值键可以使用下面的任何运算符：

=
为键指派值。如果键以前由一列值构成，键将重置，并且仅指派一个值。

+=
为包含一列项的键添加一个值。

:=
指派最终值。不允许后面的规则进行任何后续更改。

21.6.2 在 udev 规则中使用替换项

udev 规则支持使用占位符和替换项。请按照在其他任何脚本中的相同方式使用。在 udev 规则中可使用以下替换项：

%r、$root
设备目录 /dev（默认）。

%p、$devpath
DEVPATH 的值。

`%k`、`$kernel`
: `KERNEL` 的值或内部设备名称。

`%n`、`$number`
: 设备号。

`%N`、`$tempnode`
: 设备文件的临时名称。

`%M`、`$major`
: 设备的主编号。

`%m`、`$minor`
: 设备的次编号。

`%s{ATTRIBUTE}`, `$attr{ATTRIBUTE}`
: `sysfs` 属性的值（通过 `ATTRIBUTE` 指定）。

`%E{VARIABLE}`, `$env{VARIABLE}`
: 环境变量的值（通过 `VARIABLE` 指定）。

`%c`、`$result`
: `PROGRAM` 的输出。

`%%`
: `%` 字符。

`$$`
: `$` 字符。

21.6.3 使用 udev 匹配键

匹配键描述应用 `udev` 规则之前必须满足的条件。以下匹配键可用：

`ACTION`
: 事件操作的名称，如 `add` 或 `remove`（添加或删除设备时）。

`DEVPATH`
: 事件设备的设备路径，如 `DEVPATH=/bus/pci/drivers/ipw3945`，用于搜索与 ipw3945 驱动程序有关的所有事件。

KERNEL

事件设备的内部（内核）名称。

SUBSYSTEM

事件设备的子系统，如 SUBSYSTEM=usb（用于与 USB 设备有关的所有事件）。

ATTR{FILENAME}

事件设备的 sysfs 属性。例如，要匹配 vendor 属性文件名中包含的字符串，可以使用 ATTR{vendor}=="On[sS]tream"。

KERNELS

让 udev 向上搜索设备路径以查找匹配的设备名称。

SUBSYSTEMS

让 udev 向上搜索设备路径以查找匹配的设备子系统名称。

DRIVERS

让 udev 向上搜索设备路径以查找匹配的设备驱动程序名称。

ATTRS{FILENAME}

让 udev 向上搜索设备路径以查找具有匹配的 sysfs 属性值的设备。

ENV{KEY}

环境变量的值，如 ENV{ID_BUS}="ieee1394，用于搜索与该 FireWire 总线 ID 有关的所有事件。

PROGRAM

让 udev 执行外部程序。程序必须返回退出码零，才能成功。RESULT 键可使用程序的输出（打印到 stdout）。

RESULT

匹配上次 PROGRAM 调用的输出字符串。在与 PROGRAM 键相同的规则中包含该键，或在后面的一个中。

21.6.4 使用 udev 指派键

与上述匹配键相比，赋值键未描述必须满足的条件。它们将值、名称和操作指派给由 udev 维护的设备节点。

NAME

将创建的设备节点的名称。在一个规则设置节点名称之后，将对该节点忽略带有 NAME 键的其他所有规则。

SYMLINK

与要创建的节点有关的符号链接名称。多个匹配的规则可添加要使用设备节点创建的符号链接。也可以通过使用空格字符分隔符号链接名称，在一个规则中为一个节点指定多个符号链接。

OWNER, GROUP, MODE

新设备节点的权限。此处指定的值重写已编译的任何值。

ATTR{KEY}

指定要写入事件设备的 sysfs 属性的值。如果使用 == 运算符，也将使用该键匹配 sysfs 属性的值。

ENV{KEY}

告知 udev 将变量导出到环境。如果使用 == 运算符，也将使用该键匹配环境变量。

RUN

告知 udev 向程序列表添加要为该设备执行的程序。请记住，将此程序限制于很短的任务，以免妨碍此设备的后续事件。

LABEL

添加 GOTO 可跳至的标签。

GOTO

告知 udev 跳过若干规则，继续执行 GOTO 键所引用标签对应的规则。

IMPORT{TYPE}

将变量装载入外部程序输出之类的事件环境中。 udev 可导入多种类型的变量。如果未指定任何类型， udev 将尝试根据文件许可权限的可执行位来自行确定类型。

- program 告知 udev 执行外部程序并导入其输出。
- file 告知 udev 导入文本文件。
- parent 告知 udev 从父设备导入储存的键。

WAIT_FOR_SYSFS

告知 udev 等待要为某个设备创建的指定 sysfs 文件。例如，WAIT_FOR_SYSFS="ioerr_cnt" 通知 udev 等待 ioerr_cnt 文件创建完成。

OPTIONS

OPTION 键可以有多个值：

- last_rule 告知 udev 忽略后面的所有规则。
- ignore_device 告知 udev 完全忽略此事件。
- ignore_remove 告知 udev 忽略后面针对设备的所有删除事件。
- all_partitions 告知 udev 为块设备上的所有可用分区创建设备节点。

21.7 永久设备命名

动态设备目录和 udev 规则基础架构可以为所有磁盘设备提供固定名称，而不考虑它们的识别顺序或设备使用的连接。内核创建的每个相应的块设备由工具根据有关特定总线、驱动器类型或者文件系统的特殊知识进行检查。除了动态内核提供的设备节点名，udev 还维护各种指向该设备的永久符号链接：

```
/dev/disk
|-- by-id
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B -> ../../sda
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part1 -> ../../sda1
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part6 -> ../../sda6
|   |-- scsi-SATA_HTS726060M9AT00_MRH453M4HWHG7B-part7 -> ../../sda7
|   |-- usb-Generic_STORAGE_DEVICE_02773 -> ../../sdd
|   `-- usb-Generic_STORAGE_DEVICE_02773-part1 -> ../../sdd1
|-- by-label
|   |-- Photos -> ../../sdd1
|   |-- SUSE10 -> ../../sda7
|   `-- devel -> ../../sda6
|-- by-path
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0 -> ../../sda
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part1 -> ../../sda1
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part6 -> ../../sda6
```

```
|   |-- pci-0000:00:1f.2-scsi-0:0:0:0-part7 -> ../../sda7
|   |-- pci-0000:00:1f.2-scsi-1:0:0:0 -> ../../sr0
|   |-- usb-02773:0:0:2 -> ../../sdd
|   |-- usb-02773:0:0:2-part1 -> ../../sdd1
`-- by-uuid
    |-- 159a47a4-e6e6-40be-a757-a629991479ae -> ../../sda7
    |-- 3e999973-00c9-4917-9442-b7633bd95b9e -> ../../sda6
    `-- 4210-8F8C -> ../../sdd1
```

21.8 udev 使用的文件

`/sys/*`

Linux 内核提供的虚拟文件系统，用于导出所有当前已知设备。此信息由 `udev` 用于在 `/dev` 中创建设备节点

`/dev/*`

动态创建的设备节点和使用 systemd-tmpfiles 创建的静态内容。有关详细信息，请参见 `systemd-tmpfiles(8)` 手册页。

以下文件和目录包含 `udev` 基础结构的关键元素：

`/etc/udev/udev.conf`

主 `udev` 配置文件。

`/etc/udev/rules.d/*`

系统特定的 `udev` 事件匹配规则。可在此处添加自定义规则，以修改或覆盖 `/usr/lib/udev/rules.d/*` 中的默认规则。

文件将以字母数字顺序进行分析。文件中优先级较高的规则将会修改或覆盖优先级较低的规则。数字越小，优先级越高。

`/usr/lib/udev/rules.d/*`

默认的 `udev` 事件匹配规则。此目录中的文件由包拥有，将在更新时重写。请勿在此处添加、去除或编辑文件，而是应使用 `/etc/udev/rules.d`。

`/usr/lib/udev/*`

从 `udev` 规则调用的帮助程序。

`/usr/lib/tmpfiles.d/` 和 `/etc/tmpfiles.d/`

负责静态 `/dev` 内容。

21.9 更多信息

有关 `udev` 基础结构的更多信息，请参见以下手册页：

`udev`

有关 `udev`、键、规则和其他重要配置问题的常规信息。

`udevadm`

`udevadm` 可用于控制 `udev` 的运行时行为、请求内核事件、管理事件队列，以及提供简单的调试机制。

`udevd`

有关 `udev` 事件管理守护程序的信息。

22 使用 kGraft 在线增补 Linux 内核

本文档介绍 kGraft 在线增补技术的基本原理，并提供 SLE Live Patching 服务的使用准则。

kGraft 是一项在线增补技术，使用它可在运行时增补 Linux 内核，而无需停止内核。如此可以最大程度地确保系统运行时间，从而提高系统可用性，这对于任务关键型系统而言非常重要。该技术还允许动态增补内核，支持用户安装关键的安全性更新，不必将它们推迟到安排的停机时间。

kGraft 增补程序是一个内核模块，用于替换内核中的全部函数。kGraft 主要提供内核中基础结构，可在运行时将增补的代码与基本内核代码集成。

SLE Live Patching 是在常规 SUSE Linux Enterprise Server 维护基础之上提供的服务。通过 SLE Live Patching 分发的 kGraft 是对常规 SLES 维护更新的有益补充。可以使用常用的更新堆栈和过程来部署 SLE Live Patching。

本文档中提供的信息与 AMD64/Intel 64 和 POWER 体系结构相关。如果您使用的不是这些体系结构，则相关的过程可能有所不同。

22.1 kGraft 的优势

当需要对紧急情况（已知发生了应该尽力修复的严重漏洞，或者已知的修复程序出现严重的系统稳定性问题）迅速做出响应时，使用 kGraft 进行在线内核增补特别有用。该技术不适合用于非时间关键型的已安排更新。

kGraft 的典型用例包括：配有巨量 RAM，且引导时间经常长达 15 分钟或以上的内存数据库之类的系统、需要持续数周或数月不重启动的大规模仿真，或者向众多消费者持续提供服务的基础架构构建模块。

kGraft 的主要优势是它永不要求停止内核，哪怕是短暂停止。

kGraft 增补程序是 RPM 包中的一个 `.ko` 内核模块。可以在安装或更新 包时，使用 `insmod` 命令将它插入内核。kGraft 将替换内核中的全部函数，即使这些函数正在执行。如果需要，可以使用更新的 kGraft 模块替换现有增补程序。

kGraft 也很精简 - 因为利用了其他标准 Linux 技术，它只包含少量的代码。

22.2 kGraft 的底层函数

kGraft 使用 ftrace 基础结构执行增补。下面介绍了在 AMD64/Intel 64 体系结构上的实施过程。

为了增补某个内核函数，kGraft 要求该函数的开头有一定的空间，以便插入指向新函数的跳跃点。此空间是在开启函数分析的情况下，在内核编译期间由 GCC 分配的。具体而言，将在内核函数的开头注入一个 5 字节调用指令。引导此类经过检测的内核时，分析调用将替换为 5 字节 NOP（无操作）指令。

增补开始之后，第一个字节将替换为 INT3（断点）指令。这可以确保 5 字节指令替换的原子性。其他四个字节将替换为新函数的地址。最后，第一个字节将替换为 JMP（长跳跃）操作代码。

在整个过程中，将使用处理器间不可屏蔽中断 (IPI NMI) 来刷新系统中其他 CPU 的推理解码队列。这样，无需停止内核（哪怕是非常短暂的停止），就能切换到新的函数。IPI NMI 产生的中断可用毫秒为单位测量，并且不被视为服务中断，因为无论在哪种情况下，这些中断都是在内核运行时发生的。

永远不会增补调用方。相反，调用者的 NOP 将替换为指向新函数的 JMP。JMP 指令会永久保留。这种工作方式可以处理好函数指针（包括结构中的指针），并且不需要保存任何旧数据就能取消增补。

但是，这些步骤本身并不足够完善：因为函数将以非原子方式替换，内核某个部分中新修复的函数可能仍会调用其他位置的某个旧函数，反之亦然。如果函数接口的语义在增补程序中发生更改，将会造成混乱。

因此，在替换所有函数之前，kGraft 使用基于弹簧床、类似于 RCU（读取-复制-更新）的方案，来确保每个用户空间线程、内核线程和内核中断在全局视图中都保持一致。将对每个内核入口和出口设置一个基于线程的标志。这样，一个旧函数始终会调用另一个旧函数，而一个新函数始终会调用另一个新函数。为所有进程设置“new universe”标志后，增补即告完成，此时，可以去除弹簧床函数，代码可以全速运行，且不会对性能产生影响，不过，每个增补的函数需要经历超长时间的跳转。

22.3 安装 kGraft 增补程序

本节介绍如何激活 SUSE Linux Enterprise Live Patching 扩展以及如何安装 kGraft 增补程序。

22.3.1 激活 SLE Live Patching

要在您的系统上激活 SLE Live Patching，请遵循以下步骤：

1. 如果您的 SLES 系统尚未注册，现在请注册。可以在安装系统期间完成注册，或者以后使用 YaST 产品注册模块 (`yast2 registration`) 执行注册。注册后，单击是查看可用联机更新的列表。
   如果您的 SLES 系统已注册，但 SLE Live Patching 尚未激活，请打开 YaST 产品注册模块 (`yast2 registration`)，然后单击选择扩展。
2. 在可用扩展列表中选择 SUSE Linux Enterprise Live Patching 12，然后单击下一步。
3. 确认许可条款并单击下一步。
4. 输入 SLE Live Patching 注册代码并单击下一步。
5. 检查安装摘要和选定的模式。应该选择安装 `Live Patching` 模式。
6. 单击接受完成安装。这样就会在您的系统上安装 kGraft 基本组件以及初始在线增补程序。

22.3.2 更新系统

1. SLE Live Patching 更新通过允许使用标准 SLE 更新堆栈来应用增补程序的形式分发。可以使用 `zypper patch`、YaST 联机更新或等效的方法来更新初始在线增补程序。
2. 内核将在安装包的过程中自动增补。但是，在所有休眠进程都唤醒并避开前，旧内核函数的调用并不会完全消除。这可以节省大量的时间。尽管如此，使用旧内核函数的休眠进程并不被视为安全问题。不过，在最新的 kGraft 版本中，只有当所有进程都超出了内核用户空间界限时，才可以应用另一个 kGraft 增补程序来停止使用前一增补程序已增补的功能。
   要查看全局增补状态，请检查 `/sys/kernel/kgraft/in_progress` 中的标志。值 `1` 表示存在仍需唤醒的休眠进程（增补仍在进行）。值 `0` 表示所有进程都只使用了增补的函数，并且增补已经完成。或者，可以使用 `kgr status` 命令获取相同的信息。
   也可以基于每个进程检查标志。针对每个进程单独检查 `/proc/PROCESS_NUMBER/kgr_in_progress` 中的数字。同样，值 `1` 表示仍需唤醒的休眠进程。或者，可以使用 `kgr blocking` 命令输出休眠进程的列表。

22.4 增补程序生命周期

可以使用 `zypper lifecycle` 来查看在线增补程序的失效日期。确保包 `lifecycle-data-sle-live-patching` 已安装。

```
tux > zypper lifecycle

Product end of support
Codestream: SUSE Linux Enterprise Server 12             2024-10-31
SUSE Linux Enterprise Server 12 SP2                     n/a*

Extension end of support
SUSE Linux Enterprise Live Patching                     2017-10-31

Package end of support if different from product:
SUSEConnect                                Now, installed 0.2.41-18.1, update
 available 0.2.42-19.3.1
apache2-utils                              Now


*) See https://www.suse.com/lifecycle  for latest information
```

当到了增补程序的失效日期时，将不再提供此内核版本的更多在线增补程序。请计划于在线增补程序生命周期结束前更新内核。

22.5 去除 kGraft 增补程序

要去除 kGraft 增补程序，请执行以下过程：

1. 首先，使用 Zypper 去除增补程序本身：

   ```
   zypper rm kgraft-patch-3_12_32-25-default
   ```

2. 然后重引导计算机。

22.6 阻塞的内核执行线程

需要准备好内核线程才能处理 kGraft。第三方软件不一定能够配合 kGraft，并且其内核模块可能会衍生大量的内核执行线程。这些线程会无限期阻止增补过程。作为应急措施，kGraft 允许强行完成增补过程，而不等待所有执行线程跨越安全检查点。这可以通过在 `/sys/kernel/kgraft/in_progress` 中写入 `0` 来实现。在执行此过程之前，请先咨询 SUSE 支持人员。

22.7 `kgr` 工具

使用 `kgr` 工具可以简化一些 kGraft 管理任务。可用的命令为：

`kgr status`
: 显示 kGraft 增补的总体状态（`ready` 或 `in_progress`）。

`kgr patches`
: 显示已装载 kGraft 增补程序的列表。

`kgr blocking`
: 列出阻止 kGraft 完成增补的进程。默认情况下，只会列出 PID。指定 `-v` 会列显命令行（如果有）。再指定一个 `-v` 还会显示堆栈跟踪。

有关详细信息，请参见 `man kgr`。

22.8 kGraft 技术的应用范围

kGraft 的工作原理以替换函数为基础。数据结构的改动只能通过 kGraft 间接完成。因此，更改内核数据结构时需要特别小心，如果更改幅度太大，可能需要重引导。此外，kGraft 可能无法处理使用一个编译器来编译旧内核，使用另一个编译器来编译增补程序的情况。

由于 kGraft 的工作方式，对衍生大量内核线程的第三方模块的支持有限。

22.9 SLE Live Patching 的应用范围

SLE Live Patching 的应用范围包括 SUSE 通用漏洞评分系统 (CVSS) 级别 7 以上漏洞的修复，以及与系统稳定性或数据损坏相关的 Bug 修复。可能无法针对满足上述所有准则的所有修复类型生成在线增补程序。如果出于技术原因而无法生成内核在线增补程序，SUSE 有权不发布修复。有关作为 SUSE CVSS 评级基础的 CVSS 3.0 的详细信息，请参见 https://www.first.org/cvss/。

22.10 使用支持流程与我们交互

在与 SUSE 支持人员共同解决技术难题时，您可能会收到一个所谓的程序临时修复 (PTF)。我们可能会针对各种包（包括构成 SLE Live Patching 基础的包）发布 PTF。

您可以像平时一样安装符合上一节中所述条件的 kGraft PTF，SUSE 将确保无需重引导有问题的系统，并且将来的在线更新可以正常应用。

针对基础内核发布的 PTF 会中断在线增补过程。首先，安装 PTF 内核意味着需要重引导，因为在运行时无法替换整个内核。其次，需要再次重引导，以便将 PTF 替换为对其发布了在线增补程序的任何常规维护更新。

可将 SLE Live Patching 中其他包的 PTF 视为享有正常担保的常规 PTF。

23 特别的系统功能组件

本章首先提供有关各种软件包、虚拟控制台和键盘布局的信息。讨论诸如 bash、cron 和 logrotate 等软件组件，因为在最后的发行周期中已对这些组件进行了更改或增强。即使这些组件很小或者被认为不太重要，用户也应该更改它们的默认行为，因为这些组件往往与系统密切相关。本章的最后是有关语言和国家/地区特定设置（I18N 和 L10N）的内容。

23.1 特殊软件包的相关信息

程序 bash、cron、logrotate、locate、ulimit 和 free 对于系统管理员和许多用户是非常重要的。手册页和信息页是命令相关信息的两个有用来源，但是它们并不是始终可用的。GNU Emacs 是一种流行的并且非常容易配置的文本编辑器。

23.1.1 bash 包和 /etc/profile

Bash 是默认的系统外壳。在用作登录外壳时，它将读取几个初始化文件。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

在 ~/.profile 或 ~/.bashrc 中进行自定义设置。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。执行以下外壳命令可防止个人调整的损失：

```
mv ~/.bashrc ~/.bashrc.old
cp /etc/skel/.bashrc ~/.bashrc
mv ~/.profile ~/.profile.old
```

```
cp /etc/skel/.profile ~/.profile
```

然后从 `*.old` 文件将个人调整复制过来。

23.1.2 cron 包

使用 cron 可在预定义的时间自动在后台运行命令。cron 使用格式经过特别设置的时间表，并且该工具随附了几个默认的时间表。用户也可以根据需要指定自定义的表。

cron 表位于 `/var/spool/cron/tabs` 中。`/etc/crontab` 用作系统范围的 cron 表。输入在时间表之后且在此命令之前运行此命令的用户名。在例 23.1 “/etc/crontab 中的项” 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 `cron` 手册页 (`man cron`)。

例 23.1 /ETC/CRONTAB 中的项

```
1-59/5 * * * *   root   test -x /usr/sbin/atrun && /usr/sbin/atrun
```

不能通过调用命令 `crontab -e` 来编辑 `/etc/crontab`。必须直接将此文件装载到编辑器中，然后对其进行修改并保存。

许多包将外壳脚本安装到目录 `/etc/cron.hourly`、`/etc/cron.daily`、`/etc/cron.weekly` 和 `/etc/cron.monthly`，它们的执行是由 `/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。

要运行 `hourly`、`daily` 或在自定义时间运行其他周期性维护脚本，请删除通常使用 `/etc/crontab` 项的时戳文件 (请参见例 23.2 “/etc/crontab：删除时戳文件”，它删除了每个整点之前的 `hourly` 和每天凌晨 2:14 的 `daily` 等) 。

例 23.2 /ETC/CRONTAB：删除时戳文件

```
59 *  * * *     root  rm -f /var/spool/cron/lastrun/cron.hourly
14 2  * * *     root  rm -f /var/spool/cron/lastrun/cron.daily
29 2  * * 6     root  rm -f /var/spool/cron/lastrun/cron.weekly
44 2  1 * *     root  rm -f /var/spool/cron/lastrun/cron.monthly
```

或者，在 `/etc/sysconfig/cron` 中将 `DAILY_TIME` 设置为应启动 `cron.daily` 的时间。`MAX_NOT_RUN` 的设置确保即使用户很长一段时间都未在指定的 `DAILY_TIME` 打开计算机，日常任务仍被触发运行。`MAX_NOT_RUN` 的最大值为 14 天。

为了清楚起见，将日常系统维护任务分布在多个脚本中。这些脚本包含在包 aaa_base 中。例如，/etc/cron.daily 包含组件 suse.de-backup-rpmdb、suse.de-clean-tmp 或 suse.de-cron-local。

23.1.3 停止 Cron 状态讯息

要避免 cron 状态讯息导致的邮件泛滥，请将新安装产品的 /etc/sysconfig/cron 中 SEND_MAIL_ON_NO_ERROR 的默认值设为“no”。即使将此设置设为“no”，cron 数据输出仍然会发送到 MAILTO 地址，如 cron 手册页中所述。

在更新时，建议根据需要设置这些值。

23.1.4 日志文件：包 logrotate

多个系统服务（守护程序）以及内核本身会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，识别错误或故障功能，并精确诊断它们。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增长。有关详细信息，请参见《System Analysis and Tuning Guide》，第 3 章 “Analyzing and Managing System Log Files”，第 3.3 节 “Managing Log Files with logrotate”。

23.1.5 locate 命令

locate 是一个用于查找文件的命令，它不包括在已安装软件的标准范围内。如果需要，请安装 mlocate 包，它是 findutils-locate 包的后继者。updatedb 进程将在每天晚上或引导系统约 15 分钟后自动启动。

23.1.6 ulimit 命令

使用 ulimit（用户限制）命令可以为系统资源的使用设置限制并使其显示出来。ulimit 对于限制应用程序的可用内存尤其有用。设置可用内存限制后，可以防止应用程序占用过多系统资源，而导致操作系统变慢甚至挂起。

可以对 ulimit 使用多个选项。要限制使用内存，请使用表 23.1 “ulimit：为用户设置资源”中列出的选项。

表 23.1 ulimit：为用户设置资源

-m	最大驻留集大小
-v	外壳可用虚拟内存的最大量
-s	堆栈的最大大小
-c	创建的核心文件的最大大小
-a	所有当前限制均已报告

系统范围的默认项在 /etc/profile 中设置。建议不要直接编辑此文件，因为系统升级期间会覆盖所做的更改。要自定义系统范围的配置文件设置，请使用 /etc/profile.local。各用户的设置应该在 ~USER/.bashrc 中配置。

例 23.3 ULIMIT：~/.BASHRC 中的设置

```
# Limits maximum resident set size (physical memory):
ulimit -m 98304

# Limits of virtual memory:
ulimit -v 98304
```

必须以 KB 为单位指定内存分配。有关详细信息，请参见 man bash。

重要：ulimit 支持

并非所有外壳都支持 ulimit 指令。PAM（例如 pam_limits）作为替代 ulimit 的方法，提供了全面的调整功能。

23.1.7 free 命令

`free` 命令显示系统中总的可用内存、已用物理内存和交换空间，以及内核占用的缓冲区和超速缓存。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。

内核对任何应用程序或用户数据都没有直接的了解。而是在一个页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，这样，使用 `mmap` 命令便可一开始就从这些交换分区或文件中读取这些部分（请参见 `man mmap`）。

此外，内核中还包含其他缓存，如 slab 缓存，其中储存着用于网络访问的缓存。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）上述缓存。

但是如果您的目的是找出当前所用的 RAM 量，则在 `/proc/meminfo` 中查找此信息。

23.1.8 手册页和信息页

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` 选项快速查看信息页，其中提供更多深入的描述。info 是 GNU 的超文本系统。通过输入 `info info` 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 或帮助系统来查看信息页。

23.1.9 使用 `man` 命令选择手册页

要阅读手册页，请输入 `man MAN_PAGE`。如果不同章节存在同名手册页，所有手册页都会带相应部分编号列出。选择要显示的一个手册页。如果在数秒内未输入部分编号，将显示第一个手册页。

要将此行为更改为默认系统行为，请在外壳初始化文件（如 `~/.bashrc`）中设置 `MAN_POSIXLY_CORRECT=1`。

23.1.10 GNU Emacs 的设置

GNU Emacs 是一个复杂的工作环境。下面几节介绍当启动 GNU Emacs 时处理的配置文件。有关详细信息，请参见 http://www.gnu.org/software/emacs/。

启动时，Emacs 会读取包含用户、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（通过 `cp /etc/skel/.gnu-emacs ~/.gnu-emacs`）将 `.gnu-emacs` 复制到用户主目录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 Emacs 中的 `customize` 选项进行设置，则这些设置将保存到 `~/.gnu-emacs-custom` 中。

通过 SUSE Linux Enterprise Server，`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。除其他作用之外，`site-start.el` 确保自动装载通过 Emacs 扩充包分发的特殊配置文件（例如 `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更多信息：`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）：支持 X11 的程序。
- `emacs-nox`：不支持 X11 的程序。
- `emacs-info`：info 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，可安装众多附加产品包：`emacs-auctex` (LaTeX)、`psgml`（SGML 和 XML）、`gnuserv`（客户端和服务器操作）等。

23.2 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 7 个控制台是为 X 保留的，而第 10 个控制台显示内核消息。

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。要返回到 X，请按 Alt-F7。

23.3 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
/etc/X11/Xmodmap
/etc/skel/.emacs
/etc/skel/.gnu-emacs
/etc/skel/.vimrc
/etc/csh.cshrc
/etc/termcap
/usr/share/terminfo/x/xterm
/usr/share/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。不是系统附带的应用程序应该根据这些默认设置进行调整。

在 X 下，可以如 `/etc/X11/Xmodmap` 中所说明的启用 Compose 键（多键）。

可以通过“X 键盘扩展”(XKB) 进行进一步的设置。桌面环境 GNOME (gswitchit) 也使用此扩展。

提示：更多信息

有关 XKB 的信息可以在 `/usr/share/doc/packages/xkeyboard-config`（`xkeyboard-config` 包的一部分）所列的文档中找到。

23.4 语言和国家/地区特定的设置

该系统在很大程度上实现了国际化，可修改以满足本地需要。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，还指消息（语言）、字符集、排序顺序、时间和日期、数字和货币等类别。这些类别中的每一种都可以使用自己的变量直接定义，或使用 `language` 文件中的主变量间接定义（请参见 `locale` 手册页）。

`RC_LC_MESSAGES`、`RC_LC_CTYPE`、`RC_LC_COLLATE`、`RC_LC_TIME`、`RC_LC_NUMERIC`、`RC_LC_MONETARY`

这些变量以不带 `RC_` 前缀的形式传递到外壳，它们代表所列出的类别。下面列出了相关外壳配置文件。可以使用命令 `locale` 显示当前设置。

`RC_LC_ALL`

此变量（如果设置）将覆盖上述变量的值。

`RC_LANG`

如果未设置上述的任何变量，则这是后备变量。默认情况下，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。

这些变量可通过 YaST sysconfig 编辑器进行设置。此类变量的值包含语言代码、国家/地区代码、编码和修饰符。各部分之间通过特殊字符连接：

```
LANG=<language>[[_<COUNTRY>].<Encoding>[@<Modifier>]]
```

23.4.1 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 ISO 639 标准（可从 http://www.evertype.com/standards/iso639/iso639-en.html 和 http://www.loc.gov/standards/iso639-2/ 上获取）。国家/地区代码在 ISO 3166（参见 http://en.wikipedia.org/wiki/ISO_3166）中列出。

只有设置可以在 `/usr/lib/locale` 中找到其可用描述文件的值才有意义。可以使用命令 `localedef` 基于 `/usr/share/i18n` 中的文件创建更多描述文件；描述文件是 `glibc-i18ndata` 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）的描述文件：

```
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美国英语，则这是默认设置。如果选择了其他语言，则将支持该语言，但仍使用 UTF-8 作为字符编码。

`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。此设置现已过时，因为 UTF-8 也可涵盖欧元符号。仅当应用程序支持 ISO-8859-15 而不是 UTF-8 时，它才有用。

对 `/etc/sysconfig/language` 所做的更改将通过以下过程链来激活：

- 对于 Bash：`/etc/profile` 读取 `/etc/profile.d/lang.sh`，后者则分析 `/etc/sysconfig/language`。
- 对于 tcsh：`/etc/csh.login` 在登录时读取 `/etc/profile.d/lang.csh`，后者则分析 `/etc/sysconfig/language`。

这确保了对 /etc/sysconfig/language 所做的任何更改在下次登录到相应外壳时均可用，而无需手动将其激活。

用户可以通过相应地编译他们的 ~/.bashrc 覆盖系统默认值。例如，如果不想对程序讯息使用系统范围的 en_US，请加入 LC_MESSAGES=es_ES，这样讯息将以西班牙语显示。

23.4.2 ~/.i18n 中的语言环境设置

如果您对系统默认的区域设置不满意，请根据 Bash 脚本编写语法更改 ~/.i18n 中的设置。~/.i18n 中的项覆盖来自 /etc/sysconfig/language 中的系统默认值。使用相同的变量名称，但不使用 RC_ 名称空间前缀。例如，使用 LANG 而非 RC_LANG：

```
LANG=cs_CZ.UTF-8
LC_COLLATE=C
```

23.4.3 语言支持的设置

消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。如果将 LANG 设置为 en_US 并且 /usr/share/locale/en_US/LC_MESSAGES 中的消息文件不存在，则它将使用 /usr/share/locale/en/LC_MESSAGES。

还可以定义后备语言，例如，将布列塔尼语作为法语的后备语言，将加利西亚语作为葡萄牙语的后备语言。

LANGUAGE="br_FR:fr_FR"

LANGUAGE="gl_ES:es_ES:pt_PT"

如果需要，可改用挪威语变体 Nynorsk 和 Bokmal（将其他后备语言设置为 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。

可能会出现一个问题，那就是无法正确识别用于分隔成组数位的分隔符。如果 `LANG` 设置为仅两个字母的语言代码（如 `de`），但使用的定义文件 glibc 位于 `/usr/share/lib/de_DE/LC_NUMERIC`，则将出现此问题。因此必须将 `LC_NUMERIC` 设置为 `de_DE` 以使系统能够识别出分隔符定义。

23.4.4 更多信息

- 《GNU C 库参考手册》中的“区域设置和国际化”一章”。它包含在 `glibc-info` 中。该包可从 SUSE Linux Enterprise SDK 中获取。SDK 是适用于 SUSE Linux Enterprise 的模块，可以通过 SUSE Customer Center 的在线通道获得，或者，请转到 http://download.suse.com/，搜索 `SUSE Linux Enterprise Software Development Kit` 并从该处下载。有关细节，请参见《部署指南》, 第 14 章 “安装模块、扩展和第三方附加产品”
- 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 撰写的 Unicode-Howto，网址为 http://tldp.org/HOWTO/Unicode-HOWTO-1.html。

24 持久内存

本章包含有关使用配备非易失性主内存的 SUSE Linux Enterprise Server 的附加信息。此类内存称为持久内存，由一个或多个 NVDIMM 构成。

24.1 简介

持久内存是一种新型的计算机储存，其速度接近普通的动态 RAM (DRAM)，同时兼具 RAM 的按字节寻址能力以及固态硬盘 (SSD) 的性能。

与传统的 RAM 一样，持久内存直接安装在主板上的内存插槽中。因此，它的物理外形规格与 RAM 相同，以 DIMM 的形式提供。这些内存称为 NVDIMM：非易失性双列直插式内存模块。

不过，与 RAM 不同，持久内存在多个方面类似于基于闪存的 SSD。后两者采用固态内存电路的形式，但除此之外，两者都提供非易失性储存：系统断电或者重启动后，内存中的内容会得到保留。使用这两种媒体时，写入数据的速度比读取数据要慢；两者都支持有限的重新写入周期数。最后，与 SSD 一样，如果在特定的应用方案中更适合对持久内存进行扇区级别的访问，则也可以这样做。

不同的型号使用不同形式的电子储存媒体，例如 Intel 3D XPoint，或者将 NAND 闪存与 DRAM 结合使用。另外，行业正在开发新形式的非易失性 RAM。这意味着，不同的供应商和 NVDIMM 型号会提供不同的性能和持久性特征。

由于涉及的储存技术处于早期开发阶段，不同供应商的硬件可能会施加不同的限制。因此，以下叙述适用于一般性的场合。

持久内存的速度最多比 DRAM 要慢 10 倍，但比闪存要快大约 1000 倍。可在其中按字节重新写入数据，而不像在闪存中一样，需要擦除整个扇区，然后重新写入数据。最后，尽管重新写入周期数有限，但大部分形式的持久内存可以应对数百万次重新写入，相比之下，闪存只能应对数千个周期。

这会产生两种重要后果：

- 使用最新的技术无法运行仅包含持久内存的系统，因此无法实现完全非易失性的主内存。必须混合使用传统的 RAM 和 NVDIMM。操作系统和应用程序将在传统的 RAM 中执行，而 NVDIMM 可提供极速的补充性储存。
- 由于不同供应商的持久内存的性能特征不同，程序员可能需要考虑到特定服务器中 NVDIMM 的硬件规格，包括 NVDIMM 的数量，以及它们可以装入到哪些内存插槽。显然，这会对超级管理程序的使用、不同主机之间的软件迁移等造成影响。

ACPI 标准版本 6 中定义了此新型储存子系统。但是，`libnvdimm` 支持该标准颁布之前的 NVDIMM，可以相同的方式使用这些内存。

24.2 术语

区域

区域是持久内存中可划分为一个或多个名称空间的块。如果不事先将某个区域分配到名称空间，则您无法访问该区域的持久内存。

名称空间

非易失性储存的单个连续寻址范围，相当于 NVM Express SSD 的名称空间，或 SCSI 逻辑单元 (LUN)。名称空间作为单独的块设备显示在服务器的 `/dev` 目录中。根据所需的访问方法，名称空间可将多个 NVDIMM 中的储存合并成较大的卷，或者允许将这些储存分区成较小的卷。

模式

每个名称空间都有相应的模式，用于定义要为该名称空间启用的 NVDIMM 功能。同一父区域的同级名称空间始终具有相同的类型，但可将其配置为使用不同的模式。名称空间模式包括：

raw

一个内存磁盘。不支持 DAX。与其他操作系统兼容。

扇区

适用于不执行元数据校验和计算的传统文件系统。适用于小型引导卷。与其他操作系统兼容。

fsdax

文件系统 DAX 模式。如果未指定其他模式，则使用默认值。创建块设备 (`/dev/pmemX [.Y]`)，支持将 DAX 用于 `ext4` 或 `XFS`。

devdax

设备 DAX 模式。创建单字符设备文件 (`/dev/daxX.Y`)。不需要创建文件系统。

类型

每个名称空间和区域都有一种类型，该类型定义如何访问与该名称空间或区域关联的持久内存。名称空间的类型始终与其父区域的类型相同。类型分为以下两种：持久内存和块模式。

持久内存 (PMEM)

与 RAM 一样，PMEM 储存提供字节级别的访问。这会启用直接访问 (DAX)，表示访问内存时会绕过内核的页面超速缓存，直接进入到媒体中。此外，使用 PMEM 时，单个名称空间可以包含多个交错式 NVDIMM，因此这些 NVDIMM 都可作为单个设备供系统访问。

块模式 (BLK)

BLK 访问发生在扇区（通常为 512 字节）中，通过定义的访问窗口 aperture 进行。此行为更像是传统的磁盘驱动器。也就是说，读取和写入都是由内核进行超速缓存的。使用 BLK 访问模式时，每个 NVDIMM 可作为一个单独的名称空间供系统访问。

有些设备支持 PMEM 和 BLK 两种模式。此外，有些设备允许将储存区分割成多个单独的名称空间，以便能够使用 PMEM 访问其中一些名称空间，使用 BLK 访问另一些名称空间。

除了 `devdax` 名称空间以外，必须将所有其他类型格式化为 `ext2`、`ext4` 或 `XFS` 等文件系统，就像对传统的驱动器进行格式化一样。

Direct Access (DAX)

DAX 允许将持久内存直接映射成进程的地址空间（例如，使用 `mmap` 系统调用进行映射）。此功能适合用于在在未使用任何额外 RAM 的情况下直接访问大量 PMEM、注册多个适用于 RDMA 的 PMEM 块，或者直接将其指派给虚拟机。

DIMM 物理地址 (DPA)

在单个 DIMM 的内存中充当偏移量的内存地址；即，在该 DIMM 中充当最低可寻址字节的从零开始的地址。

标签

储存在 NVDIMM 中的元数据，例如名称空间定义。可以使用 DSM 访问这些数据。

设备特定的方法 (DSM)

用于访问 NVDIMM 中的固件的 ACPI 方法。

24.3 使用案例

24.3.1 将 PMEM 与 DAX 搭配使用

必须注意，这种形式的内存访问不是事务性的。如果发生断电或其他系统故障，数据可能不会完全写入到储存中。仅当应用程序可以处理部分写入数据的情况时，PMEM 储存才适用。

24.3.1.1 可受益于较大字节可寻址储存量的应用程序

如果服务器托管的某个应用程序可按字节直接使用较大的快速储存量，则程序员可以使用 `mmap` 系统调用，将持久内存块直接放入该应用程序的地址空间，而无需使用任何附加的系统 RAM。

24.3.1.2 避免使用内核页面超速缓存

您可以节省用于页面超速缓存的 RAM，而不要将 RAM 分配给应用程序。例如，可以专门使用非易失性内存来保存虚拟机 (VM) 映像。由于这些映像不会超速缓存，因此可以减少主机上的超速缓存使用量，从而可在每台主机上配置更多的 VM。

24.3.2 将 PMEM 与 BTT 搭配使用

当您想要使用一组 NVDIMM 中的持久内存作为类似于磁盘的极速储存池时，此方法非常有用。对于应用程序而言，此类设备仅显示为极速 SSD，并可像其他任何储存设备一样供您使用。例如，LVM 可以分布在非易失性储存的顶层，像往常一样正常工作。

BTT 的优点在于可以保证扇区写入的原子性，因此，即使是依赖于数据完整性的复杂应用程序也能保持正常工作。可通过标准的错误报告通道来运行媒体错误报告。

24.3.3 BLK 储存

尽管它拥有更强大的功能，可以防范单设备故障，但需要进行额外的管理，因为每个 NVDIMM 都显示为一个单独的设备。因此，通常首选带有 BTT 的 PMEM。

注意

在更高版本的 SUSE Linux Enterprise Server 中将弃用并不再支持 BLK 储存。

24.4 用于管理持久内存的工具

要管理持久内存，必须安装 `ndctl` 包。安装此包也会安装 `libndctl` 包，后者提供一组用户空间库用于配置 NVDIMM。

这些工具通过 `libnvdimm` 库运行。该库支持三种类型的 NVDIMM：

- PMEM
- BLK
- 同步 PMEM 和 BLK。

`ndctl` 实用程序提供一系列有用的手册页。可使用以下命令访问这些页面：

```
ndctl help subcommand
```

要查看可用子命令的列表，请使用：

```
ndctl --list-cmds
```

可用的子命令包括：

version

显示 NVDIMM 支持工具的当前版本。

enable-namespace
: 使指定的名称空间可供使用。

disable-namespace
: 阻止使用指定的名称空间。

create-namespace
: 从指定的储存设备创建新的名称空间。

destroy-namespace
: 去除指定的名称空间。

enable-region
: 使指定的区域可供使用。

disable-region
: 阻止使用指定的区域。

zero-labels
: 擦除设备中的元数据。

read-labels
: 检索指定设备的元数据。

list
: 显示可用的设备。

help
: 显示有关工具用法的信息。

24.5 设置持久内存

24.5.1 查看可用的 NVDIMM 储存

可以使用 `ndctl list` 命令列出系统中所有可用的 NVDIMM。

在以下示例中，系统包含三个 NVDIMM，这些 NVDIMM 位于单个三通道交错集内。

```
root # ndctl list --dimms

[
 {
  "dev":"nmem2",
  "id":"8089-00-0000-12325476"
 },
 {
  "dev":"nmem1",
  "id":"8089-00-0000-11325476"
 },
 {
  "dev":"nmem0",
  "id":"8089-00-0000-10325476"
 }
]
```

如果结合不同的参数，`ndctl list` 还可以列出可用的区域。

注意

区域可能不会按数字顺序显示。

请注意，尽管只有三个 NVDIMM，但它们却显示为四个区域。

```
root # ndctl list --regions

[
 {
  "dev":"region1",
  "size":68182605824,
  "available_size":68182605824,
  "type":"blk"
 },
 {
  "dev":"region3",
  "size":202937204736,
  "available_size":202937204736,
```

```
  "type":"pmem",
  "iset_id":5903239628671731251
  },
  {
   "dev":"region0",
   "size":68182605824,
   "available_size":68182605824,
   "type":"blk"
  },
  {
   "dev":"region2",
   "size":68182605824,
   "available_size":68182605824,
   "type":"blk"
  }
]
```

空间以两种不同的形式显示：三个 BLK 类型的独立 64 GB 区域，或者一个 PMEM 类型的合并 189 GB 区域，后者将三个交错式 NVDIMM 中的所有空间表示为单个卷。

请注意，`available_size` 的显示值与 `size` 的显示值相同。这意味着尚未分配任何空间。

24.5.2 将储存配置为使用 DAX 的单个 PMEM 名称空间

第一个示例将三个 NVDIMM 配置成使用 Direct Access (DAX) 的单个 PMEM 名称空间。

第一个步骤是创建新的名称空间。

```
root # ndctl create-namespace --type=pmem --mode=fsdax --map=memory
{
 "dev":"namespace3.0",
 "mode":"memory",
 "size":199764213760,
 "uuid":"dc8ebb84-c564-4248-9e8d-e18543c39b69",
 "blockdev":"pmem3"
}
```

这会创建支持 DAX 的块设备 `/dev/pmem3`。设备名称中的 `3` 继承自父区域编号（在本例中为 `region3`）。

`--map=memory` 选项从 NVDIMM 中设置出一部分 PMEM 储存空间，以便可以使用这些空间来分配称作 结构页面 的内部内核数据结构。这样，便可以将新的 PMEM 名称空间与 `O_DIRECT I/O` 和 `RDMA` 等功能搭配使用。

最终 PMEM 名称空间的容量之所以小于父 PMEM 区域，是因为有一部分持久内存预留给了内核数据结构。

接下来，我们校验新的块设备是否可用于操作系统：

```
root # fdisk -l /dev/pmem3
Disk /dev/pmem3: 186 GiB, 199764213760 bytes, 390164480 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

与其他任何驱动器一样，在使用该设备之前，必须先将其格式化。在本示例中，我们使用 XFS 将其格式化：

```
root # mkfs.xfs /dev/pmem3
meta-data=/dev/pmem3      isize=256    agcount=4, agsize=12192640 blks
         =                sectsz=4096  attr=2, projid32bit=1
         =                crc=0        finobt=0, sparse=0
data     =                bsize=4096   blocks=48770560, imaxpct=25
         =                sunit=0      swidth=0 blks
naming   =version 2       bsize=4096   ascii-ci=0 ftype=1
log      =internal log    bsize=4096   blocks=23813, version=2
         =                sectsz=4096  sunit=1 blks, lazy-count=1
realtime =none            extsz=4096   blocks=0, rtextents=0
```

接下来，可将新的驱动器装入到某个目录：

```
root # mount -o dax /dev/pmem3 /mnt/pmem3
```

然后，可以校验是否获得了一个支持 DAX 的设备：

```
root # mount | grep dax
/dev/pmem3 on /mnt/pmem3 type xfs (rw,relatime,attr2,dax,inode64,noquota)
```

结果是，我们已获得一个使用 XFS 文件系统格式化的，且装有 DAX 的 PMEM 名称空间。

对该文件系统中的文件进行任何 `mmap()` 调用都会返回直接映射到 NVDIMM 上的持久内存的虚拟地址，并且会完全绕过页面超速缓存。

对该文件系统中的文件进行任何 `fsync` 或 `msync` 调用仍可确保将修改后的数据完全写入到 NVDIMM。这些调用会刷新通过 `mmap` 映射在用户空间中修改的任何页面的关联处理器超速缓存行。

24.5.2.1 去除名称空间

在创建使用相同储存的其他任何类型的卷之前，我们必须卸载此 PMEM 卷，然后将其去除。

首先卸载该卷：

```
root # umount /mnt/pmem3
```

然后禁用名称空间：

```
root # ndctl disable-namespace namespace3.0
disabled 1 namespace
```

然后删除该卷：

```
root # ndctl destroy-namespace namespace3.0
destroyed 1 namespace
```

24.5.3 创建使用 BTT 的 PMEM 名称空间

在下一个示例中，我们将创建使用 BTT 的 PMEM 名称空间。

```
root # ndctl create-namespace --type=pmem --mode=sector
{
 "dev":"namespace3.0",
 "mode":"sector",
 "uuid":"51ab652d-7f20-44ea-b51d-5670454f8b9b",
 "sector_size":4096,
 "blockdev":"pmem3s"
```

```
}
```

接下来，校验新设备是否存在：

```
root # fdisk -l /dev/pmem3s
Disk /dev/pmem3s: 188.8 GiB, 202738135040 bytes, 49496615 sectors
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

与前面配置的支持 DAX 的 PMEM 名称空间一样，这个支持 BTT 的 PMEM 名称空间也会占用 NVDIMM 中的所有可用储存。

注意

设备名称 (`/dev/ pmem3s`) 尾部的 `s` 表示扇区，可用于轻松辨别配置为使用 BTT 的 PMEM 和 BLK 名称空间。

可按前一示例中所述格式化和装入卷。

此处显示的 PMEM 名称空间不能使用 DAX。它会使用 BTT 来提供扇区写入原子性。每次通过 PMEM 块驱动程序进行扇区写入时，BTT 都会分配一个新的扇区来接收新数据。完全写入新数据后，BTT 将以原子方式更新其内部映射结构，使新写入的数据可供应用程序使用。如果在此过程中的任意时间点发生电源故障，则写入内容将会完全丢失，在这种情况下，应用程序可以访问其旧数据，而这些数据仍旧保持不变。这可以防止出现所谓“扇区撕裂”的情况。

与其他任何标准块设备一样，可以使用某个文件系统格式化这个支持 BTT 的 PMEM 名称空间，并在该文件系统中使用它。无法将该名称空间与 DAX 搭配使用。但是，此块设备中的文件的 `mmap` 映射将使用页面超速缓存。

注意

在上面两个示例中，所有 NVDIMM 提供的空间都合并到一个卷中。与非冗余磁盘阵列相同，这表示如果任何单个 NVDIMM 发生错误，整个卷的内容都可能会丢失。卷中包含的 NVDIMM 越多，发生类似错误的几率就越高。

24.5.3.1 去除 PMEM 卷

如上例所述，在重新分配空间之前，必须先去除卷和名称空间：

```
root # ndctl disable-namespace namespace3.0
disabled 1 namespace

root # ndctl destroy-namespace namespace3.0
destroyed 1 namespace
```

24.5.4 创建 BLK 名称空间

在此示例中，我们将创建三个单独的 BLK 设备：每个 NVDIMM 各创建一个。

这种方法的其中一个优势是，如果任何单个 NVDIMM 发生故障，其他卷将不会受到影响。

注意

必须针对每个名称空间重复以下命令：

```
root # ndctl create-namespace --type=blk --mode=sector
{
 "dev":"namespace1.0",
 "mode":"sector",
 "uuid":"fed466bd-90f6-460b-ac81-ad1f08716602",
 "sector_size":4096,
 "blockdev":"ndblk1.0s"
}

root # ndctl create-namespace --type=blk --mode=sector
{
 "dev":"namespace0.0",
 "mode":"sector",
 "uuid":"12a29b6f-b951-4d08-8dbc-8dea1a2bb32d",
 "sector_size":4096,
 "blockdev":"ndblk0.0s"
}
```

```
root # ndctl create-namespace --type=blk --mode=sector
{
 "dev":"namespace2.0",
 "mode":"sector",
 "uuid":"7c84dab5-cc08-452a-b18d-53e430bf8833",
 "sector_size":4096,
 "blockdev":"ndblk2.0s"
}
```

然后，我们可以校验新设备是否存在：

```
root # fdisk -l /dev/ndblk*
Disk /dev/ndblk0.0s: 63.4 GiB, 68115001344 bytes, 16629639 sectors
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ndblk1.0s: 63.4 GiB, 68115001344 bytes, 16629639 sectors
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

Disk /dev/ndblk2.0s: 63.4 GiB, 68115001344 bytes, 16629639 sectors
Units: sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

针对 BLK 名称空间生成的块设备命名为 `/dev/ndblkX.Y`，其中 `X` 为父区域编号，`Y` 为该区域内唯一的名称空间编号。因此，`/dev/ndblk2.0s` 为区域 2 的子名称空间编号 0。

如上例所述，末尾的 `s` 表示该名称空间配置为使用 BTT，也就是说，它用于基于扇区的访问。由于它们是通过*块窗口*进行访问的，程序无法使用 DAX，但会对访问进行超速缓存。

与以往一样，这些设备必须格式化并装入系统后才能使用。

24.6 查错

持久内存的耐用性优于 SSD 储存，但也可能会用完。如果某个 NVDIMM 发生故障，就需要隔离出现问题的单个模块，以便能够恢复剩余数据并更换硬件。必须确定以下三条信息：

1. 发生故障的是哪个 NVDIMM 模块：有缺陷模块的物理位置。
2. 现在包含坏块的是哪个名称空间 (`/dev/pmemX`)。
3. 还有其他哪些名称空间或区域也在使用该物理模块。

确定故障模块以及使用该模块的所有名称空间和区域之后，便可以备份其他未受影响的名称空间中的数据，然后关闭服务器并更换 NVDIMM。

24.6.1 查找故障模块

服务器主板上的 DIMM 插槽中有一组 NVDIMM。

在产生的空间中，操作系统会创建一个或多个名称空间，例如 `region0`。

在这些区域内定义了某些名称空间，例如 `/dev/pmem1` 或 `/dev/dax0`。

例如，假设有一个区域由来自三个 NVDIMM 的空间构成，该区域已配置为以下三个名称空间：

NVDIMM 0	region0	/dev/pmem1	
NVDIMM 1	[X]	/dev/pmem2s	
NVDIMM 2	/dev/dax0		

在我们的示例中，标记为 [X] 的 `region0` 部分已损坏或有缺陷。

必须：

1. 确定包含受影响区域的 NVDIMM 模块。
   如果该区域跨多个 NVDIMM 交错，此步骤尤为重要。
2. 备份受影响 NVDIMM 上任何其他名称空间的内容。
   在此示例中，必须备份 `/dev/pmem2s` 的内容。
3. 确定名称空间与 NVDIMM 物理位置（位于主板的哪个内存插槽中）之间的关系。

必须关闭服务器并取下机箱盖，找到、去除并更换有缺陷模块。

24.6.2 测试持久内存

注意：故障排查的先决条件

要进行测试，需要有 `nfit_test` 内核模块。

`ndctl` 命令所对应 GitHub 页面中的单元测试部分的步骤 1 - 4 中详细介绍了测试过程。请参见本章末尾的第 24.7 节 “更多信息”。

过程 24.1 测试过程

1. 执行带有参数 `list -RM` 的 `ndctl` 命令。

   此操作将显示坏块列表。

```
tux > sudo ndctl list -RM
 :
 :
{
  "dev":"region5",
  "size":33554432,
  "available_size":33554432,
  "type":"pmem",
  "iset_id":4676476994879183020,
  "badblock_count":8,
  "badblocks":[
    {
      "offset":32768,
      "length":8,
      "dimms":[
         "nmem1"  ❶
      ]
    }
  ]
},
:
```

❶ 此处标识了这个特定的 NVDIMM。

2. 执行带有参数 `list -Du` 的 `ndctl` 命令。
   此操作将显示 DIMM 的句柄。

```
tux > sudo ndctl list -Du
{
   "dev":"nmem1",
   "id":"cdab-0a-07e0-fefffff",
   "handle":"0x1", ❶
   "phys_id":"0x1"
},
 :
 :
```

❶ 这是 NVDIMM 的句柄。

3. 执行带有参数 `list --d DIMM name` 的 `ndctl` 命令。

```
tux > sudo ndctl list -R -d nmem1
[
  {
    "dev":"region5",
    "size":33554432,
    "available_size":33554432,
    "type":"pmem",
    "iset_id":4676476994879183020,
    "badblock_count":8
   },
  :
  :
```

24.7 更多信息

可以在以下列表中找到关于此主题的更多信息：

- 持久内存 Wiki (https://nvdimm.wiki.kernel.org/)

包含有关配置 NVDIMM 系统的指导、有关测试的信息，以及有关启用 NVDIMM 的规范的链接。随着 Linux 中 NVDIMM 支持功能的不断发展，此站点的内容也会不断扩充。

- 持久内存编程 (http://pmem.io/)
  有关在 Linux 和其他操作系统中配置、使用非易失性内存，以及为使用此类内存的系统编程的信息。其中介绍了 NVM 库 (NVML)。该库旨在提供有用的 API 用于在用户空间中进行持久内存编程。

- LIBNVDIMM：非易失性设备 (https://www.kernel.org/doc/Documentation/nvdimm/nvdimm.txt)
  此文档面向内核开发人员，包含在当前 Linux 内核树的“文档”文件夹中。其中探讨了涉及启用 NVDIMM 的不同内核模块，列出了有关内核实现的一些技术细节，并介绍了 `ndctl` 工具使用的 `sysfs` 内核接口。

- GitHub：pmem/ndctl (https://github.com/pmem/ndctl)
  用于管理 Linux 内核中的 `libnvdimm` 子系统的实用程序库。另外还包含用户空间库，以及单元测试和文档。

IV 服务

25 使用 NTP 同步时间 347

26 域名系统 353

27 DHCP 377

28 通过 NFS 共享文件系统 393

29 Samba 404

30 使用 Autofs 按需装入 426

31 SLP 434

32 Apache HTTP 服务器 438

33 使用 YaST 设置 FTP 服务器 478

34 代理服务器 Squid 482

35 使用 SFCB 的基于 Web 的企业管理 505

25 使用 NTP 同步时间

NTP（网络时间协议）机制是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。

维护确切的系统时间在许多情况下都非常重要。内置硬件时钟往往不能满足数据库或群集这样的应用程序的要求。手动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。在网络中，通常需要同步所有计算机上的系统时间，但是手动调整时间是一种不好的方法。NTP 提供了解决这些问题的机制。NTP 服务通过网络中的可靠时间服务器持续调整系统时间。它还支持对本地参考时钟（如无线电控制的时钟）进行管理。

25.1 使用 YaST 配置 NTP 客户端

附于 `ntp` 包中的 NTP 守护程序 (`ntpd`) 预设置为使用本地计算机时钟作为时间参考。但是，请只在没有更精确的时间源的情况下才使用硬件时钟。YaST 简化了 NTP 客户端的配置。

25.1.1 基本配置

YaST NTP 客户端配置（网络服务 › NTP 配置）由数个选项卡组成。在常规设置选项卡上设置 `ntpd` 启动模式和要查询的服务器。

仅手工

如果您想手动启动 `ntpd` 守护程序，请选择仅手动。

不用守护程序同步

选择不用守护程序同步可定期设置系统时间，而不用永久运行 `ntpd`。您可以设置同步间隔（以分钟计）。

立即和引导时

选择立即和引导时可在引导系统时自动启动 `ntpd`。建议使用此设置。

25.1.2 更改基本配置

供客户端查询的服务器和其他时间资源列在常规设置选项卡的下半部分。使用添加、编辑和删除可按需修改此列表。显示日志使您能够查看客户端的日志文件。

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。下列选项可用：

图 25.1 YAST：NTP 服务器

服务器

在选择下拉列表（请参见图 25.1 “YaST：NTP 服务器”）中，确定是使用本地网络中的时间服务器（本地 NTP 服务器）设置时间同步，还是使用会处理您时区的因特网时间服务器（公共 NTP 服务器）设置时间同步。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或地区（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，用测试来测试所选服务器是否可用。选项使您可以指定 `ntpd` 的其他选项。

使用访问控制选项，您可限制远程计算机通过您的计算机上运行的守护程序所能执行的操作。仅在选中安全设置选项卡（请参见图 25.2 “高级 NTP 配置：安全设置”）上的限制 NTP 服务仅用于已配置的服务器后，才能启用此字段。选项对应于 `/etc/ntp.conf` 中

的 restrict 子句。例如，nomodify notrap noquery 禁止服务器修改计算机的 NTP 设置并禁止使用 NTP 守护程序的陷阱工具（一种远程事件记录功能）。建议将这些限制用于超出您控制范围的服务器（例如在因特网上）。

有关详细信息，请参见 /usr/share/doc/packages/ntp-doc（ntp-doc 包的一部分）。

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端。要在同一网络中用同级代替某个服务器，请输入系统的地址。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/ntp-doc/refclock.html。

传出广播

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。

传入广播

如果希望客户端通过广播接收信息，请在此字段中输入应接受来自哪个地址的相应数据包。

图 25.2 高级 NTP 配置：安全设置

在安全设置选项卡（请参见图 25.2 “高级 NTP 配置：安全设置”）中，确定 `ntpd` 是否应在 chroot jail 中启动。默认不会激活在 Chroot Jail 中运行 NTP 守护程序。当 `ntpd` 受到攻击时，chroot jail 选项可以提高安全性，因为该选项可以防止攻击者危害整个系统。

限制 NTP 服务用于配置过的服务器通过禁止远程计算机查看和修改您计算机的 NTP 设置以及禁止使用用于远程事件记录的陷阱工具，从而增强了系统的安全性。启用后，这些限制将应用于所有远程计算机，除非覆盖在常规设置选项卡的时间源列表中对个别计算机的访问控制选项。对于所有其他远程计算机，仅允许查询本地时间。

如果 SuSEfirewall2 处于活动状态（默认），请启用打开防火墙中的端口。如果保持端口的关闭状态，则不可能建立与事件服务器的连接。

25.2 手动配置网络中的 NTP

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，如果可以从网络访问名为 `ntp.example.com` 的时间服务器，请通过添加以下行将其名称添加到文件 `/etc/ntp.conf` 中：

```
server ntp.example.com
```

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 `systemctl start ntp` 初始化 `ntpd` 后，系统大约会花费一个小时来稳定时间以及创建用于更正本地计算机时钟的偏移文件。利用偏移文件，当计算机启动时，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 机制用作客户端：第一种方法是客户端可以定期从已知服务器查询时间。在存在许多客户端的情况下，这种方法会给服务器带来很高的负荷。第二种方法是客户端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。

25.3 运行时动态时间同步

如果在无网络连接的情况下引导系统，则 `ntpd` 将启动，但是它无法解析在配置文件中设置的时间服务器的 DNS 名称。在加密的 Wi-Fi 中使用 NetworkManager 时，可能会发生这种情况。

如果希望 `ntpd` 在运行时解析 DNS 名称，则必须设置 `dynamic` 选项。在引导后建立网络连接时，`ntpd` 将再次查找名称并可以访问时间服务器以获取时间信息。

手动编辑 `/etc/ntp.conf` 并将 `dynamic` 添加到一个或多个 `server` 项：

```
server ntp.example.com dynamic
```

您也可以使用 YaST 并按如下步骤操作：

1. 在 YaST 中单击网络服务 › NTP 配置。
2. 选择要配置的服务器。然后单击编辑。
3. 激活选项字段并添加 `dynamic`。如果已输入其他选项，请用空格分隔。
4. 单击确定关闭编辑对话框。重复之前的步骤以根据需要更改所有服务器。
5. 最后单击确定保存设置。

25.4 设置本地参考时钟

软件包 `ntpd` 包含用于连接本地参考时钟的驱动程序。`ntp-doc` 包的文件 `/usr/share/doc/packages/ntp-doc/refclock.html` 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 NTP 中，实际配置通过伪 IP 地址实现。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` 代表时钟的类型并可决定要使用的驱动程序，`U` 代表单位，可决定要使用的接口。

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/ntp-doc/drivers/driverNN.html`（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 `ntp-doc` 包之后，可以在目录 `/usr/share/doc/packages/ntp-doc` 中找到 NTP 的文档。文件 `/usr/share/doc/packages/ntp-doc/refclock.html` 提供指向描述驱动程序参数的驱动程序页的链接。

25.5 与外部时间参考 (ETR) 的时钟同步

支持与外部时间参考 (ETR) 时钟同步。外部时间参照每 2**20（2 的 20 次幂）微秒发出一次振荡器信号和同步信号，以将连接的所有服务器的 TOD 时钟保持同步。

两个 ETR 计算机可以连接到一台计算机。如果时钟偏移超过同步检查容差，所有 CPU 都会获得一次计算机检查，表示该时钟已失步。如果发生这种情况，在该时钟再次同步前，所有与支持 XRC 的设备进行的 DASD I/O 都将停止。

可通过两个 `sysfs` 属性激活 ETR 支持；请以 `root` 身份运行以下命令：

```
echo 1 > /sys/devices/system/etr/etr0/online
echo 1 > /sys/devices/system/etr/etr1/online
```

26 域名系统

需要使用 DNS（域名系统）将域名和主机名解析成 IP 地址。这样，便可以将 IP 地址指派给主机名，例如，将 192.168.2.100 指派给 `jupiter`。在设置您自己的名称服务器前，请阅读第 16.3 节 “名称解析”中有关 DNS 的一般信息。下列配置示例中使用了默认的 DNS 服务器 - BIND。

26.1 DNS 术语

区域

域名称空间由一些区域 (zone) 组成。例如，如果您拥有 `example.com`，您就拥有 `com` 域的 `example` 段（或区域）。

DNS 服务器

DNS 服务器是维护域的名称和 IP 信息的服务器。主 DNS 服务器可用于主要区域、次要服务器可用于从属区域，不含有任何区域的从属服务器可用于缓存。

主区域 DNS 服务器

主区域包含网络中的所有主机，DNS 服务器主区域储存域中所有主机的最新记录。

从属区域 DNS 服务器

从属区域是主区域的副本。从属区域 DNS 服务器将使用区域传送操作从其主服务器获取区域数据。只要从属区域 DNS 服务器具有有效（没到期）的区域数据，便会对区域作出权威响应。如果从属服务器不能获取区域数据的新副本，它将停止响应区域。

转发器

转发器是您的 DNS 服务器应将无法应答的查询发送到的 DNS 服务器。使用 `netconfig`，以便在一个配置中启用不同配置源（另请参见 `man 8 netconfig`）。

记录

记录就是有关名称和 IP 地址的信息。有关支持的记录及其语法的描述可在 BIND 文档中获取。以下是一些特殊记录：

NS 记录

NS 记录会告诉命名服务器哪些计算机负责给定域区域。

MX 记录

MX（邮件交换）记录描述在因特网中定向邮件时要联系的计算机。

SOA 记录

SOA（起始授权机构）记录是区域文件中的第一条记录。当使用 DNS 在多台计算机之间同步数据时，使用 SOA 记录。

26.2 安装

要安装 DNS 服务器，请启动 YaST 并选择软件 › 软件管理。选择视图 › 模式，然后选择 DHCP 和 DNS 服务器。确认安装相关的包来完成安装进程。

或者，在命令行中使用以下命令：

```
zypper in -t pattern dhcp_dns_server
```

26.3 使用 YaST 进行配置

使用 YaST DNS 模块为本地网络配置 DNS 服务器。第一次启动此模块时，会启动向导，提示您做出一些有关服务器管理的决定。完成此初始设置后，将产生基本服务器配置。使用专家模式可处理更高级的配置任务，例如设置 ACL、日志记录、TSIG 密钥和其他选项。

26.3.1 向导配置

向导由三个步骤或对话框组成。您可以从对话框的适当位置进入专家配置模式。

1. 第一次启动模块时，您将看到图 26.1 “DNS 服务器安装：转发器设置”中显示的转发器设置对话框将打开。本地 DNS 解析策略允许设置以下选项：

   - 已禁用合并转发器
   - 自动合并

- 已启用合并转发器
- 自定义配置 — 如果已选择自定义配置，则可以指定自定义策略；默认情况下，在已选择自动合并的情况下，自定义策略将设置为 自动 ，但是，您可以在此处设置接口名称，或者从 STATIC 与 STATIC_FALLBACK 这两个特殊策略名称中做出选择。

在本地 DNS 解析转发器中，指定要使用的服务：使用系统名称服务器、此名称服务器(bind)或本地 dnsmasq 服务器。
有关有关所有这些设置的更多信息，请参见 `man 8 netconfig`。

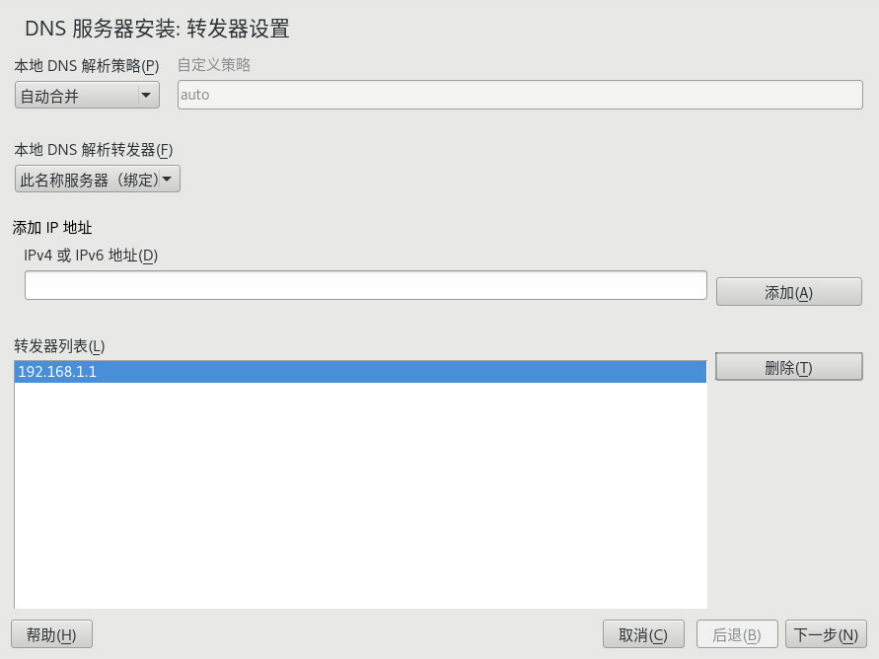


图 26.1 DNS 服务器安装：转发器设置

转发器是接收您的 DNS 服务器自己无法应答的查询的 DNS 服务器。输入它们的 IP 地址，然后单击添加。

2. DNS 区域对话框由多个部分组成，负责管理区域文件（如第 26.6 节 “区域文件”中所述）。对于新区域，请在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。最后选择类型（主、从属或转发）。参见图 26.2 “DNS 服务器安装：DNS 区域”。单击编辑可配置现有区域的其他设置。要删除区域，请单击删除。

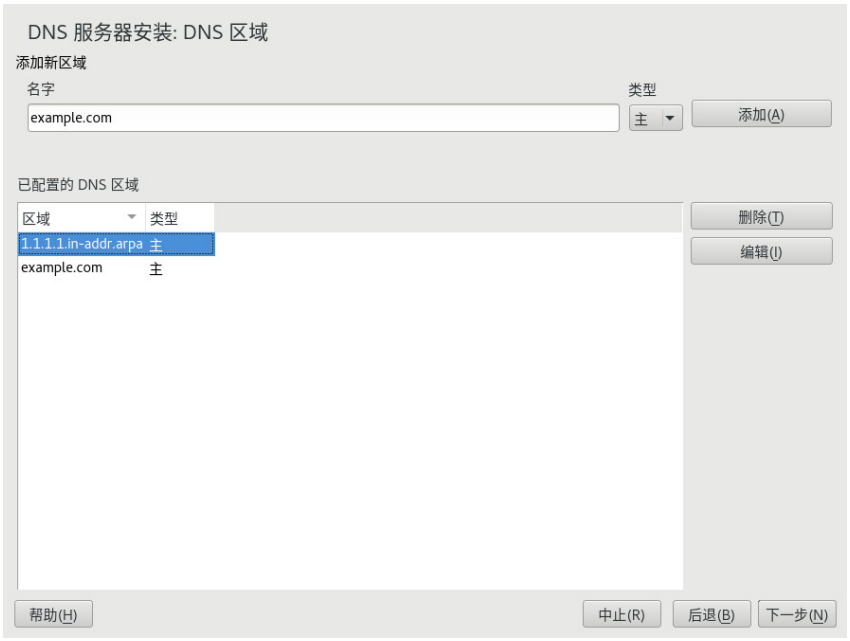


图 26.2 DNS 服务器安装：DNS 区域

3. 在最后对话框中，您可以通过单击打开防火墙中的端口打开防火墙中的 DNS 端口。然后决定是否在引导时启动 DNS 服务器（开启或关闭）。您还可以激活 LDAP 支持。请参见图 26.3 “DNS 服务器安装：完成向导”。

图 26.3 DNS 服务器安装：完成向导

26.3.2 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：

26.3.2.1 启动

在启动下，定义是要在引导系统时启动 DNS 服务器，还是手动启动 DNS 服务器。要立即启动 DNS 服务器，请单击立即启动 DNS 服务器。要停止 DNS 服务器，请单击立即停止 DNS 服务器。要保存当前设置，请选择保存设置并立即重新装载 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。

通过选择 LDAP 支持处于活动状态，让 LDAP 数据库管理区域文件。当 DNS 服务器重启动或被提示重装载其配置时，它会挑选出写入到 LDAP 数据库的任何区域数据更改。

26.3.2.2 转发器

如果您的本地 DNS 服务器无法应答请求，则会尝试将请求转发给转发器（如果进行了这样的配置）。转发器可手动添加到转发器列表。如果在拨号连接中转发器不是静态的，则 netconfig 会处理配置。有关 netconfig 的更多信息，请参见 `man 8 netconfig`。

26.3.2.3 基本选项

在这一部分，设置基本的服务器选项。从选项菜单中选择所需的项，然后在相应文本框中指定值。选择添加包括新的条目。

26.3.2.4 日志记录

要设置 DNS 服务器应该记录的内容和记录方法，请选择日志记录。在日志类型下，指定 DNS 服务器将日志数据写入的位置。选择系统日志以使用系统范围的日志，或选择文件以指定另一个文件。对于后者，请额外指定名称、最大文件大小（以兆字节为单位）以及要储存的日志文件版本数。

在附加日志记录下可以使用其他一些选项。启用记录所有的 DNS 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数据流量，请启用记录区域传送。请参见图 26.4 “DNS 服务器：日志记录”。

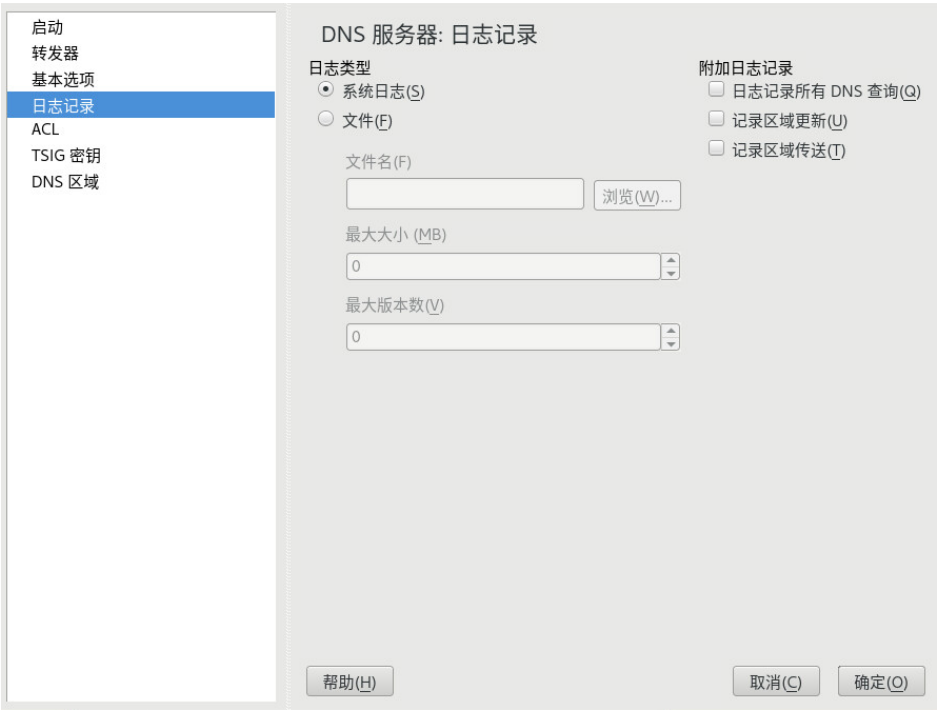


图 26.4 DNS 服务器：日志记录

26.3.2.5 ACL

使用此对话框定义 ACL（访问控制列表）来强制执行访问限制。在名称下提供不同的名称后，在值下指定具有下列形式的 IP 地址（带有或不带有网络掩码）：

```
{ 192.168.1/24; }
```

配置文件的语法要求地址以分号结尾且放在花括号中。

26.3.2.6 TSIG 密钥

TSIG（事务签名）的主要用途是保护 DHCP 和 DNS 服务器间通讯的安全性。这些内容在第 26.8 节 “安全事务”中有所介绍。

要生成 TSIG 密钥，请在标为密钥 ID 的字段中输入一个唯一名称，并指定储存密钥的文件（文件名）。用生成确认选择。

要使用以前创建的密钥，请将密钥 ID 字段保留为空，并在文件名下选择储存这个密钥 ID 的文件。选择后，请用添加按钮进行确认。

26.3.2.7 DNS 区域（添加从属区域）

要添加从属区域，请选择 DNS 区域，然后选择区域类型从属，写入新区域名称并单击添加。

在主 DNS 服务器 IP 下的区域编辑器子对话框中，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，可从列表中选择一个 ACL。

26.3.2.8 DNS 区域（添加主区域）

要添加主区域，请选择 DNS 区域，然后选择区域类型主，写入新区域名称并单击添加。当添加主区域时，也需要一个反向区域。例如，当添加区域 `example.com`（指向子集 `192.168.1.0/24` 中的主机）时，也应为包含的 IP 地址范围添加一个反时向区域。按照定义，应命名为 `1.168.192.in-addr.arpa`。

26.3.2.9 DNS 区域（编辑主区域）

要编辑主区域，请选择 DNS 区域，从表中选择主区域，最后单击编辑。该对话框包含几个页面：基本（第一个打开的页面）、NS 记录、MX 记录、SOA 和记录。

图 26.5 “DNS 服务器：区域编辑器（基本）”中显示的基本对话框用于定义动态 DNS 的设置以及指向客户端和从属名称服务器的区域传送的访问选项。要允许动态更新区域，请选择允许动态更新及相应的 TSIG 密钥。必须在更新操作开始前定义密钥。要启用区域传送，请选择相应的 ACL。必须已经定义了 ACL。

在基本对话框中，选择是否启用区域传输。使用所列 ACL 来定义谁能够下载区域。

图 26.5 DNS 服务器：区域编辑器（基本）

区域编辑器（NS 记录）

NS 记录对话框用于为指定的区域定义备用名称服务器。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 26.6 “DNS 服务器：区域编辑器（NS 记录）”。

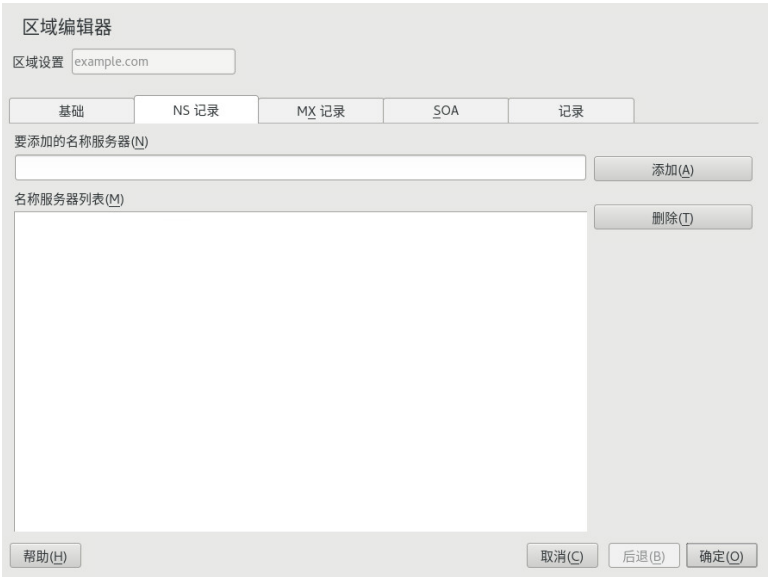


图 26.6 DNS 服务器：区域编辑器（NS 记录）

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。请参见图 26.7 “DNS 服务器：区域编辑器（MX 记录）”。

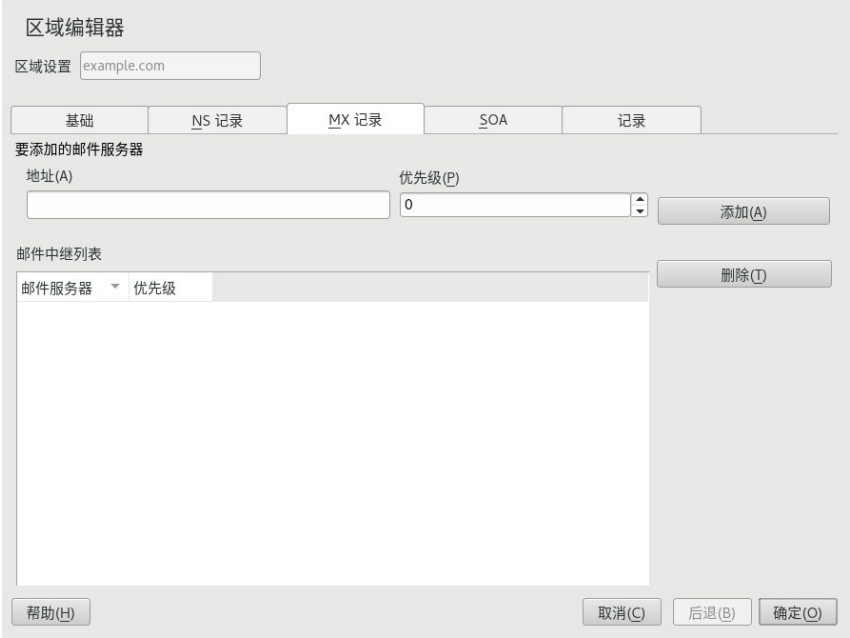


图 26.7 DNS 服务器：区域编辑器（MX 记录）

区域编辑器 (SOA)

此页用于创建 SOA（起始授权机构）记录。有关各选项的描述，请参见例 26.6 “/var/lib/named/example.com.zone 文件”。通过 LDAP 管理的动态区域不支持更改 SOA 记录。

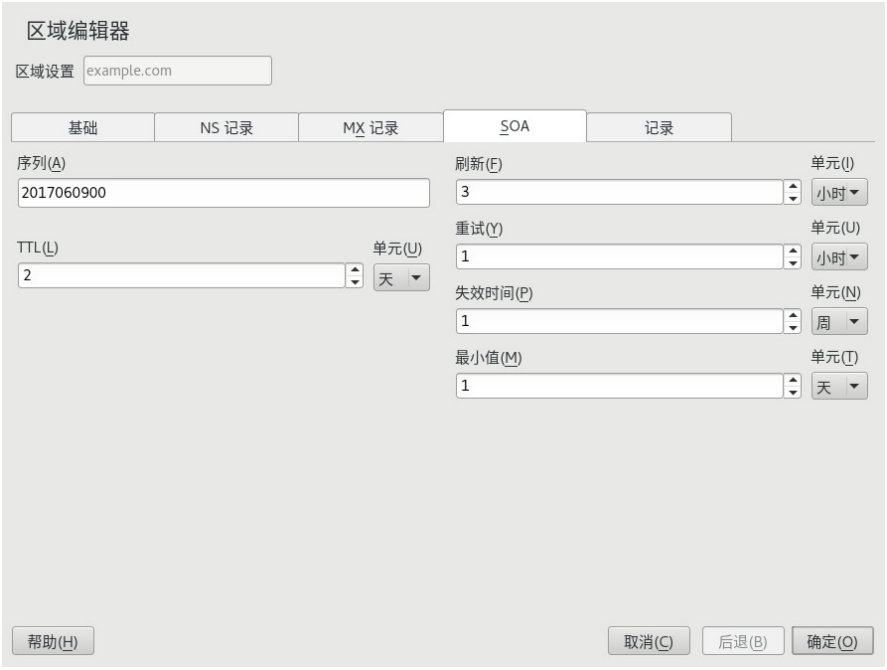


图 26.8 DNS 服务器：区域编辑器 (SOA)

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 类型表示主项。此项的值应为一个 IP 地址 (IPv4)。对于 IPv6 地址，请使用 AAAA。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，例如：

```
hostname.example.com. IN A 192.168.0.1
1.0.168.192.in-addr.arpa IN PTR hostname.example.com.
```

26.3.2.9.1 添加反向区域

要添加反向区域，请遵循以下过程：

1. 启动 YaST › DNS 服务器 › DNS 区域。
2. 如果您尚未添加主正向区域，请添加并编辑它。
3. 在记录选项卡中，填写相应的记录键和值，然后单击添加来添加记录，并单击确定来确认。如果 YaST 告知不存在某个名称服务器的记录，请在 DNS 记录选项卡中添加该记录。

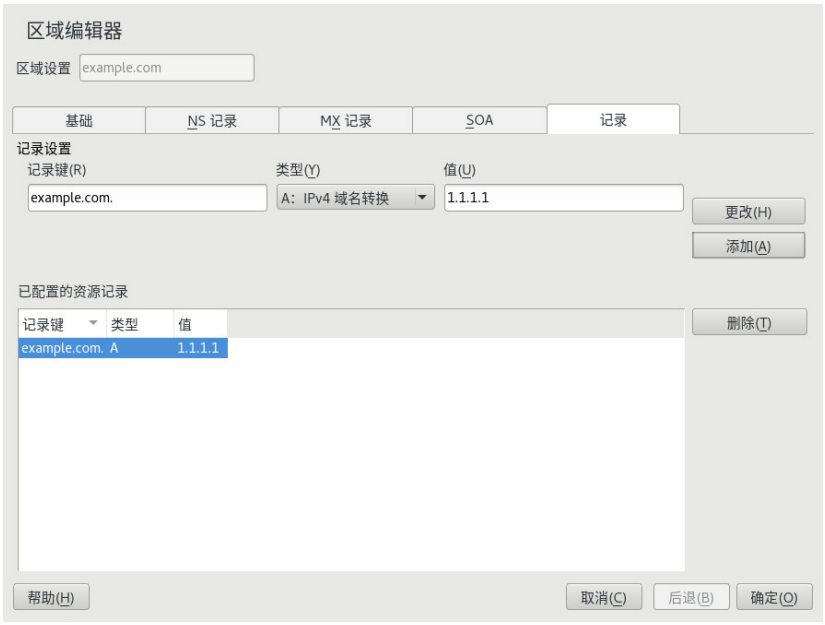


图 26.9 为主区域添加记录

4. 返回 DNS 区域窗口，添加一个主反向区域。

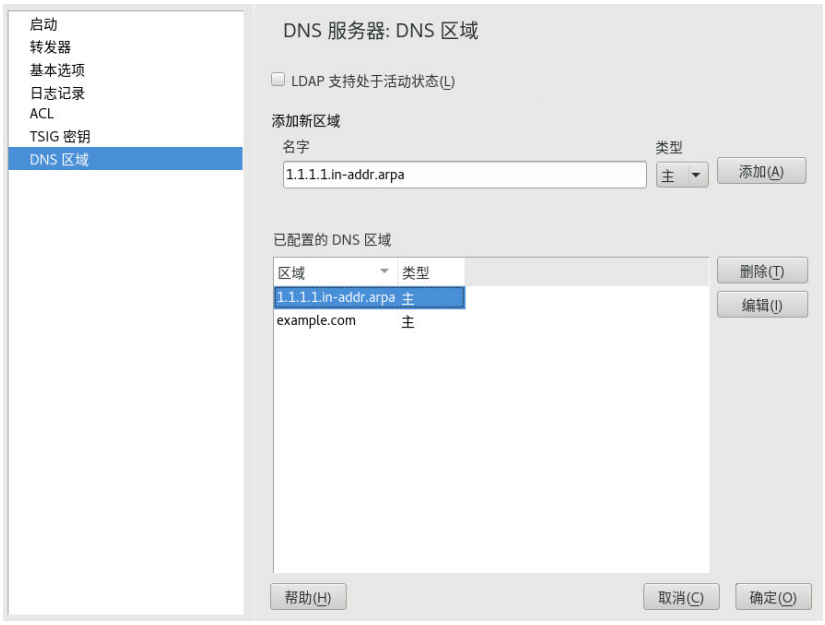


图 26.10 添加反向区域

5. 编辑该反向区域，然后在记录选项卡中，您可以看到 PTR：反向转换记录类型。添加相应的记录键和值，然后单击添加并单击确定来确认。

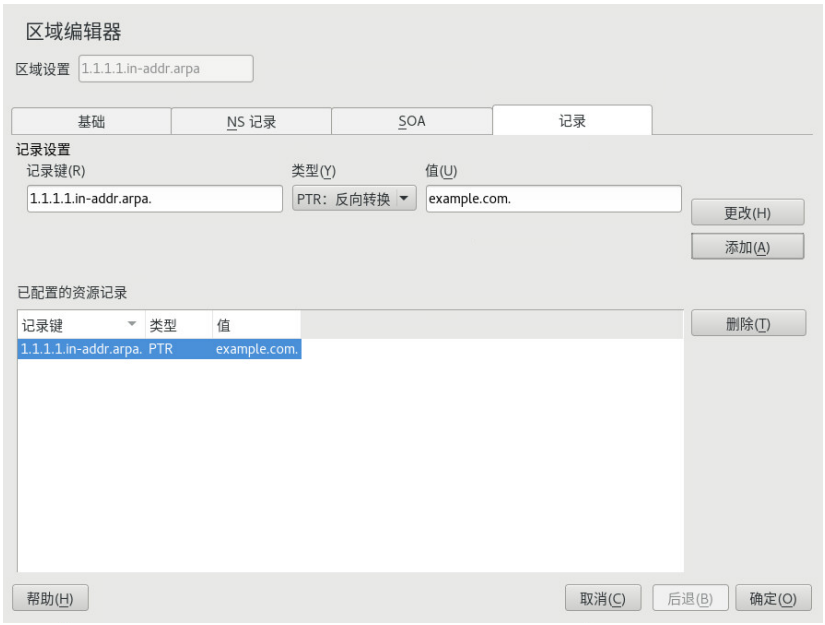


图 26.11 添加反向记录

根据需要添加名称服务器记录。

提示：编辑反向区域

添加正向区域后，返回到主菜单并选择该反向区域以进行编辑。在选项卡基础中激活复选框自动生成记录的区域，然后选择正向区域。这样，对正向区域的所有更改都会在反向区域中更新。

26.4 启动 BIND 名称服务器

在 SUSE® Linux Enterprise Server 系统上，已预先配置名称服务器 BIND（Berkeley 因特网名称域），因此在安装后可以立即启动此名称服务器，而不会出现任何问题。一般而言，如果您已有因特网连接，并在 `/etc/resolv.conf` 中输入了 `127.0.0.1` 作为 `localhost` 的名称服务器地址，则表示您已经有正常工作的名称解析功能，因而无需知道提供商的 DNS。BIND 通过 root 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，它才能成为正确的 DNS。在 `/usr/share/doc/packages/bind/config` 中可找到简单的示例。

提示：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 `/etc/sysconfig/network/config` 文件中的 `NETCONFIG_DNS_POLICY` 变量设置为 `auto`。

但是不要设置正式域，而是让负责机构指派给您。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 Web 服务器。

要启动名称服务器，请以 `root` 身份输入命令 `systemctl start named`。使用 `systemctl status named` 检查 named（当调用名称服务器进程时）是否已成功启动。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，则表明 `/etc/resolv.conf` 可能包含不正确的名称服务器项或此文件不存在。如果是第一次测试，请输入 `host 127.0.0.1`，此命令应始终有效。如果收到错误讯息，请使用 `systemctl status named` 查看服务器是否确实在运行。如果该名称服务器未启动或者出现意外的行为，请检查 `journalctl -e` 的输出。

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 26.1 “named.conf 中的转发选项”中包含的地址仅用作示例。请根据您自己的设置调整这些项。

例 26.1 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; 192.168.1.116; };
        allow-query { 127/8; 192.168/16 };
        notify no;
};
```

options 项后跟区域的项 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。更改配置文件 /etc/named.conf 或区域文件后，使用 systemctl reload named 来通知 BIND 重新读取这些文件。使用 systemctl restart named 停止然后重启动名称服务器也会获得相同的效果。您随时可以输入 systemctl stop named 来停止服务器。

26.5 /etc/named.conf 配置文件

BIND 名称服务器本身的所有设置都储存在文件 /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）储存在目录 /var/lib/named 下单独的文件中。稍后将介绍其详细信息。

/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 26.2 “基本的 /etc/named.conf” 显示了一个最小的 /etc/named.conf。

例 26.2 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";
};
```

26.5.1 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 192.168.1.116）替换 IP-ADDRESS。

forward first;

在尝试通过 root 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 root 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 通过哪些网络接口和哪个端口来接受客户端查询。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 应通过哪个端口侦听 IPv6 客户端请求。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; };

定义客户端可以自此发送 DNS 请求的网络。用地址信息（例如 192.168.2.0/24）替换 NET。末尾的 /24 是网络掩码的缩写表示（在本例中为 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 ! *. 如果没有此项，则可以从没有限制的任何位置请求区域传送。

statistics-interval 0;

如果缺少此项，则 BIND 每小时都会在系统的日记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缓存。每次清除时都会在系统的日记中触发一项。时间是以分钟为单位指定的。默认值为 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 只侦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启动时得到通知。

有关可用选项的列表，请阅读手册页 man 5 named.conf。

26.5.2 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，默认设置就已足够。例 26.3 “禁用日志记录的项”显示了此项最简单的形式，并完全抑制任何日志记录。

例 26.3 禁用日志记录的项

```
logging {
        category default { null; };
};
```

26.5.3 区域项

例 26.4 EXAMPLE.COM 的区域项

```
zone "example.com" in {
      type master;
      file "example.com.zone";
      notify no;
};
```

在 `zone` 后，指定要管理的域的名称（`example.com`），后跟 `in` 和用花括号括起来的相关选项块，如例 26.4 “example.com 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 26.5 “example.net 的区域项” 所示。

例 26.5 EXAMPLE.NET 的区域项

```
zone "example.net" in {
      type slave;
      file "slave/example.net.zone";
      masters { 10.0.0.1; };
};
```

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;
: 区域 `.`（`hint` 类型）用于设置 root 名称服务器。此区域定义可以保留原样。

file `example.com.zone` 或 file “slave/example.net.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。

26.6 区域文件

所需的区域文件有两种类型。一种类型将 IP 地址指派给主机名，另一种类型则正好相反：为 IP 地址提供主机名。

提示：在区域文件中使用点（句点）

此 "." 在区域文件中有重要的含义。如果指定的主机名末尾没有点 (.)，则会追加区域。使用完整域名指定的完整主机名必须以点 (.) 结尾，这是为了避免再次向它添加域。“.”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。

第一个要考虑的情况是负责域 example.com 的区域文件 example.com.zone，如例 26.6 “/var/lib/named/example.com.zone 文件”中所示。

例 26.6 /VAR/LIB/NAMED/EXAMPLE.COM.ZONE 文件

```
1.  $TTL 2D
2.  example.com. IN SOA      dns  root.example.com. (
3.               2003072441  ; serial
4.               1D          ; refresh
5.               2H          ; retry
6.               1W          ; expiry
7.               2D )        ; minimum
8.
9.               IN NS       dns
10.              IN MX       10 mail
11.
12. gate         IN A        192.168.5.1
```

```
13.               IN A        10.0.0.1
14. dns           IN A        192.168.1.116
15. mail          IN A        192.168.3.108
16. jupiter       IN A        192.168.2.100
17. venus         IN A        192.168.2.101
18. saturn        IN A        192.168.2.102
19. mercury       IN A        192.168.2.103
20. ntp           IN CNAME    dns
21. dns6          IN A6  0    2002:c0a8:174::
```

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：

- 最前面的 `example.com` 是要管理的域的名称。此域名以“`.`”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` 之后是用作此区域主服务器的名称服务器的名称。此名称会从 `dns` 扩展为 `dns.example.com`，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@example.com`，该项必须读作 `root.example.com.`。此 `"."` 必须包含在末尾，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，每次更改此文件时此数字都会增加。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 10 位数字格式，书写方式为 YYYYMMDDNN，这已成为惯用格式。

第 4 行：

`refresh rate` 指定二级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` 指定二级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` 指定二级名称服务器在无法重新联系上主服务器时将在多长时间后丢弃缓存的数据。在本例中为一周。

第 7 行：

SOA 记录中的最后一项指定 `negative caching TTL`，这是超速缓存来自其它服务器的未解析 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。`dns` 会扩展为 `dns.example.com`，因为它的没有以“`.`”. 可以有多个与此行类似的行，一行用于主名称服务器，其它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 `example.com` 的电子邮件的邮件服务器。在本例中，该邮件服务器是主机 `mail.example.com`。主机名称前面的数字是优先顺序值。如果有多个 MX 项，则优先选用值最小的邮件服务器。如果将邮件递送到此服务器失败，则会使用下一个值更大的项。

第 12-19 行：

这些都是实际的地址记录，在这里将一个或多个 IP 礬u216 OO稽u214 O窟u197 AE窟u248 oO鳟u187 机Auu163名称在此处列出，不带 `"."`，因为它们不包含自身的域，因此会将 `example.com` 添加到所有名称。因为主机 `gate` 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 `AAAA` 标记此项。

注意：IPv6 语法

IPv6 记录与 IPv4 记录的语法稍有不同。由于可能进行碎片整理，所以需要在寻址前提供有关缺失位的信息。要使用所需的数字“}0”填写 IPv6 地址，请在地址中的正确位置添加两个冒号。

```
pluto     AAAA 2345:00C1:CA11::1234:5678:9ABC:DEF0
pluto     AAAA 2345:00D2:DA11::1234:5678:9ABC:DEF0
```

第 20 行：

别名 `ntp` 可用于确定 `dns`（`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` 解析成 `168.192.in-addr.arpa`。参见 例 26.7 “反向查找”。

例 26.7 反向查找

```
1.  $TTL 2D
2.  168.192.in-addr.arpa.   IN SOA dns.example.com. root.example.com. (
3.                          2003072441      ; serial
4.                          1D              ; refresh
5.                          2H              ; retry
6.                          1W              ; expiry
7.                          2D )            ; minimum
8.
9.                          IN NS           dns.example.com.
10.
11. 1.5                     IN PTR          gate.example.com.
12. 100.3                   IN PTR          www.example.com.
13. 253.2                   IN PTR          cups.example.com.
```

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` 的反向查找。假设区域名为 `168.192.in-addr.arpa`，则不应将此区域添加到主机名。因此，采用完整形式输入所有主机名 — 带有它们的域并以“`.`”结尾。其余的项对应于上一个 `example.com` 示例介绍的那些内容。

第 3-7 行：

请参见上一个 `example.com` 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 的不同版本间传输区域，不会产生任何问题。

26.7 区域数据的动态更新

术语动态更新指添加、更改或删除主服务器区域文件中的项的操作。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。不应手动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，请查看关于 nsupdate 的手册页 (man 8 nsupdate)。出于安全原因，应使用 第 26.8 节 “安全事务” 中介绍的 TSIG 密钥执行此类更新。

26.8 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。

不同服务器间的通信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（有关细节，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 之间的安全通信：

```
key host1-host2 {
 algorithm hmac-md5;
 secret "ejIkuCyyGJwwuN3xAteKgg==";
};
```

警告：/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。要包括外部文件，请使用：

```
include  "filename"
```

用带有密钥的文件的绝对路径替换 filename。

要使服务器 host1 能使用 host2（在本例中，其地址为 10.1.2.3）的密钥，服务器的 /etc/named.conf 必须包含下列规则：

```
server 10.1.2.3 {
  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。

26.9 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-signzone`，您可以创建生成的密钥集（`keyset-` 文件），将它们以安全方式传送到父区域并加以签名。这会生成要包含在 `/etc/named.conf` 中的针对每个区域的文件。

26.10 更多信息

有关更多信息，请参见安装在 `/usr/share/doc/packages/bind/arm` 下的 `bind-doc` 包中的 BIND Administrator Reference Manual（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 Enterprise Server 中 BIND 的最新信息。

27 DHCP

动态主机配置协议 (DHCP) 用于从服务器集中指派网络设置，如此就不必在每个工作站本地配置这些设置。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。如果在客户端使用 NetworkManager，则无需配置客户端。在更改了环境并且一次只能使用一个活动的接口时，它才有用。请勿在运行 DHCP 服务器的机器上使用 NetworkManager。

提示：IBM Z：DHCP 支持

在 IBM Z 平台上，DHCP 仅在使用 OSA 和 OSA Express 网卡的接口上有效。现在只有这些网卡具有 MAC，因为需要 MAC 来实现 DHCP 自动配置功能。

配置 DHCP 服务器的方法之一是使用网卡的硬件地址（在大多数情况下应是固定的）来标识每个客户端，然后在客户端每次连接到服务器时为它提供相同的设置。另一种方法是对 DHCP 进行配置，从为此设置的地址池来为每个相关客户端动态指派地址。在后一种情况下，DHCP 服务器每次在收到客户端请求时都会尝试向它指派相同的地址，即使相隔较长的时间也是如此。只有在网络中包含的客户端数不超过地址数时，它才生效。

DHCP 简化了系统管理员的工作。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特别有用。

在本章中，DHCP 服务器将在工作站所在的子网内运行，即 `192.168.2.0/24`（网关为 `192.168.2.1`）。它具有固定的 IP 地址 `192.168.2.254`，并提供两个地址范围：`192.168.2.10` 至 `192.168.2.20`，`192.168.2.100` 至 `192.168.2.200`。

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 还允许您集中配置多个其他参数，例如客户端可能从中巡回检测当前时间的时间服务器，甚至是打印服务器。

27.1 使用 YaST 配置 DHCP 服务器

要安装 DHCP 服务器，请启动 YaST 并选择软件 › 软件管理。选择过滤器 › 模式，然后选择 DHCP 和 DNS 服务器。确认安装相关的包来完成安装进程。

重要：LDAP 支持

可以将 YaST DHCP 模块设置为在本地储存服务器配置（即运行 DHCP 服务器的主机上），或使其配置数据由 LDAP 服务器管理。要使用 LDAP，请在配置 DHCP 服务器前设置 LDAP 环境。

有关 LDAP 的更多信息，请参见《Security Guide》，第 5 章 “LDAP—A Directory Service”。

使用 YaST DHCP 模块（`yast2-dhcp-server`）可将自己的 DHCP 服务器设置用于本地网络。此模块能以向导模式或专家配置模式运行。

27.1.1 初始配置（向导）

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。按如下所示继续：

1. 从该列表中选择 DHCP 服务器应侦听的接口，然后单击选择。随后，请选择针对所选接口打开防火墙打开此接口的防火墙，单击下一步。请参见图 27.1 “DHCP 服务器：卡选择”。

图 27.1 DHCP 服务器：卡选择

2. 使用此复选框来确定是否由 LDAP 服务器自动储存您的 DHCP 设置。在文本框中，提供 DHCP 服务器应管理的所有客户端的网络细节。这些细节包括域名、时间服务器地址、主名称服务器和二级名称服务器的地址、打印和 WINS 服务器的地址（对于同时包含 Windows 和 Linux 客户端的混合网络）、网关地址和租用时间。请参见图 27.2 “DHCP 服务器：全局设置”。

图 27.2 DHCP 服务器：全局设置

3. 配置如何为客户端指派动态 IP 地址。为此，应首先指定服务器为 DHCP 客户端指派地址时使用的 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，在此期间客户端可以保留它的 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端保留某个 IP 地址的时间。请参见图 27.3 “DHCP 服务器：动态 DHCP”。

图 27.3 DHCP 服务器：动态 DHCP

4. 定义 DHCP 服务器应如何启动。指定 DHCP 服务器是在引导系统时自动启动还是在需要时（例如进行测试时）手动启动。单击完成以完成对服务器的配置。请参见图 27.4 “DHCP 服务器：启动”。

图 27.4 DHCP 服务器：启动

5. 除了按上文所述方式使用动态 DHCP 之外，您也可以将服务器配置为以准静态方式指派地址。使用窗口下部提供的文本框来指定要以此方式管理的一组客户端。具体地说就是提供客户端的名称和 IP 地址，以及硬件地址和网络类型（令牌环或以太网）。使用添加、编辑和从列表中删除来修改在窗口上部显示的客户端列表。请参见图 27.5 “DHCP 服务器：主机管理”。

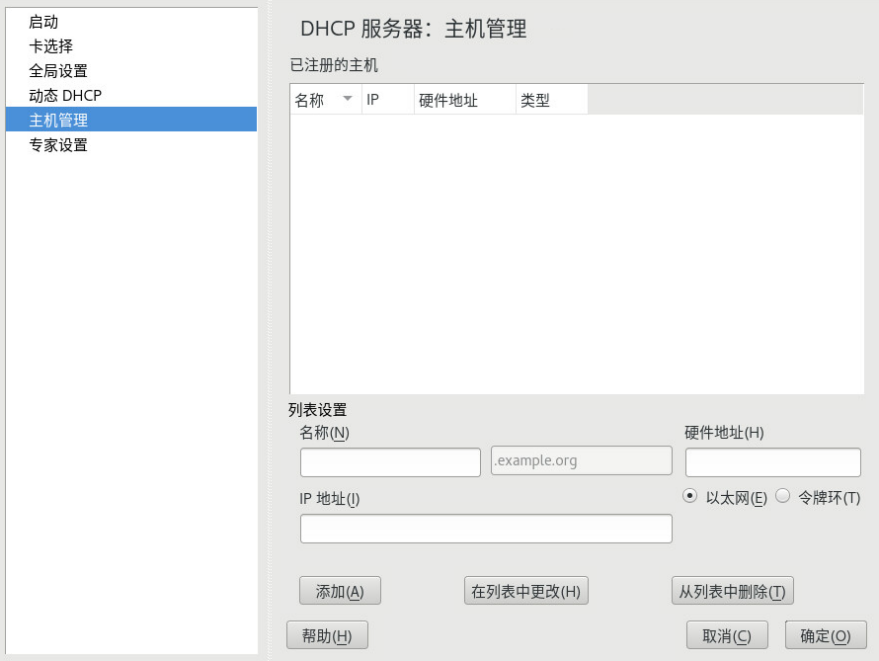


图 27.5 DHCP 服务器：主机管理

27.1.2 DHCP 服务器配置（专家）

除了前面介绍的配置方法外，还有一种专家配置方式，用于从方方面面精确调整 DHCP 服务器设置。单击启动对话框中的 DHCP 服务器专家配置（参见图 27.4 “DHCP 服务器：启动”）以启动专家配置。

Chroot 环境和声明

在第一个对话框中，选择启动 DHCP 服务器，使现有的配置可编辑。DHCP 服务器的行为的一个重要功能就是它能够在 chroot 环境（即 chroot jail）中运行，以便保证服务器主机的安全。如果 DHCP 服务器受到了外部攻击，则攻击者仍将被封锁在 chroot jail 中，从而可阻止其访问系统的其他部分。对话框的下部显示了一个树视图，列出了已经定义的声明。请使用添加、删除和编辑来修改它们。如果选择高级，就会进入其他专家对话框。参见图 27.6 “DHCP 服务器：Chroot Jail 和声明”。选择添加后，请定义要添加的声明的类型。使用高级，可查看服务器的日志文件、配置 TSIG 密钥管理以及根据 DHCP 服务器的设置调整防火墙的配置。

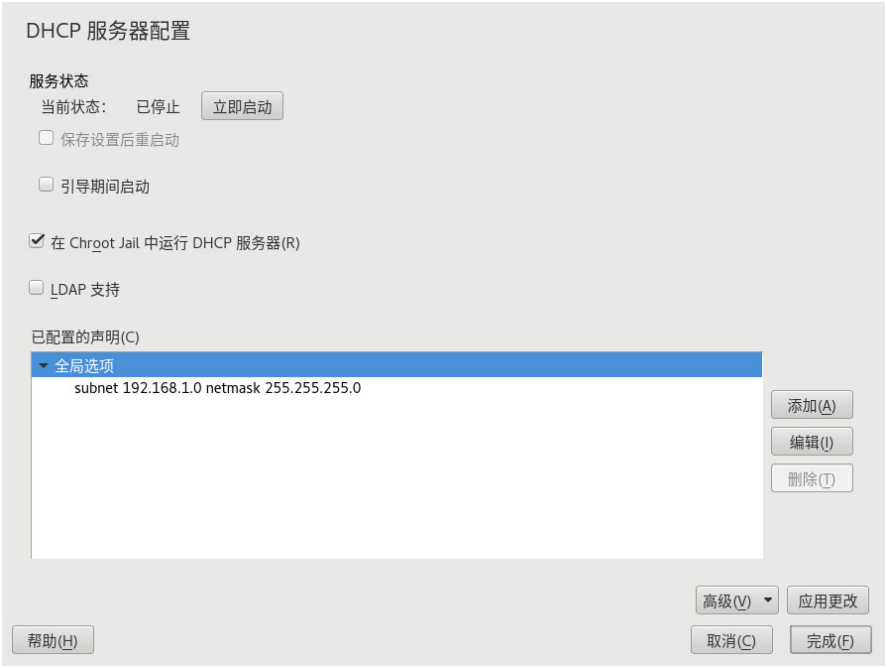


图 27.6 DHCP 服务器：CHROOT JAIL 和声明

选择声明类型

DHCP 服务器的全局选项由若干声明组成。使用此对话框可设置声明类型子网、主机、共享网络、组、地址池和类别。此示例显示了新子网的选择（请参见图 27.7 “DHCP 服务器：选择声明类型”）。

图 27.7 DHCP 服务器：选择声明类型

子网配置

此对话框用于指定新子网的 IP 地址和网络掩码。在对话框的中部，使用添加、编辑和删除修改所选子网的 DHCP 服务器启动选项。要为子网设置动态 DNS，请选择动态 DNS。

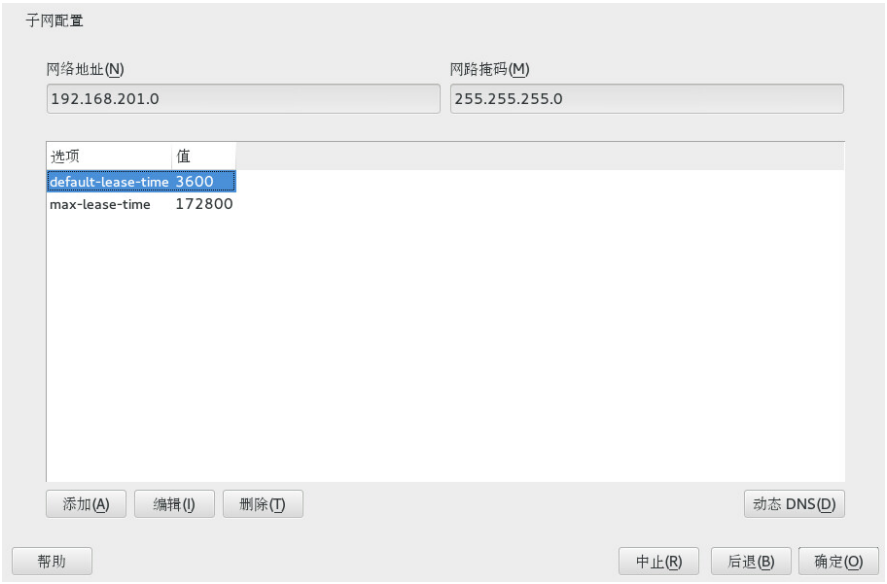


图 27.8 DHCP 服务器：配置子网

TSIG 密钥管理

如果在前面的对话框中选择了配置动态 DNS，现在就可以配置密钥管理来实现安全区域传送。选择确定将进入另一个对话框，在其中可以配置动态 DNS 的接口（请参见图 27.10 “DHCP 服务器：动态 DNS 的接口配置”）。

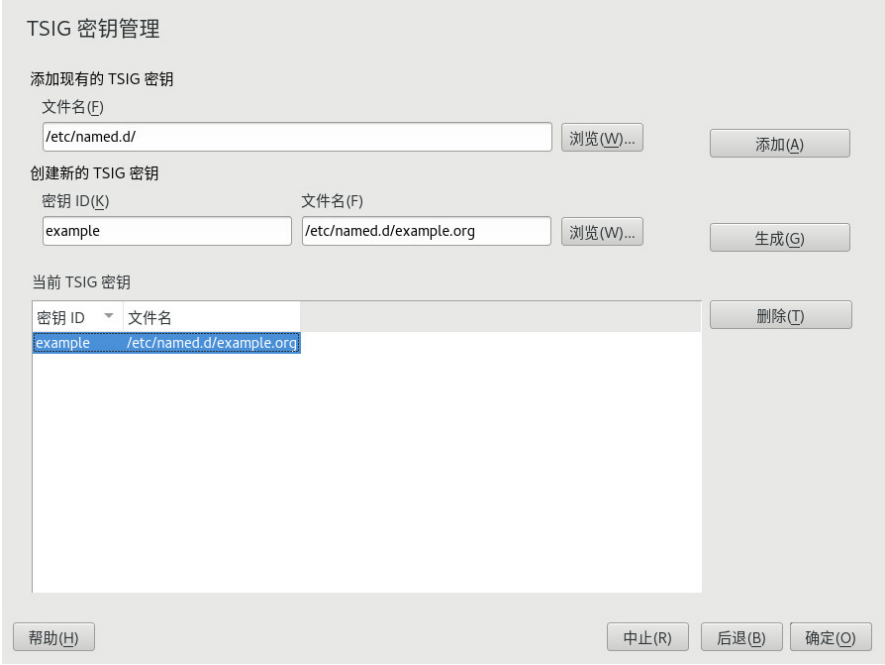


图 27.9 DHCP 服务器：TSIG 配置

动态 DNS：接口配置

通过选择为此子网启用动态 DNS，可以为子网激活动态 DNS。完成激活后，请使用下拉框来激活正向和反向区域的 TSIG 密钥，确保为 DNS 和 DHCP 服务器选择相同密钥。使用更新全局动态 DNS 设置，您可以根据动态 DNS 环境自动更新和调节全局 DHCP 服务器设置。最后需要定义每个动态 DNS 应更新哪些正向和反向区域，同时分别为两个区域指定主名称服务器的名称。选择确定返回子网配置对话框（请参见图 27.8 “DHCP 服务器：配置子网”）。再次选择确定将返回最初的专家配置对话框。

图 27.10 DHCP 服务器：动态 DNS 的接口配置

网络接口配置

要定义 DHCP 服务器应侦听的接口并调整防火墙配置，请从专家配置对话框中选择 高级 › 接口配置 。从所显示的接口列表中，选择一个或多个应由 DHCP 服务器处理的接口。如果希望使所有子网中的客户端都能够与服务器通讯，同时如果服务器主机也运行防火墙，则相应调整防火墙。要执行此操作，选择修改防火墙设置。然后，YaST 将按照新的条件调整 SuSEFirewall2 的规则（请参见图 27.11 “DHCP 服务器：网络接口和防火墙” ），之后您可以通过选择确定返回到原始对话框。

图 27.11 DHCP 服务器：网络接口和防火墙

在完成所有配置步骤后，选择确定关闭对话框。服务器现在将以新配置启动。

27.2 DHCP 软件包

DHCP 服务器和 DHCP 客户端都适用于 SUSE Linux Enterprise Server。可用的 DHCP 服务器是 `dhcpd`（由因特网系统联盟发布）。客户端提供了 `dhcp-client`（ISC 中也有提供）及 `wicked` 包附带的工具。

默认情况下，`wicked` 工具会连同 `wickedd-dhcp4` 和 `wickedd-dhcp6` 服务一起安装。系统每次引导时，会自动启动它们，以监视 DHCP 服务器。它们不需要配置文件来执行操作，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 `dhcp-client`，它是通过配置文件 `/etc/dhclient.conf` 和 `/etc/dhclient6.conf` 来控制的。

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。

例 27.1 配置文件 /ETC/DHCPD.CONF

```
default-lease-time 600;         # 10 minutes
max-lease-time 7200;            # 2  hours

option domain-name "example.com";
option domain-name-servers 192.168.1.116;
option broadcast-address 192.168.2.255;
option routers 192.168.2.1;
option subnet-mask 255.255.255.0;

subnet 192.168.2.0 netmask 255.255.255.0
 {
  range 192.168.2.10 192.168.2.20;
  range 192.168.2.100 192.168.2.200;
 }
```

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。

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端的默认秒数 (`default-lease-time`)，超过此时间就应申请续期。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。
- 使用 `option domain-name-servers` 项，最多可指定三个 DNS 服务器值，用来将 IP 地址解析为主机名，或将主机名解析为 IP 地址。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，请参见第 26 章 “域名系统”。
- 行 `option broadcast-address` 定义了请求客户端应该使用的广播地址。
- `option routers` 用于设置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）。通常情况下，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` 用于指定为客户端指派的网络掩码。

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端指派 IP 地址。在例 27.1 “配置文件 /etc/dhcpd.conf”中，可以为客户端分配 `192.168.2.10` 到 `192.168.2.20` 之间或 `192.168.2.100` 到 `192.168.2.200` 之间的任何地址。

编辑这些行后，便可以使用命令 `systemctl start dhcpd` 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` 来执行简单的语法检查。如果配置时出现任何意外问题（例如服务器由于错误而中止或在启动时不返回 `done`），您可以使用命令 `journalctl` 查询主系统日志中的信息，找出问题所在（有关更多信息，请参见第 15 章 “journalctl: 查询 systemd 日记”）。

在默认的 SUSE Linux Enterprise Server 系统上，出于安全考虑，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 `systemctl start dhcpd` 会自动复制这些文件。

27.3.1 具有固定 IP 地址的客户端

DHCP 可用来向特定客户端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。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，服务器需要在客户端之间重新分发这些地址。

要标识使用静态地址配置的客户端，dhcpd 会使用硬件地址（这是一个全球唯一的固定数字代码，其中包含 6 对八位组），用于标识所有网络设备（例如 `00:30:6E:08:EC:80`）。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 守护程序会将相同的一组数据指派到相应的客户端。

例 27.2 配置文件的添加项

```
host jupiter {
hardware ethernet 00:30:6E:08:EC:80;
fixed-address 192.168.2.100;
}
```

在第一行输入相应客户端的名称（`host HOSTNAME`，在本例中为 `jupiter`），在第二行输入 MAC 地址。在 Linux 主机上，使用命令 `ip link show` 后跟网络设备（例如 `eth0`）来查找 MAC 地址。输出应包含如下内容：

```
link/ether 00:30:6E:08:EC:80
```

在上面的示例中，会为网卡的 MAC 地址为 `00:30:6E:08:EC:80` 的客户端自动指派 IP 地址 `192.168.2.100` 和主机名 `jupiter`。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 `以太网`，

27.3.2 SUSE Linux Enterprise Server 版本

为了提高安全性，SUSE Linux Enterprise Server 版本的 ISC DHCP 服务器已应用了 Ari Edelkind 的 non-root/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` 下设置它们。为了确保 DHCP 日志记录功能即使在 syslog 守护程序重启动后仍然起作用，文件 `/etc/sysconfig/syslog` 中必须有附加项 `SYSLOGD_ADDITIONAL_SOCKET_DHCP`。

27.4 更多信息

有关 DHCP 的更多信息，请访问因特网系统联盟网站 (https://www.isc.org/dhcp/)。也可在 `dhcpd`、`dhcpd.conf`、`dhcpd.leases` 和 `dhcp-options` 手册页中获得相关信息。

28 通过 NFS 共享文件系统

网络文件系统 (NFS) 是允许访问服务器上的文件的协议，访问方式与访问本地文件非常相似。

28.1 概述

网络文件系统 (NFS) 是久经考验且广泛支持的标准化网络协议，它允许在单独的主机之间共享文件。

网络信息服务 (NIS) 可用于在网络中进行集中式用户管理。将 NFS 和 NIS 结合使用可通过文件和目录权限在网络中进行访问控制。NFS 与 NIS 一起使用时网络面向用户是透明的。

在默认配置中，NFS 完全信任网络，因此会信任连接到可信网络的任何计算机。在可通过物理方式访问 NFS 服务器所信任的任何网络的任何计算机上，任何具有管理员特权的用户都可以访问该服务器提供的所有文件。

在许多情况下，此安全性级别非常适于以下情形：所信任的网络是真正的专用网络，通常局限于单个计算机机柜或机房，并且无法进行未经授权的访问。将整个子网作为一个整体信任的其他情形限制较多，需要更精密的信任机制。为了满足这些情形的需要，NFS 使用 Kerberos 基础架构来支持各种安全性级别。Kerberos 需要 NFSv4（默认使用该协议）。有关细节，请参见《Security Guide》, 第 6 章 “Network Authentication with Kerberos”。

下面是 YaST 模块中使用的术语。

导出
: 由 NFS 服务器导出的目录，客户端可将其集成到系统中。

NFS 客户端
: NFS 客户端是通过网络文件系统协议使用来自 NFS 服务器的 NFS 服务的系统。TCP/IP 协议已集成到 Linux 内核中；无需再安装任何其他软件。

NFS 服务器
: NFS 服务器向客户端提供 NFS 服务。运行中的服务器依赖于以下守护程序：`nfsd` (worker)、`idmapd`（用于 NFSv4 的 ID 到名称映射，仅在某些场景下需要）、`statd`（文件锁定）和 `mountd`（装入请求）。

NFSv3

NFSv3 是版本 3 实施，支持客户端身份验证的“旧版”无状态 NFS。

NFSv4

NFSv4 是新的版本 4 实施，支持通过 kerberos 进行安全用户身份验证。NFSv4 只需要一个端口，因此，它比 NFSv3 更适合用于防火墙后的环境。

协议指定为 http://tools.ietf.org/html/rfc3530 。

pNFS

并行 NFS，属于 NFSv4 的一种协议扩展。任何 pNFS 客户端都可以直接访问 NFS 服务器上的数据。

重要：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。为避免超时，您需要一个有效的 DNS 系统。至少为了日志记录目的也应使用 DNS，因为 mountd 守护程序执行反向查找。

28.2 安装 NFS 服务器

默认不会安装 NFS 服务器。要使用 YaST 安装 NFS 服务器，请依次选择软件 › 软件管理、模式，然后启用服务器功能部分的文件服务器选项。按接受安装所需的包。

与 NIS 一样，NFS 也是一个客户端/服务器系统。但是，一台计算机可充当这两种角色：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）。

注意：在导出服务器上本地装入 NFS 卷

SUSE Linux Enterprise Server 上不支持在导出服务器本地装入 NFS 卷。

28.3 配置 NFS 服务器

可通过 YaST 配置 NFS 服务器，也可以手动配置它。NFS 还可与 Kerberos 结合来进行身份验证。

28.3.1 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，此服务器可将目录和文件导出到所有有权访问它的主机或导出到某个组的所有成员。因此，无需在每台主机本地安装应用程序，服务器也能提供应用程序。

要设置此类服务器，请继续执行以下步骤：

过程 28.1 设置 NFS 服务器

1. 启动 YaST 并选择网络服务 › NFS 服务器；请参见图 28.1 “NFS 服务器配置工具”。系统会提示您安装其他软件。

图 28.1 NFS 服务器配置工具

2. 激活启动单选按钮。

3. 如果防火墙在您的系统 (SuSEfirewall2) 中处于活动状态，请选中在防火墙中打开端口。YaST 会针对 NFS 服务器更改其配置，方法是启用 `nfs` 服务。

4. 选中是否启用 NFSv4。如果您停用 NFSv4，YaST 将只支持 NFSv3。有关启用 NFSv2 的信息，请参见注意：NFSv2。

- 如果选择 NFSv4，另外还请输入相应的 NFSv4 域名。`idmapd` 守护程序会使用此参数。Kerberos 设置需要该守护程序，当客户端无法处理数字用户名时，也需要使用该守护程序。如果您不运行 `idmapd` 或无任何特殊要求，请将它保留为 `localdomain`（默认值）。有关 `idmapd` 守护程序的详细信息，请参见 `/etc/idmapd.conf`。

5. 如果您需要安全访问服务器，请单击启用 GSS 安全性。先决条件是您的域中安装了 Kerberos 并且服务器和客户端都已采用 Kerberos 系统。单击下一步继续执行下一个配置对话框。

6. 单击对话框上半部分中的添加目录以导出您的目录。

7. 如果您尚未配置允许的主机，系统会自动弹出另一个对话框及相应的选项，供您输入客户端信息。输入主机通配符（通常您可以保留默认值不变）。
   可以为每个主机设置四类主机通配符：单主机（名称或 IP 地址）、网络组、通配符（如 `*` 表示所有计算机都能访问服务器）和 IP 网络。
   有关这些选项的更多信息，请参见 `exports` 手册页。

8. 单击完成以完成配置。

28.3.2 手动导出文件系统

NFS 导出服务的配置文件是 `/etc/exports` 和 `/etc/sysconfig/nfs`。如果 NFSv4 服务器配置包含经过 Kerberos 身份验证的 NFS，或者客户端不能使用数字用户名，则除了这些文件外，还需要 `/etc/idmapd.conf`。

要启动或重启动服务，请运行命令 `systemctl restart nfsserver`。此命令还会将 NFS 服务器必需的 RPC portmapper 重启动。

为确保 NFS 服务器始终在引导时启动，请运行 `sudo systemctl enable nfsserver`。

注意：NFSv4

NFSv4 是 SUSE Linux Enterprise Server 上可用的最新版 NFS 协议。现在，通过 NFSv4 导出所用的配置目录与通过 NFSv3 导出所用的目录相同。

在 SUSE Linux Enterprise Server 11 上，必须在 `/etc/exports` 中指定绑定装入。该设置仍然受支持，但现在已弃用。

`/etc/exports`

`/etc/exports` 文件包含项列表。每个条目表示共享的目录以及共享的方式。`/etc/exports` 中的条目通常包含：

```
/SHARED/DIRECTORY   HOST(OPTION_LIST)
```

例如：

```
/export/data   192.168.1.2(rw,sync)
```

在此，使用 IP 地址 `192.168.1.2` 标识允许的客户端。您可以使用主机名、表示一组主机的通配符（`*.abc.com`、`*` 等）或网络组（`@my-hosts`）。
有关所有选项及其含义的详细说明，请参见 `exports` 的手册页（`man exports`）。
如果您在 NFS 服务器运行时修改了 `/etc/exports`，则需使用 `sudo systemctl restart nfsserver` 命令重启动 NFS 服务器，以使更改生效。

`/etc/sysconfig/nfs`

`/etc/sysconfig/nfs` 文件包含一些决定 NFSv4 服务器守护程序行为的参数。请务必将参数 `NFS4_SUPPORT` 设置为 `yes`（默认值）。`NFS4_SUPPORT` 决定 NFS 服务器是否支持 NFSv4 导出和客户端。
如果您在 NFS 服务器运行时修改了 `/etc/sysconfig/nfs`，则需使用 `sudo systemctl restart nfsserver` 命令重启动 NFS 服务器，以使更改生效。

提示：装入选项

在 SUSE Linux Enterprise Server 11 上，必须在 `/etc/exports` 中指定 `--bind` 装入。该设置仍然受支持，但现在已弃用。现在，通过 NFSv4 导出所用的配置目录与通过 NFSv3 导出所用的目录相同。

注意：NFSv2

如果 NFS 客户端仍依赖于 NFSv2，请在服务器的 `/etc/sysconfig/nfs` 中设置以下几项启用该协议：

```
NFSD_OPTIONS="-V2"
MOUNTD_OPTIONS="-V2"
```

重启动服务后，请使用以下命令检查版本 2 是否可用：

```
tux > cat /proc/fs/nfsd/versions
+2 +3 +4 +4.1 -4.2
```

`/etc/idmapd.conf`

从 SLE 12 SP1 开始，仅当使用 Kerberos 身份验证或客户端不能使用数字用户名时，才需要 `idmapd` 守护程序。自 Linux 内核 2.6.39 起，Linux 客户端可以使用数字用户名。`idmapd` 守护程序会将发送到服务器的 NFSv4 请求进行名称到 ID 的映射，并答复客户端。

如果需要，`idmapd` 需在 NFSv4 服务器上运行。客户端上的名称到 ID 映射将由以下包提供的 `nfsidmap` 来执行：`nfs-client`。

对于可能使用 NFS 来共享文件系统的计算机，请确保以统一的方式在这些计算机间为用户指定用户名和 ID (UID)。这可以使用 NIS、LDAP 或域中的任何统一的域身份验证机制来实现。

对于客户端和服务器，必须在 `/etc/idmapd.conf` 文件中将参数 `Domain` 设为相同值。如果您不确定，请在服务器和客户端文件中将域保留为 `localdomain`。配置文件样本如下：

```
[General]
Verbosity = 0
Pipefs-Directory = /var/lib/nfs/rpc_pipefs
Domain = localdomain

[Mapping]
Nobody-User = nobody
Nobody-Group = nobody
```

要启动 `idmapd` 守护程序，请运行 `systemctl start nfs-idmapd`。如果您在守护程序运行时修改了 `/etc/idmapd.conf`，则需使用 `systemctl start nfs-idmapd` 命令重启动守护程序，以使更改生效。

有关更多信息，请参见 `idmapd` 和 `idmapd.conf` 的手册页（`man idmapd` 和 `man idmapd.conf`）。

28.3.3 采用 Kerberos 的 NFS

要对 NFS 使用 Kerberos 身份验证，必须启用通用安全服务 (GSS)。在初始 YaST NFS 服务器对话框中选择启用 GSS 安全。必须具有一个有效的 Kerberos 服务器才能使用此功能。YaST 不会设置服务器，而只使用所提供的功能。如果希望使用 Kerberos 进行身份验证，则除了 YaST 配置外，还必须首先至少完成以下步骤，才能运行 NFS 配置：

1. 请确保服务器和客户端都在同一 Kerberos 域中。它们必须访问相同的 KDC（密钥分发中心）服务器并共享其 `krb5.keytab` 文件（在任何计算机上的默认位置是 `/etc/krb5.keytab`）。有关 Kerberos 的更多信息，请参见《Security Guide》，第 6 章 “Network Authentication with Kerberos”。
2. 在客户端上运行 `systemctl start rpc-gssd.service` 启动 gssd 服务。
3. 在服务器上运行 `systemctl start rpc-svcgssd.service` 启动 svcgssd 服务。

要进行 Kerberos 身份验证，也需要在服务器上运行 `idmapd` 守护程序。有关详细信息，请参见 `/etc/idmapd.conf`。

有关配置采用 Kerberos 的 NFS 的更多信息，请参见第 28.5 节 “更多信息” 中的链接。

28.4 配置客户端

要将主机配置为 NFS 客户端，无需安装其他软件。将默认安装所有需要的包。

28.4.1 使用 Yast 导入文件系统

授权用户可以用 YaST NFS 客户端模块从 NFS 服务器将 NFS 目录装入本地文件树。按如下所示继续：

过程 28.2 导入 NFS 目录

1. 启动 YaST NFS 客户端模块。

2. 单击 NFS 共享选项卡中的添加。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。
3. 使用 NFSv4 时，在 NFS 设置选项卡中选择启用 NFSv4。另外，NFSv4 域名必须包含 NFSv4 服务器所用的相同值。默认域为 `localdomain`。
4. 要对 NFS 使用 Kerberos 身份验证，必须启用 GSS 安全性。选择启用 GSS 安全。
5. 若要使用防火墙并允许从远程计算机访问服务，请启用 NFS 设置选项卡中的打开防火墙中的端口。防火墙状态将显示在复选框旁边。
6. 单击确定保存更改。

配置写入 `/etc/fstab`，并将装入指定的文件系统。当您稍后启动 YaST 配置客户端时，它还将读取此文件中的现有配置。

提示：NFS 用作根文件系统

在通过网络以 NFS 共享形式装入根分区的（无磁盘）系统中，配置可供访问 NFS 共享的网络设备时需保持谨慎。

关闭或重引导系统时，默认的处理顺序是关闭网络连接，然后卸载根分区。对于 NFS 根分区，这种顺序会产生问题，因为在尚未激活与 NFS 共享的网络连接的情况下，根分区无法完全卸载。为防止系统停用相关的网络设备，请按第 16.4.1.2.5 节 “激活网络设备”中所述打开网络设备配置选项卡，然后在设备激活窗格中选择通过 NFSroot。

28.4.2 手动导入文件系统

手动从 NFS 服务器导入文件系统的先决条件是运行 RPC 端口映射器。`nfs` 服务负责正确启动该程序；因此，请以 `root` 身份输入 `systemctl start nfs` 来启动该服务。然后就可以使用 `mount` 将远程文件系统像本地分区那样装入文件系统中：

```
tux > sudo mount HOST:REMOTE-PATHLOCAL-PATH
```

例如，要从 `nfs.example.com` 计算机导入用户目录，请使用：

```
tux > sudo mount nfs.example.com:/home /home
```

28.4.2.1 使用自动装入服务

autofs 守护程序可用于自动装入远程文件系统。请在 `/etc/auto.master` 文件中添加以下条目：

```
/nfsmounts /etc/auto.nfs
```

如果 `auto.nfs` 文件正确填充，`/nfsmounts` 目录将作为客户端上所有 NFS 装入的 root 目录。选择 `auto.nfs` 这个名称是为了方便起见，您可以选择任何名称。在 `auto.nfs` 中为所有 NFS 装入添加条目，如下所示：

```
localdata -fstype=nfs server1:/data
nfs4mount -fstype=nfs4 server2:/
```

以 `root` 身份运行 `systemctl start autofs` 来激活该设置。对于此示例，`/nfsmounts/localdata`，`server1` 的 `/data` 目录将通过 NFS 装入，`server2` 的 `/nfsmounts/nfs4mount` 将通过 NFSv4 装入。

如果在 autofs 服务正在运行时编辑了 `/etc/auto.master` 文件，则必须使用 `systemctl restart autofs` 重启动自动装载器才能使更改生效。

28.4.2.2 手动编辑 /etc/fstab

通常，`/etc/fstab` 中的 NFSv3 装入项如下：

```
nfs.example.com:/data /local/path nfs rw,noauto 0 0
```

对于 NFSv4 装入，请在第三列中使用 `nfs4` 而不是 `nfs`：

```
nfs.example.com:/data /local/pathv4 nfs4 rw,noauto 0 0
```

`noauto` 选项可禁止在启动时自动装入文件系统。如果您要手动安装各文件系统，可以缩短只指定安装点的安装命令：

```
tux > sudo mount /local/path
```

注意：启动时装入

如果您没有输入 `noauto` 选项，系统的 init 脚本将在启动时处理这些文件系统的装入。

28.4.3 并行 NFS (pNFS)

NFS 是最老的协议之一，开发于上世纪八十年代。因此，如果您要共享小文件，NFS 通常能够满足需求。但是，当您要传送大文件或有大量的客户端要访问数据时，NFS 服务器会成为瓶颈，严重影响系统性能。这是因为文件迅速变大，而以太网的相对速度没有完全跟上这一变化。

当您向普通 NFS 服务器请求文件时，服务器会查找文件元数据、收集所有数据并通过网络将数据传送到您的客户端。但是，无论文件的大小如何，性能瓶颈都会凸显出来：

- 如果是小文件，则大部分时间都花在收集元数据上。
- 如果是大文件，则大部分时间花在将数据从服务器传送到客户端上。

pNFS 或并行 NFS 则突破了此种限制，因为它将文件系统元数据从数据位置分离出来。因此，pNFS 需要两类服务器：

- 一个元数据或控制服务器，用于处理所有非数据通讯
- 一个或多个储存服务器，用于存放数据

元数据和储存服务器组成单独一个逻辑 NFS 服务器。当客户端要读取或写入时，元数据服务器会告诉 NFSv4 客户端使用哪个储存服务器访问文件块。客户端可以直接访问该服务器上的数据。

SUSE Linux Enterprise Server 仅在客户端上支持 pNFS。

28.4.3.1 使用 YaST 配置 pNFS 客户端

请执行过程 28.2 “导入 NFS 目录”中所述的步骤，但选中 pNFS (v4.1) 复选框以及可选的 NFSv4 共享。YaST 会执行所有必需的步骤，并且会在文件 `/etc/exports` 中写入所有必要选项。

28.4.3.2 手动配置 pNFS 客户端

请参阅第 28.4.2 节 “手动导入文件系统”着手配置。大多数配置通过 NFSv4 服务器完成。对于 pNFS，唯一的区别是将 `minorversion` 选项和元数据服务器 `MDS_服务器` 添加到您的 `mount` 命令：

```
tux > sudo mount -t nfs4 -o minorversion=1 MDS_SERVER MOUNTPOINT
```

为方便调试，请更改 `/proc` 文件系统中的值：

```
tux > sudo echo 32767 > /proc/sys/sunrpc/nfsd_debug
tux > sudo echo 32767 > /proc/sys/sunrpc/nfs_debug
```

28.5 更多信息

除了 `exports`、`nfs` 和 `mount` 的手册页外，还可在 `/usr/share/doc/packages/nfsidmap/README` 中找到关于配置 NFS 服务器和客户端的信息。有关更多联机文档，请参见以下网站：

- 在 SourceForge (http://nfs.sourceforge.net/) 上联机查找详细的技术文档。
- 关于设置采用 Kerberos 的 NFS 的描述，请参见 NFS Version 4 Open Source Reference Implementation (http://www.citi.umich.edu/projects/nfsv4/linux/krb5-setup.html)。
- 如果您对 NFSv4 有疑问，请参考 Linux NFSv4 FAQ (http://www.citi.umich.edu/projects/nfsv4/linux/faq/)（Linux NFSv4 常见问题）。

29 Samba

使用 Samba 可以将 Unix 计算机配置为 macOS、Windows 和 OS/2 计算机的文件和打印服务器。Samba 已经发展成为一个功能完备且相当复杂的产品。使用 YaST 或手动编辑配置文件来配置 Samba。

29.1 术语

以下是 Samba 文档和 YaST 模块中使用的一些术语。

SMB 协议

Samba 使用基于 NetBIOS 服务的 SMB（服务器消息块）协议。Microsoft 发布该协议以便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端上安装 TCP/IP 协议。

提示：IBM Z：NetBIOS 支持

IBM Z 仅支持基于 TCP/IP 的 SMB。这些系统上不提供 NetBIOS 支持。

CIFS 协议

（常用因特网文件系统）协议是 Samba 支持的另一种协议。CIFS 定义网络中使用的标准远程文件系统访问协议，使用户组能够一起工作并在网络中共享文档。

NetBIOS

NetBIOS 是为用于提供名称服务的计算机之间进行通讯而设计的软件接口 (API)。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均可以保留所需数量的名称，前提是这些名称均未使用。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，建议您使用与 DNS 主机名对应的名称，或本机使用 DNS。Samba 默认采用这种方式。

Samba 服务器

Samba 服务器向客户端提供 SMB/CIFS 服务和 NetBIOS over IP 命名服务。对于 Linux，Samba 服务器有三个守护程序：smbd 用于 SMB/CIFS 服务，nmbd 用于命名服务，winbind 用于身份验证。

Samba 客户端

Samba 客户端是一种能够通过 SMB 协议从 Samba 服务器使用 Samba 服务的系统。常见操作系统（例如 Windows 和 macOS）都支持 SMB 协议。必须在所有计算机上安装 TCP/IP 协议。Samba 提供适用于多种不同类型 UNIX 的客户端。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。不需要对 Samba 客户端运行任何守护程序。

共享

SMB 服务器通过共享为其客户端提供资源。共享就是服务器上的打印机和目录及其子目录。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 — 不一定是导出目录的名称。也可以为打印机指派一个名称。客户端可以根据打印机的名称来访问打印机。

DC

域控制器 (DC) 是处理域中帐户的服务器。为了复制数据，一个域中可有更多域控制器可用。

29.2 安装 Samba 服务器

要安装 Samba 服务器，请启动 YaST 并选择软件 › 软件管理。选择视图 › 模式，然后选择文件服务器。确认已安装完成安装进程所需的包。

29.3 启动和停止 Samba

（引导时）可以自动启动或停止 Samba 服务器，或者手动执行这两个操作。启动和停止策略是第 29.4.1 节 “使用 YaST 配置 Samba 服务器”中所述的 YaST Samba 服务器配置的一部分。

在命令行中使用 `systemctl stop smb nmb` 可停止 Samba 所需的服务，使用 `systemctl start nmb smb` 可启动它们。`smb` 服务会根据需要处理 `winbind`。

提示：winbind

`winbind` 是一项独立服务，同样也是以单独的 `samba-winbind` 包提供。

29.4 配置 Samba 服务器

SUSE® Linux Enterprise Server 中的 Samba 服务器可通过两种不同方式进行配置：用 YaST 或手动方式。手工配置可提供更详细的信息，但没有 YaST GUI 方便。

29.4.1 使用 YaST 配置 Samba 服务器

要配置 Samba 服务器，请启动 YaST 并选择网络服务 › Samba 服务器。

29.4.1.1 初始 Samba 配置

第一次启动此模块时，系统会启动 Samba 安装对话框，提示您做一些涉及服务器管理的基本设置。配置结束时，系统会提示您输入 Samba 管理员口令（Samba root 口令）。以后启动时，会显示 Samba 配置对话框。

Samba 安装对话框包括两个步骤和详细设置（可选）：

工作组名或域名

在工作组名或域名中选择一个现有名称或输入一个新的名称，然后单击下一步。

Samba 服务器类型

在下一步中，指定服务器是应该充当主域控制器 (PDC)、备用域服务器 (BDC) 还是不充当域控制器。按下一步继续。

如果不想再继续详细的服务器配置，请单击确定确认。然后在最后的弹出框中，设置 Samba root 口令。

稍后可以在 Samba 配置对话框的启动、共享、身份、可信域和 LDAP 设置选项卡中更改所有设置。

29.4.1.2 高级 Samba 配置

在 Samba 服务器模块第一次启动时，Samba 配置对话框会在两个初始步骤后立即显示，如第 29.4.1.1 节 “初始 Samba 配置”所述。使用它调整您的 Samba 服务器配置。

编辑配置之后，单击确定保存设置。

29.4.1.2.1 启动服务器

在启动选项卡中，配置 Samba 服务器的启动。若想在每次系统引导时启动服务，请选择引导时。要激活手动启动，请选择手动。有关启动 Samba 服务器的更多信息，请参见第 29.3 节 “启动和停止 Samba”。

在此选项卡中，还可以打开防火墙中的端口。为此应选择打开防火墙中的端口。如果有多个网络接口，则请通过单击防火墙细节、选择接口并单击确定来为 Samba 服务选择网络接口。

29.4.1.2.2 共享

在共享选项卡中，确定要激活的 Samba 共享。存在一些预定义的共享，例如主页和打印机。使用切换状态可在活动和不活动之间进行切换。单击添加可添加新共享，单击删除可删除选中共享。

允许用户共享目录使允许的组中的组成员可以与其他用户共享他们拥有的目录。例如，`users` 用于本地范围，`DOMAIN\Users` 用于域范围。该用户必须还确保文件系统权限允许访问。最大共享数可限制可以创建的共享的总数。要允许访问用户共享而无需身份验证，请启用允许来宾访问。

29.4.1.2.3 身份

在身份选项卡中，确定与主机关联的域（基本设置）以及是否在网络中使用备用主机名（NetBIOS 主机名）。可以使用 Microsoft Windows Internet Name Service (WINS) 进行名称解析。在这种情况下，激活使用 WINS 进行主机名解析，并确定是否通过 DHCP 检索 WINS 服务器。要设置专家全局设置或设置用户身份验证源，例如 LDAP 而不是 TDB 数据库，请单击高级设置。

29.4.1.2.4 可信域(T)

要使其他域的用户能够访问您的域，在可信域选项卡中进行适当的设置。要添加新域，请单击添加。要除去所选的域，请单击删除。

29.4.1.2.5 LDAP 设置

在选项卡 LDAP 设置中，您可以确定要用于身份验证的 LDAP 服务器。要测试到 LDAP 服务器的连接，请单击测试连接。要设置专家 LDAP 设置或使用默认值，请单击高级设置。

有关 LDAP 配置的更多信息，请参见《Security Guide》, 第 5 章 “LDAP—A Directory Service”。

29.4.2 手动配置服务器

如果想将 Samba 用作服务器，请安装 `samba`。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。以下默认部分包含各个文件和打印机共享：

- [homes]
- [profiles]
- [users]
- [groups]
- [printers]
- [print$]

通过此方法，您可以设置不同的共享选项，或在 [global] 部分设置全局共享选项，这使得配置文件更容易理解。

29.4.2.1 global 部分

应该修改 [global] 部分的以下参数来满足网络设置的要求，以使其他计算机能在 Windows 环境中通过 SMB 访问 Samba 服务器。

workgroup = WORKGROUP

此行将 Samba 服务器指派到工作组。将 WORKGROUP 替换为您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，除非此名称已指派给网络中的其他计算机。如果 DNS 名称不可用，请使用 netbiosname=MYNAME 设置服务器名称。有关此参数的更多细节，请参见 smb.conf 手册页。

os level = 20

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 网络因 Samba 服务器配置不当而中断，应选择非常低的值，如 2。有关此主题的更多信息，可以在《Samba 3 Howto》（Samba 3 操作指南）的“Network Browsing”（网络浏览）一章中找到；有关《Samba 3 Howto》的更多信息，请参见第 29.9 节 “更多信息”。

如果网络中没有其他 SMB 服务器（如 Windows 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。

如果您的各 Windows 计算机连接到不同的子网，而它们又需要看到彼此，您必须设置一个 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。

29.4.2.2 共享

以下示例描述了如何使 CD-ROM 驱动器和用户目录 (`homes`) 对 SMB 客户端可用。

[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。删除第一列中的分号，以便与 Samba 共享 CD-ROM 驱动器。

例 29.1 CD-ROM 共享

```
[cdrom]
       comment = Linux CD-ROM
       path = /media/cdrom
       locking = No
```

`[cdrom]` 和 `comment`

`[cdrom]` 部分项是网络上的所有 SMB 客户端均可看到的共享的名称。可以添加一个附加 `comment` 来进一步描述此共享。

`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。

例 29.2 [HOMES] 共享

```
[homes]
        comment = Home Directories
        valid users = %S
        browseable = No
        read only = No
        inherit acls = Yes
```

[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。生成的共享名称就是用户名。

`valid users = %S`

成功建立连接后，会使用具体的共享名称替换 `%S`。对于 `[homes]` 共享，此名称始终是用户名。这样就可以将用户的共享访问权仅限制于此用户。

`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。

警告：不要与 Samba 共享 NFS 载具

与 Samba 共享 `NFS` 载具可能导致数据丢失，并且不支持这样做。请直接在文件服务器上安装 Samba，或者考虑使用替代方式，例如 `iSCSI`。

29.4.2.3 安全性级别

要提高安全性，可以使用口令来保护每个共享访问。SMB 提供以下检查许可权限的方式：

用户级安全性 (`security = user`)

此变体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。

ADS 级安全性 (`security = ADS`)

在该模式中，Samba 将在 Active Directory 环境中充当域成员。要在该模式中工作，运行 Samba 的计算机需要安装并配置 Kerberos。必须使用 Samba 将该计算机加入到 ADS 领域。此步骤可通过使用 YaST Windows 域成员资格模块完成。

域级安全性 (`security = domain`)

仅当计算机已加入到 Windows NT 域中时，该模式才能正常工作。Samba 将尝试验证用户名和口令，方法是将其传递到 Windows NT 主域控制器或备份域控制器。与 Windows NT 服务器所采用的方式相同。它期望将加密口令参数设置为 `yes`。

选择共享、用户或域级安全性适用于整个服务器。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。

有关此主题的更多信息，可以在《Samba 3 操作指南》中找到。对于一个系统上的多个服务器，应注意选项 `interfaces` 和 `bind interfaces only`。

29.5 配置客户端

客户端只能通过 TCP/IP 访问 Samba 服务器。NetBEUI 和通过 IPX 的 NetBIOS 不能与 Samba 共用。

29.5.1 使用 YaST 配置 Samba 客户端

配置 Samba 客户端来访问 Samba 或 Windows 服务器上的资源（文件或打印机）。在 网络服务 › Windows 域成员资格对话框中输入 NT 或 Active Directory 域或工作组。如果激活将 SMB 信息也用于 Linux 身份验证，则用户身份验证将在 Samba、NT 或 Kerberos 服务器上运行。

单击专家设置获取高级配置选项。例如，使用装入服务器目录表启用自动装入服务器用户主目录和身份验证。这样用户就能访问他们在 CIFS 上的主目录。有关细节，请参见 `pam_mount` 手册页。

完成所有设置后，请确认对话框以完成配置。

29.6 将 Samba 用作登录服务器

在主要由 Windows 客户端组成的网络中，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。在基于 Windows 的网络中，此任务由主域控制器 (PDC) 来处理。您可以使用配置为 PDC 的 Windows NT 服务器，但是此任务也可以借助 Samba 服务器来完成。例 29.3 “smb.conf 中的 global 部分”中显示了必须在 `smb.conf` 的 `[global]` 部分设置的项。

例 29.3 SMB.CONF 中的 GLOBAL 部分

```
[global]
    workgroup = WORKGROUP
    domain logons = Yes
    domain master = Yes
```

需要准备适合 Windows 加密格式的用户帐户和口令。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 域概念要求的域帐户：

```
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。

```
add machine script = /usr/sbin/useradd -g nogroup -c "NT Machine Account" \
-s /bin/false %m\$
```

要确保 Samba 能够正确执行此脚本，请选择具有必需的管理员权限的 Samba 用户，并将其添加到 `ntadmin` 组中。然后可以使用以下命令为属于此 Linux 组的所有用户指派 `Domain Admin` 状态：

```
net groupmap add ntgroup="Domain Admins" unixgroup=ntadmin
```

29.7 带有 Active Directory 的网络中的 Samba 服务器

如果您同时运行 Linux 服务器和 Windows 服务器，则可以构建两个独立的身份验证系统和网络，或者将服务器连接到使用一个中央身份验证系统的网络。由于 Samba 可以与 Active Directory 域协作，因此您可以将 SUSE Linux Enterprise Server 加入 Active Directory（AD）。要加入到 AD 域，请执行以下操作：

1. 作为 `root` 登录并启动 YaST。
2. 启动 网络服务 › Windows 域成员资格。
3. 在 Windows 域成员资格屏幕上的域或工作组中输入要加入的域。

图 29.1 确定 WINDOWS 域成员资格

4. 选中同时使用 SMB 信息进行 Linux 身份验证，以在服务器上使用 SMB 源进行 Linux 身份验证。
5. 单击确定并在提示时确认域连接。
6. 在 AD 服务器上提供 Windows Administrator 的口令，并单击确定。

现在您的服务器已经设置了从 Active Directory 域控制器获取认证数据。

提示：标识映射

在有多个 Samba 服务器的环境中，将不会采用一致的方式创建 UID 和 GID。指派给用户的 UID 将取决于用户首次登录的顺序，而这会导致在服务器间产生 UID 冲突。要解决此问题，您需要使用身份映射。有关详细信息，请参见https://www.samba.org/samba/docs/man/Samba-HOWTO-Collection/idmapper.html。

29.8 高级主题

本节介绍用于管理 Samba 套件中客户端部分与服务器部分的高级方法。

29.8.1 Btrfs 上的透明文件压缩

Samba 允许客户端针对 Btrfs 文件系统中的共享远程操作文件与目录压缩标志。Windows 资源管理器可让用户通过文件 › 属性 › 高级对话框来标记要进行透明压缩的文件/目录：

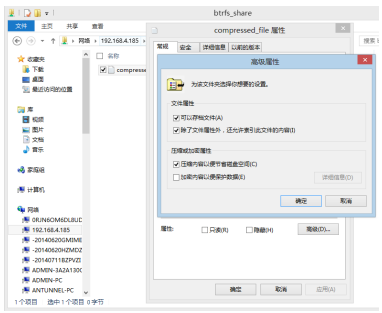


图 29.2 WINDOWS 资源管理器高级属性对话框

带有压缩标志的文件将以透明方式进行压缩，当用户访问或修改这些文件时，底层文件系统会将其解压缩。这通常可以节省储存容量，不过，在访问文件时会造成额外的 CPU 开销。除非新文件和目录是使用 FILE_NO_COMPRESSION 选项创建的，否则，它们将继承父目录的压缩标志。

Windows 资源管理器以不同的显示方式区分压缩文件和未压缩文件：

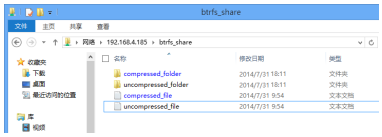


图 29.3 列有压缩文件的 WINDOWS 资源管理器目录

要启用 Samba 共享压缩，您可以将以下内容

```
vfs objects = btrfs
```

手动添加到 `/etc/samba/smb.conf` 中的共享配置，或者使用 YaST：网络服务 › Samba 服务器 › 添加，然后选中使用 Btrfs 功能。

有关 Btrfs 上的压缩功能的一般概述，请参见《储存管理指南》，第 1 章 “Linux 中的文件系统的概述”，第 1.2.2.1 节 “装入压缩的 Btrfs 文件系统”。

29.8.2 快照

快照也称为阴影副本，是指某个文件系统子卷在特定时间点的状态副本。在 Linux 中，可以使用 Snapper 工具来管理这些快照。Btrfs 文件系统或精简配置的 LVM 卷支持快照。Samba 套件支持通过服务器端和客户端的 FSRVP 协议管理远程快照。

29.8.2.1 先前版本

Samba 服务器上的快照可以作为文件或目录的先前版本公开给远程 Windows 客户端。

要在 Samba 服务器上启用快照，必须符合以下条件：

- SMB 网络共享位于 Btrfs 子卷上。
- SMB 网络共享路径中包含相关的 snapper 配置文件。可以使用以下命令创建 snapper 文件

  ```
  snapper -c <cfg_name> create-config /path/to/share
  ```

  有关 snapper 的更多信息，请参见第 7 章 “通过 Snapper 进行系统恢复和快照管理”。
- 必须允许相关用户访问快照目录树。有关更多信息，请参见 vfs_snapper 手册页 (`man 8 vfs_snapper`) 的 PERMISSIONS (权限) 部分。

要支持远程快照，需要修改 `/etc/samba/smb.conf` 文件。要完成此操作，您可以选择 YaST › 网络服务 › Samba 服务器，或者使用以下命令手动增强相关的共享部分

```
vfs objects = snapper
```

请注意，只有在重启动 Samba 服务后，对 `smb.conf` 所做的手动更改才能生效：

```
systemctl restart nmb smb
```

图 29.4 在启用快照的情况下添加新的 SAMBA 共享

经过配置后，可以通过 Windows 资源管理器中某个文件或目录的以前的版本选项卡访问 snapper 为 Samba 共享路径创建的快照。

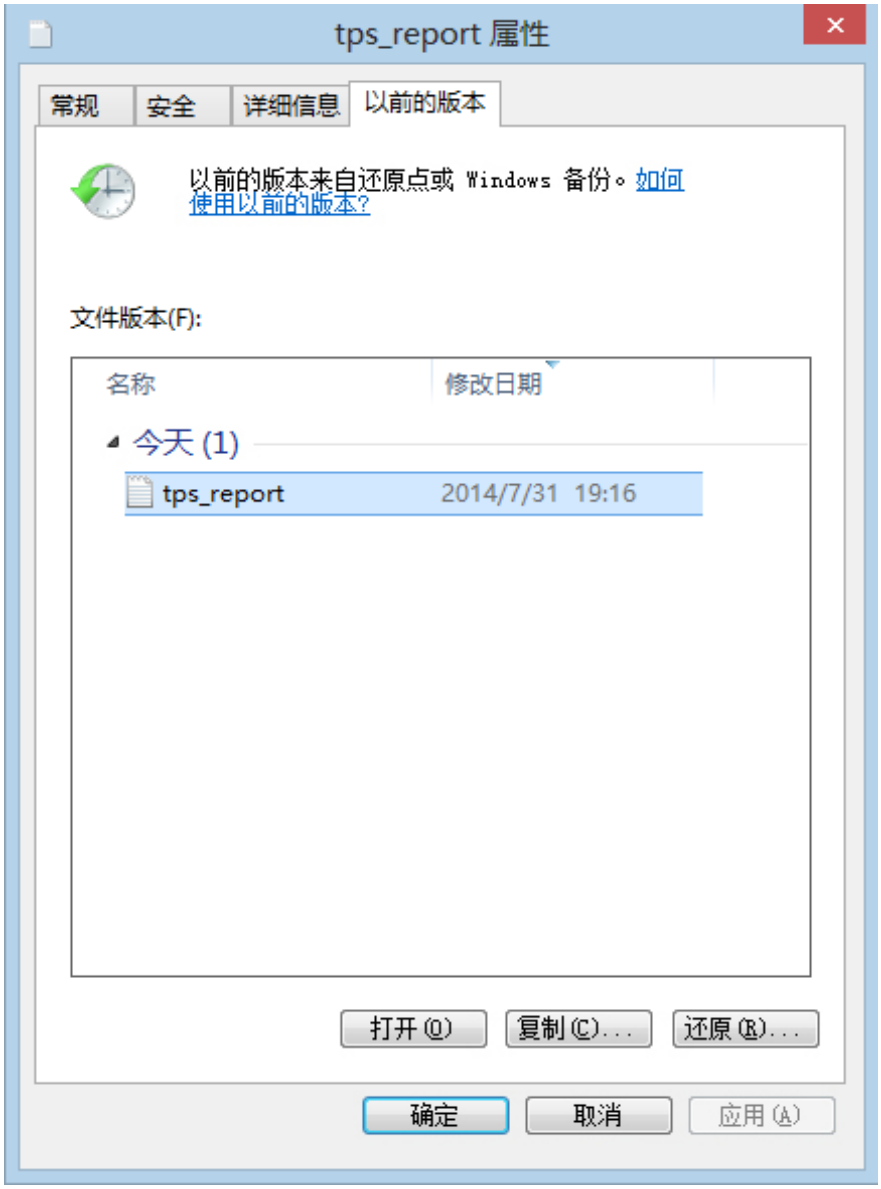


图 29.5 WINDOWS 资源管理器中的以前的版本选项卡

29.8.2.2 远程共享快照

默认情况下，只能在 Samba 服务器本地通过 snapper 命令行实用程序或者使用 snapper 时间轴功能来创建和删除快照。

可将 Samba 配置为使用文件服务器远程 VSS 协议 (FSRVP) 处理远程主机发出的共享快照创建和删除请求。

除了第 29.8.2.1 节 “先前版本”中所述的配置和先决条件以外，还需要在 `/etc/samba/smb.conf` 中指定以下全局配置：

```
[global]
rpc_daemon:fssd = fork
registry shares = yes
include = registry
```

然后，FSRVP 客户端（包括 Samba 的 `rpcclient` 以及 Windows Server 2012 `DiskShadow.exe`）便可以指示 Samba 为指定的共享创建或删除快照，并将该快照公开为新共享。

29.8.2.3 使用 `rpcclient` 从 Linux 远程管理快照

`samba-client` 包中有一个 FSRVP 客户端，它可以远程请求 Windows/Samba 服务器创建并公开指定共享的快照。然后，您可以使用 SUSE Linux Enterprise Server 中的现有工具装入公开的共享并备份其文件。向服务器发出的请求将使用 `rpcclient` 二进制文件发送。

例 29.4 使用 `rpcclient` 请求 WINDOWS SERVER 2012 共享快照

以 `EXAMPLE` 域中管理员的身份连接到 `win-server.example.com` 服务器：

```
# rpcclient -U 'EXAMPLE\Administrator' ncacn_np:win-
server.example.com[ndr64,sign]
Enter EXAMPLE/Administrator's password:
```

检查 SMB 共享是否对于 `rpcclient` 可见：

```
rpcclient $> netshareenum
netname: windows_server_2012_share
remark:
path:   C:\Shares\windows_server_2012_share
password:       (null)
```

检查 SMB 共享是否支持创建快照：

```
rpcclient $> fss_is_path_sup windows_server_2012_share \
UNC \\WIN-SERVER\windows_server_2012_share\ supports shadow copy requests
```

请求创建共享快照：

```
rpcclient $> fss_create_expose backup ro windows_server_2012_share
13fe880e-e232-493d-87e9-402f21019fb6: shadow-copy set created
13fe880e-e232-493d-87e9-402f21019fb6(1c26544e-8251-445f-be89-d1e0a3938777):
 \
\\WIN-SERVER\windows_server_2012_share\ shadow-copy added to set
13fe880e-e232-493d-87e9-402f21019fb6: prepare completed in 0 secs
13fe880e-e232-493d-87e9-402f21019fb6: commit completed in 1 secs
13fe880e-e232-493d-87e9-402f21019fb6(1c26544e-8251-445f-be89-d1e0a3938777):
 \
share windows_server_2012_share@{1C26544E-8251-445F-BE89-D1E0A3938777} \
exposed as a snapshot of \\WIN-SERVER\windows_server_2012_share\
```

确认服务器是否已公开快照共享：

```
rpcclient $> netshareenum
netname: windows_server_2012_share
remark:
path:   C:\Shares\windows_server_2012_share
password:       (null)

netname: windows_server_2012_share@{1C26544E-8251-445F-BE89-D1E0A3938777}
remark: (null)
path:   \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy{F6E6507E-
F537-11E3-9404-B8AC6F927453}\Shares\windows_server_2012_share\
password:       (null)
```

尝试删除快照共享：

```
rpcclient $> fss_delete windows_server_2012_share \
13fe880e-e232-493d-87e9-402f21019fb6 1c26544e-8251-445f-be89-d1e0a3938777
13fe880e-e232-493d-87e9-402f21019fb6(1c26544e-8251-445f-be89-d1e0a3938777):
 \
\\WIN-SERVER\windows_server_2012_share\ shadow-copy deleted
```

确认服务器是否已去除快照共享：

```
rpcclient $> netshareenum
```

```
netname: windows_server_2012_share
remark:
path:   C:\Shares\windows_server_2012_share
password:       (null)
```

29.8.2.4 使用 DiskShadow.exe 从 Windows 远程管理快照

您也可以在充当客户端的 Windows 环境中，管理 Linux Samba 服务器上的 SMB 共享的快照。Windows Server 2012 提供了 `DiskShadow.exe` 实用程序，该程序可以像第 29.8.2.3 节 “使用 rpcclient 从 Linux 远程管理快照”中所述的 `rpcclient` 那样管理远程共享。请注意，首先您需要妥善设置 Samba 服务器。

以下示例步骤描述了如何设置 Samba 服务器，使 Windows Server 客户端能够管理其共享的快照。请注意，`EXAMPLE` 是在测试环境中使用的 Active Directory 域，`fsrvp-server.example.com` 是 Samba 服务器的主机名，`/srv/smb` 是 SMB 共享的路径。

过程 29.1 SAMBA 服务器配置详细说明

1. 通过 YaST 加入到 Active Directory 域。 详细信息, 第 29.7 节 “带有 Active Directory 的网络中的 Samba 服务器”.
2. 确保“活动域 DNS”项正确：

   ```
   fsrvp-server:~ # net -U 'Administrator' ads dns register \
   fsrvp-server.example.com <IP address>
   Successfully registered hostname with DNS
   ```
3. 在 `/srv/smb` 位置创建 Btrfs 子卷

   ```
   fsrvp-server:~ # btrfs subvolume create /srv/smb
   ```
4. 为路径 `/srv/smb` 创建 snapper 配置文件

   ```
   fsrvp-server:~ # snapper -c <snapper_config> create-config /srv/smb
   ```
5. 创建路径为 `/srv/smb` 的新共享，并启用 YaST 公开快照复选框。确保将以下代码片段添加到 `/etc/samba/smb.conf` 的 global 部分，如第 29.8.2.2 节 “远程共享快照”中所述：

```
[global]
 rpc_daemon:fssd = fork
 registry shares = yes
 include = registry
```

6. 使用 `systemctl restart nmb smb` 重启动 Samba

7. 配置 snapper 权限：

```
fsrvp-server:~ # snapper -c <snapper_config> set-config \
ALLOW_USERS="EXAMPLE\\\\Administrator EXAMPLE\\\\win-client$"
```

确保也允许任何 ALLOW_USERS 浏览 `.snapshots` 子目录。

```
fsrvp-server:~ # snapper -c <snapper_config> set-config SYNC_ACL=yes
```

重要：路径转义

请小心使用“\”转义！请转义两次，以确保 `/etc/snapper/configs/<snapper_config>` 中储存的值转义一次。

"EXAMPLE\win-client$" 对应于 Windows 客户端计算机帐户。对此帐户进行验证后，Windows 将发出初始 FSRVP 请求。

8. 授予 Windows 客户端帐户必要的特权：

```
fsrvp-server:~ # net -U 'Administrator' rpc rights grant \
"EXAMPLE\\win-client$" SeBackupPrivilege
Successfully granted rights.
```

不需要对 "EXAMPLE\Administrator" 用户执行上一条命令，因为已授予该用户特权。

过程 29.2 运行 WINDOWS 客户端设置和 `DiskShadow.exe`

1. 引导 Windows Server 2012（示例主机名为 WIN-CLIENT）。

2. 就像在 SUSE Linux Enterprise Server 上那样，加入到同一个 Active Directory 域 EXAMPLE。

3. 重引导。
4. 打开 Powershell。
5. 启动 `DiskShadow.exe`，然后开始执行备份过程：

```
PS C:\Users\Administrator.EXAMPLE> diskshadow.exe
Microsoft DiskShadow version 1.0
Copyright (C) 2012 Microsoft Corporation
On computer:  WIN-CLIENT,  6/17/2014 3:53:54 PM

DISKSHADOW> begin backup
```

6. 指定每次程序退出、重设置或重引导时要保留的阴影副本：

```
DISKSHADOW> set context PERSISTENT
```

7. 检查指定的共享是否支持快照，然后创建一个快照：

```
DISKSHADOW> add volume \\fsrvp-server\sles_snapper

DISKSHADOW> create
Alias VSS_SHADOW_1 for shadow ID {de4ddca4-4978-4805-8776-cdf82d190a4a} set
 as \
 environment variable.
Alias VSS_SHADOW_SET for shadow set ID {c58e1452-c554-400e-a266-
d11d5c837cb1} \
 set as environment variable.

Querying all shadow copies with the shadow copy set ID \
 {c58e1452-c554-400e-a266-d11d5c837cb1}

 * Shadow copy ID = {de4ddca4-4978-4805-8776-cdf82d190a4a}
 %VSS_SHADOW_1%
    - Shadow copy set: {c58e1452-c554-400e-a266-d11d5c837cb1}
 %VSS_SHADOW_SET%
    - Original count of shadow copies = 1
    - Original volume name: \\FSRVP-SERVER\SLES_SNAPPER\ \
      [volume not on this machine]
    - Creation time: 6/17/2014 3:54:43 PM
```

```
    - Shadow copy device name:
      \\FSRVP-SERVER\SLES_SNAPPER@{31afd84a-44a7-41be-b9b0-751898756faa}
    - Originating machine: FSRVP-SERVER
    - Service machine: win-client.example.com
    - Not exposed
    - Provider ID: {89300202-3cec-4981-9171-19f59559e0f2}
    - Attributes:  No_Auto_Release Persistent FileShare

Number of shadow copies listed: 1
```

8. 完成备份过程：

```
DISKSHADOW> end backup
```

9. 创建快照后，尝试将它删除，并确认删除结果：

```
DISKSHADOW> delete shadows volume \\FSRVP-SERVER\SLES_SNAPPER\
Deleting shadow copy {de4ddca4-4978-4805-8776-cdf82d190a4a} on volume \
 \\FSRVP-SERVER\SLES_SNAPPER\ from provider \
{89300202-3cec-4981-9171-19f59559e0f2} [Attributes: 0x04000009]...

Number of shadow copies deleted: 1

DISKSHADOW> list shadows all

Querying all shadow copies on the computer ...
No shadow copies found in system.
```

29.9 更多信息

Samba 文档包含在 `samba-doc` 包中，默认情况下不会安装该包。您可以使用 `zypper install samba-doc` 进行安装。在命令行中输入 `apropos samba` 可显示一些手册页；或浏览 `/usr/share/doc/packages/samba` 目录以获取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`)。另一个可以查看 Samba 相关信息的文件是 `/usr/share/doc/packages/samba/README.SUSE`。

Samba 小组提供的《Samba 操作指南》（请参见 https://wiki.samba.org ）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。

30 使用 Autofs 按需装入

autofs 是一个可根据需要自动装入指定目录的程序。它基于一个内核模块运行以实现高效率，并且可以同时管理本地目录和网络共享。这些自动安装点仅会在被访问时装入，一定时间内不活动后即会被卸载。这种按需行为可节省带宽，并实现比 /etc/fstab 管理的静态装入更高的性能。虽然 autofs 是控制脚本，但 automount 才是实际执行自动装入的命令（守护程序）。

30.1 安装

SUSE Linux Enterprise Server 上默认未安装 autofs。要使用它的自动装载功能，请先使用下面的命令安装该程序

```
sudo zypper install autofs
```

30.2 配置

您需要使用 vim 等文本编辑器编辑 autofs 的配置文件来手动配置它。配置 autofs 有两个基本步骤 — master 映射文件和特定映射文件。

30.2.1 Master 映射文件

autofs 的默认 master 配置文件是 /etc/auto.master。可通过在 /etc/sysconfig/autofs 文件中更改 DEFAULT_MASTER_MAP_NAME 选项的值来更改其位置。以下是 SUSE Linux Enterprise Server 中默认 master 映射文件的内容：

```
#
# Sample auto.master file
# This is an automounter map and it has the following format
# key [ -mount-options-separated-by-comma ] location
# For details of the format look at autofs(5).
```

```
#
#/misc  /etc/auto.misc ❷
#/net -hosts
#
# Include /etc/auto.master.d/*.autofs ❸
#
#+dir:/etc/auto.master.d
#
# Include central master map if it can be found using
# nsswitch sources.
#
# Note that if there are entries for /net or /misc (as
# above) in the included master map any keys that are the
# same will not be seen as the first read key seen takes
# precedence.
#
+auto.master ❹
```

❶ `autofs` 手册页 (`man 5 autofs`) 提供了许多有关该自动装入器映射格式的重要信息。

❷ 虽然这些内容默认会被注释掉 (#)，但它依然是简单的自动装入器映射语法示例。

❸ 如果您需要将 master 映射分割成几个文件，请将该行取消注释，并将映射（后缀为 `.autofs`）置于 `/etc/auto.master.d/` 目录中。

❹ `+auto.master` 可确保使用 NIS（请参见《Security Guide》, 第 3 章 “Using NIS”, 第 3.1 节 “Configuring NIS Servers”了解 NIS 的更多信息）的用户仍可找到其 master 映射。

`auto.master` 中的项有三个字段，语法如下：

```
mount point      map name      options
```

mount point

要在其中装入 `autofs` 文件系统的基本位置，例如 `/home`。

map name

装入时所用映射源的名称。有关映射文件的语法，请参见第 30.2.2 节 “映射文件”。

options

这些选项（如指定）将作为默认值应用于给定映射中的所有项。

提示：更多信息

有关选用 `map-type`、`format` 和 `options` 的特定值的更多详细信息，请参见 auto.master 手册页 (`man 5 auto.master`)。

`auto.master` 中下面的项指示 `autofs` 查看 `/etc/auto.smb`，并在 `/smb` 目录中创建安装点。

```
/smb    /etc/auto.smb
```

30.2.1.1 直接装入

直接装入会在相关映射文件内的指定路径创建安装点。这种方式不是在 `auto.master` 中指定安装点，而是用 `/-` 替换安装点字段。例如，下行指示 `autofs` 在 `auto.smb` 中的指定位置创建安装点：

```
/-        /etc/auto.smb
```

提示：不含完整路径的映射

如果指定映射文件时未包含其完整本地或网络路径，系统会使用名称服务转换 (NSS) 配置寻找该映射文件。

```
/-        auto.smb
```

30.2.2 映射文件

重要：其他映射类型

虽然文件是使用 `autofs` 自动装入的最常见的映射类型，但是还有其他一些类型。映射规范可以是命令的输出，也可以是 LDAP 或数据库中查询的结果。有关映射类型的更多详细信息，请参见 `man 5 auto.master` 手册页。

映射文件指定（本地或网络）来源位置，以及在本地装入来源的安装点。映射的一般格式与 master 映射相似。区别在于 options 位于 mount point 与 location 之间，而不是该项的末尾：

```
mount point      options      location
```

确保映射文件未标记为可执行文件。可通过执行 `chmod -x MAP_FILE` 去除可执行文件位。

mount point

指定将来源位置装入到何处。这可以是要添加到 auto.master 中所指定基础安装点的单个目录名称（所谓的 间接 装入），也可以是安装点的完整路径（直接装入，请参见第 30.2.1.1 节 “直接装入”）。

options

指定相关项的安装点逗号分隔列表（可选）。如果 `auto.master` 还包含此映射文件的选项，这些选项会附加在后面。

location

指定要装入的文件系统来自何处。通常是 NFS 或 SMB 卷，常用表示法是 主机名:路径名称。如果要装入的文件系统以“/”开头（例如本地 `/dev` 项或 smbfs 共享），需要在前面加一个冒号“:”，例如 `:/dev/sda1`。

30.3 操作和调试

本节介绍如何控制 `autofs` 服务操作，以及如何在调整该自动装入器操作时查看更多调试信息。

30.3.1 控制 autofs 服务

`autofs` 服务的操作由 `systemd` 控制。`autofs` 的 `systemctl` 命令的一般语法为

```
sudo systemctl SUB_COMMAND autofs
```

其中，`SUB_COMMAND` 是下列项目之一：

enable

在引导时启动该自动装入器守护程序。

start

启动该自动装入器守护程序。

stop

停止该自动装入器守护程序。自动安装点将不再可访问。

status

打印 `autofs` 服务的当前状态以及相关日志文件的部分内容。

restart

停止然后启动该自动装入器，以便终止所有正在运行的守护程序，然后再启动新的守护程序。

reload

检查当前的 `auto.master` 映射，重启动项已更改的守护程序，并为新项启动新守护程序。

30.3.2 调试自动装入器问题

如果您在使用 `autofs` 装入目录时遇到问题，手动运行 `automount` 守护程序并查看其输出讯息将非常有用：

1. 停止 `autofs`。

   ```
   sudo systemctl stop autofs
   ```

2. 从一个终端的前台手动运行 `automount`，生成详细输出。

   ```
   sudo automount -f -v
   ```

3. 从另一个终端上尝试通过访问安装点（例如，通过 `cd` 或 `ls`）装入自动装入文件系统。

4. 从第一个终端上检查 `automount` 的输出以了解更多信息，例如装入为何失败，或者为何甚至未尝试执行装入。

30.4 自动装入 NFS 共享

下面的过程说明了如何配置 `autofs` 以自动装入网络上可用的 NFS 共享。该过程要用到前面提到的信息，并假设您熟悉 NFS 导出步骤。有关 NFS 的更多信息，请参见第 28 章 “通过 NFS 共享文件系统”。

1. 编辑 master 映射文件 `/etc/auto.master`：

   ```
   sudo vim /etc/auto.master
   ```

   在 `/etc/auto.master` 末尾为新的 NFS 装入添加一条新项：

   ```
   /nfs      /etc/auto.nfs      --timeout=10
   ```

   它告诉 `autofs` 基本安装点是 `/nfs`，NFS 共享在 `/etc/auto.nfs` 映射中指定，并且此映射中的所有共享将在 10 秒不活动后自动卸载。

2. 为 NFS 共享创建新的映射文件：

   ```
   sudo vim /etc/auto.nfs
   ```

   对每个 NFS 共享，`/etc/auto.nfs` 通常都会包含单独的一行。有关其格式，请参见第 30.2.2 节 “映射文件”。添加下行，指出安装点及 NFS 共享网络地址：

   ```
   export      jupiter.com:/home/geeko/doc/export
   ```

   上面的行表示当收到请求时，系统会将 `jupiter.com` 主机上的 `/home/geeko/doc/export` 目录自动装入到本地主机上的 `/nfs/export` 目录（`/nfs` 取自 `auto.master` 映射）。`/nfs/export` 目录将由 `autofs` 自动创建。

3. （选择性）如果您先前以静态方式装入了该 NFS 共享，请将 `/etc/fstab` 中的相关行注释掉。该行应类似于：

   ```
   #jupiter.com:/home/geeko/doc/export /nfs/export nfs defaults 0 0
   ```

4. 重新装载 `autofs` 并检查它是否正常工作：

   ```
   sudo systemctl restart autofs
   ```

```
# ls -l /nfs/export
total 20
drwxr-xr-x  6 1001 users 4096 Oct 25 08:56 ./
drwxr-xr-x  3 root root     0 Apr  1 09:47 ../
drwxr-xr-x  5 1001 users 4096 Jan 14  2013 .images/
drwxr-xr-x 10 1001 users 4096 Aug 16  2013 .profiled/
drwxr-xr-x  3 1001 users 4096 Aug 30  2013 .tmp/
drwxr-xr-x  4 1001 users 4096 Oct 25 08:56 SLE-12-manual/
```

如果您能看到远程共享上的文件列表，则表示 `autofs` 工作正常。

30.5 高级主题

本节讨论的主题超出了 `autofs` 基本介绍的范畴 — 自动装入网络上可用的 NFS 共享、在映射文件中使用通配符，以及特定于 CIFS 文件系统的信息。

30.5.1 /net 安装点

如果您使用了许多 NFS 共享，这个助手安装点将非常有用。`/net` 会根据需要自动装入本地网络上的所有 NFS 共享。该项在 `auto.master` 文件中已经存在，因此，您只需将其取消注释，然后重启动 `autofs` 即可：

```
/net      -hosts
```

```
systemctl restart autofs
```

例如，如果您有名为 `jupiter` 的服务器以及名为 `/export` 的 NFS 共享，您可以在命令行上键入

```
# cd /net/jupiter/export
```

来装入它。

30.5.2 使用通配符自动装入子目录

如果您的某个目录含有多个子目录，并且您需要将这些子目录单个自动装入（一般情况下，该目录是包含各个用户主目录的 `/home` 目录），`autofs` 提供了便捷的解决方案。

如果这些子目录是主目录，则在 `auto.master` 中添加下行：

```
/home      /etc/auto.home
```

现在，您需要在 `/etc/auto.home` 文件中添加正确的映射，以便自动装入用户的主目录。一种方法是为每个目录创建单独的项：

```
wilber      jupiter.com:/home/wilber
penguin      jupiter.com:/home/penguin
tux      jupiter.com:/home/tux
[...]
```

这种方法非常麻烦，因为您需要在 `auto.home` 中管理用户列表。您可以使用星号“*”取代安装点，使用符号“&”取代要装入的目录。

```
*      jupiter:/home/&
```

30.5.3 自动装入 CIFS 文件系统

如果想自动装入 SMB/CIFS 共享（有关 SMB/CIFS 协议的更多信息，请参见第 29 章 “Samba”了解），需要修改映射文件的语法。在 option 字段中添加 `-fstype=cifs`，并在共享位置前面加上一个冒号“:”。

```
mount point      -fstype=cifs      ://jupiter.com/export
```

31 SLP

要配置网络客户端，需要了解关于通过网络提供的服务（例如打印或 LDAP）的详细知识。为了简化在网络客户端上配置此类服务的工作，“服务定位协议”(SLP）应运而生。SLP 可向本地网络中的所有客户端告知所选服务的可用性和配置数据。支持 SLP 的应用程序可以利用这一信息来进行自动配置。

SUSE® Linux Enterprise Server 支持使用 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。您可以使用 SLP 为联网客户端（如系统上的安装服务器、文件服务器或打印服务器）提供核心功能。提供 SLP 支持的服务包括 cupsd、login、ntp、openldap2、postfix、rpasswd、rsyncd、saned、sshd（通过 fish）、vnc 和 ypserv。

默认情况下，系统会安装在网络客户端上使用 SLP 服务所需的所有包。但如果您要通过 SLP 提供服务，请检查 `openslp-server` 包是否已安装。

31.1 SLP 前端 `slptool`

`slptool` 是一个命令行工具，用于查询和注册 SLP 服务。在进行诊断时，查询功能非常有用。下面列出了最重要的 `slptool` 子命令。`slptool --help` 列出所有可用的选项和函数。

findsrvtypes

列出网络上可用的所有服务类型。

```
tux > slptool findsrvtypes
service:install.suse:nfs
service:install.suse:ftp
service:install.suse:http
service:install.suse:smb
service:ssh
service:fish
service:YaST.installation.suse:vnc
service:smtp
service:domain
```

```
service:management-software.IBM:hardware-management-console
service:rsync
service:ntp
service:ypserv
```

findsrvs SERVICE_TYPE

列出提供 SERVICE_TYPE 的所有服务器

```
tux > slptool findsrvs service:ntp
service:ntp://ntp.example.com:123,57810
service:ntp://ntp2.example.com:123,57810
```

findattrs SERVICE_TYPE//HOST

列出 HOST 上 SERVICE_TYPE 的属性

```
tux > slptool findattrs service:ntp://ntp.example.com
(owner=tux),(email=tux@example.com)
```

register SERVICE type//HOST:PORT "(ATTRIBUTE=VALUE),(ATTRIBUTE=VALUE)"

使用可选属性列表在 HOST 上注册 SERVICE_TYPE

```
slptool register service:ntp://ntp.example.com:57810 \
"(owner=tux),(email=tux@example.com)"
```

deregister SERVICE_TYPE//host

在 HOST 上取消注册 SERVICE_TYPE

```
slptool deregister service:ntp://ntp.example.com
```

有关更多信息，请运行 `slptool --help`。

31.2 通过 SLP 提供服务

要提供 SLP 服务，SLP 守护程序 (`slpd`) 必须处于运行状态。就像 SUSE Linux Enterprise Server 中的大多数系统服务一样，`slpd` 通过单独的启动脚本来控制。安装后，默认情况下停用守护程序。要为当前的会话激活它，请运行 `sudo systemctl start slpd`。如果 `slpd` 应该在系统启动时激活，请运行 `sudo systemctl enable slpd`。

SUSE Linux Enterprise Server 中的许多应用程序都已通过 `libslp` 库集成了 SLP 支持。如果服务未符合 SLP 支持，请使用以下方法之一使其可通过 SLP 发布。

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。下面的示例会注册一个扫描仪服务：

```
## Register a saned service on this system
## en means english language
## 65535 disables the timeout, so the service registration does
## not need refreshes
service:scanner.sane://$HOSTNAME:6566,en,65535
watch-port-tcp=6566
description=SANE scanner daemon
```

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` 会自动替换为完整主机名。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。

该注册文件还包含 `watch-port-tcp` 和 `description` 这两个变量。`watch-port-tcp` 通过使 `slpd` 检查相关服务的状态，链接 SLP 服务对该服务是否活动的发布。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描述。

提示：YaST 和 SLP

在模块对话框中激活 SLP 后，由 YaST 代理的某些服务（如安装服务器或 YOU 服务器）会为您自动执行此注册。然后，YaST 为这些服务创建注册文件。

通过 `/etc/slp.reg` 进行的静态注册

此方法与使用 `/etc/slp.reg.d` 的步骤之间唯一的区别在于，所有服务在中心文件中分组。

使用 `slptool` 进行的动态注册

如果某个服务须动态注册，而无需配置文件，请使用 slptool 命令行实用程序。该实用程序还可用于取消注册某个现有服务产品，而无需重启动 `slpd`。有关详细信息，请参见第 31.1 节 “SLP 前端 `slptool`”。

31.2.1 设置 SLP 安装服务器

在网络中通过 SLP 公告安装数据可简化网络安装过程，因为系统会通过 SLP 查询自动请求服务器 IP 地址或安装媒体路径等安装数据。有关指导，请参见《部署指南》, 第 8 章 “设置存放安装源的服务器”。

31.3 更多信息

RFC 2608、2609、2610

RFC 2608 主要描述了 SLP 的定义。RFC 2609 更详细地描述了所用服务 URL 的语法；RFC 2610 则对通过 SLP 的 DHCP 进行了描述。

http://www.openslp.org

OpenSLP 项目的主页。

`/usr/share/doc/packages/openslp`

此目录包含 `openslp-server` 包随附的 SLP 文档，其中的 `README.SUSE` 文件包含 SUSE Linux Enterprise Server 细节、RFC 和两个介绍性的 HTML 文档。要使用 SLP 功能的程序员可参见 SUSE 软件开发包附带的 `openslp-devel` 包中的《Programmers Guide》（编程指南），以了解更多信息。

32 Apache HTTP 服务器

http://www.netcraft.com/ 的调查表明，Apache HTTP 服务器 (Apache) 是世界上应用最广泛的 Web 服务器。Apache 由 Apache 软件基金会 (http://www.apache.org/) 开发，适用于大多数操作系统。SUSE® Linux Enterprise Server 包含 Apache 版本 2.4。本章将介绍如何安装、配置和设置 Web 服务器；如何使用 SSL、CGI 和其他模块；以及如何对 Apache 进行查错。

32.1 快速入门

借助本节内容，可快速设置并启动 Apache。您必须是 `root` 用户才能安装和配置 Apache。

32.1.1 要求

在尝试设置 Apache Web 服务器之前，请确保满足以下要求：

1. 计算机的网络配置正确。有关该主题的详细信息，请参见第 16 章 “基本联网知识”。
2. 通过与时间服务器同步来维护计算机的准确系统时间。这一点是必需的，因为 HTTP 协议的多个部分依赖于正确的时间。请参见第 25 章 “使用 NTP 同步时间”来了解该主题的更多信息。
3. 将安装最新的安全更新。如果存在疑问，请运行 YaST 联机更新。
4. 默认 Web 服务器端口 (80) 将在防火墙中打开。为此，请将 SuSEFirewall2 配置为允许外部区域中的服务 HTTP 服务器。这可以使用 YaST 完成。有关详细信息，请参见《Security Guide》, 第 16 章 “Masquerading and Firewalls”, 第 16.4.1 节 “Configuring the Firewall with YaST”。

32.1.2 安装

SUSE Linux Enterprise Server 中的 Apache 默认不会安装到系统中。要用“即装即用”的标准预定义配置来安装它，请按如下所示继续：

过程 32.1 用默认配置安装 APACHE

1. 启动 YaST，然后选择 软件 › 软件管理 。
2. 选择视图 › 模式，然后选择 Web 和 LAMP 服务器。
3. 确认安装相关的包来完成安装进程。

32.1.3 开始

可以自动在引导时启动 Apache 或手动启动它。

要确保 Apache 在引导期间自动启动，请在目标 `multi-user.target` 和 `graphical.target` 中执行以下命令：

```
root # systemctl enable apache2
```

有关 SUSE Linux Enterprise Server 中 systemd 目标的更多信息以及 YaST 服务管理器 的说明，请参见第 13.4 节 “使用 YaST 管理 服务”。

要使用外壳手动启动 Apache，请运行 `systemctl start apache2`。

过程 32.2 检查 APACHE 是否正在运行

如果在启动 Apache 时没有收到错误消息，这通常表示 Web 服务器正在运行。测试 Apache 是否正在运行：

1. 启动浏览器，然后打开 http://localhost/ 。
   如果 Apache 已启动并正在运行，您将看到一个测试页，指示“它正在运行！”。
2. 如果看不到此页面，请参见第 32.9 节 “查错”。

既然 Web 服务器已在运行，因此可以添加您自己的文档、根据需要调整配置或通过安装模块来添加功能。

32.2 配置 Apache

SUSE Linux Enterprise Server 提供了两个配置选项：

- 手动配置 Apache
- 使用 YaST 配置 Apache

手工配置可提供更详细的信息，但没有 YaST GUI 方便。

重要：配置更改后重新装载或重启动 Apache

大多数配置更改需要重新装载（有些还需要重启动）Apache 后才能生效。使用 `systemctl reload apache2` 或第 32.3 节 “启动和停止 Apache”中所述的某个重启动选项手动重新装载 Apache。

如果用 YaST 配置 Apache，按第 32.2.3.2 节 “HTTP 服务器配置”中所述将 HTTP 服务设置为已启用即可让上述操作自动完成。

32.2.1 Apache 配置文件

本部分概述了 Apache 配置文件。如果使用 YaST 进行配置，则不需要使用这些文件；但如果以后要切换到手动配置，则此信息可能有用。

Apache 配置文件可在两个不同位置处获取：

- `/etc/sysconfig/apache2`
- `/etc/apache2/`

32.2.1.1 `/etc/sysconfig/apache2`

`/etc/sysconfig/apache2` 控制 Apache 的某些全局设置，例如要装载的模块、要包含的其他配置文件、启动服务器时应同时启动的标志，以及应添加到命令行的标志。此文件中的每个配置选项都有详细记录，因此在此不再描述。对于一般用途的 Web 服务器，`/etc/sysconfig/apache2` 中的设置应足以满足所有配置需要。

32.2.1.2 /etc/apache2/

`/etc/apache2/` 托管 Apache 的所有配置文件。下面描述了每个文件的用途。每个文件均包含几个配置选项（也称为指令）。这些文件中的每个配置选项都有详细记录，因此在此不再描述。

Apache 配置文件按如下所示组织：

```
/etc/apache2/
     |
     |- charset.conv
     |- conf.d/
     |   |
     |   |- *.conf
     |
     |- default-server.conf
     |- errors.conf
     |- httpd.conf
     |- listen.conf
     |- magic
     |- mime.types
     |- mod_*.conf
     |- server-tuning.conf
     |- ssl.*
     |- ssl-global.conf
     |- sysconfig.d
     |   |
     |   |- global.conf
     |   |- include.conf
     |   |- loadmodule.conf . .
     |
     |- uid.conf
     |- vhosts.d
     |   |- *.conf
```

/ETC/APACHE2/ 中的 APACHE 配置文件

`charset.conv`

指定要用于不同语言的字符集。不要编辑此文件。

`conf.d/*.conf`

其他模块添加的配置文件。可在需要时将这些配置包含进虚拟主机配置。有关示例请参见 `vhosts.d/vhost.template`。如此操作后，可以为不同的虚拟主机提供不同的模块集。

`default-server.conf`

具有合理默认值的所有虚拟主机的全局配置。除了更改值之外，还可以使用虚拟主机配置来覆盖它们。

`errors.conf`

定义 Apache 如何响应错误。要为所有虚拟主机自定义这些消息，请编辑此文件。否则在您的虚拟主机配置中覆盖这些指令。

`httpd.conf`

主 Apache 服务器配置文件。请勿更改此文件。它主要包含 include 语句和全局设置。重写此处列出的相关配置文件中的全局设置。更改您的虚拟主机配置中的特定于主机的设置（例如文档根目录）。

`listen.conf`

将 Apache 绑定到特定的 IP 地址和端口。基于名称的虚拟主机也在此处配置。有关细节，请参见第 32.2.2.1.1 节 “基于名称的虚拟主机”。

`magic`

mime_magic 模块的数据帮助 Apache 自动确定 MIME 类型的未知文件。不要更改此文件。

`mime.types`

MIME 类型可由系统识别（它实际上是一个指向 `/etc/mime.types` 的链接）。不要编辑此文件。如果需要添加此处没有列出的 MIME 类型，那么请将它们添加到 `mod_mime-defaults.conf`。

`mod_*.conf`

默认情况下安装的模块的配置文件。有关细节，请参见第 32.4 节 “安装、激活和配置模块”。注意，可选模块的配置文件储存在目录 `conf.d` 中。

`server-tuning.conf`

包含不同 MPM（请参见第 32.4.4 节 “多处理模块”）的配置指令以及控制 Apache 性能的一般配置选项。在此处更改时，请对 Web 服务器进行合理的测试。

`ssl-global.conf` 和 `ssl.*`

全局 SSL 配置和 SSL 证书数据。有关细节，请参见第 32.6 节 “使用 SSL 设置安全性 Web 服务器”。

`sysconfig.d/*.conf`

从 `/etc/sysconfig/apache2` 自动生成的配置文件。请勿更改这些文件，而应编辑 `/etc/sysconfig/apache2`。不要在此目录中放置其他配置文件。

`uid.conf`

指定运行 Apache 的用户和组 ID。不要更改此文件。

`vhosts.d/*.conf`

虚拟主机配置应位于此处。该目录包含使用和不使用 SSL 的虚拟主机的模板文件。该目录中以 `.conf` 结尾的所有文件均自动包含在 Apache 配置中。有关详细信息，请参见第 32.2.2.1 节 “虚拟主机配置”。

32.2.2 手动配置 Apache

手动配置 Apache 包括作为 `root` 用户来编辑纯文本配置文件。

32.2.2.1 虚拟主机配置

术语虚拟主机指的是 Apache 在一台物理计算机上为多个统一资源标识符 (URI) 提供服务的能力。这意味着在一个物理计算机上的一个 Web 服务器可以运行几个域（例如 www.example.com 和 www.example.net）。

通常的做法是使用虚拟主机来节省管理精力（只需维护一个 Web 服务器即可）和硬件费用（每个域不需要专用的服务器）。虚拟主机可以是基于名称、基于 IP 或基于端口的。

要列出所有现有的虚拟主机，请使用命令 `apache2ctl -S`。这将输出一个列表，显示默认服务器和所有虚拟主机以及它们的 IP 地址和侦听端口。此外，该列表还针对每个虚拟主机包含一项，显示其在配置文件中的位置。

可以通过 YaST（如第 32.2.3.1.4 节 “虚拟主机”中所述）或通过手动编辑配置文件来配置虚拟主机。默认情况下，系统会根据 /etc/apache2/vhosts.d/ 中每个虚拟主机一个配置文件的设置，为 SUSE Linux Enterprise Server 中的 Apache 做好准备。该目录中扩展名为 .conf 的所有文件均会自动包含到配置中。虚拟主机的基本模板将在目录 vhost.template 或 vhost-ssl.template 中提供，以用于带有 SSL 支持的虚拟主机。

提示：始终创建虚拟主机配置

建议您始终创建虚拟主机配置文件，即使您的 Web 服务器仅主管一个域。这样不但可以将特定于域的配置保存在一个文件中，还可以只需移动、删除或重命名虚拟主机的配置文件就能始终回退到有效的基本配置。因此，还应该为每个虚拟主机创建单独的配置文件。

使用基于名称的虚拟主机时，建议设置将在域名与虚拟主机配置不匹配时使用的默认配置。默认虚拟主机即最先装载其配置的虚拟主机。由于配置文件的装载顺序取决于文件名，因此请在默认虚拟主机配置的文件名的开头使用下划线字符 (_)，以确保最先装载它（例如：_default_vhost.conf）。

<VirtualHost> </VirtualHost> 块保存适用于特定域的信息。当 Apache 接收到客户端对某已定义虚拟主机的请求时，将使用此部分包含的指令。几乎所有指令均可用在虚拟主机环境中。请参见 http://httpd.apache.org/docs/2.4/mod/quickreference.html 来获取有关 Apache 的配置指令的进一步信息。

32.2.2.1.1 基于名称的虚拟主机

使用基于名称的虚拟主机，每个 IP 地址能服务于多个网站。Apache 使用客户端发送的 HTTP 报头中的主机字段来将请求连接到某个虚拟主机声明中匹配的 ServerName 项。如果找不到匹配的 ServerName，则默认使用第一个指定的虚拟主机。

第一步是为您要提供服务的每个基于名称的不同主机创建 <VirtualHost> 块。在每个 <VirtualHost> 块内，至少需要有一个 ServerName 指令来指定要为哪个主机提供服务，还需要有一个 DocumentRoot 指令指出该主机的内容位于文件系统中的哪个位置。

例 32.1 基于名称的 VirtualHost 项的基本示例

```
<VirtualHost *:80>
```

```
# This first-listed virtual host is also the default for *:80
ServerName www.example.com
ServerAlias example.com
DocumentRoot /srv/www/htdocs/domain
</VirtualHost>

<VirtualHost *:80>
ServerName other.example.com
DocumentRoot /srv/www/htdocs/otherdomain
</VirtualHost>
```

`VirtualHost` 开始标记会使该 IP 地址（或完全限定的域名）在基于名称的虚拟主机配置中作为自变量。端口号指令为可选项目。

允许使用通配符 `*` 代替 IP 地址。当使用 IPv6 地址时，地址必须括在方括号中。

例 32.2 基于名称的 VirtualHost 指令

```
<VirtualHost 192.168.3.100:80>
  ...
</VirtualHost>

<VirtualHost 192.168.3.100>
  ...
</VirtualHost>

<VirtualHost *:80>
  ...
</VirtualHost>

<VirtualHost *>
  ...
</VirtualHost>

<VirtualHost [2002:c0a8:364::]>
  ...
</VirtualHost>
```

32.2.2.1.2 基于 IP 的虚拟主机

这种备选的虚拟主机配置要求为计算机设置多个 IP。Apache 的一个实例储存多个域，并为每个域指派一个不同的 IP。

物理服务器必须为每个基于 IP 的虚拟主机指定一个 IP 地址。如果计算机没有多个网卡，也可以使用虚拟网络接口（IP 别名）。

以下示例显示，Apache 在 IP 为 `192.168.3.100` 且储存着其它两个 IP 为 `192.168.3.101` 和 `192.168.3.102` 的域的计算机上运行的情况。请为每个虚拟服务器指定一个单独的 `VirtualHost` 块。

例 32.3 基于 IP 的 VirtualHost 指令

```
<VirtualHost 192.168.3.101>
  ...
</VirtualHost>

<VirtualHost 192.168.3.102>
  ...
</VirtualHost>
```

在此，`VirtualHost` 指令只针对除 `192.168.3.100` 以外的接口。在也为 `192.168.3.100` 配置 `监听` 指令时，必须创建单独的、基于 IP 的虚拟主机才能答复对该接口的 HTTP 请求，否则应用在默认服务器配置 (`/etc/apache2/default-server.conf`) 中找到的指令。

32.2.2.1.3 基本虚拟主机配置

每个虚拟主机配置中至少要有以下指令，才能设置虚拟主机。请参见 `/etc/apache2/vhosts.d/vhost.template` 获取更多选项。

`ServerName`
: 主机所在的全限定域名。

`DocumentRoot`
: Apache 应该为此主机提供文件的目录路径。出于安全考虑，在默认情况下禁止访问整个文件系统，所以必须在 `目录` 容器中显示地解锁此目录。

`ServerAdmin`

服务器管理员的电子邮件地址。例如，此地址将显示在 Apache 创建的错误页面上。

`ErrorLog`

该虚拟主机的错误日志文件。尽管不必为每个虚拟主机创建单独的错误日志文件，但是通常建议执行此操作，因为这样能使错误调试变得容易些。`/var/log/apache2/` 是 Apache 日志文件的默认目录。

`CustomLog`

该虚拟主机的访问日志文件。尽管不必为每个虚拟主机创建单独的访问日志文件，但是通常建议执行此操作，因为这样可单独分析每个主机的访问统计数字。`/var/log/apache2/` 是 Apache 日志文件的默认目录。

综上所述，出于安全考虑，在默认情况下禁止访问整个文件系统。因此，明确对您放置了 Apache 应为其提供服务的文件所在的目录解除锁 定，例如 `DocumentRoot`：

```
<Directory "/srv/www/www.example.com/htdocs">
  Require all granted
</Directory>
```

注意：Require all granted

在旧版 Apache 中，`Require all granted` 语句表达为：

```
Order allow,deny
Allow from all
```

`mod_access_compat` 模块仍然支持该旧语法。

完整的配置文件外观如下所示：

例 32.4 基本 VirtualHost 配置

```
<VirtualHost 192.168.3.100>
  ServerName www.example.com
  DocumentRoot /srv/www/www.example.com/htdocs
  ServerAdmin webmaster@example.com
  ErrorLog /var/log/apache2/www.example.com_log
  CustomLog /var/log/apache2/www.example.com-access_log common
```

```
  <Directory "/srv/www/www.example.com/htdocs">
  Require all granted
  </Directory>
</VirtualHost>
```

32.2.3 使用 YaST 配置 Apache

要使用 YaST 配置 Web 服务器，请启动 YaST，并选择网络服务 › HTTP 服务器。第一次启动此模块时，HTTP 服务器向导会启动，提示您做出一些有关服务器管理的基本决定。完成向导后，在您每次调用 HTTP 服务器模块时，HTTP 服务器配置对话框都会启动。有关详细信息，请参见第 32.2.3.2 节 “HTTP 服务器配置”。

32.2.3.1 HTTP 服务器向导

HTTP 服务器向导包括五个步骤。在对话框的最后一步中，您可以进入专家配置模式进行更具体的设置。

32.2.3.1.1 网络设备选择

在这里，指定 Apache 用以侦听传入请求的网络接口和端口。可以选择现有网络接口及其各自 IP 地址的任意组合。可以使用其他服务未预留的所有三个范围内的端口（公认端口、注册端口和动态或私用端口）。默认设置是在端口 80 上侦听所有网络接口（IP 地址）。

选中打开防火墙中的端口，在防火墙中打开 Web 服务器侦听的端口。要使 Web 服务器在网络（LAN、WAN 或公共因特网）中可用，这样做是必要的。仅在测试时不必对 Web 服务器进行外部访问的情况下，关闭端口是有用的。如果有多个网络接口，请单击防火墙细节以指定要在哪些接口上打开端口。

单击下一步继续配置。

32.2.3.1.2 模块

模块配置选项允许激活或停用 Web 服务器应支持的脚本语言。要激活或停用其他模块，请参见第 32.2.3.2.2 节 “服务器模块”。单击下一步进入下一个对话框。

32.2.3.1.3 默认主机

该选项与默认的 Web 服务器相关。正如第 32.2.2.1 节 “虚拟主机配置”中所述，Apache 可以在一台物理计算机上为多台虚拟主机提供服务。配置文件中最先声明的虚拟主机通常称为默认主机。每个虚拟主机都将继承默认主机的配置。

要编辑主机设置（也称为指令），请在表中选择相应的项，然后单击编辑。要添加新指令，请单击添加。要删除指令，请选择该主机，然后单击删除。

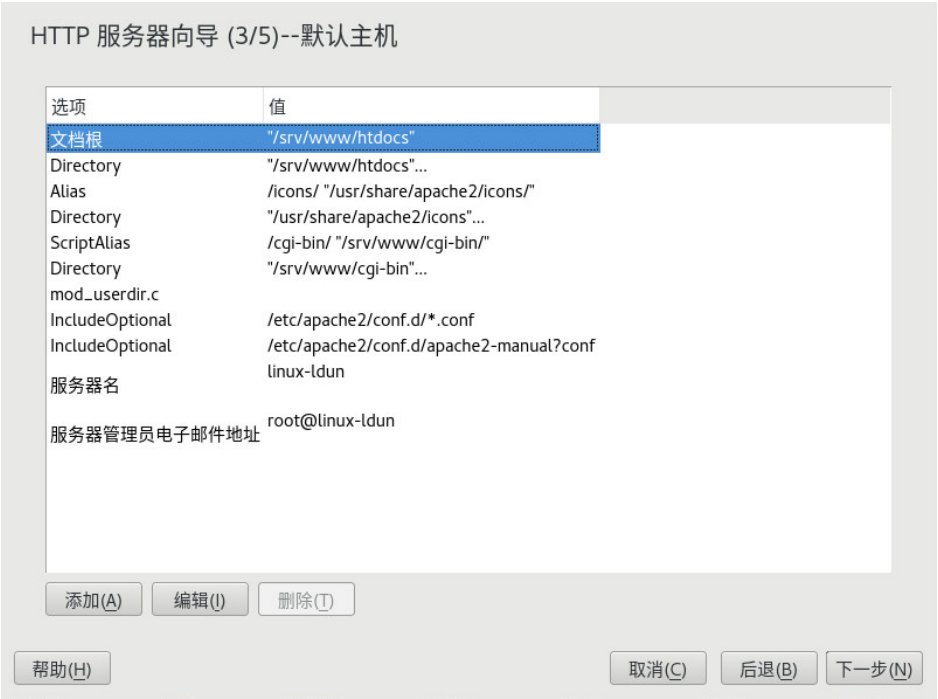


图 32.1 HTTP 服务器向导：默认主机

这里是服务器默认设置的列表：

`Document Root`

Apache 为此主机提供文件的目录路径。`/srv/www/htdocs` 是默认位置。

`Alias`

使用 `Alias` 指令可以将 URL 映射到物理文件系统位置。这意味着可以通过对某路径进行 URL 别名判别来访问该路径（即使是在文件系统中文档根目录之外的路径）。

默认的 SUSE Linux Enterprise Server `Alias /icons` 指向 `/usr/share/apache2/icons`，作为显示在目录索引视图中的 Apache 图标。

`ScriptAlias`

和 `Alias` 指令类似，`ScriptAlias` 指令将 URL 映射到文件系统位置。不同之处在于 `ScriptAlias` 将目标目录指定为 CGI 位置，意味着 CGI 脚本应该在此位置执行。

`Directory`

设置 `Directory` 后，便可包含一组只能应用于指定目录的配置选项。目录 `/srv/www/htdocs`、`/usr/share/apache2/icons` 和 `/srv/www/cgi-bin` 的访问和显示选项是在此处配置的。不需要更改默认值。

`Include`

使用 include，还可指定其他配置文件。已预配置两个 `Include` 指令：`/etc/apache2/conf.d/` 是包含与外部模块一起提供的配置文件的目录。使用此指令可包含该目录中以 `.conf` 结尾的所有文件。使用第二个指令可包含 `/etc/apache2/conf.d/apache2-manual.conf`（`apache2-manual` 配置文件）。

`Server Name`

这指定了客户端用来联系 Web 服务器的默认 URL。使用完全限定的域名 (FQDN) 到达 Web 服务器（位于 `http://FQDN/`）或其 IP 地址。不能在此处随意选择名称 – 服务器在此名称下必须是“已知”的。

`Server Administrator E-Mail`

服务器管理员的电子邮件地址。例如，此地址将显示在 Apache 创建的错误页面上。

完成默认主机步骤后，单击下一步继续完成配置。

32.2.3.1.4 虚拟主机

在本步骤中，向导显示已配置的虚拟主机（请参见第 32.2.2.1 节 “虚拟主机配置”）的列表。如果启动 YaST HTTP 向导前未进行手动更改，将不显示虚拟主机。

要添加主机，请单击添加以打开一个对话框，可在其中输入有关该主机的基本信息，如服务器名称、服务器内容根 (`DocumentRoot`) 和管理员电子邮件。服务器解析用来确定如何识别主机（基于名称或基于 IP）。通过更改虚拟主机 ID 指定名称或 IP 地址

单击下一步进入虚拟主机配置对话框的第二部分。

在虚拟主机配置的第二部分中，可以指定是否启用 CGI 脚本以及用于这些脚本的目录。还可启用 SSL。如果要启用，还必须指定证书的路径。请参见第 32.6.2 节 “使用 SSL 配置 Apache”了解有关 SSL 和证书的细节。使用目录索引选项，可指定在客户端请求目录时所显示的文件（默认情况下为 `index.html`）。添加一个或多个文件名（用空格分隔）可更改此设置。使用启用公用 HTML，便可在服务器的 `http://www.example.com/~USER` 下访问用户公共目录（`~USER/public_html/`）的内容。

重要：创建虚拟主机

不能随意添加虚拟主机。如果使用基于名称的虚拟主机，必须在网络上解析每个主机名。如果使用基于 IP 的虚拟主机，则仅可向每个可用的 IP 地址指定一个主机。

32.2.3.1.5 摘要

这是本向导的最后一步。在此，确定 Apache 服务器启动的方式和时间：何时引导或手动引导。另请参见迄今为止所作配置的简短摘要。如果对设置满意，单击完成以完成配置。要进行更改，请单击后退直至显示所需的对话框。单击 HTTP 服务器专家配置打开第 32.2.3.2 节 “HTTP 服务器配置”中所述的对话框。

图 32.2 HTTP 服务器向导：摘要

32.2.3.2 HTTP 服务器配置

HTTP 服务器配置对话框还允许您对配置进行比在向导（它只在您首次配置 Web 服务器时运行）中更多的调整。它由四个如下所述的选项卡组成。在此处更改的任何配置选项都不会立即生效，总是需要使用完成来确认更改从而使其生效。单击中止退出配置模块并丢弃所作更改。

32.2.3.2.1 监听端口和地址

在 HTTP Service 中，选择应该运行（启用）还是停止（禁用）Apache。在侦听端口中，添加、编辑或删除服务器可用的地址和端口。默认设置是在端口 80 上侦听所有接口。应始终选中打开防火墙中的端口，否则无法从外部访问 Web 服务器。仅在测试时不必对 Web 服务器进行外部访问的情况下，关闭端口是有用的。如果有多个网络接口，请单击防火墙细节以指定要在哪些接口上打开端口。

使用日志文件查阅访问日志文件或错误日志文件。如果要测试配置，这很有用。该日志文件将在单独的窗口中打开，您还可从该窗口重启动或重新装载 Web 服务器。有关详细信息，请参见第 32.3 节 “启动和停止 Apache”。这些命令将立即生效，并且其日志消息也会立即显示。

图 32.3 HTTP 服务器配置：侦听端口和地址

32.2.3.2.2 服务器模块

可以通过单击切换状态来更改 Apache2 模块的状态（启用或禁用）。单击添加模块可添加已安装但还未列出的新模块。要了解模块的更多信息，请参见第 32.4 节 “安装、激活和配置模块”。

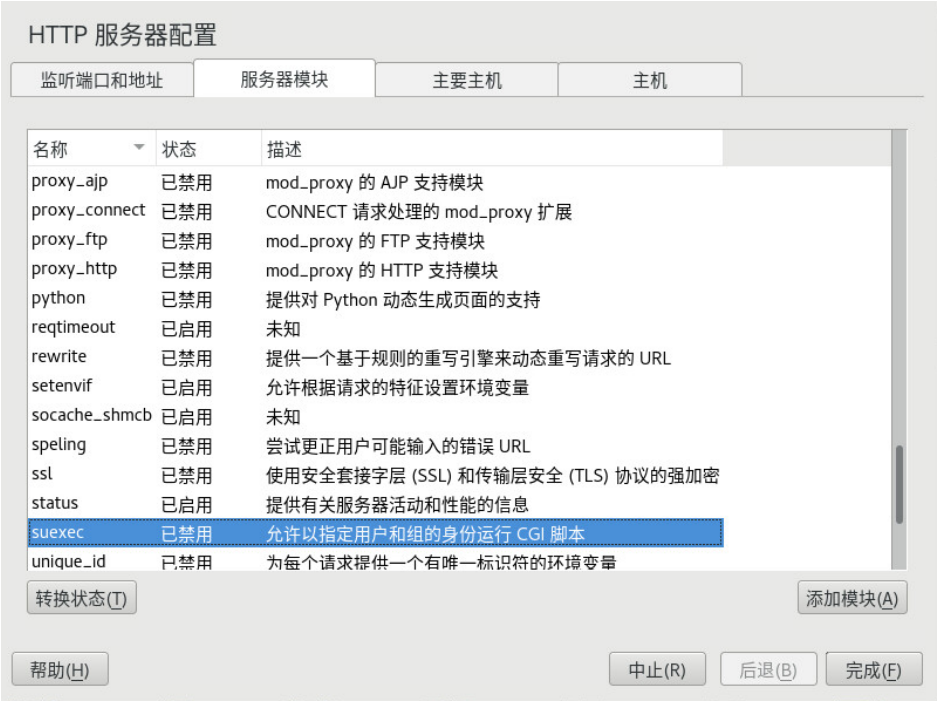


图 32.4 HTTP 服务器配置：服务器模块

32.2.3.2.3 主要主机

这些对话框与上述对话框相同。请参见第 32.2.3.1.3 节 “默认主机”和第 32.2.3.1.4 节 “虚拟主机”。

32.3 启动和停止 Apache

如果按第 32.2.3 节 “使用 YaST 配置 Apache”中所述使用 YaST 配置，Apache 会在引导时在 `multi-user.target` 和 `graphical.target` 中启动。您可以使用 YaST 的 服务管理器或借助 `systemctl` 命令行工具（`systemctl enable` 或 `systemctl disable`）更改此行为。

要在正在运行的系统上启动、停止或操作 Apache，请使用 `systemctl` 或 `apachectl` 命令，详见下面的说明。

有关 `systemctl` 命令的一般信息，请参考第 13.2.1 节 “管理正在运行的系统中的服务”。

`systemctl status apache2`

请检查 Apache 是否已启动。

`systemctl start apache2`

如果 Apache 未在运行，则启动它。

`systemctl stop apache2`

通过终止父进程来停止 Apache。

`systemctl restart apache2`

停止然后重启动 Apache。如果 Web 服务器没有预先运行，则启动它。

`systemctl try-restart apache2`

仅当 Apache 已在运行时才停止并重启动它。

`systemctl reload apache2`

停止 Web 服务器时，应建议所有生成的 Apache 进程在关闭之前首先完成它们的请求。每个进程终止时，会替换为一个新启动的进程，继而导致 Apache 完全“重启动”。

提示：在生产环境中重启动 Apache

使用此命令，无需中断连接，即可激活 Apache 配置中的更改。

`systemctl stop apache2`

在经过 `GracefulShutdownTimeout` 所配置的指定时间段后停止 Web 服务器，以确保现有请求可以完成。

`apachectl configtest`

在不影响运行的 Web 服务器的情况下检查配置文件的语法。由于此检查是在服务器每次启动时强制执行的，所以通常不需要显式运行测试（如果发现配置错误，则 Web 服务器将不启动、重装载或重启动）。

`apachectl status` 和 `apachectl fullstatus`

分别转储不全或完整状态屏幕。需要启用 `mod_status` 模块，以及安装基于文本的浏览器（例如 `links` 或 `w3m`）。此外，还必须将状态添加到文件 `/etc/sysconfig/apache2` 中的 `APACHE_SERVER_FLAGS`。

提示：其他标志

如果您为命令指定了其他标志，这些标志就会传递给 Web 服务器。

32.4 安装、激活和配置模块

Apache 软件是以模块化方式构建的：除某些核心任务外的所有功能都是通过模块处理的。这方面的发展很快，甚至连 HTTP 都是由模块（`http_core`）处理的。

Apache 模块可以在构建时编译成 Apache 二进制文件，或在运行时动态装载。请参见第 32.4.2 节 “激活和停用”以获取有关如何动态装载模块的详细信息。

Apache 模块可以划分为四个不同的类别：

基础模块

默认情况下，基础模块将编译到 Apache 中。SUSE Linux Enterprise Server 上的 Apache 中仅编译了 `mod_so`（装载其他模块时需要）和 `http_core`。所有其他对象都可用作共享对象：它们可在运行时被包含，而不是包含在服务器二进制文件本。

扩展模块

通常，扩展模块包含在 Apache 软件包中，但一般不静态编译到服务器中。在 SUSE Linux Enterprise Server 中，它们以共享对象方式提供，在运行时装载到 Apache 中。

外部模块

标注为外部的模块不包含在正式 Apache 发行版中。不过，SUSE Linux Enterprise Server 提供了其中的几个模块。

多处理模块 (MPM)

MPM 负责接受和处理对 Web 服务器的请求，代表 Web 服务器软件的核心。

32.4.1 模块安装

如果您按第 32.1.2 节 “安装”中所述执行了默认安装，那么以下模块已经安装：所有基本和扩展模块、多处理模块 Prefork MPM 以及外部模块 `mod_python`。

您可以启动 YaST，然后选择软件 › 软件管理，来安装其他外部模块。现在请选择视图 › 搜索并搜索 apache。在其他包中，结果列表将包含所有可用的外部 Apache 模块。

32.4.2 激活和停用

手动或用 YaST 激活或停用特定模块。在 YaST 中，需要使用第 32.2.3.1 节 “HTTP 服务器向导”中所述的模块配置启用或禁用脚本语言模块（PHP5、Perl 和 Python）。可以按第 32.2.3.2.2 节 “服务器模块”中所述启用或禁用所有其他模块。

如果您想手动激活或停用这些模块，请分别使用命令 `a2enmod MODULE` 或 `a2dismod MODULE`。`a2enmod -l` 会输出所有当前活动的模块列表。

重要：包含外部模块的配置文件

如果已经手动激活外部模块，则确保在所有虚拟主机配置中装载其配置文件。外部模块的配置文件位于 `/etc/apache2/conf.d/` 下，并且默认装载到 `/etc/apache2/default-server.conf` 中。要获取更精密的控制，您可以注释掉 `/etc/apache2/default-server.conf` 中的内容，并仅将该文件添加到特定虚拟主机。请参见 `/etc/apache2/vhosts.d/vhost.template` 获取示例。

32.4.3 基础模块和扩展模块

Apache 文档中对所有基础模块和扩展模块均进行了详细的描述。此处仅提供大多数重要模块的简短描述。请参见 http://httpd.apache.org/docs/2.4/mod/ 以了解有关每个模块的详细信息。

`mod_actions`

请求某个特定 MIME 类型（如 `application/pdf`）、带特定扩展名的文件（如 `.rpm`）或某个特定请求方法（如 `GET`）时，提供执行脚本的方法。默认情况下启用此模块。

mod_alias

提供 Alias 和 Redirect 指令，供您将 URL 映射到特定目录 (Alias)，或将请求的 URL 重定向到另一个位置。默认情况下启用此模块。

mod_auth*

身份验证模块提供不同的身份验证方法：使用 mod_auth_basic 的基本身份验证或使用 mod_auth_digest 的摘要身份验证。

mod_auth_basic 和 mod_auth_digest 必须与身份验证提供程序模块 mod_authn_* (例如，用于基于文本文件的身份验证的 mod_authn_file)，以及授权模块 mod_authz_* (例如，用于用户授权的 mod_authz_user) 结合使用。

有关该主题的更多信息可以从Authentication HOWTO中获取，网址是 http://httpd.apache.org/docs/2.4/howto/auth.html↗.

mod_autoindex

当不存在索引文件 (例如 index.html) 时，Autoindex 将生成目录列表。这些索引的外观是可配置的。默认情况下启用此模块。但是，在默认情况下，目录列表将通过 选项 指令禁用，覆盖虚拟主机配置中的此设置。此模块的默认配置文件位于 /etc/apache2/mod_autoindex-defaults.conf 处。

mod_cgi

执行 CGI 脚本时需要有 mod_cgi 。默认情况下启用此模块。

mod_deflate

可使用此模块配置 Apache，使其在传递给定文件类型之前实时压缩这些文件类型。

mod_dir

mod_dir 提供 DirectoryIndex 指令，它可用来配置在请求目录时自动传递的文件 (默认使用 index.html)。当目录请求不包含尾部斜杠时，它还能自动重定向到正确的 URL。默认情况下启用此模块。

mod_env

控制传递到 CGI 脚本或 SSI 页面的环境。环境变量可设置或取消设置，或者从调用 httpd 进程的外壳传递。默认情况下启用此模块。

mod_expires

使用 mod_expires，便可通过发送 Expires 报头来控制代理和浏览器缓存刷新文档的频率。默认情况下启用此模块。

`mod_http2`

通过 `mod_http2`，Apache 可获取对 HTTP/2 协议的支持。这可以通过在 `VirtualHost` 中指定 `Protocols h2 http/1.1` 来实现。

`mod_include`

`mod_include` 允许您使用服务器端包含 (SSI)，它能提供动态生成 HTML 页面的基本功能。默认情况下启用此模块。

`mod_info`

在 http://localhost/server-info/ 下提供服务器配置的完整概述。出于安全考虑，始终应该限制对此 URL 的访问。默认情况下，仅允许 `localhost` 访问此 URL。`mod_info` 是在 `/etc/apache2/mod_info.conf` 中配置的。

`mod_log_config`

使用此模块可配置 Apache 日志文件的外观。默认情况下启用此模块。

`mod_mime`

Mime 模块会根据所传递文件的扩展名（例如，HTML 文档的扩展名为 `text/html`）来确定文件是否具有正确的 MIME 报头。默认情况下启用此模块。

`mod_negotiation`

对于内容协商是必需的。请参见 http://httpd.apache.org/docs/2.4/content-negotiation.html 获取更多信息。默认情况下启用此模块。

`mod_rewrite`

提供 `mod_alias` 的功能，但功能更全且更为灵活。使用 `mod_rewrite`，便可根据多个规则、请求报头等来重定向 URL。

`mod_setenvif`

基于客户端的请求细节（如客户端发送的浏览器字符串或客户端的 IP 地址）来设置环境变量。默认情况下启用此模块。

`mod_spelling`

`mod_speling` 会尝试自动更正 URL 中的打字错误，例如大小写错误。

`mod_ssl`

在 Web 服务器和客户端之间启用加密连接。有关详细信息，请参见第 32.6 节 “使用 SSL 设置安全性 Web 服务器”。默认情况下启用此模块。

`mod_status`

在 http://localhost/server-status/ 下提供有关服务器活动和性能的信息。出于安全考虑，始终应该限制对此 URL 的访问。默认情况下，仅允许 `localhost` 访问此 URL。`mod_status` 是在 `/etc/apache2/mod_status.conf` 中配置的。

`mod_suexec`

`mod_suexec` 允许您在不同的用户和组下运行 CGI 脚本。默认情况下启用此模块。

`mod_userdir`

启用 `~USER/` 下特定于用户的目录。必须在配置中指定 `UserDir` 指令。默认情况下启用此模块。

32.4.4 多处理模块

SUSE Linux Enterprise Server 提供了两个不同的多处理模块 (MPM) 来结合 Apache 使用：

- Prefork MPM
- Worker MPM

32.4.4.1 Prefork MPM

prefork MPM 实施了一个非线程的预生成 Web 服务器。它使 Web 服务器在行为上类似于 Apache 版本 1.x。在该版本中，它隔离每个请求并通过派生单独的子进程来处理请求。这样，有问题的请求就不会影响其他请求，避免了 Web 服务器被锁定。

此基于进程的方法 prefork MPM 虽然提供了稳定性，但比相应的 worker MPM 消耗更多的系统资源。prefork MPM 被视为是基于 Unix 操作系统的默认 MPM。

重要：本文档中的 MPM

本文档假设 Apache 使用 prefork MPM。

32.4.4.2 Worker MPM

worker MPM 提供一种多线程 Web 服务器。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。worker MPM 并非仅生成子进程，还通过在服务器进程中使用线程来处理请求。预派生的子进程是多线程的。此方法相比 prefork MPM，使 Apache 消耗更少的系统资源，从而提高了 Apache 的执行效率。

一个主要缺点是 worker MPM 的稳定性：如果一个线程损坏，进程的所有线程都会受影响。最严重的情况会导致服务器崩溃。尤其是，如果在高负载下将通用网关接口 (CGI) 与 Apache 一起使用，就可能由于线程无法与系统资源通讯而发生内部服务器错误。将 worker MPM 与 Apache 搭配使用的另一个争议是，并非所有可用的 Apache 模块都是线程安全的，因此它不能与 worker MPM 搭配使用。

警告：将 PHP 模块与 MPM 一起使用

并非所有可用的 PHP 模块都是线程安全的。强烈建议不要将 worker MPM 与 `mod_php` 一起使用。

32.4.5 外部模块

此处提供了 SUSE Linux Enterprise Server 随附的所有外部模块的列表。在列出的目录中查找模块的文档。

`mod_apparmor`

为 Apache 提供额外支持，以便对由 `mod_php5` 和 `mod_perl` 等模块处理的各个 CGI 脚本设置 AppArmor 限制。

包名称：`apache2-mod_apparmor`
更多信息：《Security Guide》

`mod_perl`

`mod_perl` 使您能够在嵌入的解释器中运行 Perl 脚本。服务器中嵌入的持久解释器能够避免启动外部解释器并且不会损失 Perl 启动时间。

包名称：`apache2-mod_perl`

配置文件：`/etc/apache2/conf.d/mod_perl.conf`

更多信息：`/usr/share/doc/packages/apache2-mod_perl`

`mod_php5`

PHP 是一种服务器端、跨平台 HTML 嵌入式脚本编写语言。

包名称：`apache2-mod_php5`

配置文件：`/etc/apache2/conf.d/php5.conf`

更多信息：`/usr/share/doc/packages/apache2-mod_php5`

`mod_python`

`mod_python` 允许将 Python 嵌入到 Apache HTTP 服务器中以增强性能并使基于 Web 的应用程序的设计更为灵活。

包名称：`apache2-mod_python`

更多信息：`/usr/share/doc/packages/apache2-mod_python`

`mod_security`

`mod_security` 提供用于保护 Web 应用程序免受一系列攻击的 Web 应用程序防火墙。它可以实现对 HTTP 流量的监控和实时分析。

包名称：`apache2-mod_security2`

配置文件：`/etc/apache2/conf.d/mod_security2.conf`

更多信息：`/usr/share/doc/packages/apache2-mod_security2`

文档：http://modsecurity.org/documentation/

32.4.6 编译

高级用户可以通过编写自定义模块来扩展 Apache。要开发 Apache 模块或编译第三方模块，就需要 `apache2-devel` 包以及相应的开发工具。`apache2-devel` 还包含 `apxs2` 工具，此工具是编译其他 Apache 模块所必需的。

`apxs2` 允许从源代码编译和安装模块（包括对配置文件进行必要的更改），这将创建可在运行时装载入 Apache 的动态共享对象 (DSO)。

`apxs2` 二进制文件在 `/usr/sbin` 中：

- `/usr/sbin/apxs2` — 适合用来构建可搭配任何 MPM 使用的扩展模块。安装位置为 `/usr/lib64/apache2`。
- `/usr/sbin/apxs2-prefork` — 适用于 prefork MPM 模块。安装位置为 `/usr/lib64/apache2-prefork`。
- `/usr/sbin/apxs2-worker` — 适用于 worker MPM 模块。安装位置为 `/usr/lib64/apache2-worker`。

使用以下命令从源代码安装并激活模块：

```
cd /path/to/module/source
apxs2 -cia MODULE.c
```

其中，`-c` 编译该模块，`-i` 安装该模块，`-a` 激活该模块。`apxs2` 的其他选项在 `apxs2(1)` 手册页中有描述。

32.5 启用 CGI 脚本

Apache 的通用网关接口 (CGI) 允许您使用程序或脚本（通常称为 CGI 脚本）创建动态内容。可以用任何编程语言来编写 CGI 脚本。通常使用诸如 Perl 或 PHP 之类的脚本语言。

为了使 Apache 能够递送由 CGI 脚本创建的内容，需要激活 `mod_cgi`。另外还需要 `mod_alias`。默认情况下启用这两种模块。请参见第 32.4.2 节 “激活和停用”来获取有关激活模块的详细信息。

警告：CGI 安全性

允许服务器执行 CGI 脚本是一项潜在的安全性漏洞。请参见第 32.8 节 “避免安全性问题”以了解更多信息。

32.5.1 Apache 配置

在 SUSE Linux Enterprise Server 中，仅允许在目录 `/srv/www/cgi-bin/` 中执行 CGI 脚本。已配置此位置来执行 CGI 脚本。如果已经创建了虚拟主机配置（请参见第 32.2.2.1 节 “虚拟主机配置”）并且想将脚本放置在特定于主机的目录中，必须解锁并配置此目录。

例 32.5 VIRTUALHOST CGI 配置

```
ScriptAlias /cgi-bin/ "/srv/www/www.example.com/cgi-bin/" ❶

<Directory "/srv/www/www.example.com/cgi-bin/">
 Options +ExecCGI ❷
 AddHandler cgi-script .cgi .pl ❸
 Require all granted ❹
</Directory>
```

❶ 指示 Apache 在此目录中将所有文件作为 CGI 脚本处理。

❷ 启用 CGI 脚本执行

❸ 指示服务器将扩展名为 .pl 和 .cgi 的文件视为 CGI 脚本。根据需要进行调整。

❹ `Require` 指令控制默认访问状态。在此例中，授予对指定目录的访问权且无任何限制。有关身份验证和授权的更多信息，请参见http://httpd.apache.org/docs/2.4/howto/auth.html。

32.5.2 运行示例脚本

CGI 编程不同于“常规”编程，因为 CGI 程序和脚本前面必须有一个 MIME 类型的报头，例如 `Content-type: text/html`。此报头将发送到客户端，所以它知道所接收内容的类型。其次，脚本的输出必须是客户端（通常是 Web 浏览器）所知道的东西，比如 HTML（通常情况）、纯文本或图像。

在 `/usr/share/doc/packages/apache2/test-cgi` 下提供的简单测试脚本是 Apache 包的一部分。它将某些环境变量的内容输出为纯文本。将此脚本复制到 `/srv/www/cgi-bin/` 或您虚拟主机的脚本目录（`/srv/www/www.example.com/cgi-bin/`）中，并将它命名为 `test.cgi`。编辑该文件，将 `#!/bin/sh` 作为第一行。

可通过 Web 服务器访问的文件应由用户 `root` 拥有。有关更多信息，请参见第 32.8 节 “避免安全性问题”。由于该 Web 服务器是由不同用户运行的，所以 CGI 脚本必须可被世界各地的用户执行和读取。更改为 CGI 目录并使用命令 `chmod 755 test.cgi` 来应用正确的权限。

现在调用 `http://localhost/cgi-bin/test.cgi` 或 `http://www.example.com/cgi-bin/test.cgi`。应该能看到 “CGI/1.0 测试脚本报告”。

32.5.3 CGI 查错

如果没有看到测试程序的输出而是看到了错误消息，则请检查以下项：

CGI 查错

- 是否在更改配置后重装载了服务器？如果没有，请使用 `systemctl reload apache2` 重新装载
- 如果已经配置了自定义 CGI 目录，那么该配置是否正确？如果不确定，请尝试默认 CGI 目录 `/srv/www/cgi-bin/` 中的脚本并用 `http://localhost/cgi-bin/test.cgi` 调用它。
- 文件权限是否正确？更改为 CGI 目录并执行 `ls -l test.cgi`。输出应该以下面的字符串开头

```
-rwxr-xr-x  1 root root
```

- 确保脚本中没有编程错误。如果还未更改 `test.cgi`，则问题应该不大，但是如果正在使用您自己的程序，则始终要确保它们没有编程错误。

32.6 使用 SSL 设置安全性 Web 服务器

只要在 Web 服务器和客户端之间传送敏感数据（如信用卡信息），就需要具有带身份验证的安全的加密连接。`mod_ssl` 使用安全套接字层（SSL）和传输层安全（TLS）协议来为客户端和 Web 服务器之间的 HTTP 通信提供强有力的加密机制。使用 SSL/TLS 时，将在 Web 服务器和客户端之间建立专用连接。如此可确保数据完整性，并且客户端和服务器能够彼此验证。

基于此目的，服务器在回答对 URL 的任何请求之前，会发送一个 SSL 证书，其中包含证明服务器有效身份的信息。反过来，这保证了该服务器对于通信来说是唯一正确的终端。此外，证书使得在客户端和服务器之间建立起加密连接，确保在不泄露敏感的明文内容的情况下传输信息。

`mod_ssl` 不会实施 SSL/TLS 协议本身，而是充当 Apache 和 SSL 库之间的接口。在 SUSE Linux Enterprise Server 中，将使用 OpenSSL 库。OpenSSL 将自动随 Apache 安装。

将 `mod_ssl` 与 Apache 一起使用的最明显效果就是 URL 的前缀为 `https://`（而不是 `http://`）。

32.6.1 创建 SSL 证书

要将 SSL/TLS 与 Web 服务器搭配使用，您需要创建 SSL 证书。在 Web 服务器和客户端之间授权时需要此证书，以便每一方都能明确地识别另一方。为了确保证书的完整性，证书必须由所有用户都信任的一方签署。

您可创建三种类型的证书：“虚设”证书（仅用于测试）、自我签名证书（用于信任您的指定用户群）和由独立的、众所周知的证书颁发机构 (CA) 签署的证书。

创建证书分两步执行。首先，生成证书颁发机构的私用密钥，然后使用此密钥签署服务器证书。

提示：更多信息

要想更多地了解 SSL/TLS 的概念和定义，请参见 http://httpd.apache.org/docs/2.4/ssl/ssl_intro.html。

32.6.1.1 创建“虚拟”证书

要生成虚设证书，请调用脚本 `/usr/bin/gensslcert`。它创建或重写下列文件。使用 `gensslcert` 的可选开关调整证书。调用 `/usr/bin/gensslcert -h` 了解更多信息。

- `/etc/apache2/ssl.crt/ca.crt`
- `/etc/apache2/ssl.crt/server.crt`
- `/etc/apache2/ssl.key/server.key`
- `/etc/apache2/ssl.csr/server.csr`

还会将 `ca.crt` 的副本放在 `/srv/www/htdocs/CA.crt` 下以供下载。

重要：仅供测试

不能在生产系统上使用虚设证书。它只能用来测试。

32.6.1.2 创建自签署证书

如果要为内部网或指定用户群设置安全的 Web 服务器，通过您自己的证书颁发机构 (CA) 对证书签名可能就足矣。请注意，访问此类网站的用户将会看到类似“此网站不可信”的警告，因为 Web 浏览器不能识别自我签名证书。

重要：自我签名证书

仅在 Web 服务器上使用自签署证书，此证书必须可由知道并相信您是证书授权者的人员访问。例如，不建议在公共商店使用此类证书。

首先，您需要生成证书签名请求 (CSR)。您将需要使用 `openssl`，并采用 `PEM` 证书格式。在执行此步骤期间，系统将要求您输入通行口令并回答几个问题。请记住您输入的通行口令，将来还要使用它。

```
sudo openssl req -new > new.cert.csr
Generating a 1024 bit RSA private key
..++++++
.........++++++
writing new private key to 'privkey.pem'
Enter PEM pass phrase: ❶
Verifying - Enter PEM pass phrase: ❷
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]: ❸
State or Province Name (full name) [Some-State]: ❹
Locality Name (eg, city) []: ❺
Organization Name (eg, company) [Internet Widgits Pty Ltd]: ❻
Organizational Unit Name (eg, section) []: ❼
Common Name (for example server FQDN, or YOUR name) []: ❽
Email Address []: ❾
```

```
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: 10
An optional company name []: 11
```

1. 填写您的通行口令，
2. ...再次填写通行口令（并记住它）。
3. 填写两个字母的国家/地区代码，例如 `GB` 或 `CZ`。
4. 填写您所在的省/自治区/直辖市名称。
5. 填写城市名称，例如 `Prague`。
6. 填写您的工作单位名称。
7. 填写您的组织单位，没有则保留为空白。
8. 填写服务器的域名，或者您的名字和姓氏。
9. 填写您的办公电子邮件地址。
10. 将询问口令保留为空白，否则您每次重启动 Apache Web 服务器都需要输入该口令。
11. 填写选填的公司名称，或保留为空白。

现在可以生成证书。您将再次使用 `openssl`，并且证书的格式是默认的 `PEM`。

过程 32.3 生成证书

1. 将密钥的私用部分导出到 `new.cert.key`。系统将提示您输入您在创建证书签名请求 (CSR) 时所输入的通行口令。

   ```
   sudo openssl rsa -in privkey.pem -out new.cert.key
   ```

2. 根据您在签名请求中填写的信息生成证书的公共部分。`-days` 选项指定证书多长时间以后失效。您可以在证书失效前将其撤消或更换。

   ```
   sudo openssl x509 -in new.cert.csr -out new.cert.cert -req \
   -signkey new.cert.key -days 365
   ```

3. 将证书文件复制到相关的目录，以便 Apache 服务器可以读取它们。确保私用密钥 `/etc/apache2/ssl.key/server.key` 不是人们可以看懂的格式，而公共 PEM 证书 `/etc/apache2/ssl.crt/server.crt` 则是。

```
sudo cp new.cert.cert /etc/apache2/ssl.crt/server.crt
sudo cp new.cert.key /etc/apache2/ssl.key/server.key
```

提示：公共证书位置

最后一步是将公共证书文件从 `/etc/apache2/ssl.crt/server.crt` 复制到用户可以访问的位置，以便将它合并到用户 Web 浏览器中已知且可信的 CA 列表中。否则，浏览器将指示证书是由未知授权者发出的。

32.6.1.3 获取正式签署的证书

签署证书的正式证书颁发机构有多个。证书是由值得信任的第三方签署的，所以可以完全相信。公共操作安全 Web 服务器通常具有正式签署的证书。有关最常用的证书颁发机构 (CA) 列表，请访问 https://en.wikipedia.org/wiki/Certificate_authority#Providers。

请求正式签署的证书时，无需向 CA 发送证书。相反，请发出证书签署请求 (CSR)。要创建 CSR，请运行以下命令：

```
openssl req -new -newkey rsa:2048 -nodes -keyout newkey.pem -out newreq.pem
```

系统将要求您输入判别名。这要求您回答几个问题，例如国家/地区名称或组织名称。输入有效的数据，在此处输入的所有内容稍后都会显示在证书中并检查。无需回答所有问题。如果有问题不适用于您或者您不想回答，请使用“.”. 常用名就是 CA，请选择一个重要的名称，例如 我的公司 CA。最后，必须输入询问口令和备用的公司名称。

在调用脚本的目录中查找 CSR。文件名是 `newreq.pem`。

32.6.2 使用 SSL 配置 Apache

Web 服务器端的 SSL 和 TLS 请求的默认端口是 443。在端口 80 上的“普通” Apache 侦听和端口 443 上支持 SSL/TLS 的 Apache 侦听之间没有冲突。事实上，HTTP 和 HTTPS 可以使用相同的 Apache 实例运行。通常使用一个虚拟主机将请求发送到端口 80 和端口 443 以区分虚拟服务器。

重要：防火墙配置

记住在端口 443 上为支持 SSL 的 Apache 打开防火墙。可以按《Security Guide》，第 16 章 “Masquerading and Firewalls”，第 16.4.1 节 “Configuring the Firewall with YaST”中的描述使用 YaST 来完成此操作。

在全局服务器配置中，SSL 模块默认情况下处于启用状态。如果它在您的主机上已禁用，请使用以下命令激活它：`a2enmod ssl`。要最终启用 SSL，需要使用标志“SSL”启动服务器。要执行此操作，请调用 `a2enflag SSL`（区分大小写！）。如果打算使用口令加密服务器证书，则还应增加 `/etc/sysconfig/apache2` 中 `APACHE_TIMEOUT` 的值，这样在 Apache 启动时，您就有足够的时间输入通行口令。重启动服务器可使这些更改生效。仅重装载是不够的。

虚拟主机配置目录中包含模板 `/etc/apache2/vhosts.d/vhost-ssl.template`，该模板带有详细记录的特定于 SSL 的指令。请参见第 32.2.2.1 节 “虚拟主机配置”了解通用虚拟主机配置。

要开始配置，请将模板复制到 `/etc/apache2/vhosts.d/mySSL-host.conf`，并对其进行编辑。调整以下指令的值应该就足够了：

- `DocumentRoot`
- `ServerName`
- `ServerAdmin`
- `ErrorLog`
- `TransferLog`

32.6.2.1 基于名称的虚拟主机和 SSL

默认情况下，不能在仅具有一个 IP 地址的服务器上运行多个启用了 SSL 的虚拟主机。基于名称的虚拟主机要求 Apache 了解已请求了哪些服务器名称。SSL 连接问题在于，此类请求只能在已建立 SSL 连接之后读取（通过使用默认虚拟主机）。因此用户将收到警告消息，指示证书与服务器名称不匹配。

SUSE Linux Enterprise Server 提供了一个 SSL 协议扩展：服务器名称指示 (SNI)。该协议会在 SSL 协商过程中发送虚拟域的名称，从而解决了这一问题。这样服务器就能提前“切换”到正确的虚拟域，并向浏览器显示正确的证书。

SUSE Linux Enterprise Server 上默认会启用 SNI。为了使基于名称的虚拟主机能够使用 SSL，可按第 32.2.2.1.1 节 “基于名称的虚拟主机”中所述配置服务器（请注意，需要将端口 443 而不是端口 80 用于 SSL）。

重要：SNI 浏览器支持

客户端也必须支持 SNI。不过，只有大部分浏览器支持 SNI，某些较旧的浏览器不支持。更多信息请参见https://en.wikipedia.org/wiki/Server_Name_Indication#Support。

要配置对不支持 SNI 的浏览器的处理方式，请使用指令 SSLStrictSNIVHostCheck。在服务器配置中设置为 on 时，所有虚拟主机都将拒绝不支持 SNI 的浏览器。如果 VirtualHost 指令中设置为 on，对此特定主机的访问将被拒。

在服务器配置中设置为 off 时，服务器的行为类似于不支持 SNI。SSL 请求将由定义的第一个虚拟主机（端口 443）处理。

32.7 在同一服务器上运行多个 Apache 实例

从 SUSE® Linux Enterprise Server 12 SP1 开始，您可以在同一服务器上运行多个 Apache 实例。与运行多个虚拟主机相比，这提供了诸多优势（请参见第 32.2.2.1 节 “虚拟主机配置”）：

- 如果需要将虚拟主机禁用一段时间，您需要更改 Web 服务器配置，并将其重启动以使更改生效。
- 如果一个虚拟主机出现问题，您需要重启动所有的虚拟主机。

您可以照常运行默认的 Apache 实例：

```
sytemctl start apache2
```

它会读取默认的 /etc/sysconfig/apache2 文件。如果该文件不存在，或者存在但未设置 APACHE_HTTPD_CONF 变量，则该实例将读取 /etc/apache2/httpd.conf。

要激活另一个 Apache 实例，请运行：

```
systemctl start apache2@INSTANCE_NAME
```

例如：

```
systemctl start apache2@example_web.org
```

默认情况下，该实例会使用 `/etc/apache2@example_web.org/httpd.conf` 作为主要配置文件，您可以在 `/etc/sysconfig/apache2@example_web.org` 中设置 `APACHE_HTTPD_CONF` 来重写此设置。

下面显示了一个设置更多 Apache 实例的示例。请注意，您需要以 `root` 身份执行所有命令。

过程 32.4 配置其他 APACHE 实例

1. 在 `/etc/sysconfig/apache2` 的基础上创建一个新的配置文件，例如 `/etc/sysconfig/apache2@example_web.org`：

   ```
   cp /etc/sysconfig/apache2 /etc/sysconfig/apache2@example_web.org
   ```

2. 编辑文件 `/etc/sysconfig/apache2@example_web.org`，将包含以下内容的行

   ```
   APACHE_HTTPD_CONF
   ```

   更改为

   ```
   APACHE_HTTPD_CONF="/etc/apache2/httpd@example_web.org.conf"
   ```

3. 在 `/etc/apache2/httpd.conf` 的基础上创建文件 `/etc/apache2/httpd@example_web.org.conf`。

   ```
   cp /etc/apache2/httpd.conf /etc/apache2/httpd@example_web.org.conf
   ```

4. 编辑 `/etc/apache2/httpd@example_web.org.conf`，将

   ```
   Include /etc/apache2/listen.conf
   ```

   更改为

   ```
   Include /etc/apache2/listen@example_web.org.conf
   ```

   检查所有指令，并根据需要予以更改。您可能需要更改

```
Include /etc/apache2/global.conf
```

并为每个实例创建新的 `global@example_web.org.conf`。建议将

```
ErrorLog /var/log/apache2/error_log
```

更改为

```
ErrorLog /var/log/apache2/error@example_web.org_log
```

以便每个实例都有独立的日志。

5. 在 `/etc/apache2/listen.conf` 的基础上创建 `/etc/apache2/listen@example_web.org.conf`。

```
cp /etc/apache2/listen.conf /etc/apache2/listen@example_web.org.conf
```

6. 编辑 `/etc/apache2/listen@example_web.org.conf`，将

```
Listen 80
```

更改为要用于运行新实例的端口号，例如 82：

```
Listen 82
```

要通过安全协议（请参见第 32.6 节 “使用 SSL 设置安全性 Web 服务器”）运行新的 Apache 实例，还需将下面一行

```
Listen 443
```

更改为（示例）

```
Listen 445
```

7. 启动新的 Apache 实例：

```
systemctl start apache2@example_web.org
```

8. 在 Web 浏览器中打开 `http://server_name:82`，检查服务器是否正在运行。如果以前更改了新实例的错误日志文件名，您可以检查这项更改：

```
tail -f /var/log/apache2/error@example_web.org_log
```

下面是在同一服务器上设置多个 Apache 实例时要注意的几点：

- `/etc/sysconfig/apache2@INSTANCE_NAME` 文件可以包含与 `/etc/sysconfig/apache2` 相同的变量，包括模块装载和 MPM 设置。
- 当有其他实例在运行时，默认的 Apache 实例就无需运行。
- 如果未使用 `HTTPD_INSTANCE` 环境变量另行指定，Apache 助手实用程序 `a2enmod`、`a2dismod` 和 `apachectl` 将在默认的 Apache 实例上运行。下面的示例

```
export HTTPD_INSTANCE=example_web.org
a2enmod access_compat
a2enmod status
apachectl start
```

会将 `access_compat` 和 `status` 模块添加到 `/etc/sysconfig/apache2@example_web.org` 的 `APACHE_MODULES` 变量，然后启动 `example_web.org` 实例。

32.8 避免安全性问题

对公共因特网开放的 Web 服务器需要不断加强管理。对于软件和意外的错误配置，安全问题似乎都是不可避免的。有关如何处理这些问题，在此有一些提示。

32.8.1 最新软件

在 Apache 软件中发现漏洞时，SUSE 将会发出安全忠告。其中包含修复漏洞的相关指导，用户应该在情况允许时予以采纳。SUSE 安全性声明可以从以下位置处获取：

- 网页： http://www.suse.com/support/security/
- 邮件列表存档： http://lists.opensuse.org/opensuse-security-announce/
- 安全性声明列表： http://www.suse.com/support/update/

32.8.2 DocumentRoot 权限

在 SUSE Linux Enterprise Server 中，默认情况下， `DocumentRoot` 目录 (`/srv/www/htdocs`) 和 CGI 目录 (`/srv/www/cgi-bin`) 的所有权属于 `root` 用户和组。您不能更改这些权限。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。之后，具有 `wwwrun` 权限 (该权限允许用户随意访问文件系统资源) 的 Apache 可能会执行这些文件。使用 `/srv/www` 的子目录可存放虚拟主机的 `DocumentRoot` 和 CGI 指令，并确保目录和文件属于用户和组 `root` 。

32.8.3 文件系统访问权

默认情况下，在 `/etc/apache2/httpd.conf` 中拒绝对整个文件系统的访问。不应该重写这些指令，而是要明确启用对 Apache 可读的所有目录的访问权。有关详细信息，请参见第 32.2.2.1.3 节 “基本虚拟主机配置”。如此操作后，请确保任何重要文件 (例如口令或系统配置文件) 均不能从外部读取。

32.8.4 CGI 脚本

Perl、PHP、SSI 或任何其他编程语言中的交互脚本基本上可以运行任意命令，因此存在通常的安全性问题。从服务器执行的脚本只能从服务器管理员信任的源安装，允许用户运行他们拥有的所有脚本通常不是好的做法。还建议对所有脚本执行安全性审计。

为了尽可能简化脚本的管理，通常会将 CGI 脚本的执行限制于特定目录而不是全局使用它们。指令 `ScriptAlias` 和 `Option ExecCGI` 用于配置。SUSE Linux Enterprise Server 的默认配置不允许随处执行 CGI 脚本。

所有 CGI 脚本都会作为同一个用户运行，所以不同的脚本可能会彼此冲突。模块 suEXEC 允许您在不同的用户和组下运行 CGI 脚本。

32.8.5 用户目录

启用用户目录（使用 `mod_userdir` 或 `mod_rewrite`）时，一定不要使用 `.htaccess` 文件，这些文件允许用户重写安全设置。至少应该使用指令 `AllowOverRide` 来限制用户的注册。在 SUSE Linux Enterprise Server 中，`.htaccess` 文件默认处于启用状态，但用户在使用 `mod_userdir`（请参见 `/etc/apache2/mod_userdir.conf` 配置文件）时不允许覆盖任何 `Option` 指令。

32.9 查错

如果 Apache 不启动、网页不可访问或用户无法连接到 Web 服务器，那么找出问题的原因是很重要的。下面是几处查找错误描述的常见位置和需要检查的重要事项：

`apache2.service` 子命令的输出：

不要使用 `/usr/sbin/apache2ctl` 二进制文件启动和停止 Web 服务器，而应使用 `systemctl` 命令（如第 32.3 节 “启动和停止 Apache”中所述）。`systemctl status apache2` 详细描述了错误，甚至还提供了解决配置错误的提示。

日志文件和详细程度

不管是致命错误还是非致命错误，都请检查 Apache 日志文件了解原因，主要是默认位于 `/var/log/apache2/error_log` 的错误日志文件。此外，如果需要日志文件记录得更详细一些，可以使用 `LogLevel` 指令来控制所记录消息的详细程度。

提示：简单测试

使用 `tail -F /var/log/apache2/MY_ERROR_LOG` 命令查看 Apache 日志讯息。然后运行 `systemctl restart apache2`。现在，请尝试连接浏览器并检查输出。

防火墙和端口

常见错误之一是在服务器的防火墙配置中未打开针对 Apache 的端口。如果使用 YaST 配置 Apache，有一个单独的选项用于这个具体问题（请参见第 32.2.3 节 “使用 YaST 配置 Apache”）。如果正在手工配置 Apache，则请通过 YaST 的防火墙模块打开 HTTP 和 HTTPS 的防火墙端口。

如果使用以上任何信息均无法找到错误原因，请检查联机 Apache Bug 数据库（网址为 http://httpd.apache.org/bug_report.html ）。此外，可以通过 http://httpd.apache.org/userslist.html 上的邮件列表联系 Apache 用户社区。

32.10 更多信息

包 `apache2-doc` 中包含有关本地安装和参考的多种本地化版本的完整 Apache 手册。它在默认情况下是不安装的，最快的安装方法是使用命令 `zypper in apache2-doc`。完成安装之后，http://localhost/manual/ 中将会有 Apach 手册可供使用。还可在 Web 上的 http://httpd.apache.org/docs-2.4/ 访问它。特定于 SUSE 的配置提示可以在目录 `/usr/share/doc/packages/apache2/README.*` 中获得。

32.10.1 Apache 2.4

有关 Apache 2.4 中新功能的列表，请参见 http://httpd.apache.org/docs/2.4/new_features_2_4.html。可以在 http://httpd.apache.org/docs-2.4/upgrading.html 获得有关从版本 2.2 升级到 2.4 的信息。

32.10.2 Apache 模块

有关第 32.4.5 节 “外部模块”中简述的外部 Apache 模块的更多信息，可在以下位置找到：

`mod_apparmor`
: http://en.opensuse.org/SDB:AppArmor

`mod_auth_kerb`
: http://modauthkerb.sourceforge.net/

`mod_perl`
: http://perl.apache.org/

`mod_php5`
: http://www.php.net/manual/en/install.unix.apache2.php

`mod_python`

http://www.modpython.org/

`mod_security`

http://modsecurity.org/

32.10.3 开发

有关开发 Apache 模块和涉及 Apache Web 服务器项目的更多信息，可以从以下位置处获得：

Apache 开发人员信息

http://httpd.apache.org/dev/

Apache 开发人员文档

http://httpd.apache.org/docs/2.4/developer/

32.10.4 其他来源

如果遇到特定于 SUSE Linux Enterprise Server 中的 Apache 的问题，请查看“技术信息搜索”，网址为：http://www.suse.com/support。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。

33 使用 YaST 设置 FTP 服务器

使用 YaST FTP 服务器模块，可以将计算机配置为 FTP（文件传输协议）服务器。匿名用户和/或经身份验证的用户可以连接到您的计算机并使用 FTP 协议下载文件。根据配置，可能还可以将文件上载到 FTP 服务器。YaST 使用 vsftpd（非常安全的 FTP 守护程序）。

如果 YaST FTP 服务器模块在您的系统中不可用，请安装 `yast2-ftp-server` 包。

要使用 YaST 配置 FTP 服务器，请执行以下步骤：

1. 打开 YaST 控制中心并选择网络服务 › FTP 服务器，或以 `root` 身份运行 `yast2 ftp-server` 命令。
2. 如果您的系统未安装任何 FTP 服务器，YaST FTP 服务器模块启动时将询问您要安装哪个服务器。选择 vsftpd 服务器并确认该对话框中。
3. 在启动对话框中，配置 FTP 服务器的启动选项。有关详细信息，请参见第 33.1 节 “启动 FTP 服务器”。

   在常规对话框中，配置 FTP 目录、欢迎讯息、文件创建掩码和其他参数。有关详细信息，请参见第 33.2 节 “FTP 常规设置”。

   在性能对话框中，设置会影响 FTP 服务器负载的参数。有关详细信息，请参见第 33.3 节 “FTP 性能设置”。

   在身份验证对话框中，设置匿名用户和/或已通过身份验证的用户是否可以使用 FTP 服务器。有关详细信息，请参见第 33.4 节 “身份验证”。

   在专家设置对话框中，配置 FTP 服务器的操作模式、SSL 连接和防火墙设置。有关详细信息，请参见第 33.5 节 “专家设置”。
4. 按完成以保存配置。

33.1 启动 FTP 服务器

在 FTP 启动对话框的服务启动框架中，设置 FTP 的启动方式。可以在系统引导期间自动启动服务器和手动启动服务器之间选择。如果只应在请求 FTP 连接后启动 FTP 服务器，请选择通过 xinetd。

FTP 服务器的当前状态显示在 FTP 启动对话框的打开和关闭框架中。通过单击立即启动 FTP 来启动 FTP 服务器。要停止服务器，请单击立即停止 FTP。更改服务器设置后，单击立即保存设置并重启动 FTP。使用完成离开配置模块也将保存您的配置。

图 33.1 FTP 服务器配置 — 启动

33.2 FTP 常规设置

在 FTP 常规设置对话框的常规设置框架中，可以设置连接到 FTP 服务器后会显示的欢迎消息。

如果选中 Chroot 任何人选项，登录后会将所有本地用户放入其用户主目录的 chroot jail 中。此选项会牵涉到安全性问题，尤其是在用户拥有上载许可权限或进行外壳访问时，所以启用此选项时请务必小心。

如果选中详细日志记录选项，就会将所有 FTP 请求和响应记录在日志中。

可以限制由匿名用户和/或已通过身份验证的用户使用 umask 创建的文件的许可权限。在匿名 Umask 中为匿名用户设置文件创建掩码，在已通过身份验证的用户的 Umask 中为已通过身份验证的用户设置文件创建掩码。掩码应输入为带有前导零的八进制数字。有关 umask 的更多信息，请参见 umask 手册页 (`man 1p umask`)。

在 FTP 目录框架中，设置用于匿名用户和已通过身份验证的用户的目录。按浏览可以从本地文件系统选择要使用的目录。匿名用户的默认 FTP 目录为 `/srv/ftp`。请注意，vsftpd 并未允许所有用户都可对此目录进行写操作。将改为创建带有匿名用户的写许可权限的子目录 `upload`。

33.3 FTP 性能设置

在性能对话框中，设置会影响 FTP 服务器负载的参数。最大闲置时间是远程客户端在 FTP 命令间可能会花费的最长时间（以分钟为单位）。如果长时间处于非活动状态，远程客户端会断开。单 IP 最大连接数确定了可从单个 IP 地址连接的最大客户端数。最大用户数确定了可连接的最大客户端数。任何其他客户端都将收到错误消息。

最大数据传送速度（以 KB/s 为单位）分别是在本地已通过身份验证的用户的本地最大速度和匿名客户端的匿名用户最高速度中设置的。速度设置的默认值为 `0`，表示数据传送速度不受限制。

33.4 身份验证

在身份验证对话框的启用/禁用匿名用户和本地用户框架中，可以设置允许访问您的 FTP 服务器的用户。您可以在以下选项中选择：仅为匿名用户授予访问权限、仅为经身份验证的用户授予访问权限或为两种类型的用户授予访问权限。

要允许用户将文件上载到 FTP 服务器，请选中身份验证对话框上载框架中的启用上载。在此处选中相应的框，您就可以允许包括匿名用户在内的用户上载或创建目录。

注意：vsftp — 允许匿名用户上载文件

如果使用的是 vsftpd 服务器且希望匿名用户可以上载文件或创建目录，则需要在匿名 FTP 目录中创建所有用户都有写权限的子目录。

33.5 专家设置

FTP 服务器可以在主动或被动模式下运行。默认情况下，服务器在被动模式下运行。要切换到主动模式中，请取消选择专家设置对话框中的启用被动模式选项。也可更改服务器上用于数据流的端口范围，方法是：调整被动模式最小端口和被动模式最大端口选项。

如果想在客户端和服务器之间进行加密通讯，可以启用 SSL。检查要支持的协议版本，指定用于 SSL 加密连接的 DSA 证书。

如果您的系统受到防火墙的保护，请选中打开防火墙中的端口以启用至 FTP 服务器的连接。

33.6 更多信息

有关 FTP 服务器的更多信息，请参见 `vsftpd` 和 `vsftpd.conf` 的手册页。

34 代理服务器 Squid

Squid 是广泛用于 Linux 和 UNIX 平台的代理超速缓存。这表示它会将请求的因特网对象（例如 Web 或 FTP 服务器上的数据）储存在离请求工作站更近（与服务器相比）的计算机上。可以在多个层次中设置它，以确保即使在对最终用户来说是透明的模式下，也能达到最佳的响应速度和较低的带宽用量。可以使用其他诸如 squidGuard 的软件来过滤 Web 内容。

Squid 可以充当代理缓存。它将来自客户端（这里指来自 Web 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端并在硬盘缓存中保存对象副本。超速缓存的其中一个优点是，当多个客户端请求同一对象时，可以从硬盘超速缓存提供该对象。这样客户端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 还提供众多其他功能：

- 在代理服务器的互通层次之间分配负载
- 为所有访问代理的客户端定义严密的访问控制列表
- 允许或拒绝使用其他应用程序访问特定网页
- 生成有关频繁访问的网页的统计数字以评估用户的浏览习惯

Squid 不是通用代理。通常只充当 HTTP 连接的代理。它支持 FTP、Gopher、SSL 和 WAIS 协议，但不支持其他因特网协议，如新闻协议或视频会议协议。由于 Squid 只支持将 UDP 协议用于在不同的超速缓存间提供通讯功能，因此很多多媒体程序都不受支持。

34.1 有关代理缓存的一些事实

作为代理缓存，Squid 的使用方法分为几种。与防火墙结合使用时，能够提高安全性。可以一起使用多个代理。还能确定应该缓存的对象的类型以及缓存的时间。

34.1.1 Squid 和安全性

Squid 可以与防火墙结合起来，通过使用代理缓存防止内部网络遭受外部攻击。防火墙会拒绝 Squid 之外的所有客户端对外部服务的访问。所有 Web 连接都必须通过代理方式建立。经过此配置后，Squid 便可全面控制 Web 访问。

如果防火墙配置中包含 DMZ，代理应该在此区域内操作。第 34.6 节 “配置透明代理”介绍如何实施透明代理。它能简化客户端的配置，因为在这种情况下，它们不需要有关代理的任何信息。

34.1.2 多个缓存

可以配置 Squid 的几个实例从而在它们之间交换对象。这样会降低系统总负载，提高从本地网络检索对象的几率。还可以配置超速缓存层次，以便超速缓存能够将对象请求转发给同级超速缓存或父超速缓存，如此其可从本地网络中的其他超速缓存或直接从数据源请求对象。

为了不给网络增加总体数据流量，为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（因特网缓存协议）来处理。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。

为了找到最适合向其请求对象的服务器，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 响应回复这些请求。如果检测到对象，它们会使用代码 `HIT`，若未检测到，则会使用 `MISS`。

如果发现多个 `HIT` 响应，代理服务器会根据哪个超速缓存发送回复的速度最快或哪个服务器较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，该请求会发送至父超速缓存。

注意：Squid 如何避免对象重复

为了避免网络中不同缓存中的对象重复，还会使用其他 ICP 协议，如 CARP（缓存阵列路由协议）或 HTCP（超文本缓存协议）。网络中维护的对象越多，找到所需对象的可能性就越大。

34.1.3 缓存因特网对象

网络中提供的许多对象都不是静态的，例如动态生成的页面和 TLS/SSL 加密内容。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。

为确定对象应保留在超速缓存中的期限，系统会为对象指派多个状态中的一种。Web 和代理服务器会通过为这些对象添加报头来找出对象的状态（如“Last modified”或“Expires”）以及相应的日期。也可以使用指定不能超速缓存的对象的其他报头。

超速缓存中的对象通常会因缺少可用磁盘空间而通过 LRU（最近最少使用）之类的算法进行替换。这意味着代理会删除未被请求时间最长的对象。

34.2 系统要求

系统要求主要取决于系统必须承受的最大网络负载。由于负载在高峰期间可能是日均值的四倍以上，因此需要检查负载峰值。如果不能确定，请将系统要求稍微高估一点。让 Squid 在濒临其处理能力上限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：

1. RAM 大小
2. CPU 速度/物理 CPU 内核
3. 磁盘超速缓存的大小
4. 硬盘/SSD 及其体系结构

34.2.1 RAM

Squid 所需内存 (RAM) 大小与超速缓存中的对象数有直接的关系。随机存取内存比硬盘/SSD 快得多。因此，请务必让 Squid 进程拥有充足的内存，因为如果使用交换磁盘，系统性能会显著降低。

Squid 还会在主储存器中储存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。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、准确域名缓存、最常请求的对象、访问控制列表、缓冲区等等。

34.2.2 CPU

Squid 已经过优化，最适合处理器内核较少的环境（4–8 个物理内核），这样每个内核都能提供高性能。提供虚拟核心的技术（如超线程）可能会降低性能。

要充分利用多个 CPU 内核，必须设置多个写入不同超速缓存设备的工作线程。默认情况下，多核心支持通常都是禁用的。

34.2.3 磁盘缓存的大小

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 1 GB，而用户每天浏览所用的空间只有 10 MB，那么至少需要 100 多天才会占满超速缓存。

确定所需超速缓存大小的最简单的方法就是参考连接的最大传输速度。1 Mbit/s 连接的最大传输速度为 128 KB/s。如果所有流量都进入超速缓存，1 小时内累计可达到 460 MB。假设所有流量都是在 8 小时工作时间生成的，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，所以可以认为缓存处理的数据总量约为 2 GB。因此，在本示例中，Squid 需要 2 GB 的磁盘空间来保存一天内超速缓存的数据浏览量。

34.2.4 硬盘/SSD 体系结构

速度在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/SSD，此参数通过以毫秒度量的随机搜索时间或随机读取性能来描述。Squid 从硬盘/SSD 读取或写入其中的数据块一般都较小，因此硬盘/SSD 的搜索时间/读取性能比其数据吞吐量更重要。

如果要充当代理，高转速硬盘或者 SSD 会是最好的选择。使用硬盘时，使用多个较小硬盘的效果可能更好，这样每个硬盘都有单独的超速缓存目录，可避免读取次数过多。

如果使用 RAID 系统，可以通过牺牲速度来提高可靠性。不过，出于性能原因，请避免使用（软件）RAID5 及类似设置。

文件系统的选择通常无关紧要。不过，使用装入选项 `noatime` 可以提高性能 - Squid 提供其自己的时间戳，如此便不需要文件系统来跟踪访问时间。

34.3 Squid 的基本用法

如果尚未安装，请安装包 `squid` 。`SUSE® Linux Enterprise Server` 上默认不会安装 squid 包。

Squid 已在 SUSE Linux Enterprise Server 中预先配置好，因此安装完后即可直接启动。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，则可能出现问题。在这种情况下，至少应该指定名称服务器，因为如果在 `/etc/resolv.conf` 中未检测到 DNS 服务器，Squid 便不会启动。

34.3.1 启动 Squid

要启动 Squid，请使用：

```
tux > sudo systemctl start squid
```

如果想让 Squid 随系统一起启动，请使用 `systemctl enable squid` 启用该服务。

34.3.2 检查 Squid 是否正在工作

要检查 Squid 是否在运行，请选择以下其中一种方法：

- 使用 `systemctl`：

  ```
  tux > systemctl status squid
  ```

  此命令的输出应指出 Squid 已装载且处于活动状态（运行中）。

- 使用 Squid 本身：

  ```
  tux > sudo squid -k check | echo $?
  ```

  此命令的输出应当为 `0`，但也可能包含其他警告或讯息。

要测试 Squid 在本地系统上的功能，请选择以下其中一种方法：

- 要进行测试，您可以使用命令行工具 `squidclient`，它可向 Web 请求输出响应，类似于 `wget` 或 `curl`。

与这些工具不同的是，`squidclient` 将自动连接到 Squid 的默认代理设置 `localhost:3128`。不过，如果您更改过 Squid 的配置，则需要通过命令行选项将 `squidclient` 配置为使用不同的设置。有关详细信息，请参见 `squidclient --help`。

例 34.1 通过 `squidclient` 提交的请求

```
tux > squidclient http://www.example.org
HTTP/1.1 200 OK
Cache-Control: max-age=604800
Content-Type: text/html
Date: Fri, 22 Jun 2016 12:00:00 GMT
Expires: Fri, 29 Jun 2016 12:00:00 GMT
Last-Modified: Fri, 09 Aug 2013 23:54:35 GMT
Server: ECS (iad/182A)
Vary: Accept-Encoding
X-Cache: HIT
x-ec-custom-error: 1
Content-Length: 1270
X-Cache: MISS from moon (1)
X-Cache-Lookup: MISS from moon:3128
Via: 1.1 moon (squid/3.5.16) (2)
Connection: close

<!doctype html>
<html>
<head>
    <title>Example Domain</title>
[...]
</body>
</html>
```

例 34.1 “通过 `squidclient` 提交的请求”中显示的输出可分成两个部分：

1. 响应的协议报头：空白行前面的行。
2. 响应的实际内容：空白行后面的行。

要校验是否使用了 Squid，请参见所选的报头行：

(1) 报头 X-Cache 的值告诉您请求的文档不在计算机 moon 的 Squid 超速缓存中 (MISS)。
上面的示例包含两行 X-Cache。您可以忽略第一个 X-Cache 报头。它是由源 Web 服务器的内部超速缓存软件生成的。

(2) 报头 Via 的值告诉您 HTTP 版本、计算机名称以及所使用的 Squid 版本。

- 使用浏览器：将代理设置为 localhost 并将端口设置为 3128。然后您可以装载页面，并在浏览器的检查器或开发人员工具的网络面板中检查响应报头。报头应该会以例 34.1 “通过 squidclient 提交的请求”中所示的类似方式再现。

要允许用户从本地系统和其他系统访问 Squid 和因特网，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，需定义可控制对代理的访问权限的 ACL（访问控制列表）。修改配置文件后，必须重新装载或重启动 Squid。有关 ACL 的详细信息，请参见第 34.5.2 节 “访问控制选项”。

如果 Squid 在成功启动后不久就退出，请检查名称服务器项是否有误，或者是否缺少 /etc/resolv.conf 文件。Squid 会在 /var/log/squid/cache.log 文件中记录启动失败的原因。

34.3.3 停止、重新装载和重启动 Squid

要重新装载 Squid，请选择以下其中一种方法：

- 使用 systemctl：

```
root # systemctl reload squid
```

或者

```
root # systemctl restart squid
```

- 使用 YaST:
在 Squid 模块中，单击立即保存设置并重启动 Squid按钮.

要停止 Squid，请选择以下其中一种方法：

- 使用 `systemctl`：

```
root # systemctl stop squid
```

- 使用 YaST

  在 Squid 模块中，单击立即停止 Squid按钮.

关闭 Squid 可能需要一些时间，因为 Squid 会等待最长半分钟时间，来中断与客户端的连接并将其数据写入磁盘（请参见 `/etc/squid/squid.conf` 中的 `shutdown_lifetime` 选项）。

警告：终止 Squid

使用 `kill` 或 `killall` 终止 Squid 可能会损坏超速缓存。要能够重启动 Squid，必须删除损坏的超速缓存。

34.3.4 去除 Squid

从系统中去除 Squid 并不会去除超速缓存层次和日志文件。要删除这些内容，请手动删除 `/var/cache/squid` 目录。

34.3.5 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 root 名称服务器解析 DNS 请求（请参见 第 26.4 节 “启动 BIND 名称服务器”）。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`。可以在文件 `/etc/sysconfig/network/config` 中使用以下 sysconfig 变量控制此行为： `NETCONFIG_DNS_POLICY` 。设置 `NETCONFIG_DNS_POLICY` 更改为 `""` 。

然后在 `/etc/resolv.conf` 文件中添加本地 DNS 服务器，并将 `localhost` 的 IP 地址设置为 `127.0.0.1`。这样，Squid 每次一启动就能找到本地名称服务器。

为了使提供商的名称服务器可访问，请在配置文件 `/etc/named.conf` 中的 `forwarders` 下指定该名称服务器及其 IP 地址。如果使用动态 DNS，则可在建立连接时自动完成此操作，方法是将 sysconfig 变量 `NETCONFIG_DNS_POLICY` 更改为 `auto`。

静态 DNS

有了静态 DNS，在建立连接时自动 DNS 调整便不会发生，所以不需要更改任何 sysconfig 变量。但是，必须按动态 DNS中所述在 `/etc/resolv.conf` 文件中指定本地 DNS 服务器。此外，还必须在 `/etc/named.conf` 文件中的 `forwarders` 下手动指定提供商的静态名称服务器及其 IP 地址。

提示：DNS 和防火墙

如果运行了防火墙，应确保 DNS 请求能够通过。

34.4 YaST Squid 模块

YaST Squid 模块包含以下选项卡：

入门

指定启动 Squid 的方式，以及在哪些接口上打开哪个防火墙端口。

HTTP 端口

定义 Squid 将用来侦听客户端 HTTP 请求的所有端口。

刷新模式

定义 Squid 如何处理超速缓存中的对象。

超速缓存设置

定义有关超速缓存内存、最大和最小对象大小等的设置。

超速缓存目录

定义 Squid 将用来储存所有超速缓存交换文件的顶层目录。

访问控制

通过 ACL 组控制对 Squid 服务器的访问。

日志记录和超时

定义用于访问、超速缓存和超速缓存储存日志文件的路径，以及连接超时和客户端有效期。

杂项

设置管理员的语言和电子邮件地址。

34.5 Squid 配置文件

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，但是外部客户端最初不具备访问权。代理可供 `localhost` 使用。默认端口为 `3128`。预装的配置文件 `/etc/squid/squid.conf` 提供了有关选项的详细信息和许多示例。

许多条目都已注释掉，因此以注释字符 `#` 开头。行尾处提供了相关规范。给定值通常与默认值相关，因此仅去除注释符号而不更改任何参数通常没有什么影响。如果可能，请保留原始的注释行，在该行下方插入选项及修改过的值。这样，便可容易地恢复默认值，并将其与所作更改进行比较。

提示：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。

有时，该文件中会添加、去除或修改 Squid 选项。因此，如果尝试使用旧的 `squid.conf`，Squid 可能会无法正常工作。

34.5.1 一般配置选项

下面列出了 Squid 的一些配置选项，但并不详尽。`/etc/squid/squid.conf.documented` 中列出了 Squid 包的完整选项列表，其中仅做了简要记录。

`http_port` 端口

这是 Squid 侦听客户端请求所用的端口。默认端口为 `3128`，但也常使用 `8080`。

`cache_peer 主机名 类型 代理端口 ICP 端口`

此选项允许创建能够协同工作的超速缓存网络。超速缓存对等是一台也托管了网络超速缓存并与您自己的计算机有某种关系的计算机。*类型* 指定关系的类型。类型可以是 `parent` 或 `sibling`。

对于 *主机名*，请指定要使用的代理的名称或 IP 地址。对于 *代理端口*，请指定浏览器中要使用的端口号（通常为 `8080`）。请将 *ICP 端口* 设置为 `7`，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，则设置为 `0`。

要让 Squid 以 Web 浏览器而非代理的方式工作，请禁止使用 ICP 协议。您可以通过追加 `default` 和 `no-query` 选项来实现此目的。

`cache_mem 大小`

此选项定义 Squid 可用于常用答复的内存大小。默认为 `8 MB`。它不指定 Squid 的内存使用率，并且可能已经超过。

`cache_dir 储存类型 超速缓存目录 超速缓存大小 1 级目录 2 级目录`

`cache_dir` 选项定义磁盘超速缓存的目录。在 SUSE Linux Enterprise Server 上的默认配置中，Squid 不会创建磁盘超速缓存。

占位符 *储存类型* 可以是以下其中一个：

- 基于目录的储存类型：`ufs`、`aufs`（默认值）、`diskd`。这三种类型都是 `ufs` 储存形式的变体。不过，虽然 `ufs` 作为核心 Squid 线程的一部分运行，但 `aufs` 是在单独的线程中运行，而 `diskd` 则使用单独的进程。这表示后两种类型可避免因磁盘 I/O 而阻止 Squid。
- 基于数据库的储存系统：`rock`。此储存格式依赖于单一数据库文件，其中每个对象占用一个或多个固定大小的内存单元（“槽”）。

下文将只介绍基于 `ufs` 的储存类型的参数。`rock` 的参数有些不同。

超速缓存目录 是磁盘超速缓存的目录，默认为 `/var/cache/squid`。*超速缓存大小* 是该目录的最大大小（以兆字节为单位），默认设置为 100 MB。请将其设置为可用磁盘空间的 50% 到最大 80% 之间的值。

最后两个值（*1 级目录* 和 *2 级目录*）指定 *超速缓存目录* 中创建的子目录数量。默认情况下，在 *超速缓存目录* 下的第一级会创建 16 个子目录，在其中的每个子目录下创建 256 个子目录。提高这些值时应谨慎，因为创建过多的目录可能会导致性能问题。

如果有多个磁盘共享一个超速缓存，请指定数行 `cache_dir`。

`cache_access_log` 日志文件，
`cache_log` 日志文件，
`cache_store_log` 日志文件

这三个选项指定 Squid 记录其所有操作的路径。通常无需在这里进行任何更改。如果 Squid 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。

`client_netmask` 网络掩码

此选项允许通过应用子网掩码在日志文件中屏蔽客户端的 IP 地址。例如，要将 IP 地址的最后一位设置为 `0`，请指定 `255.255.255.0`。

`ftp_user` 电子邮件

此选项允许设置 Squid 应该用于匿名 FTP 登录的口令。请在此处指定有效的电子邮件地址，因为有些 FTP 服务器会检查这些信息来验证有效性。

`cache_mgr` 电子邮件

如果 Squid 意外崩溃，将会向此电子邮件地址发送一封邮件。默认为 webmaster。

`logfile_rotate` 值

如果运行 `squid -k rotate`，`Squid` 可以轮转日志文件。在此过程中会给文件编号，并且在达到指定值后重写最旧的文件。默认值为 `10`，即轮转编号为 0 到 9 的日志文件。不过，在 SUSE Linux Enterprise Server 上，日志文件的轮转是通过使用 `logrotate` 和配置文件 `/etc/logrotate.d/squid` 自动执行的。

`append_domain` 域

使用 append_domain 可指定当未指定域时自动追加的域。通常在此处指定您自己的域，因此在浏览器中指定 www 将访问您自己的 Web 服务器。

`forwarded_for` 状态

如果此选项设置为 `on`，将会向报头添加如下所示的一行：

```
X-Forwarded-For: 192.168.0.1
```

如果将此选项设置为 `off`，Squid 会将客户端的 IP 地址和系统名称从 HTTP 请求中去除。

`negative_ttl` 时间，
`negative_dns_ttl` 时间

如果设置了这些选项，Squid 将会超速缓存某些类型的失败，例如 `404` 响应。以后，它将拒绝发出新请求，即使当时资源可用。

默认情况下，negative_ttl 设置为 0，negative_dns_ttl 设置为 1 minute。这表示默认不会超速缓存 Web 请求的负响应，但会将 DNS 请求的负响应超速缓存 1 分钟。

never_direct allow ACL 名称

为防止 Squid 接受直接来自因特网的请求，可使用选项 never_direct 强制连接连到另一个代理。事先必须已在 cache_peer 中指定该代理。如果将 ACL 名称 指定为 all，则所有请求都将直接转发给 父代理。有时这可能是必要的，例如，您使用的提供商规定了其代理的使用方式或拒绝通过其防火墙直接访问因特网时。

34.5.2 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。这些访问控制列表 (ACL) 都是包含按顺序处理的规则的列表。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。只有存在相应的 http_acces 规则时，才会应用。

选项 acl 的语法如下所示：

```
acl ACL_NAME TYPE DATA
```

此语法中的占位符含义如下：

- 名称 ACL_NAME 可以任意选择。
- 对于 TYPE，/etc/squid/squid.conf 文件的 ACCESS CONTROLS 部分提供了多个不同的选项，您可以从中加以选择。
- DATA 的规格取决于单个 ACL 类型，也可从文件中读取。例如，“通过”主机名、IP 地址或 URL。

要在 YaST Squid 模块中添加规则，请打开该模块，然后单击访问控制选项卡。在“ACL 组”列表下单击添加，输入规则的名称、类型及其参数。

有关 ACL 规则类型的详细信息，请参见 http://www.squid-cache.org/Versions/v3/3.5/cfgman/acl.html 上的 Squid 文档。

例 34.2 定义 ACL 规则

```
acl mysurfers srcdomain .example.com ❶
```

```
acl teachers src 192.168.1.0/255.255.255.0 ❷
acl students src 192.168.7.0-192.168.9.0/255.255.255.0 ❸
acl lunch time MTWHF 12:00-15:00 ❹
```

❶ 此 ACL 将 `mysurfers` 定义为来自 `.example.com` 中的所有用户（由 IP 的反向查找确定）。

❷ 此 ACL 将 `teachers` 定义为计算机 IP 地址以 `192.168.1.` 开头的用户。

❸ 此 ACL 将 `students` 定义为计算机 IP 地址以 `192.168.7.`、`192.168.8.` 或 `192.168.9.` 开头的用户。

❹ 此 ACL 将 `lunch` 定义为星期一至星期五的中午到下午 3 点之间的某个时间。

http_access allow ACL 名称

`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。对于此选项，必须定义 ACL。上文中已经定义了 `localhost` 和 `all`，您可以通过 `deny` 或 `allow` 拒绝或允许对它们的访问。可以创建一个包含任意数量 `http_access` 条目的列表，按从上到下的顺序处理各条目。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，而所有其他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 students lunch time
http_access deny all
```

为提高可读性，请在配置文件 `/etc/squid/squid.conf` 中将所有 `http_access` 选项指定在一个段落中。

`url_rewrite_program 路径`

使用此选项可以指定 URL 重写程序。例如，它可以是 squidGuard (`/usr/sbin/squidGuard`)，用于阻止不需要的 URL。通过此选项，可以使用代理身份验证和适当的 ACL 针对不同用户组分别控制因特网访问。

有关 squidGuard 的详细信息，请参见第 34.8 节 “squidGuard”。

`auth_param basic program 路径`

如果必须在代理上对用户进行身份验证，请设置一个相应的程序，如 `/usr/sbin/pam_auth`。当首次访问 `pam_auth` 时，用户会看到一个需要指定用户名和口令的登录窗口。此外，您还需要一个 ACL，以便让只有有效登录的客户端才能使用因特网：

```
acl password proxy_auth REQUIRED

http_access allow password
http_access deny all
```

在 `acl proxy_auth` 选项中，使用 `REQUIRED` 表示接受所有有效的用户名。`REQUIRED` 还可替换为允许的用户名列表。

`ident_lookup_access allow ACL 名称`

使用此选项，可以运行 ident 请求，以便为类型为 `src` 的 ACL 定义的所有客户端确定各个用户的身份。或者，可对所有客户端使用此选项，至于 `ACL 名称`，则应用预定义的 ACL `all`。

`ident_lookup_access` 涵盖的所有客户端都必须运行 ident 守护程序。在 Linux 上，您可以将 `pidentd`（包 `pidentd`）用作 ident 守护程序。对于其他操作系统，通常有免费的软件可供使用。为确保只有 ident 查找成功的客户端才有权访问，请定义相应的 ACL：

```
acl identhosts ident REQUIRED

http_access allow identhosts
http_access deny all
```

在 `acl identhosts ident` 选项中，使用 `REQUIRED` 表示接受所有有效的用户名。`REQUIRED` 还可替换为允许的用户名列表。

使用 `ident` 会延缓访问时间，因为对每个请求都要重复 ident 查找过程。

34.6 配置透明代理

使用代理服务器的一般方式如下：Web 浏览器向代理服务器的某端口发送请求，代理始终会提供要求的这些对象，而不论它们是否在其超速缓存中。不过，在某些情况下，Squid 的透明代理模式很有效：

- 如果出于安全考虑，建议所有客户端都使用代理来浏览因特网。
- 如果所有客户端都必须使用代理，无论客户端是否清楚这一点。
- 如果网络中的代理已转移，但是现有客户端需要保留其原有配置。

透明代理会截获并回复 Web 浏览器的请求，因此 Web 浏览器可接收到所请求的页面，但并不知道它们来自何处。顾名思义，整个处理过程对用户而言是透明的。

过程 34.1 SQUID 充当透明代理 (命令行)

1. 在 `/etc/squid/squid.conf` 的选项 `http_port` 行中添加参数 `transparent`：

   ```
   http_port 3128 transparent
   ```

2. 重启动 Squid：

   ```
   tux > sudo systemctl restart squid
   ```

3. 设置 SuSEFirewall2 以将 HTTP 流量重定向到 `http_proxy` 中指定的端口（在上面的示例中为端口 3128）。为此，请编辑配置文件 `/etc/sysconfig/SuSEfirewall2`。本示例假设您使用的是以下设备：

   - 设备指向因特网：`FW_DEV_EXT="eth1"`
   - 设备指向网络：`FW_DEV_INT="eth0"`

   在防火墙上定义供不可信的（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 Web 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 Web 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。否则，只需从上面的条目中去除 `domain` 并将下面的选项设置为 `no` 即可：

```
FW_SERVICE_DNS="yes"
```

选项 `FW_REDIRECT` 非常重要，因为它用于将 HTTP 流量实际重定向到特定端口。配置文件中选项上方的注释中解释了其语法：

```
# Format:
# list of <source network>[,<destination
 network>,<protocol>[,dport[:lport]]
# Where protocol is either tcp or udp. dport is the original
# destination port and lport the port on the local machine to
# redirect the traffic to
#
# An exclamation mark in front of source or destination network
# means everything EXCEPT the specified network
```

即：

1. 指定访问代理防火墙的内部网络的 IP 地址和网络掩码。
2. 指定这些客户端的请求发往的 IP 地址和网络掩码。如果使用的是 Web 浏览器，请指定网络 `0/0`（表示“至任意地址的通配符”）。
3. 指定这些请求最初发送到的端口。
4. 指定所有这些请求要重定向到的端口。在下面的示例中，Web 服务（端口 `80`）重定向到代理端口（端口 `3128`）。如果要添加更多网络或服务，请在相应条目中用空格分隔它们。
   由于 Squid 支持 HTTP 以外的协议，您也可以将请求从其他端口重定向到该代理。例如，还可以重定向端口 21 （FTP) 和端口 443（HTTPS 或 SSL）。

因此，对于 Squid 配置，您可以使用：

```
FW_REDIRECT="192.168.0.0/16,0/0,tcp,80,3128"
```

4. 在配置文件 `/etc/sysconfig/SuSEfirewall2` 中，请确保 `START_FW` 这一项设置为 `"yes"`。

5. 重启动 SuSEFirewall2：

```
tux > sudo systemctl restart SuSEfirewall2
```

6. 要确认是否一切正常，请检查 `/var/log/squid/access.log` 中的 Squid 日志文件。要校验所有端口是否都已正确配置，请从网络外的任意计算机对本计算机上的端口进行扫描。只有 Web 服务（端口 80）应该是打开的。要使用 `nmap` 扫描端口，请使用：

```
nmap -O IP_ADDRESS
```

过程 34.2 SQUID 充当透明代理 (YAST)

1. 启动 YaST Squid 模块：

   a. 在启动选项卡中，启用在防火墙中打开端口。单击防火墙细节以选择要在其上打开端口的接口。此选项只有在启用防火墙后才可用。

   b. 在 HTTP 端口选项卡中，选择包含端口 `3128` 的第一行。

   c. 单击编辑按钮。此时会显示一个小窗口，您可在其中编辑当前的 HTTP 端口。选择透明。

   d. 单击 确认 完成。

2. 按照过程 34.1 “Squid 充当透明代理 (命令行)”的步骤 3 中所述配置防火墙设置。

34.7 使用 Squid 超速缓存管理器 CGI 接口 (cachemgr.cgi)

Squid 超速缓存管理器 CGI 接口 (`cachemgr.cgi`) 是一个 CGI 实用程序，用于显示正运行的 Squid 进程所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便捷方式。

过程 34.3 设置 `cachemgr.cgi`

1. 确保 Apache Web 服务器正在系统上运行。按第 32 章 “Apache HTTP 服务器”中所示配置 Apache。请着重参见第 32.5 节 “启用 CGI 脚本”。要检查 Apache 是否已在运行，请使用：

   ```
   tux > sudo systemctl status apache2
   ```

   如果显示 `inactive`，您可以使用 SUSE Linux Enterprise Server 默认设置启动 Apache：

   ```
   tux > sudo systemctl start apache2
   ```

2. 现在，在 Apache 中启用 `cachemgr.cgi`。要执行此操作，请为 `ScriptAlias` 创建一个配置文件。
   在目录 `/etc/apache2/conf.d` 中创建该文件并将其命名为 `cachemgr.conf`。在其中添加以下内容：

   ```
   ScriptAlias /squid/cgi-bin/ /usr/lib64/squid/

   <Directory "/usr/lib64/squid/">
   Options +ExecCGI
   AddHandler cgi-script .cgi
   Require host HOST_NAME
   </Directory>
   ```

   将 `HOST_NAME` 替换为您要从中访问 `cachemgr.cgi` 的计算机的主机名。这样，只有您的计算机才能访问 `cachemgr.cgi`。要允许从任何位置访问该文件，请改为使用 `Require all granted`。

3.
   - 如果 Squid 与 Apache Web 服务器在同一台计算机上运行，则无需对 `/etc/squid/squid.conf` 进行任何更改。但需校验 `/etc/squid/squid.conf` 是否包含以下几行：

     ```
     http_access allow manager localhost
     http_access deny manager
     ```

     这几行允许您从您自己的计算机 (`localhost`) 访问管理器界面，但不允许从其他位置访问。

   - 如果 Squid 与 Apache Web 服务器在不同的计算机上运行，您需要添加额外的规则以允许从 Squid 的 CGI 脚本访问。定义服务器的 ACL（将 `WEB_SERVER_IP` 替换为您 Web 服务器的 IP 地址）：

     ```
     acl webserver src WEB_SERVER_IP/255.255.255.255
     ```

     确保以下规则都包含在配置文件中。与默认配置相比，只有中间的规则是新的。不过，顺序很重要。

     ```
     http_access allow manager localhost
     http_access allow manager webserver
     http_access deny manager
     ```

4. （可选）您可以为 `cachemgr.cgi` 配置一个或多个口令。这样还可允许执行更多操作，例如远程关闭超速缓存或查看超速缓存的更多信息。为此，请使用管理器的一个或多个口令和允许的操作列表配置选项 `cache_mgr` 和 `cachemgr_passwd`。
   例如，若要明确允许不进行身份验证即可查看索引页、菜单、60 分钟计数器的平均值，允许使用口令 `secretpassword` 切换脱机模式，以及完全禁用任何其他功能，请使用以下配置：

   ```
   cache_mgr user
   cachemgr_passwd none index menu 60min
   cachemgr_passwd secretpassword offline_toggle
   cachemgr_passwd disable all
   ```

   `cache_mgr` 定义用户名。`cache_mgr` 定义使用哪个口令允许哪些操作。

`none` 和 `disable` 是特殊关键字：`none` 表示无需输入口令，`disable` 会完全禁用功能。

登录 `cachemgr.cgi` 后便可全面查看完整的操作列表。要了解配置文件中需如何参照操作，请查看操作页 URL 中 `&operation=` 后面的字符串。`all` 是一个特殊关键字，表示所有操作。

5. 配置文件更改后重新装载 Squid 和 Apache：

```
tux > sudo systemctl reload squid
```

6. 要查看统计数字，请转到先前设置的 `cachemgr.cgi` 页。例如 `http://webserver.example.org/squid/cgi-bin/cachemgr.cgi`。

选择正确的服务器，如已设置，请指定用户名和口令。然后单击继续浏览不同的统计数字。

34.8 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，请参见 squidGuard 的 网站 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 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 可以执行以下操作：

- 将某些用户的 Web 访问权限制为只能访问一组可接受的或知名 Web 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 Web 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、过滤或检查 HTML 嵌入脚本语言，如 JavaScript。

过程 34.4 设置 SQUIDGUARD

1. 在使用 squidGuard 之前，请先安装 `squidGuard` 。
2. 提供最小的配置文件，如 `/etc/squidguard.conf` 。可在 http://www.squidguard.org/Doc/examples.html 中找到配置示例。以后可尝试更为复杂的配置设置。
3. 接下来，创建一个“拒绝访问”HTML 页面或 CGI 页面，以便在客户端请求列在黑名单中的网站时，Squid 可重定向到该页面。强烈建议使用 Apache。
4. 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
```

5. 名为 `redirect_children` 的另一选项配置在该计算机上运行的“重定向”（在此例中是 squidGuard）进程数。设置的进程越多，所需的 RAM 就越多。先尝试使用较小的数字，例如 `4` 。

```
redirect_children 4
```

6. 最后，通过运行 `systemctl reload squid` 让 Squid 装载新配置。现在，可以通过浏览器测试这些设置。

34.9 使用 Calamaris 生成缓存报告

Calamaris 是一个 Perl 脚本，用来以 ASCII 或 HTML 格式生成缓存活动的报告。它可以处理本机 Squid 访问日志文件。Calamaris 的主页为 http://cord.de/calamaris-english 。此工具不属于 SUSE Linux Enterprise Server 默认安装范围 — 要使用它，请安装 `calamaris` 包。

以 `root` 身份登录，然后输入：

```
cat access1.log [access2.log access3.log] | calamaris OPTIONS > reportfile
```

使用多个日志文件时，请确保这些文件按时间顺序排列，较早的文件列在前面。这可以通过以下两种方式来实现：如上例一样逐个列出文件，或使用 `access{1..3}.log`。

`calamaris` 可使用以下选项：

`-a`
: 输出所有可用报告

`-w`
: 以 HTML 格式输出报告

`-l`
: 在报告标题处包含消息或徽标

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.{10..1} access.log | calamaris -a -w \
> /usr/local/httpd/htdocs/Squid/squidreport.html
```

这会将报告放入 Web 服务器目录。需要通过 Apache 来查看这些报告。

34.10 更多信息

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。

此外还通过 http://www.squid-cache.org/Support/mailing-lists.html 提供 Squid 的邮件列表。

35 使用 SFCB 的基于 Web 的企业管理

35.1 简介和基本概念

SUSE® Linux Enterprise Server (SLES) 提供了一组基于开放标准的工具，用于统一管理不同的计算系统和环境。我们的企业解决方案实现 Distributed Management Task Force 所提出的标准。以下段落描述了它们的基本组件。

Distributed Management Task Force, Inc (DMTF) 是业内引领企业和因特网环境管理标准开发的公司。他们的目标是统一管理标准和计划，开发出集成度更高、成本更低、可互操作性更好的管理解决方案。DMTF 标准为控制和通讯提供通用系统管理组件。他们的解决方案是独立于平台和技术的。基于 Web 的企业管理和通用信息模型是其关键技术之一。

基于 Web 的企业管理 (WBEM) 是一整套管理和因特网标准技术。开发 WBEM 的意图是统一企业计算环境的管理。它为该行业提供了使用 Web 技术并且良好集成的管理工具集。WBEM 由以下标准组成：

- 数据模型：通用信息模型 (CIM) 标准
- 编码规范：CIM-XML 编码规范
- 传输机制：通过 HTTP 的 CIM 操作

通用信息模型是描述系统管理的概念信息模型。它并不局限于特定的实施，能在管理系统、网络、服务和应用程序之间交换管理信息。CIM 有两部分 — CIM 规范和 CIM 纲要。

- CIM 规范描述语言、命名和元纲要。元模式是模型的正式定义。它定义用来表示模型及其用途以及语义的术语。元模式的元素有类、属性和方法。元纲要也支持使用指令和关联作为类的类型，使用参考作为属性的类型。
- CIM 纲要可提供实际模型描述。它提供具有属性和关联的类集合，这些类可提供易理解的概念框架，在该框架内可组织关于受管环境的可用信息。

“通用信息模型对象管理器”（CIMOM）是一种 CIM 对象管理器，或更确切地说，它是一个根据 CIM 标准管理对象的应用程序。CIMOM 管理 CIMOM 提供程序与 CIM 客户端（管理员管理系统的位置）之间的通讯。

CIMOM 提供程序是在 CIMOM 内执行客户端应用程序请求的特定任务的软件。每个提供者实施 CIMOM 模式的一个或多个方面。这些提供程序直接与硬件交互。

Standards Based Linux Instrumentation for Manageability (SBLIM) 是设计用来支持基于 Web 的企业管理 (WBEM) 的工具集。SUSE® Linux Enterprise Server 使用名为小规模 CIM 中介程序的 SBLIM 项目提供的开放源 CIMOM（或 CIM 服务器）。

小规模 CIM 中介程序是设计用于资源有限或嵌入式环境中的 CIM 服务器。它设计为同时保持模块化与轻量级。它是基于开放标准的，支持 CMPI 提供程序、CIM-XML 编码和管理对象格式 (MOF)。它可以灵活配置，并且即便提供程序崩溃也仍然表现稳定。它也很容易访问，因为它支持各种传输协议，例如 HTTP、HTTPS、Unix 域套接字、Service Location Protocol (SLP) 和 Java Database Connectivity (JDBC)。

35.2 设置 SFCB

要设置小规模 CIM 中介程序 (SFCB) 环境，请确保 SUSE Linux Enterprise Server 安装期间选择了 YaST 中的基于 Web 的企业管理模式。或者选择它作为组件安装到已在运行的服务器上。确保您的系统上安装了以下包：

cim-schema，通用信息模型 (CIM) 纲要
: 包含通用信息模型 (CIM)。CIM 是描述网络或企业环境中总体管理信息的模型。CIM 由规范和模式组成。规范定义与其他管理模型集成的详细信息。模式提供实际模型说明。

cmpi-bindings-pywbem
: 包含在 Python 中写入和运行 CMPI 类型的 CIM 提供程序的适配器。

cmpi-pywbem-base
: 包含基本系统 CIM 提供程序。

cmpi-pywbem-power-management
: 包含基于 DSP1027 的电源管理提供程序。

python-pywbem
: 包含通过 WBEM 协议调用 CIM 操作来查询和更新受管对象的 Phthon 模块。

cmpi-provider-register，CIMOM 中性提供程序注册实用程序
: 包含的实用程序允许 CMPI 提供程序包用 CIMOM 存在于系统上的任何内容注册。

sblim-sfcb，小规模 CIM 中介程序

包含小规模 CIM 中介程序。它是通过 HTTP 协议与 CIM 操作保持一致的 CIM 服务器。它功能强大，占用资源又少，因此很适合嵌入式和资源有限的环境。SFCB 支持通过 Common Manageability Programming Interface (CMPI) 写入的提供程序。

sblim-sfcc

包含小规模 CIM 客户端库运行时库。

sblim-wbemcli

包含 WBEM 命令行界面。它是一个独立的命令行 WBEM 客户端，特别适合基本的系统管理任务。

smis-providers

包含用于在 Linux 文件系统上处理卷和快照的提供程序。这些分别基于 SNIA 的 SMI-S 卷管理配置文件和 Copy Services 配置文件。

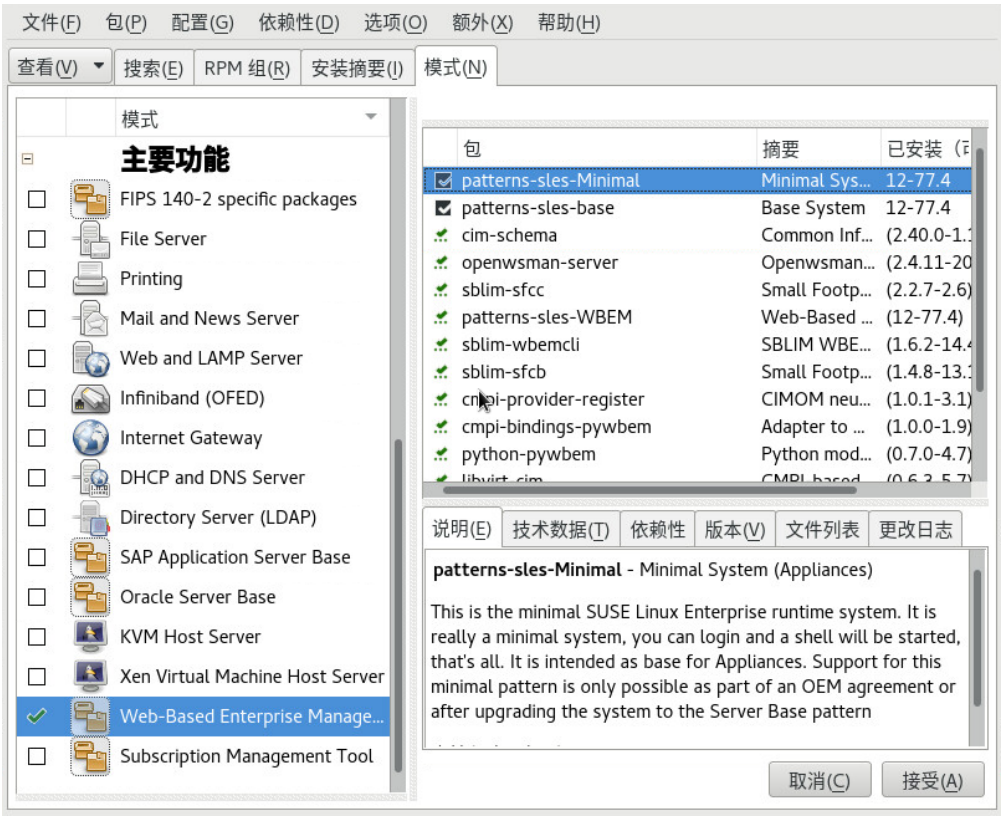


图 35.1 基于 WEB 的企业管理模式的包选择

35.2.1 安装其他提供程序

SUSE® Linux Enterprise Server 软件储存库包含基于 Web 的企业管理安装模式中没有的额外 CIM 提供程序。您可以在 YaST 软件安装模块中搜索模式 `sblim-cmpi-` 来方便地获取其列表和安装状态。这些提供程序涵盖了各种系统管理任务，例如 DHCP、NFS 或内核参数设置。安装要用于 SFCB 的那些提供程序是很有用的。

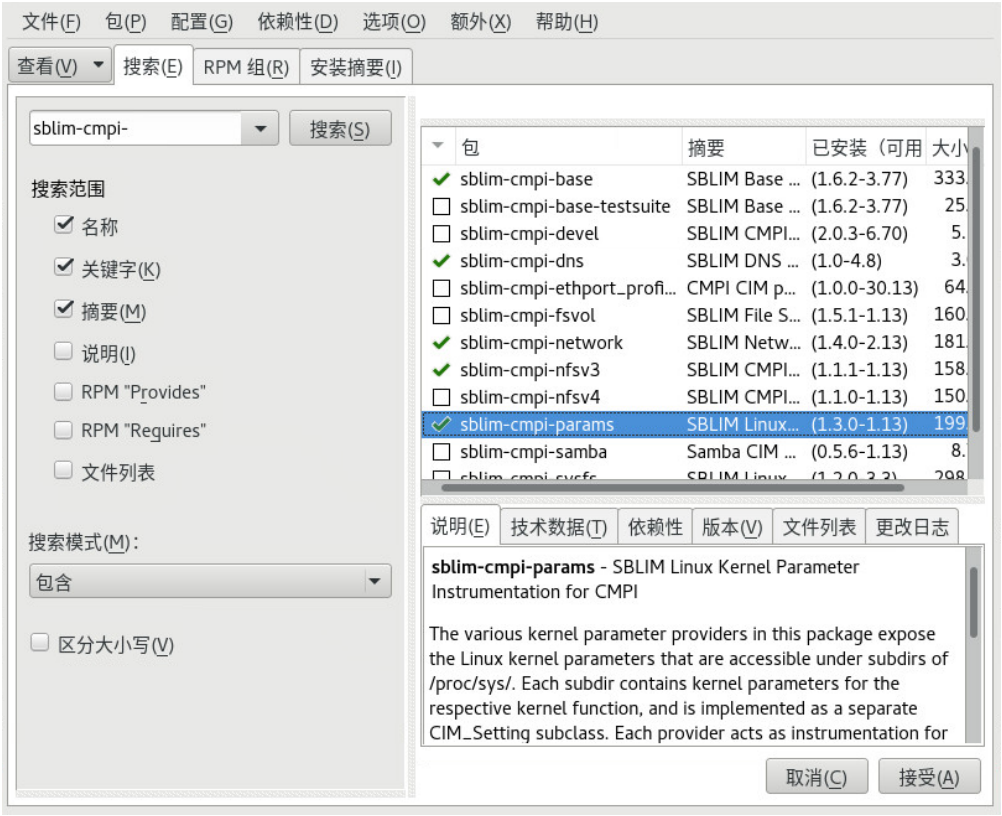


图 35.2 其他 CIM 提供程序的包选择

35.2.2 启动、停止 SFCB 和检查其状态

CIM 服务器 sfcbd 守护程序是和基于 Web 的企业管理软件一起安装的，默认会在系统启动时启动。下表描述如何启动、停止 sfcbd 和检查其状态。

表 35.1 用于管理 SFCBD 的命令

任务	Linux命令
启动 sfcbd	以 `root` 身份在命令行中输入 `systemctl start sfcb`。
停止 sfcbd	以 `root` 身份在命令行中输入 `systemctl stop sfcb`。
检查 sfcbd 状态	以 `root` 身份在命令行中输入 `systemctl status sfcb`。

35.2.3 确保安全访问

SFCB 的默认设置相对来说比较安全。但是也要检查对 SFCB 组件访问的安全性是否符合您的组织要求。

35.2.3.1 证书

安全套接字层 (SSL) 传输需要保证安全通讯的证书。安装 SFCB 时，会生成自我签名的证书。

您可以通过更改 `/etc/sfcb/sfcb.cfg` 中的 `sslCertificateFilePath: PATH_FILENAME` 设置，将默认证书的路径替换成商用证书或自我签名证书的路径。该文件必须是 PEM 格式。

默认生成的服务器证书位置如下：

`/etc/sfcb/server.pem`

注意：SSL 证书的路径

默认生成的证书文件 `servercert.pem` 和 `serverkey.pem` 储存在 `/etc/ssl/servercerts` 目录下。文件 `/etc/sfcb/client.pem`、`/etc/sfcb/file.pem` 和 `/etc/sfcb/server.pem` 是这些文件的符号链接。

要生成新证书，请以 `root` 身份在命令行中输入以下命令：

```
tux > sh /usr/share/sfcb/genSslCert.sh
Generating SSL certificates in .
Generating a 2048 bit RSA private key
..............................................................+++
.+++
writing new private key to '/var/tmp/sfcb.OBjt69/key.pem'
-----
```

默认情况下，该脚本在当前工作目录下生成证书 `client.pem`、`file.pem` 和 `server.pem`。如果希望脚本在 `/etc/sfcb` 目录中生成证书，需要将它追加到命令后。如这些文件已存在，将显示一条警告消息，而不会重写旧的证书。

```
tux > sh /usr/share/sfcb/genSslCert.sh /etc/sfcb
Generating SSL certificates in .
WARNING: server.pem SSL Certificate file already exists.
         old file will be kept intact.
WARNING: client.pem SSL Certificate trust store already exists.
         old file will be kept intact.
```

必须从文件系统删除旧的证书并重新运行命令。

要更改 SFCB 使用证书的方式，请参见第 35.2.3.3 节 “身份验证”。

35.2.3.2 端口

默认情况下，将 SFCB 配置为接受通过安全端口 5989 的所有通讯。以下段落描述通讯端口的设置和建议的配置。

端口 5989（安全）

SFCB 通讯通过 HTTPS 服务使用的安全端口。这是默认选项。通过此设置，当通过因特网在服务器和工作站之间发送通信时，将加密 CIMOM 和客户程序应用程序之间的所有通信。用户必须以客户端应用程序进行身份验证，才能连接到 SFCB 服务器。建议您保留该设置。如果客户端应用程序与所监视的节点之间存在路由器和防火墙，为了使 SFCB CIMOM 能够与必要的应用程序通讯，必须在路由器和防火墙上打开此端口。

端口 5988（不安全）

SFCB 通讯通过 HTTP 服务使用的不安全端口。在默认情况下会禁用该设置。通过此设置，当任何人在未经身份验证的情况下通过因特网在服务器和工作站之间发送通讯时，将打开并查看 CIMOM 和客户端应用程序之间的所有通讯。建议仅当尝试调试 CIMOM 问题时才使用此设置。在问题解决后，请再次禁用非安全端口选项。如果客户端应用程序与所监视的节点之间存在路由器和防火墙，为了使 SFCB CIMOM 能够与需要非安全访问的必要应用程序通讯，必须在路由器和防火墙中打开此端口。

如果希望更改默认端口指派，请参见第 35.2.3.2 节 “端口”。

35.2.3.3 身份验证

SFCB 支持 HTTP 基本身份验证和基于客户端证书的身份验证（通过 SSL 连接的 HTTP）。基本 HTTP 身份验证通过在 SFCB 配置文件（默认是 `/etc/sfcb/sfcb.cfg`）中指定 `doBasicAuth =true` 来启用。SFCB 的 SUSE® Linux Enterprise Server 安装支持可嵌入身份验证模块 (PAM) 方式；因此本地 root 用户可以用本地 root 身份凭证向 SFCB CIMOM 验证。

如果 `sslClientCertificate` 配置属性设置为 `accept` 或 `require`，SFCB HTTP 适配器通过 HTTP over SSL (HTTPS) 连接时，会从客户端请求证书。如果指定了 `require`，客户端必须提供有效的证书（根据通过 `sslClientTrustStore` 指定的客户端信任储存）。如果客户端未能提供，CIM 服务器将拒绝该连接。

`sslClientCertificate =accept` 设置可能不是显式的。如果同时允许基本和客户端证书身份验证，它将很有用。如果客户端能提供有效的证书，将建立 HTTPS 连接，并且不会执行基本身份验证过程。如果该功能不能校验证书，则会发生 HTTP 基本身份验证。

35.3 SFCB CIMOM 配置

SFCB 是 CIM 服务器的“轻量级”实现，但也是可以灵活配置的。有若干选项可控制其行为。您可以通过三种方式控制 SFCB 服务器：

- 设置相应的环境变量
- 使用命令行选项
- 更改其配置文件

35.3.1 环境变量

有若干环境变量会直接影响 SFCB 的行为。您必须使用 `systemctl restart sfcb` 重启动 SFCB 守护程序才能使这些更改生效。

`PATH`

指定 `sfcbd` 守护程序和实用程序的路径。

`LD_LIBRARY_PATH`

指定 sfcb 运行时库的路径。或者可以将此路径添加到系统范围的动态加载器配置文件 `/etc/ld.so.conf` 中。

`SFCB_PAUSE_PROVIDER`

指定提供程序的名称。SFCB 服务器会在第一次装载提供程序后暂停。然后您就可以将运行时调试程序加到提供程序的进程中，以便调试。

`SFCB_PAUSE_CODEC`

指定 SFCB 编解码器（目前仅支持 `http`）的名称。SFCB 服务器将在第一次装载编解码器后暂停。然后您就可以将运行时调试程序加到进程中。

`SFCB_TRACE`

为 SFCB 指定调试消息的级别。有效值有 0（无调试消息）、1（关键调试消息）和 4（全部调试消息）。默认值是 1。

`SFCB_TRACE_FILE`

默认情况下，SFCB 将其调试消息输出到标准错误输出 (STDERR)。设置该值会将调试消息写入指定的文件。

`SBLIM_TRACE`

为 SBLIM 提供程序指定调试消息级别。有效值有 0（无调试消息）、1（关键调试消息）和 4（全部调试消息）。

`SBLIM_TRACE_FILE`

默认情况下，SBLIM 提供程序将把跟踪消息输出到 STDERR。设置该值会将调试消息写入指定的文件。

35.3.2 命令行选项

SFCB 守护程序 `sfcbd` 有若干命令行选项，可以打开或关闭特定运行时功能。SFCB 守护程序启动时输入这些选项。

`-c, --config-file=FILE`

SFCB 守护程序启动时，默认会从 `/etc/sfcb/sfcb.cfg` 读取其配置。您可以用该选项指定备用配置文件。

`-d, --daemon`

强制 sfcbd 及其子进程在后台运行。

`-s, --collect-stats`

打开运行时统计数字收集。各种 sfcbd 运行时统计数字都会写入当前工作目录下的 `sfcbStat` 文件。默认情况下，不收集统计数字。

`-l, --syslog-level=LOGLEVEL`

指定系统日志记录工具的详细级别。`LOGLEVEL` 可以是 LOG_INFO、LOG_DEBUG 或 LOG_ERR（默认）。

`-k, --color-trace=日志级别`

以不同颜色打印每个进程的跟踪输出，以方便调试。

`-t, --trace-components=NUM`

激活组件级跟踪消息，其中 `NUM` 是进行 OR 运算的位掩码整数，它定义要跟踪哪个组件。您指定 `-t ?` 后，它会列出所有组件及其相关整数位掩码：

```
tux > sfcbd -t ?
---   Traceable Components:     Int       Hex
---            providerMgr:           1  0x0000001
---            providerDrv:           2  0x0000002
---             cimxmlProc:           4  0x0000004
---             httpDaemon:           8  0x0000008
---                upCalls:          16  0x0000010
---               encCalls:          32  0x0000020
---        ProviderInstMgr:          64  0x0000040
---       providerAssocMgr:         128  0x0000080
---              providers:         256  0x0000100
```

```
---             indProvider:        512  0x0000200
---       internalProvider:       1024  0x0000400
---              objectImpl:       2048  0x0000800
---                   xmlIn:       4096  0x0001000
---                  xmlOut:       8192  0x0002000
---                 sockets:      16384  0x0004000
---              memoryMgr:       32768  0x0008000
---                msgQueue:       65536  0x0010000
---              xmlParsing:      131072  0x0020000
---           responseTiming:      262144  0x0040000
---                dbpdaemon:      524288  0x0080000
---                      slp:     1048576  0x0100000
```

能反映 sfcbd 的内部功能，但不会产生过多消息的有用值是 `-t 2019`。

35.3.3 SFCB 配置文件

SFCB 启动后从配置文件 `/etc/sfcb/sfcb.cfg` 读取其运行时配置。该行为在启动时可以用 `-c` 选项覆盖。

配置文件包含 `option:VALUE` 对，一行一对。当对此文件执行更改时，如果文本编辑器保存文件的格式是所使用环境的本机格式，则可以使用此文本编辑器更改文件。

由井号 (#) 注释的选项的所有设置都使用默认设置。

以下选项列表可能不完整。查看 `/etc/sfcb/sfcb.cfg` 和 `/usr/share/doc/packages/sblim-sfcb/README` 的内容以获取其完整列表。

35.3.3.1 httpPort

用途

指定 sfcbd 用于侦听接收自 CIM 客户端的 HTTP（非安全）请求的本地端口值。默认值是 `5988`。

语法

`httpPort: PORT_NUMBER`

35.3.3.2 enableHttp

用途

指定 SFCB 是否接受 HTTP 客户端的连接。默认为 `false`。

语法

`enableHttp: OPTION`

选项	描述
true	启用 HTTP 连接。
false	禁用 HTTP 连接。

35.3.3.3 httpProcs

用途

指定在阻止新传入 HTTP 请求之前，HTTP 客户端连接的最大并行数。默认值是 `8`。

语法

`httpProcs: MAX_NUMBER_OF_CONNECTIONS`

35.3.3.4 httpUserSFCB、httpUser

用途

这些选项控制将运行 HTTP 服务器的用户。如果 `httpUserSFCB` 为 `true`，HTTP 将以 SFCB 主进程的同一用户身份运行。如果为 `false`，将使用为 `httpUser` 指定的用户名。此设置用于 HTTP 和 HTTPS 服务器。如果 `httpUserSFCB` 设置为 false，则必须指定 `httpUser`。默认值为 `true`。

语法

```
httpUserSFCB: true
```

35.3.3.5 httpLocalOnly

用途

指定是否将 HTTP 请求限制为仅 localhost。默认为 `false`。

语法

```
httpLocalOnly: false
```

35.3.3.6 httpsPort

用途

指定 sfcbd 侦听来自 CIM 客户端的 HTTPS 请求的本地端口值。默认值是 `5989`。

语法

```
httpsPort: 端口号
```

35.3.3.7 enableHttps

用途

指定 SFCB 是否接受 HTTPS 客户端连接。默认值是 `true`。

语法

`enableHttps: 选项`

选项	描述
true	启用 HTTPS 连接。
false	禁用 HTTPS 连接。

35.3.3.8 httpsProcs

用途

指定在阻止新传入 HTTPS 请求之前，HTTPS 客户端连接的最大并行数。默认为 `8`。

语法

`httpsProcs: MAX_NUMBER_OF_CONNECTIONS`

35.3.3.9 enableInterOp

用途

指定 SFCB 是否为指示支持提供 interop 名称空间。默认值是 `true`。

语法

`enableInterOp: OPTION`

选项	描述
true	启用 interop 名称空间。
false	禁用 interop 名称空间。

35.3.3.10 provProcs

用途

指定并发提供程序进程的最大数。达到这点之后，如果有新传入请求要求装载新提供程序，则现有提供程序之一将自动卸载。默认值是 `32`。

语法

`provProcs: MAX_NUMBER_OF_PROCS`

35.3.3.11 doBasicAuth

用途

根据客户端用户标识符打开或关闭它接受请求之前所做的基本身份验证。默认值为 `true`，表示执行基本客户端身份验证。

语法

`doBasicAuth: OPTION`

选项	描述
true	启用基本身份验证。
false	禁用基本身份验证。

35.3.3.12 basicAuthLib

用途

指定本地库名称。SFCB 服务器将装载该库以验证客户端用户标识符。默认值是 `sfcBasicPAMAuthentication`。

语法

`provProcs: MAX_NUMBER_OF_PROCS`

35.3.3.13 useChunking

用途

该选项启用或禁用 HTTP/HTTPS“分块”。如果启用，服务器将通过多个较小的“块”将大量响应数据返回客户端，而不是缓存数据并集中在一个块中全部送回。默认值是 `true`。

语法

`useChunking: OPTION`

选项	描述
true	启用 HTTP/HTTPS 数据分块。

选项	描述
false	禁用 HTTP/HTTPS 数据分块。

35.3.3.14 keepaliveTimeout

用途

指定一个连接上的 SFCB HTTP 进程在两次请求之间最多等待多少秒就终止。将它设置为 `0` 将禁用 HTTP keep-alive。默认值是 `0`。

语法

```
keepaliveTimeout: SECS
```

35.3.3.15 keepaliveMaxRequest

用途

指定一个连接上连续请求的最大数。将它设置为 `0` 将禁用 HTTP keep-alive。默认值是 `10`。

语法

```
keepaliveMaxRequest: NUMBER_OF_CONNECTIONS
```

35.3.3.16 registrationDir

用途

指定注册目录，它包含提供程序的注册数据、分阶段区域和静态储存库。默认值是 `/var/lib/sfcb/registration`。

语法

`registrationDir: DIR`

35.3.3.17 providerDirs

用途

指定 SFCB 搜索提供程序库的目录的空格分隔列表。默认值是 `/usr/lib64 /usr/lib64 /usr/lib64/cmpi`。

语法

`providerDirs: DIR`

35.3.3.18 providerSampleInterval

用途

指定提供程序管理器间隔多少秒检查空闲的提供程序。默认值是 `30`。

语法

`providerSampleInterval: SECS`

35.3.3.19 providerTimeoutInterval

用途

指定提供程序管理器经过多少秒的间隔就卸载空闲的提供程序。默认值是 `60`。

语法

`providerTimeoutInterval: SECS`

35.3.3.20 providerAutoGroup

用途

如果提供程序注册文件未指定任何其他组，而该选项设置为 `true`，则同一共享库内的所有提供程序都将在同一进程中执行。

语法

`providerAutoGroup: OPTION`

选项	描述
true	启用提供程序分组。
false	禁用提供程序分组。

35.3.3.21 sslCertificateFilePath

用途

指定包含服务器证书的文件的名称。该文件必须是 PEM (保密邮件，RFC 1421 和 RFC 1424) 格式。只有当 `enableHttps` 设置为 `true` 时才需要该文件。默认值是 `/etc/sfcb/server.pem`。

语法

`sslCertificateFilePath: PATH`

35.3.3.22 sslKeyFilePath

用途

指定包含服务器证书私用密钥的文件的名称。该文件必须是 PEM 格式且不能有通行口令的保护。该文件仅当 `enableHttps` 设置为 `true` 时才需要。默认值是 `/etc/sfcb/file.pem`。

语法

`sslKeyFilePath: PATH`

35.3.3.23 sslClientTrustStore

用途

指定包含客户端的 CA 或自我签名证书的文件的名称。该文件必须是 PEM 格式，只有当 `sslClientCertificate` 设置为 `accept` 或 `require` 时才需要。默认值是 `/etc/sfcb/client.pem`。

语法

`sslClientTrustStore: PATH`

35.3.3.24 sslClientCertificate

用途

指定 SFCB 处理基于客户端证书的身份验证的方式。如果设置为 `ignore`，将不会从客户端请求证书。如果设置为 `accept`，它会从客户端请求证书，但即使客户端没有证书也不会失败。如果设置为 `require`，会在客户端不存在证书时拒绝客户端连接。默认值是 `ignore`。

语法

`sslClientCertificate: OPTION`

选项	描述
ignore	禁止请求客户端证书。
accept	禁止请求客户端证书。 如果证书不存在，不会失败。
require	拒绝无有效证书的客户端连接。

35.3.3.25 certificateAuthLib

用途

指定用于请求基于客户端证书的用户身份验证的本地库名称。只有当 `sslClientCertificate` 未设置成 `ignore` 时才这样请求。默认值是 `sfcCertificateAuthentication`。

语法

`certificateAuthLib: FILE`

35.3.3.26 traceLevel

用途

指定 SFCB 的跟踪级别您可以通过设置环境变量 `SFCB_TRACE_LEVEL` 覆盖它。默认值是 `0`。

语法

`traceLevel: NUM_LEVEL`

35.3.3.27 traceMask

用途

指定 SFCB 的跟踪掩码。您可以用命令行选项 `--trace-components` 覆盖它。默认值是 `0`。

语法

`traceMask: MASK`

35.3.3.28 tracefile

用途

为 SFCB 指定跟踪文件。您可以通过设置环境变量 `SFCB_TRACE_FILE` 覆盖它。默认值是 `stderr`（标准错误输出）。

语法

`traceFile: OUTPUT`

35.4 高级 SFCB 任务

本章涵盖了与 SFCB 使用相关的更多高级主题。要了解这些主题，您需要有 Linux 文件系统的基础知识，并具有使用 Linux 命令行的经验。本章包括以下任务:

- 安装 CMPI 提供程序
- 测试 SFCB
- 使用 `wbemcli` CIM 客户端

35.4.1 安装 CMPI 提供程序

要安装 CMPI 提供程序，您必须确保其共享库已复制到 `providerDirs` 配置选项所指定的目录之一，参见第 35.3.3.17 节 “providerDirs”。提供程序还必须用 `sfcbstage` 和 `sfcbrepos` 命令正确注册。

提供程序包通常是为 SFCB 准备的，为此其安装过程负责进行正确的注册。多数 SBLIM 提供程序是为 SFCB 准备的。

35.4.1.1 类储存库

类储存库就是 SFCB 储存 CIM 类信息的位置。它通常由包含名称空间组件的目录树组成。典型的 CIM 名称空间如 `root/cimv2` 或 `root/interop`，它们会分别转换为文件系统上的类储存库目录路径

`/var/lib/sfcb/registration/repository/root/cimv2`

和

`/var/lib/sfcb/registration/repository/root/interop`

每个名称空间目录都包含文件 `classSchemas`。该文件中有该名称空间下注册的所有 CIM 类的已编译二进制表示。它还包含有关其 CIM 超类的必要信息。

此外，每个名称空间目录都可能包含文件 `qualifiers`，其中包含了该名称空间的所有限定符。sfcbd 重启动时，类提供程序会扫描目录 `/var/lib/sfcb/registration/repository/` 及其全部子目录，以确定注册的名称空间。然后将解码 `classSchemas` 文件，为每个名称空间构建类的层次结构。

35.4.1.2 添加新类

SFCB 不能在线处理 CIM 类。您需要脱机添加、更改或去除类，然后使用 `systemctl restart sfcb` 重启动 SFCB 服务来注册这些更改。

为储存提供程序类和注册信息，SFCB 使用名为分阶段区域的位置。在 SUSE® Linux Enterprise Server 系统上，它就是 `/var/lib/sfcb/stage/` 下的目录结构。

为添加新的提供程序，您必须：

- 将提供程序类定义文件复制到分阶段区域目录的子目录 `./mofs`（`/var/lib/sfcb/stage/mofs`）。
- 将包含类名、提供程序类型和可执行库文件名的注册文件复制到 `./regs` 子目录。

分阶段目录中有两个默认的 “mof”（类定义）文件：`indication.mof` 和 `interop.mof`。运行 `sfcbrepos` 命令后，root 阶段目录 `/var/lib/sfcb/stage/mofs` 下的 MOF 文件将复制到每个名称空间。`interop.mof` 将只编译到 interop 名称空间。

目录布局看上去如下所示：

```
tux > ls /var/lib/sfcb/stage
default.reg  mofs  regs
```

```
tux > ls /var/lib/sfcb/stage/mofs
indication.mof  root
```

```
tux > ls /var/lib/sfcb/stage/mofs/root
cimv2  interop  suse  virt
```

```
tux > ls -1 /var/lib/sfcb/stage/mofs/root/cimv2 | less
Linux_ABIParameter.mof
Linux_BaseIndication.mof
Linux_Base.mof
Linux_DHCPElementConformsToProfile.mof
Linux_DHCPEntity.mof
[..]
OMC_StorageSettingWithHints.mof
OMC_StorageVolumeDevice.mof
OMC_StorageVolume.mof
OMC_StorageVolumeStorageSynchronized.mof
```

```
OMC_SystemStorageCapabilities.mof
```

```
tux > ls -1 /var/lib/sfcb/stage/mofs/root/interop
ComputerSystem.mof
ElementConformsToProfile.mof
HostSystem.mof
interop.mof
Linux_DHCPElementConformsToProfile.mof
[..]
OMC_SMIElementSoftwareIdentity.mof
OMC_SMISubProfileRequiresProfile.mof
OMC_SMIVolumeManagementSoftware.mof
ReferencedProfile.mof
RegisteredProfile.mof
```

```
tux > ls -1 /var/lib/sfcb/stage/regs
AllocationCapabilities.reg
Linux_ABIParameter.reg
Linux_BaseIndication.reg
Linux_DHCPGlobal.reg
Linux_DHCPRegisteredProfile.reg
[..]
OMC_Base.sfcb.reg
OMC_CopyServices.sfcb.reg
OMC_PowerManagement.sfcb.reg
OMC_Server.sfcb.reg
RegisteredProfile.reg
```

```
tux > cat /var/lib/sfcb/stage/regs/Linux_DHCPRegisteredProfile.reg
[Linux_DHCPRegisteredProfile]
   provider: Linux_DHCPRegisteredProfileProvider
   location: cmpiLinux_DHCPRegisteredProfile
   type: instance
   namespace: root/interop
#
[Linux_DHCPElementConformsToProfile]
   provider: Linux_DHCPElementConformsToProfileProvider
   location: cmpiLinux_DHCPElementConformsToProfile
```

```
   type: instance association
   namespace: root/cimv2
#
[Linux_DHCPElementConformsToProfile]
   provider: Linux_DHCPElementConformsToProfileProvider
   location: cmpiLinux_DHCPElementConformsToProfile
   type: instance association
   namespace: root/interop
```

SFCB 对每个提供程序使用自定义提供程序注册文件。

注意：SBLIM 提供程序注册文件

SBLIM 网站上的所有 SBLIM 提供程序都已包括用于生成 SFCB 的 .reg 文件的注册文件。

SFCB 注册文件的格式是：

```
[<class-name>]
   provider: <provide-name>
   location: <library-name>
   type: [instance] [association] [method] [indication]
   group: <group-name>
   unload: never
   namespace: <namespace-for-class> ...
```

其中：

`<class-name>`

CIM 类名（必需）

`<provider-name>`

CMPI 提供程序名称（必需）

`<location-name>`

提供程序库的名称（必需）

`type`

提供程序的类型（必需）它可以是以下的任意组合：`instance`、`association`、`method` 或 `indication`。

`<group-name>`

可以将多个提供程序组合起来，在单一进程下运行，以进一步最小化运行时资源的占用。所有注册在同一 <group-name> 下的提供程序都将在同一进程下执行。默认情况下每个提供程序作为单独的进程运行。

`unload`

指定提供程序的卸载策略。目前唯一支持的选项是 `never`，即指定不监视提供程序的空闲次数，也从不卸载。默认情况下，每个提供程序只要空闲次数超过配置文件中指定的值就卸载。

`namespace`

可执行本提供程序的名称空间的列表。这是必需的，尽管对于多数提供程序来说它就是 `root/cimv2`。

所有类定义和提供程序注册文件都储存在分阶段区域后，就需要用命令 `sfcbrepos -f` 重构建 SFCB 类储存库。

您可以用这种方式添加、更改或删除类。重构建类储存库后，用命令 `systemctl restart sfcb` 重启动 SFCB。

或者，SFCB 包包含一个实用程序，它会将提供程序类 mof 文件和注册文件复制到分阶段区域中正确的位置。

`sfcbstage -r [provider.reg] [class1.mof] [class2.mof] ...`

运行该命令后，您还需要重构建类储存库并重启动 SFCB 服务。

35.4.2 测试 SFCB

SFCB 包包括两个测试脚本：`wbemcat` 和 `xmltest`。

`wbemcat` 通过 HTTP 协议将原始 CIM-XML 数据发送到端口 5988 上正在侦听的指定 SFCB 主机（默认为本地主机）。然后它会显示返回的结果以下文件包含标准 EnumerateClasses 请求的 CIM-XML 表示：

```
<?xml version="1.0" encoding="utf-8"?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
  <MESSAGE ID="4711" PROTOCOLVERSION="1.0">
    <SIMPLEREQ>
      <IMETHODCALL NAME="EnumerateClasses">
```

```
        <LOCALNAMESPACEPATH>
          <NAMESPACE NAME="root"/>
          <NAMESPACE NAME="cimv2"/>
        </LOCALNAMESPACEPATH>
        <IPARAMVALUE NAME="ClassName">
          <CLASSNAME NAME=""/>
        </IPARAMVALUE>
        <IPARAMVALUE NAME="DeepInheritance">
          <VALUE>TRUE</VALUE>
        </IPARAMVALUE>
        <IPARAMVALUE NAME="LocalOnly">
          <VALUE>FALSE</VALUE>
        </IPARAMVALUE>
        <IPARAMVALUE NAME="IncludeQualifiers">
          <VALUE>FALSE</VALUE>
        </IPARAMVALUE>
        <IPARAMVALUE NAME="IncludeClassOrigin">
          <VALUE>TRUE</VALUE>
        </IPARAMVALUE>
      </IMETHODCALL>
    </SIMPLEREQ>
  </MESSAGE>
</CIM>
```

将该请求发送到 SFCB CIMOM 后，会返回具有已注册提供程序的所有支持类的列表。假定您将文件保存为 `cim_xml_test.xml`。

```
tux > wbemcat cim_xml_test.xml | less
HTTP/1.1 200 OK
Content-Type: application/xml; charset="utf-8"
Content-Length: 337565
Cache-Control: no-cache
CIMOperation: MethodResponse

<?xml version="1.0" encoding="utf-8" ?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
<MESSAGE ID="4711" PROTOCOLVERSION="1.0">
<SIMPLERSP>
<IMETHODRESPONSE NAME="EnumerateClasses">
```

```
[..]
<CLASS NAME="Linux_DHCPParamsForEntity" SUPERCLASS="CIM_Component">
<PROPERTY.REFERENCE NAME="GroupComponent" REFERENCECLASS="Linux_DHCPEntity">
</PROPERTY.REFERENCE>
<PROPERTY.REFERENCE NAME="PartComponent" REFERENCECLASS="Linux_DHCPParams">
</PROPERTY.REFERENCE>
</CLASS>
</IRETURNVALUE>
</IMETHODRESPONSE>
</SIMPLERSP>
</MESSAGE>
</CIM>
```

列出的类会随您在系统上安装的提供程序的不同而不同。

第二个脚本 `xmltest` 也用于将原始 CIM-XML 测试文件发送到 SFCB CIMOM。它会随即比较返回的结果和以前保存的“OK”结果文件。如果还不存在对应的“OK”文件，将创建它以备后用：

```
tux > xmltest cim_xml_test.xml
Running test cim_xml_test.xml ... OK
        Saving response as cim_xml_test.OK
root # xmltest cim_xml_test.xml
Running test cim_xml_test.xml ... Passed
```

35.4.3 命令行 CIM 客户端：wbemcli

除了 `wbemcat` 和 `xmltest` 外，SBLIM 项目包含更高级的命令行 CIM 客户端 `wbemcli`。该客户端用于将 CIM 请求发送到 SFCB 服务器，并显示返回的结果。它独立于 CIMOM 库，可与所有 WBEM 兼容实现一起使用。

例如，如果您需要列出注册到您的 SFCB 的 SBLIM 提供程序实现的所有类，则将“EnumerateClasses” (ec) 请求发送到 SFCB：

```
tux > wbemcli -dx ec http://localhost/root/cimv2
To server: <?xml version="1.0" encoding="utf-8" ?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
<MESSAGE ID="4711" PROTOCOLVERSION="1.0"><SIMPLEREQ><IMETHODCALL \
```

```
    NAME="EnumerateClasses"><LOCALNAMESPACEPATH><NAMESPACE NAME="root"> \
    </NAMESPACE><NAMESPACE NAME="cimv2"></NAMESPACE> \
    </LOCALNAMESPACEPATH>
<IPARAMVALUE NAME="DeepInheritance"><VALUE>TRUE</VALUE> \
    </IPARAMVALUE>
<IPARAMVALUE NAME="LocalOnly"><VALUE>FALSE</VALUE></IPARAMVALUE>
<IPARAMVALUE NAME="IncludeQualifiers"><VALUE>FALSE</VALUE> \
    </IPARAMVALUE>
<IPARAMVALUE NAME="IncludeClassOrigin"><VALUE>TRUE</VALUE> \
    </IPARAMVALUE>
</IMETHODCALL></SIMPLEREQ>
</MESSAGE></CIM>
From server: Content-Type: application/xml; charset="utf-8"
From server: Content-Length: 337565
From server: Cache-Control: no-cache
From server: CIMOperation: MethodResponse
From server: <?xml version="1.0" encoding="utf-8" ?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
<MESSAGE ID="4711" PROTOCOLVERSION="1.0">
<SIMPLERSP>
<IMETHODRESPONSE NAME="EnumerateClasses">
<IRETURNVALUE>
<CLASS NAME="CIM_ResourcePool" SUPERCLASS="CIM_LogicalElement">
<PROPERTY NAME="Generation" TYPE="uint64">
</PROPERTY>
<PROPERTY NAME="ElementName" TYPE="string">
</PROPERTY>
<PROPERTY NAME="Description" TYPE="string">
</PROPERTY>
<PROPERTY NAME="Caption" TYPE="string">
</PROPERTY>
<PROPERTY NAME="InstallDate" TYPE="datetime">
</PROPERTY>
[..]
<CLASS NAME="Linux_ReiserFileSystem" SUPERCLASS="CIM_UnixLocalFileSystem">
<PROPERTY NAME="FSReservedCapacity" TYPE="uint64">
</PROPERTY>
<PROPERTY NAME="TotalInodes" TYPE="uint64">
```

```
</PROPERTY>
<PROPERTY NAME="FreeInodes" TYPE="uint64">
</PROPERTY>
<PROPERTY NAME="ResizeIncrement" TYPE="uint64">
<VALUE>0</VALUE>
</PROPERTY>
<PROPERTY NAME="IsFixedSize" TYPE="uint16">
<VALUE>0</VALUE>
</PROPERTY>
[..]
```

`-dx` 选项向您显示 `wbemcli` 发送到 SFCB 的实际 XML，以及收到的实际 XML。在以上示例中，众多返回类中的第一个就是 `CIM_ResourcePool`，后面是 `Linux_ReiserFileSystem`。类似的条目会对所有其他注册的类显示。

如果您省略 `-dx` 选项，`wbemcli` 会显示返回数据的简化表示：

```
tux > wbemcli ec http://localhost/root/cimv2
localhost:5988/root/cimv2:CIM_ResourcePool Generation=,ElementName=, \
    Description=,Caption=,InstallDate=,Name=,OperationalStatus=, \
    StatusDescriptions=,Status=,HealthState=,PrimaryStatus=, \
    DetailedStatus=,OperatingStatus=,CommunicationStatus=,InstanceID=, \
    PoolID=,Primordial=,Capacity=,Reserved=,ResourceType=, \
    OtherResourceType=,ResourceSubType=, \AllocationUnits=
localhost:5988/root/cimv2:Linux_ReiserFileSystem FSReservedCapacity=, \
    TotalInodes=,FreeInodes=,ResizeIncrement=,IsFixedSize=,NumberOfFiles=, \
    OtherPersistenceType=,PersistenceType=,FileSystemType=,ClusterSize=, \
    MaxFileNameLength=,CodeSet=,CasePreserved=,CaseSensitive=, \
    CompressionMethod=,EncryptionMethod=,ReadOnly=,AvailableSpace=, \
    FileSystemSize=,BlockSize=,Root=,Name=,CreationClassName=,CSName=, \
    CSCreationClassName=,Generation=,ElementName=,Description=,Caption=, \
    InstanceID=,InstallDate=,OperationalStatus=,StatusDescriptions=, \
    Status=,HealthState=,PrimaryStatus=,DetailedStatus=,OperatingStatus= \
    ,CommunicationStatus=,EnabledState=,OtherEnabledState=,RequestedState= \
    ,EnabledDefault=,TimeOfLastStateChange=,AvailableRequestedStates=, \
    TransitioningToState=,PercentageSpaceUse=
    [..]
```

35.5 更多信息

有关 WBEM 和 SFCB 的细节，请参见以下来源：

http://www.dmtf.org

Distributed Management Task Force 网站

http://www.dmtf.org/standards/wbem/

Web-Based Enterprise Management (WBEM) 网站

http://www.dmtf.org/standards/cim/

通用信息模型 (CIM) 网站

http://sblim.wiki.sourceforge.net/

Standards Based Linux Instrumentation (SBLIM) 网站

http://sblim.sourceforge.net/wiki/index.php/Sfcb

小规模 CIM 中介程序 (SFCB) 网站

http://sblim.sourceforge.net/wiki/index.php/Providers

SBLIM 提供程序包

V 移动计算机

36 Linux 中的移动计算 537

37 使用 NetworkManager 547

38 电源管理 557

36 Linux 中的移动计算

移动计算主要与便携式计算机、PDA 和手提电话（以及它们之间的数据交换）关联。移动硬件部件（如外部硬盘、闪存盘或数码相机）可连接到便携式计算机或台式机。移动计算方案中涉及了许多软件组件，一些应用程序是专门为移动定制的。

36.1 便携式计算机

便携式计算机的硬件不同于普通台式机的硬件。这是因为必须考虑可交换性、空间要求和能耗等条件。移动硬件的制造商已开发了标准接口，如可用于扩展便携式计算机硬件的 PCMCIA（个人计算机内存卡国际协会）、迷你 PCI 和迷你 PCIe。这些标准涉及内存卡、网络接口卡和外部硬盘。

36.1.1 省电

由于在制造便携式计算机时加入了能量优化系统组件，这使得不必连接电源线即可使用便携式计算机。这些部件在省电方面所起的作用并不亚于操作系统。SUSE® Linux Enterprise Server 支持各种控制便携式计算机能耗的方法，在使用电池供电时，这些方法对计算机运行时间的影响各不相同。下面的列表按照省电作用从高到低排列：

- 节制 CPU 流速。
- 在暂停期间关闭显示器。
- 手动调节显示器亮度。
- 将不使用的可热插拔配件（USB CD-ROM、外部鼠标、不使用的 PCMCIA 卡、Wi-Fi 等）断开连接。
- 在硬盘闲置时降低其转速。

有关 SUSE Linux Enterprise Server 电源管理的详细背景信息，请参见第 38 章 “电源管理”。

36.1.2 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。很多服务都依赖于环境，而且必须重配置底层客户端。SUSE Linux Enterprise Server 将为您处理这项任务。

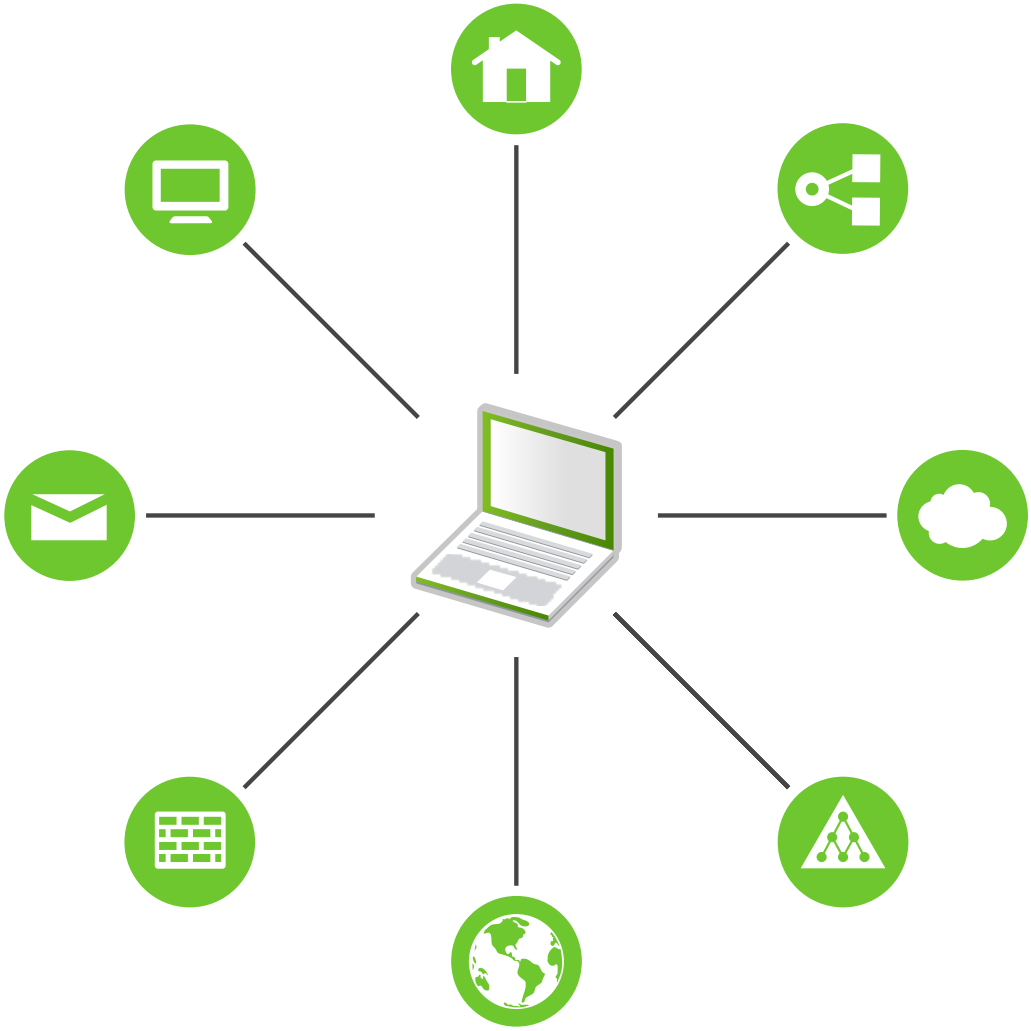


图 36.1 在现有环境中集成移动计算机

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

这包括 IP 地址分配、名称解析、因特网连接以及与其他网络的连接。

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。

X（图形环境）

如果您的便携式计算机暂时连接到投影仪或外部监视器，则需要有其他显示配置。

SUSE Linux Enterprise Server 提供了多种将便携式计算机集成到现有操作环境的方法：

NetworkManager

它让您能够轻松地在网络环境之间或不同网络类型之间进行自动切换，例如，移动宽带（如 GPRS、EDGE 或 3G）、无线 LAN 和以太网。NetworkManager 在无线 LAN 中支持 WEP 和 WPA-PSK 加密。它也支持拨号连接。GNOME 桌面包含 NetworkManager 的前端。有关详细信息，请参见第 37.3 节 “配置网络连接”。

表 36.1 NETWORKMANAGER 的用例

我的计算机…	使用 NetworkManager
是便携式计算机	是
有时与不同网络连接	是
提供网络服务（例如 DNS 或 DHCP）	否
仅使用静态 IP 地址	否

只要不应该使用 NetworkManager 来处理网络配置，便使用 YaST 工具配置联网。

提示：DNS 配置和各种类型的网络连接

如果您经常携带便携式计算机出行并需要更改不同类型的网络连接，NetworkManager 将是您的好助手，只要所有 DNS 地址均已使用 DHCP 正确指派。如果一些连接使用静态 DNS 地址，请将其添加到 `/etc/sysconfig/network/config` 中的 `NETCONFIG_DNS_STATIC_SERVERS` 选项。

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端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 还可用于安装系统，从而最大限度地减少搜索合适安装源的工作量。有关 SLP 的更多详细信息，请参见第 31 章 “SLP”。

36.1.3 软件选择

在移动使用场合有各种任务领域，它们由专用软件实现：系统监控（特别是电池充电）、数据同步和与外围设备及因特网的无线通讯。以下章节介绍了 SUSE Linux Enterprise Server 为各项任务提供的最为重要的应用程序。

36.1.3.1 系统监视

SUSE Linux Enterprise Server 提供了两个系统监视工具：

电源管理

电源管理是可让您调整 GNOME 桌面的节能行为的应用程序。通常您可以通过计算机 › 控制中心 › 系统 › 电源管理来访问该应用程序。

系统监视程序

系统监视器将可测量的系统参数收集到一个监视环境中。它默认会在三个选项卡中显示输出信息。进程提供有关当前正在运行的进程的详细信息，比如，CPU 负载、内存使用情况或进程 ID 号和优先级。您可以自定义所收集数据的显示和过滤方式 — 要添加新类型的进程信息，请左键单击进程表标题，并选择要隐藏或要添加到视图中的列。也可以监视不同数据页中的不同系统参数，或跨网络并行收集不同计算机上的数据。资源选项卡显示 CPU、内存和网络历史的图表，文件系统选项卡列出所有分区及其使用情况。

36.1.3.2 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，则需要在所有实例间保持同步处理数据。这些数据可能包括出差时以及在公司内部需要用到的电子邮件文件夹、目录和单个文件。适用于这两种情况的解决方案如下：

同步电子邮件

在办公室网络中使用 IMAP 帐户储存电子邮件。随后可以从工作站使用任意断开连接但支持 IMAP 的电子邮件客户端（如 Mozilla Thunderbird 或 Evolution）来访问这些电子邮件，如《GNOME 用户指南》中所述。必须对电子邮件客户端进行配置，以便始终从同一文件夹访问 已发送邮件。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。使用邮件客户端中实施的 SMTP 服务器（而非系统范围的 MTA Postfix 或 Sendmail）来发送邮件，从而可收到有关未发送邮件的可靠反馈。

同步文件和目录

有多个实用程序适合在便携式计算机和工作站之间同步数据。使用最广泛的一种实用程序是称为 `rsync` 的命令行工具。有关更多信息，请参见其手册页 (`man 1 rsync`)。

36.1.3.3 无线通讯：Wi-Fi

Wi-Fi 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网络（有时甚至是在空间上分离的网络）的操作技术。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。称为接入点的设备是作为支持 Wi-Fi 的设备的基站，而且充当着访问因特网的中介角色。移动用户可以在多个接入点之间切换，这取决于所在位置以及哪个接入点提供的连接最佳。类似移动电话的情况，Wi-Fi 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。

Wi-Fi 卡使用 IEEE 组织提供的 802.11 标准通讯。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细节（请参见表 36.2 “各种 Wi-Fi 标准的概述”）。此外，许多公司实施了带专有或设计功能的硬件。

表 36.2 各种 WI-FI 标准的概述

名称 (802.11)	频率 (GHz)	最大传送速率 (MBit/s)	记事
a	5	54	不易受干扰
b	2.4	11	较少使用
g	2.4	54	广泛采用，向后兼容 11b

名称 (802.11)	频率 (GHz)	最大传送速率 (MBit/s)	记事
n	2.4 和/或 5	300	常用
ac	5	最高大约 865	预期在 2015 年会广泛使用
ad	60	最高大约 7000	2012 年发行，当前较少使用；不受 SUSE Linux Enterprise Server 支持

SUSE® Linux Enterprise Server 不支持 802.11 旧式卡。支持使用 802.11 a/b/g/n 的大多数卡。新卡通常符合 802.11n 标准，但是使用 802.11g 的卡仍然可用。

36.1.3.3.1 操作方式

在无线联网中，会使用各种技术和配置来确保连接的快速、高质量和安全。Wi-Fi 卡通常在受管模式下运行。但是，不同的操作类型需要不同的设置。无线网络可分为四种网络模式：

受管模式（基础架构模式），通过接入点（默认模式）

受管网络具有一个管理元素，即接入点。在此模式（又称为基础架构模式或默认模式）下，网络中 Wi-Fi 站的所有连接都会通过接入点，接入点也充当以太网的连接点。此模式使用了多种身份验证机制（WPA 等），以确保只有经过授权的工作站才能连接。这也是能耗最低的主模式。

对等模式（对等网络）

对等网络没有接入点。各站之间直接通讯，因此对等网络通常比受管网络速度更慢。但是，在对等网络中，传送范围和参与工作站的数目都受到很大限制。它们也不支持 WPA 身份验证。此外，并非所有卡都能可靠地支持对等模式。

主模式

在主模式下，Wi-Fi 卡将用作接入点（如果您的卡支持此模式）。有关 Wi-Fi 卡的细节，请访问 http://linux-wless.passys.nl。

网状模式

无线网状网络是通过网状拓扑组织的。无线网状网络的连接分散在所有无线网状节点之间。属于此网络的每个节点都与其他节点相连以共享连接，这种连接的覆盖区域可能很大。(在 SLE12 中不受支持)。

36.1.3.3.2 身份验证

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身份验证和加密方法。

早期的 Wi-Fi 卡仅支持 WEP（有线等效加密）。但是，WEP 经证明是不安全的，因此 Wi-Fi 行业制订了一个名为 WPA 的扩展，用以弥补 WEP 的缺陷。WPA 有时与 WPA2 同义，应将其用作默认的身份验证方法。

用户通常无法选择身份验证方法。例如，当卡以受管模式工作时，身份验证由接入点设置。NetworkManager 会显示身份验证方法。

36.1.3.3.3 加密

有多种加密方法可确保所有未授权用户不能读取无线网络中交换的数据包并且不能访问网络：

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，后来密钥也可为 104 位。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 总比不加密网络要好。某些供应商实施了非标准的“动态 WEP”。它与 WEP 的工作完全相同，也具有相同弱点，不同之处在于密钥管理设备会定期更改密钥。

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。由于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 对密钥管理进行了描述。通常，它用于与 WPA-EAP 连接，但也可以与 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。

36.1.3.4 无线通讯：蓝牙

蓝牙技术是所有无线技术中应用范围最广的技术。与 IrDA 一样，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信。它还可用于连接一定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组件。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。此时就可选用 Wi-Fi 技术来穿越墙壁之类的有形障碍物进行通讯。

36.1.3.5 无线通讯：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。向收件人远距离传输文件的任务由移动网络来处理。IrDA 的另一种应用方式就是在办公室中无线传送打印任务。

36.1.4 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。

强大的身份验证

除了通过登录名和口令的标准身份验证外，还使用生物特征身份验证。SUSE Linux Enterprise Server 支持指纹身份验证。

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。中介绍了如何使用 SUSE Linux Enterprise Server《Security Guide》, 第 12 章 “Encrypting Partitions and Files” 创建加密分区。在使用 YaST 添加用户时还可以创建加密的主目录。

重要：数据安全性和暂挂到磁盘

在发生暂挂到磁盘事件期间，不会卸装加密的分区。因此，任何人只需窃取硬件然后对硬盘发出 resume 命令就可以获取这些分区上的所有数据。

网络安全

不管使用的方式如何，任何数据传输都应该是安全的。有关 Linux 和网络的常见安全性问题，请参见《Security Guide》, 第 1 章 “Security and Confidentiality”。

36.2 移动硬件

SUSE Linux Enterprise Server 支持自动检测 FireWire (IEEE 1394) 或 USB 上的移动储存设备。术语移动储存设备适用于任何种类的防火墙或 USB 硬盘、闪存盘或数码相机。这些设备在通过相应的接口与系统连接后，系统将会自动检测并配置这些设备。GNOME 的文件管理器提供了灵活的移动硬件项目处理方式。要安全卸载任何此类媒体，请使用文件管理器的卸载卷 (GNOME) 功能。有关详细信息，请参见《GNOME 用户指南》。

外部硬盘（USB 和火线）

系统正确识别外部硬盘后，其图标会显示在文件管理器中。单击该图标将显示该驱动器的内容。可以在此创建目录和文件，并执行编辑或删除操作。要重命名某个硬盘，请从右键单击上下文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。根据其将设备装入 `/media` 中的描述符将不受影响。

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。

36.3 移动设备（智能手机和平板电脑）

桌面系统或便携式计算机可通过蓝牙、Wi-Fi 或 USB 直接连接与移动设备进行通讯。根据移动设备型号和您的特定需求选择一种连接方法。通常，通过 USB 将移动设备连接到台式计算机或便携式计算机时，可将 USB 作为常规外部储存使用。通过设置蓝牙或 Wi-Fi 连接，您可以与

移动设备交互，并直接从台式计算机或便携式计算机中控制其功能。您可以使用若干个开源图形实用程序（特别是 KDE Connect (https://community.kde.org/KDEConnect) 和 GSConnect (https://extensions.gnome.org/extension/1319/gsconnect/) ）控制连接的移动设备。

37 使用 NetworkManager

NetworkManager 是用于便携式计算机和其他可移动计算机的理想解决方案。它支持网络连接的顶级加密类型和标准，包括 802.1x 保护的网络的连接。802.1X 是“基于端口的网络访问控制的本地和城域网 IEEE 标准”。使用 NetworkManager，在外出时，您就无需担心配置网络接口以及在有线或无线网络之间切换的问题。NetworkManager 可自动连接到已知无线网络或并行管理多个网络连接 – 然后将最快的连接用作默认连接。而且，您还可手动在可用网络之间切换，并使用系统盘中的小程序管理网络连接。

不只可激活一个连接，也可同时激活多个。这样您可以将便携式计算机从以太网连接拔出后仍通过无线连接保持连接状态。

37.1 NetworkManager 的用例

NetworkManager 提供了完善且直观的用户界面，可使用户轻松地切换其网络环境。但是，在以下情况下，NetworkManager 解决方案不适用：

- 您的计算机将为网络中的其他计算机（例如，DHCP 或 DNS 服务器）提供网络服务。
- 您的计算机为 Xen 服务器或您的系统是 Xen 内的虚拟系统。

37.2 启用或禁用 NetworkManager

在便携式计算机上，默认情况下 NetworkManager 处于启用状态。但是，任何时候都可以在 YaST 网络设置模块中启用或禁用它。

1. 运行 YaST，然后转到系统 › 网络设置。
2. 将打开网络设置对话框。转到全局选项选项卡。
3. 要通过 NetworkManager 配置和管理您的网络连接，请执行以下操作：
    a. 在网络设置方法字段中选择通过 NetworkManager 的用户控制方法。
    b. 单击确定并关闭 YaST。

c. 按照第 37.3 节 “配置网络连接”中所述通过 NetworkManager 配置您的网络连接。

4. 停用 NetworkManager 并使用您自己的配置控制网络

   a. 在网络设置方法字段中选择通过 wicked 进行控制。

   b. 单击确定。

   c. 通过 YaST 设置您的网卡，即通过 DHCP 或静态 IP 地址进行自动配置。
   第 16.4 节 “使用 YaST 配置网络连接”中提供了使用 YaST 进行网络配置的详细说明。

37.3 配置网络连接

在 YaST 中启用 NetworkManager 后，使用 GNOME 中提供的 NetworkManager 前端配置网络连接。它会显示所有网络连接类型对应的选项卡，例如有线、无线、移动宽带、DSL 和 VPN 连接。

要在 GNOME 中打开网络配置对话框，请通过状态菜单打开设置菜单，然后单击网络项。

注意：选项的可用性

根据您的系统设置，可能不允许您配置连接。在安全环境中，某些选项可能会被锁定或需要 `root` 权限。请咨询系统管理员以了解细节。

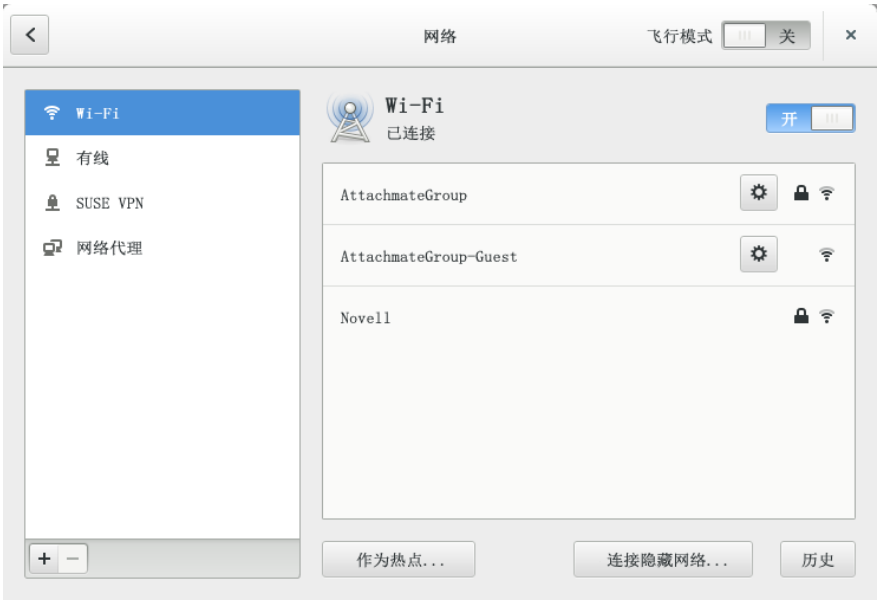


图 37.1 GNOME 网络连接对话框

过程 37.1 添加和编辑连接

1. 打开 NetworkManager 配置对话框。
2. 添加连接：
    a. 单击左下角的 + 图标。
    b. 选择首选连接类型并按照指示操作。
    c. 完成后，单击添加。
    d. 确认您的更改后，打开“状态菜单”，便会在显示的可用网络列表中看到新配置的网络连接。
3. 编辑连接：
    a. 选择要编辑的项目。
    b. 单击齿轮图标打开连接设置对话框。
    c. 插入您的更改，然后单击应用保存更改。
    d. 要让您的连接可用作系统连接，请转到身份选项卡，选中复选框可用于其他用户。有关用户和系统连接的详细信息，请参见第 37.4.1 节 “用户和系统连接”。

37.3.1 管理有线网络连接

如果您的计算机连接的是有线网络，请使用 NetworkManager 小程序管理连接。

1. 打开“状态菜单”，然后单击有线以更改连接细节或将其关闭。
2. 要更改设置，请单击有线设置，然后单击齿轮图标。
3. 要关闭所有网络连接，请激活飞行模式设置。

37.3.2 管理无线网络连接

可见的无线网络在无线网络下的 GNOME NetworkManager 小程序菜单中列出。每个网络的信号强度也会显示在菜单中。加密无线网络是用保护物图标标记的。

过程 37.2 连接到可见无线网络

1. 要连接可见无线网络，请打开“状态菜单”然后单击 Wi-Fi。
2. 单击开启将其启用。
3. 单击选择网络，选择 Wi-Fi 网络然后单击连接。
4. 如果网络已加密，一个配置对话框将会打开。其中会显示网络使用的加密类型以及用于输入登录凭证的文本框。

过程 37.3 连接到不可见无线网络

1. 要连接到未广播其服务集标识符（SSID 或 ESSID）因而无法自动检测到的网络，请打开“状态菜单”，然后单击 Wi-Fi。
2. 单击 Wi-Fi 设置打开详细设置菜单。
3. 确保您的 Wi-Fi 处于启用状态，然后单击连接到隐藏网络。
4. 在打开的对话框中的网络名称中输入 SSID 或 ESSID，并视需要设置加密参数。

显式选中的无线网络将尽可能始终保持连接。如果在此期间插入网线，则会连接任何设置为尽可能保持连接的连接，而无线连接也会保持连接状态。

37.3.3 将 Wi-Fi/蓝牙网卡配置为接入点

如果您的 Wi-Fi/蓝牙网卡支持接入点模式，则可以使用 NetworkManager 进行配置。

1. 打开“状态菜单”，然后单击 Wi-Fi。
2. 单击 Wi-Fi 设置打开详细设置菜单。
3. 单击用作热点并按照指示操作。
4. 使用随后出现的对话框中显示的身份凭证来连接远程计算机的热点。

37.3.4 NetworkManager 和 VPN

NetworkManager 支持多种虚拟专用网 (VPN) 技术。对于每种技术，SUSE Linux Enterprise Server 都随附了提供 NetworkManager 常规支持的基础包。此外，还需要为您的小程序安装特定于桌面的包。

OpenVPN

要使用此 VPN 技术，请安装:

- `NetworkManager-openvpn`
- `NetworkManager-openvpn-gnome`

vpnc (Cisco AnyConnect)

要使用此 VPN 技术，请安装:

- `NetworkManager-vpnc`
- `NetworkManager-vpnc-gnome`

PPTP（点对点隧道协议）

要使用此 VPN 技术，请安装：

- `NetworkManager-pptp`
- `NetworkManager-pptp-gnome`

以下过程介绍如何使用 NetworkManager 将您的计算机设置为 OpenVPN 客户端。设置其他类型 VPN 的过程与此类似。

开始之前，请确保包 `NetworkManager-openvpn-gnome` 已安装，并且所有依赖项均已解析。

过程 37.4 使用 NETWORKMANAGER 设置 OPENVPN

1. 依次单击位于面板右端的状态图标和扳手和螺丝刀图标，打开应用程序设置。在所有设置窗口中，选择网络。
2. 单击 + 图标。
3. 依次选择 VPN 和 OpenVPN。
4. 选择身份验证类型。根据 OpenVPN 服务器的设置，选择证书 (TLS) 或口令和证书 (TLS)。
5. 将所需的值插入到相应文本框中。对于我们的示例配置，值如下所示：

网关	VPN 服务器的远程端点
用户名	用户（仅当选择了口令和证书 (TLS) 时才可用）
口令	用户口令（仅当选择了口令和证书 (TLS) 时才可用）
用户证书	`/etc/openvpn/client1.crt`
CA 证书	`/etc/openvpn/ca.crt`
私用密钥	`/etc/openvpn/client1.key`

6. 单击添加完成配置。
7. 要启用连接，请在设置应用程序的网络面板中，单击切换按钮。也可以单击位于面板右端的状态图标，然后依次单击您的 VPN 名称和连接。

37.4 NetworkManager 和安全性

NetworkManager 区分两种类型的无线连接，即可信和不可信。可信连接是您过去明确选择的任何网络。所有其他连接均为不可信连接。可信连接用接入点的名称和 MAC 地址识别。使用 MAC 地址可以确保带有可信连接名称的不同接入点不可使用。

NetworkManager 会定期扫描是否存在可用的无线网络。如果找到多个可信网络，则自动选择最近使用的可信网络。如果所有网络均不可信，NetworkManager 将等待您做出选择。

如果加密设置改变，但名称和 MAC 地址不变，则 NetworkManager 将尝试连接，但首先会要求您确认新的加密设置并提供所有更新（如新密钥）。

如果您从使用无线连接切换到脱机模式，则 NetworkManager 会将 SSID 或 ESSID 设为空白。这可以确保断开网卡连接。

37.4.1 用户和系统连接

NetworkManager 可识别两种类型的连接：*用户*连接和*系统*连接。用户连接是第一个用户登录时对 NetworkManager 可用的连接。会要求该用户提供所有必需的身份凭证，当该用户注销时，连接会断开并从 NetworkManager 中去除。定义为系统连接的连接可以由所有用户共享，并在启动 NetworkManager 之后、任何用户登录之前即可使用。如果是系统连接，必须在创建连接时提供所有身份凭证。此类系统连接可用于自动连接到要求授权的网络。有关如何用 NetworkManager 配置用户连接或系统连接的信息，请参见第 37.3 节 “配置网络连接”。

37.4.2 储存密码和身份凭证

如果不想每次连接到加密网络时都要再次输入身份凭证，则可以使用 GNOME 密钥环管理器将身份凭证加密存储在磁盘上，并用主密码保护。

NetworkManager 还可从证书储存检索用于安全连接（例如，加密的有线、无线或 VPN 连接）的证书。有关更多信息，请参见《Security Guide》, 第 13 章 “Certificate Store”。

37.5 常见问题 (FAQ)

下面是关于使用 NetworkManager 配置特殊网络选项的一些常见问题。

37.5.1.如何将连接绑定到特定设备？

默认情况下，NetworkManager 中的连接是特定于设备类型的：它们适用于同一类型的所有物理设备。如果每个连接类型有多台物理设备可用（例如您的计算机装有两块以太网卡），您可以将一个连接绑定到特定设备。

要在 GNOME 中执行此操作，请先查找设备的 MAC 地址（使用小程序中提供的连接信息，或者使用 `nm-tool` 或 `wicked show all` 等命令行工具的输出）。然后启动配置网络连接的对话框，选择您要修改的连接。在有线或无线选项卡上，输入设备的 MAC 地址，并确认更改。

37.5.2. 如果检测到同一 ESSID 有多个接入点，如何指定特定接入点？

当有不同无线波段 (a/b/g/n) 的多个接入点时，默认情况下会自动选择信号最强的接入点。要覆盖此值，配置无线连接时请使用 BSSID 字段。

基本服务集标识 (BSSID) 可唯一标识每个基本服务集。在基础结构基本服务集中，BSSID 是无线接入点的 MAC 地址。在独立（特别）基本服务集中，BSSID 是本地管理的 MAC 地址（从 46 位数字随机生成）。

如第 37.3 节 “配置网络连接” 中所述启动配置网络连接的对话框。选择要修改的无线连接，然后单击编辑。在无线选项卡上，输入 BSSID。

37.5.3. 如何将网络连接与其他计算机共享？

主设备（连接到因特网的设备）不需要任何特殊配置。但是，需要如下配置连接到本地集线器或计算机的设备：

1. 如第 37.3 节 “配置网络连接” 中所述启动配置网络连接的对话框。选择要修改的连接，然后单击编辑。切换到 IPv4 设置选项卡，并激活方法下拉框中的共享给其他计算机。这将启用 IP 通讯转发并运行该设备上的 DHCP 服务器。确认在 NetworkManager 中所做更改。

2. 由于 DCHP 服务器使用端口 67，请确保该端口没有被防火墙阻止：在共享连接的计算机上，启动 YaST 并选择安全和用户 › 防火墙。切换到允许的服务类别。如果 DHCP 服务器尚未显示为允许的服务，请从待允许的服务中选择 DHCP 服务器，然后单击添加。确认在 YaST 中所做更改。

37.5.4. 如何对自动（DHCP、PPP、VPN）地址提供静态 DNS 信息？

如果 DHCP 服务器提供无效的 DNS 信息（和/或路由），则可以覆盖它。如第 37.3 节 “配置网络连接” 中所述启动配置网络连接的对话框。选择要修改的连接，然后单击编辑。切换到 IPv4 设置选项卡，并激活方法下拉框中的仅自动 (DHCP) 地址。在 DNS 服务器和搜索域字段中输入 DNS 信息。要忽略自动取得的路由，请单击路由，然后激活相应的复选框。确认更改。

37.5.5.如何在用户登录之前将 NetworkManager 连接到受口令保护的网络？

定义可以用于此类用途的系统连接。有关更多信息，请参考第 37.4.1 节 “用户和系统连接”。

37.6 查错

可能出现连接问题。与 NetworkManager 相关的一些常见问题包括小程序不启动或缺少 VPN 选项。解决方法和预防这些问题的方法随使用的工具而定。

NetworkManager 桌面小程序没有启动

如果设置了 NetworkManager 控制的网络，小程序会自动启动。如果小程序未启动，请按照第 37.2 节 “启用或禁用 NetworkManager”中所述检查是否在 YaST 中启用了 NetworkManager。然后，请确保 NetworkManager-gnome 包也已安装。

如果桌面小程序已经安装，但出于某种原因没有运行，则手动启动它。如果该桌面小程序已安装但由于某些原因没有运行，请用命令 `nm-applet` 手动启动它。

NetworkManager 小程序不包括 VPN 选项

对 NetworkManager 的支持、小程序以及适用于 NetworkManager 的 VPN 已在不同的包中分发。如果 NetworkManager 小程序不包括 VPN 选项，请检查带有 NetworkManager 支持的 VPN 技术的包是否已经安装。有关详细信息，请参见第 37.3.4 节 “NetworkManager 和 VPN”。

没有可用的网络连接

如果您已正确配置网络连接并且网络连接的所有其他组件（路由器等等）也已启动并在正常运行，则重启动计算机上的网络接口有时可能有帮助。要实现此目的，请以 `root` 身份登录到命令行，然后运行 `systemctl restart wicked`。

37.7 更多信息

可在以下网站和目录中找到有关 NetworkManager 的更多信息：

NetworkManager 项目页

http://projects.gnome.org/NetworkManager/

包文档

还可以在以下目录中找到有关 NetworkManager 和 GNOME 小程序的最新信息：

- `/usr/share/doc/packages/NetworkManager/`，
- `/usr/share/doc/packages/NetworkManager-gnome/`。

38 电源管理

System z IBM Z 上不提供本章所述的功能和硬件，因此本章内容与这些平台不相关。

电源管理对于便携式计算机特别重要，但对于其他系统也是有用的。ACPI（高级配置和电源接口）在所有通用计算机（便携式计算机、台式机和服务器）上都可用。电源管理技术需要合适的硬件和 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。还可以通过控制 CPU 频率调节以达到省电或降低噪音的目的。

38.1 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。ACPI 中的主要功能和它们的用法为：

待机
: 不支持。

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启动应用程序就可以在数秒内将工作恢复到原来的进度。此功能对应于 ACPI 状态 S3。

休眠（暂挂到磁盘）
: 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。至少要有一个像 RAM 一样大的交换分区才能写入所有活动的数据。从该状态重激活大约需要 30 至 90 秒的时间。将恢复到暂停之前的状态。某些制造商提供这种方式的有用的混合变体（例如 IBM Thinkpad 中的 RediSafe）。对应的 ACPI 状态是 S4。在 Linux 中，由独立于 ACPI 的内核例程执行暂挂到磁盘。

注意：通过 mkswap 进行格式化时更改了交换分区的 UUID

如果可能，请不要使用 `mkswap` 重新设置现有交换分区的格式。使用 `mkswap` 重新设置格式将会更改交换分区的 UUID 值。请通过 YaST 重新设置格式（将更新 `/etc/fstab`），或者手动调整 `/etc/fstab`。

电池监视

ACPI 检查电池充电状态并提供相关信息。另外，当达到临界电量状态时，它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。

处理器速度控制

在 CPU 方面，有三种方法可以节省电能：频率和电压调节（也称为 PowerNow! 或 Speedstep）、限制和使处理器进入休眠 (C-state)。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。

38.2 高级配置和电源接口 (ACPI)

ACPI 旨在支持操作系统设置和控制各个硬件组件。ACPI 取代即插即用电源管理 (PnP) 和高级电源管理 (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停用部件等任务。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。journald 中报告了 ACPI 能够检测并装载的表。有关查看日记日志消息的更多信息，请参见第 15 章 “journalctl：查询 systemd 日记”。有关对 ACPI 问题进行故障诊断的详细信息，请参见第 38.2.2 节 “故障诊断”。

38.2.1 控制 CPU 性能

CPU 可以采用三种节能方法：

- 频率和电压调节
- 限制时钟频率 (T-state)
- 使处理器进入休眠 (C-state)

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。

仅当处理器忙时，才需要进行频率调节和限制，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-state。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，最佳方法是由内核按需调节器控制动态频率调节。

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。

有关详细信息，请参见《System Analysis and Tuning Guide》, 第 11 章 “Power Management”。

38.2.2 故障诊断

问题有两种不同的类型。一种是内核的 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。更多情况下，问题是由 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。如果计算机未引导，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=ht
: 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。

警告：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。

有时，计算机会对通过 USB 或 FireWire 挂接的硬件感到困惑。如果一台计算机拒绝引导，请拔下所有不需要的硬件，然后再次重试。

引导后，使用命令 `dmesg -T | grep -2i acpi` 来监视系统的引导消息（或所有消息，因为问题也可能是 ACPI 以外的因素所导致）。如果在分析 ACPI 表时出错，则最重要的表 DSDT（区分系统描述表）可替换为改进的版本。在这种情况下，将忽略 BIOS 中有问题的 DSDT。第 38.4 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 调试消息。如果编译和安装的是带有 ACPI 调试功能的内核，则会显示详细信息。

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。

38.2.2.1 更多信息

- http://tldp.org/HOWTO/ACPI-HOWTO/（详细的 ACPI HOWTO 文档，包含 DSDT 增补程序）
- http://www.acpi.info（高级配置和电源接口规范）
- http://acpi.sourceforge.net/dsdt/index.php（Bruno Ducrot 开发的 DSDT 增补程序）

38.3 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，您无需手动关闭硬盘，因为硬盘会在不运行时自动进入经济的运行方式。但是，如果要最大程度地节能，请使用 `hdparm` 命令测试以下某些方法。

它可用于修改各种磁盘设置。选项 `-y` 将硬盘立即切换到待机方式。 `-Y` 会让硬盘进入休眠状态。 `hdparm -S X` 会让硬盘闲置一段时间后减慢运行速度。使用以下值替换 `X`： `0` 表示禁用此机制，会使硬盘持续运行。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 `pdflush` 守护程序监视。当数据达到一定的有效期限制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 pdflush 设置为较短的时间间隔可以获得最好的数据完整性。它会每 5 秒钟检查一次缓冲区并将数据写入硬盘。以下变量很有用：

`/proc/sys/vm/dirty_writeback_centisecs`
: 包含截至 pdflush 线程唤醒的延迟（以百分之一秒为单位）。

`/proc/sys/vm/dirty_expire_centisecs`
: 定义最晚在什么时间范围之后应写出未写入页。默认值是 `3000`，表示 30 秒。

`/proc/sys/vm/dirty_background_ratio`
: pdflush 开始写入未写入页之前未写入页的最大百分比。默认值是 `5` %。

`/proc/sys/vm/dirty_ratio`
: 当未写入页超出总内存的此百分比后，将强制进程在其时间范围内写入未写入缓冲区，而不是继续写入。

警告：对数据完整性的损害

更改为 `pdflush` 守护程序设置将损害数据完整性。

除了这些进程之外，`Btrfs`、`Ext3`、`Ext4` 等日记文件系统会独立于 `pdflush` 写入它们的元数据，这也会妨碍硬盘降速。

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。

38.4 查错

所有错误消息和警报都记录在可以使用 `journalctl` 命令查询的系统日记中（有关更多信息，请参见第 15 章 “journalctl：查询 systemd 日记”）。以下几个部分介绍最常见的问题。

38.4.1 CPU 频率不工作

请参见内核源以确认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。如果安装了 `kernel-source` 包，则在 `/usr/src/linux/Documentation/cpu-freq/*` 中可找到此信息。

38.5 更多信息

- http://en.opensuse.org/SDB:Suspend_to_RAM — 如何使“暂挂到 RAM”正常工作
- http://old-en.opensuse.org/Pm-utils — 如何修改常规暂挂框架

VI 查错

39 帮助和文档 564

40 收集用于支持的系统信息 569

41 常见问题及其解决方案 594

39 帮助和文档

SUSE® Linux Enterprise Server 提供了各种信息和文档来源，其中绝大部分已经集成在您的已安装系统中。

`/usr/share/doc` 中的文档

这一传统帮助目录包含各种文档文件以及系统的发行说明。它还包含子目录 `packages` 中的已安装包的信息。有关详细信息可以在第 39.1 节 “文档目录” 中找到。

外壳命令的手册页和信息页

使用外壳时，您不需要了解命令选项。外壳往往是通过手册页和信息页来提供集成帮助的。有关详细信息，请参见第 39.2 节 “手册页”和第 39.3 节 “信息页”。

桌面帮助中心

GNOME 桌面的帮助中心（帮助）按可搜索的形式提供了对系统上最重要的文档资源的集中访问途径。这些资源包括已安装应用程序、手册页、信息页以及随产品提供的 SUSE 手册的联机帮助。

某些应用程序的独立帮助包

当用 YaST 安装新软件时，系统通常会自动安装软件文档，并且这些文档会出现在您桌面的帮助中心中。但是，某些应用程序（如 GIMP）可能具有不同的联机帮助包，可与 YaST 分开安装，并且不集成到帮助中心。

39.1 文档目录

在您安装的 Linux 系统上查找文档的传统目录是 `/usr/share/doc`。目录通常包含有关您的系统上已安装的包和发行说明、手册等等的信息。

注意：内容取决于所安装的包

在 Linux 系统中，许多手册和其他种类的文档都可以像软件一样以包的形式获取。`/usr/share/docs` 中的信息量和信息内容还取决于安装的（文档）包。如果您找不到这里提到的子目录，请检查相应的包是否已安装到您的系统上，并根据需要使用 YaST 添加它们。

39.1.1 SUSE 手册

我们的手册提供了不同语言的 HTML 和 PDF 版本。在 `manual` 子目录下有您的产品适用的大多数 HTML 版 SUSE 手册。有关对您的产品可用的所有文档的概述，请参见这些手册的前言。

如果安装了多种语言，`/usr/share/doc/manual` 可能包含这些手册的不同语言版本。两个桌面的帮助中心内也提供 HTML 版本的 SUSE 手册。有关在安装媒体的何处能找到这些书的 PDF 和 HTML 版本的信息，请参见 SUSE Linux Enterprise Server 发行说明。它们位于所安装系统的 `/usr/share/doc/release-notes/` 下，或者也可以联机访问您的产品专属网站：http://www.suse.com/releasenotes/。

39.1.2 包文档

在 `packages` 下可找到系统上安装的软件包中包含的文档。对每个包，都会创建子目录 `/usr/share/doc/packages/PACKAGENAME`。它经常包含该包的自述文件，有时还包含示例、配置文件或其他脚本。下表列出了 `/usr/share/doc/packages` 下常见的文件。以下每项不一定都存在，许多包中可能只包含其中的一部分。

`AUTHORS`
: 主要开发者列表。

`BUGS`
: 已知 bug 或故障。还可能包含到 Bugzilla 网页的链接，您可以在该页面上搜索所有 bug。

`CHANGES`，
`ChangeLog`
: 每个版本的更改摘要。通常开发人员会对此感兴趣，因为它非常详细。

`COPYING`，
`LICENSE`
: 许可信息。

`FAQ`
: 从邮件列表或新闻组收集的问题和回答。

`INSTALL`
: 如何在系统上安装此包。因为您读到该文件前该包已安装，您可以放心地忽略该文件的内容。

`README`、`README.*`

软件的常规信息。例如用途和用法。

`TODO`

尚未实施但是将来可能要实施的操作。

`MANIFEST`

带有简述的文件列表。

`NEWS`

描述此版本中的新增内容。

39.2 手册页

手册页是任何 Linux 系统的基本组成部分。它们介绍命令的用法以及所有可用的选项和参数。许多页面都可以用 `man` 后面跟命令名来访问，例如 `man ls`。

手册页直接显示在外壳中。可以用 Page ↑ 和 Page ↓ 上下移动来浏览它们。用 Home 和 End 可以切换显示文档的开头和结尾。按 Q 可以结束这种查看模式。使用 `man man` 可以了解有关 `man` 命令本身的更多信息。手册页如表 39.1 “手册页—类别和说明” 所示按类别进行排序（取自 man 命令本身的手册页）。

表 39.1 手册页—类别和说明

号码	描述
1	可执行程序或外壳命令
2	系统调用（内核提供的函数）
3	库调用（程序库内的函数）
4	特殊文件（通常位于 `/dev`）
5	文件格式和约定（`/etc/fstab`）
6	游戏

号码	描述
7	其他（包括宏包和约定），如 man(7)、groff(7)
8	系统管理命令（通常仅供 `root` 使用）
9	内核例程（非标准）

每个手册页包括标为 NAME、SYNOPSIS、DESCRIPTION、SEE ALSO、LICENSING 和 AUTHOR 的几个部分。根据具体的命令类型，可能还有其他部分。

39.3 信息页

信息页是系统上另一个重要的信息来源。它们通常比手册页更为详细。它们包含多个命令行选项，有时还包含完整的教学课程或参考文档。要查看特定命令的信息页，请输入 `info` 后面跟命令名，例如 `info ls`。您可以在外壳中直接用查看器浏览信息页，并显示名为“节点”的各个部分。使用 Space 可前移，使用 <– 可后移。在节点内，您也可以用 Page ↑ 和 Page ↓ 浏览，但只有 Space 和 <– 仍会带您到上个或下个节点。按 Q 结束查看模式。并非每个命令都有对应的信息页，反之亦然。

39.4 联机资源

除了安装在 `/usr/share/doc` 下的联机版 SUSE 手册之外，您还可以访问网上的产品特定手册和文档。有关 SUSE Linux Enterprise Server 所有适用文档的概述，请查看产品特定的文档网页，网址为：http://www.suse.com/doc/。

如果要搜索更多产品相关信息，您也可以参见以下网站：

SUSE 技术支持

如果您有技术方面的疑问或需要相应的解决方案，可在 http://www.suse.com/support/ 上找到 SUSE 技术支持。

SUSE 论坛

SUSE 有多个论坛，您可以进入其中讨论有关该产品的话题。如需获取一份列表，请参见 http://forums.suse.com/。

SUSE 交流园地

联机社区提供文章、提示、问答和可供免费下载的工具：http://www.suse.com/communities/conversations/

GNOME 文档

适用于 GNOME 用户、管理员和开发人员的文档可在 http://library.gnome.org/ 上找到。

Linux 文档计划

Linux 文档计划（Linux Documentation Project，TLDP）由编写 Linux 相关文档的志愿者团队负责管理（请参见 http://www.tldp.org）。它可能是 Linux 相关的最全面的文档资源。这套文档包括初学者教程，但主要侧重于有经验的用户和职业系统管理员。TLDP 发布了免费的操作指南、常见问题和其他指南（手册）。SUSE Linux Enterprise Server 上也有来自 TLDP 的部分文档。

您还可以尝试通用搜索引擎。例如，如果在刻录 CD 或转换 LibreOffice 文件时遇到问题，可以使用搜索词“`Linux CD-RW 帮助`”或“`OpenOffice 文件转换问题`”。

40 收集用于支持的系统信息

为了让用户快速概览计算机的所有系统相关信息，SUSE Linux Enterprise Server 提供了 `hostinfo` 包。该包还可以帮助系统管理员检查污染的（不支持的）内核，或者计算机上安装的任何第三方包。

出现问题时，可以使用 `supportconfig` 命令行工具或 YaST 支持模块创建详细的系统报告。这两种方法都会收集系统的相关信息，包括当前内核版本、硬件、已安装包、分区设置及其他信息。最后会生成一个包含多个文件的 TAR 存档。在建立服务请求 (SR) 后，您可以将该 TAR 存档上载至全球技术支持。该存档有助于跟踪您所报告的问题，并可以帮助您解决问题。

此外，您可以分析 `supportconfig` 输出来发现已知问题，以帮助快速解决问题。为此，SUSE Linux Enterprise Server 提供了一个设备和一个命令行工具用于进行 `Supportconfig 分析` (SCA)。

40.1 显示当前系统信息

在登录到服务器时，要想快速方便地概览所有相关系统信息，请使用包 `hostinfo`。在计算机上安装该包后，控制台将会向登录到该计算机的任何 `root` 用户显示以下信息：

例 40.1 以 root 身份登录时的 hostinfo 输出

```
Hostname:                earth
Current As Of:           Fri 28 Sep 2018 03:18:57 PM CEST
Distribution:            SUSE Linux Enterprise Server 12
 -Service Pack:          4
Architecture:            x86_64
Kernel Version:          4.12.14-94.37-default
 -Installed:             Mon 24 Sep 2018 10:43:46 AM CEST
 -Status:                Not Tainted
Last Updated Package:    Fri 28 Sep 2018 03:18:55 PM CEST
 -Patches Needed:        0
 -Security:              0
```

```
 -3rd Party Packages:     0
IPv4 Address:             eth0 192.168.1.1
Total/Free/+Cache Memory: 1812/360/1275 MB (70% Free)
Hard Disk:                /dev/sda 32 GB
```

如果输出显示 `tainted` 内核状态，请参见第 40.6 节 “内核模块支持”以了解更多细节。

40.2 使用 Supportconfig 收集系统信息

要创建包含详细系统信息的 TAR 存档以提交给全球技术支持，请直接使用 `supportconfig` 命令行工具，或者使用 YaST 支持模块。该命令行工具由默认安装的包 `supportutils` 提供。YaST 支持模块也基于该命令行工具。

40.2.1 创建服务请求编号

系统随时都可以生成 Supportconfig 存档。但是，要将 supportconfig 数据提交给全球技术支持，首先需要生成一个服务请求编号。上载存档以获取支持时，您需要使用此编号。

要创建服务请求，请访问 https://scc.suse.com/support/requests 并遵照屏幕上的说明操作。写下您的 12 位服务请求编号。

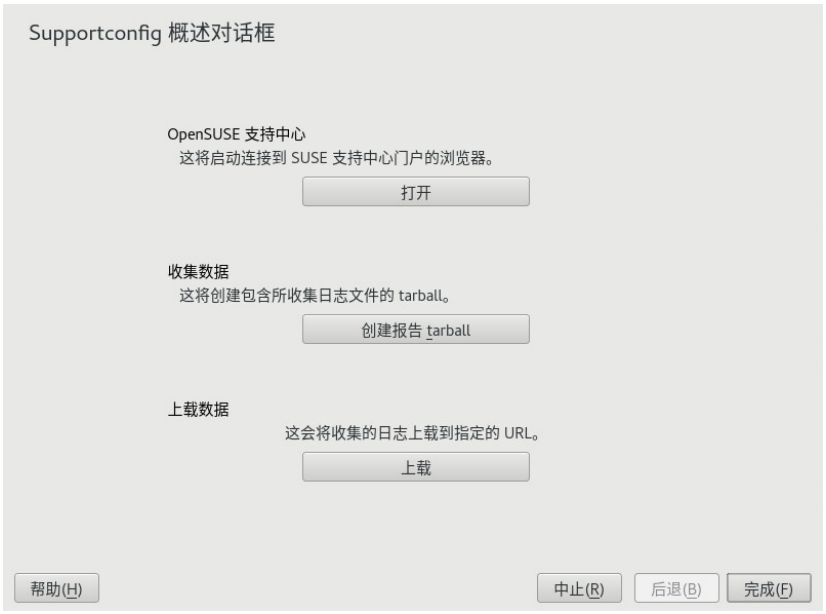
注意：隐私声明

SUSE 和 Micro Focus 将系统报告视为机密数据。有关我们在隐私方面所做承诺的详细信息，请参见 https://www.suse.com/company/policies/privacy/。

40.2.2 使用 YaST 创建 Supportconfig 存档

要使用 YaST 收集系统信息，请按如下所述操作：

1. 启动 YaST 并打开支持模块。



2. 单击创建报告 tarball。

3. 在随后出现的窗口中，从单选按钮列表中选择一个 supportconfig 选项。系统默认会预先选择使用自定义（专家）设置。如果要先测试报告功能，请使用仅收集最少量的信息。有关其他选项的某些背景信息，请参考 `supportconfig` 手册页。
   单击下一步继续。

4. 输入您的联系信息。该信息将写入名为 `basic-environment.txt` 的文件，并包含在要创建的存档中。

5. 如果要在结束信息收集过程时将存档提交到全球技术支持，则需要指定上载信息。YaST 会自动推荐一个上载服务器。
   如果希望稍后提交存档，则可以暂时将上载信息留空。

6. 单击下一步继续。

7. 开始收集信息。

该过程完成后，单击下一步继续。

8. 检查数据收集：选择日志文件的文件名可以在 YaST 中查看其内容。在将 TAR 存档提交到支持部门之前，如果要去除您不希望包含在该存档中的文件，请使用从数据中去除。按下一步继续。

9. 保存该 TAR 存档。如果您以 `root` 用户身份启动了 YaST 模块，则默认情况下，YaST 会建议将该存档保存到 `/var/log`（否则将保存到您的主目录）。文件名格式为 `nts_主机_日期_时间.tbz`。

10. 如果要直接将该存档上载到支持部门，请确保将日志文件 tarball 上载到 URL 已激活。此处显示的上载目标是步骤 5 中 YaST 建议的上载目标。

11. 如果要跳过上载步骤，请停用将日志文件 tarball 上载到 URL。

12. 确认更改以关闭 YaST 模块。

40.2.3 从命令行创建 Supportconfig 存档

以下过程显示了如何创建 supportconfig 存档但不将它直接提交到支持部门。要上载该存档，需要根据过程 40.2 “从命令行向支持部门提交信息”中所述，结合某些选项运行命令。

1. 打开外壳并转换为 `root` 用户。
2. 运行 `supportconfig`，不使用任何选项。这会收集默认的系统信息。
3. 等待工具完成操作。
4. 默认的存档位置为 `/var/log`，文件名格式为 `nts_主机_日期_时间.tbz`

40.2.4 常用的 supportconfig 选项

`supportconfig` 实用程序在调用时通常不带任何选项。使用 `supportconfig -h` 显示所有选项的列表或参见手册页。下面的列表提供了某些常见用例的简述：

减少所收集信息的大小

使用最少量选项 (`-m`)：

```
supportconfig -m
```

将信息限制为特定的主题

如果已使用默认的 `supportconfig` 输出找到问题所在，并发现该问题只与特定的区域或功能集相关，则您在下一次运行 `supportconfig` 时，应将收集的信息限制为特定的区域。例如，如果您检测到 LVM 出现问题并想要测试最近对 LVM 配置所做的更改，则合适的做法是只收集有关 LVM 的最少量 supportconfig 信息：

```
supportconfig -i LVM
```

要查看可用于将所收集信息限制为特定区域的功能关键字的完整列表，请运行

```
supportconfig -F
```

在输出中包含其他联系信息：

```
supportconfig -E tux@example.org -N "Tux Penguin" -O "Penguin Inc." ...
```

（在一行中输入所有命令）

收集已经过轮换的日志文件

```
supportconfig -l
```

这对日志记录量较大的环境，或重引导后 syslog 轮换日志文件时发生内核崩溃的情况尤为有用。

40.3 将信息提交到全球技术支持

可以使用 YaST 支持模块或 `supportconfig` 命令行实用程序向全球技术支持提交系统信息。如果您遇到服务器问题并想要获得支持人员的帮助，则首先需要建立一个服务请求。有关细节，请参见第 40.2.1 节 “创建服务请求编号”。

以下示例使用 12345678901 作为服务请求编号的占位符。请将 12345678901 替换为您在第 40.2.1 节 “创建服务请求编号”中创建的服务请求编号。

过程 40.1 使用 YAST 向支持部门提交信息

以下过程假设您已创建但尚未上载某个 supportconfig 存档。请确保已按第 40.2.2 节 “使用 YaST 创建 Supportconfig 存档”中的步骤 4 所述在存档中包含了您的联系信息。有关如何通过一个步骤生成并提交 supportconfig 存档的说明，请参见第 40.2.2 节 “使用 YaST 创建 Supportconfig 存档”。

1. 启动 YaST 并打开支持模块。
2. 单击上载。
3. 在有日志文件的包中，指定现有 supportconfig 存档的路径，或者单击浏览找到该存档。
4. YaST 会自动推荐一个上载服务器。

单击下一步继续。

5. 单击完成。

过程 40.2 从命令行向支持部门提交信息

以下过程假设您已创建但尚未上载某个 supportconfig 存档。有关如何通过一个步骤生成并提交 supportconfig 存档的说明，请参见第 40.2.2 节 “使用 YaST 创建 Supportconfig 存档”。

1. 具有因特网连接的服务器：

   a. 要使用默认上载目标，请运行：

   ```
   supportconfig -ur 12345678901
   ```

   b. 对于安全上载目标，请使用以下命令：

   ```
   supportconfig -ar 12345678901
   ```

2. 不具有因特网连接的服务器：

   a. 运行以下命令：

   ```
   supportconfig -r 12345678901
   ```

b. 将 `/var/log/nts_SR12345678901*tbz` 存档手动上载到我们的 FTP 服务器之一。有关详细信息，请参见 `supportconfig` 的手册页。

3. 将 TAR 存档上载到我们 FTP 服务器的接收目录后，它会自动附加到您的服务请求中。

40.4 分析系统信息

您可以分析使用 `supportconfig` 创建的系统报告来发现已知问题，以帮助快速解决问题。为此，SUSE Linux Enterprise Server 提供了一个设备和一个命令行工具用于进行 `Supportconfig 分析` (SCA)。SCA 设备是一个非交互式服务器端工具。SCA 工具 (`scatool`) 在客户端运行，并通过命令行执行。这两个工具都能分析来自受影响服务器的 supportconfig 存档。初始服务器分析在 SCA 设备或运行 scatool 的工作站上进行。生产服务器上不发生任何分析周期。

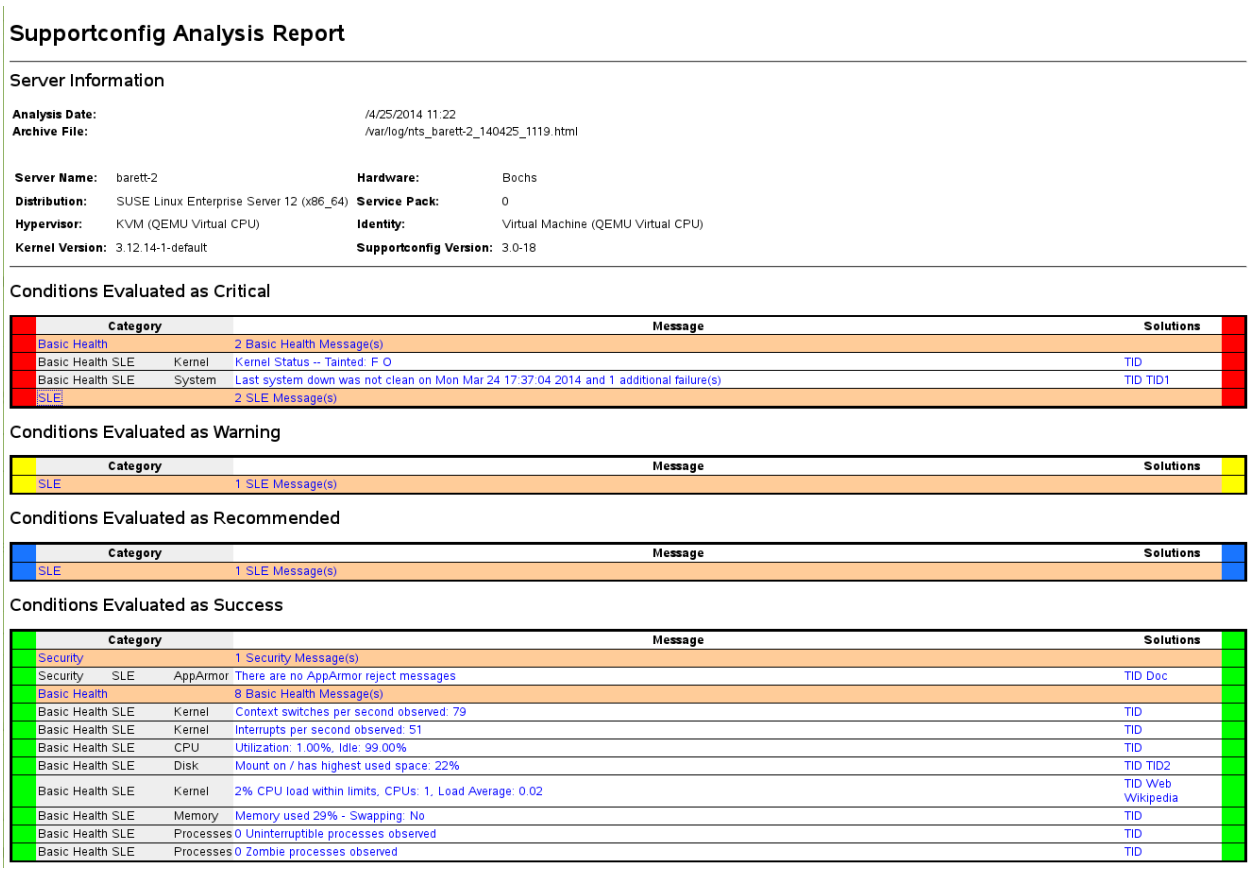
此外，该设备和命令行工具还需要产品特定的模式，这样它们才能分析关联产品的 supportconfig 输出。每个模式是一个脚本，用于针对某个已知问题分析和评估 supportconfig 存档。模式以 RPM 包的形式提供。

例如，如果您想要分析 SUSE Linux Enterprise 11 计算机上生成的 supportconfig 存档，则需要将 `sca-patterns-sle11` 包随 SCA 工具一起安装（或者，在您想要用作 SCA 设备服务器的计算机上安装该包）。要分析 SUSE Linux Enterprise 10 计算机上生成的 supportconfig 存档，需要安装 `sca-patterns-sle10` 包。

您也可以根据第 40.4.3 节 “开发自定义分析模式”中的简述开发自己的模式。

40.4.1 SCA 命令行工具

SCA 命令行工具可让您既可使用 `supportconfig` 又可使用本地计算机上安装的特定产品的分析模式来分析该本地计算机。该工具将创建一份 HTML 报告来显示分析结果。有关示例，请参见图 40.1 “SCA 工具生成的 HTML 报告”。



Supportconfig Analysis Report

Server Information

Analysis Date: /4/25/2014 11:22
Archive File: /var/log/nts_barett-2_140425_1119.html

Server Name:	barett-2	**Hardware:**	Bochs
Distribution:	SUSE Linux Enterprise Server 12 (x86_64)	**Service Pack:**	0
Hypervisor:	KVM (QEMU Virtual CPU)	**Identity:**	Virtual Machine (QEMU Virtual CPU)
Kernel Version:	3.12.14-1-default	**Supportconfig Version:**	3.0-18

Conditions Evaluated as Critical

Category			Message	Solutions
Basic Health			2 Basic Health Message(s)	
Basic Health	SLE	Kernel	Kernel Status -- Tainted: F O	TID
Basic Health	SLE	System	Last system down was not clean on Mon Mar 24 17:37:04 2014 and 1 additional failure(s)	TID TID1
SLE			2 SLE Message(s)	

Conditions Evaluated as Warning

Category			Message	Solutions
SLE			1 SLE Message(s)	

Conditions Evaluated as Recommended

Category			Message	Solutions
SLE			1 SLE Message(s)	

Conditions Evaluated as Success

Category			Message	Solutions
Security			1 Security Message(s)	
Security	SLE	AppArmor	There are no AppArmor reject messages	TID Doc
Basic Health			8 Basic Health Message(s)	
Basic Health	SLE	Kernel	Context switches per second observed: 79	TID
Basic Health	SLE	Kernel	Interrupts per second observed: 51	TID
Basic Health	SLE	CPU	Utilization: 1.00%, Idle: 99.00%	TID
Basic Health	SLE	Disk	Mount on / has highest used space: 22%	TID TID2
Basic Health	SLE	Kernel	2% CPU load within limits, CPUs: 1, Load Average: 0.02	TID Web Wikipedia
Basic Health	SLE	Memory	Memory used 29% - Swapping: No	TID
Basic Health	SLE	Processes	0 Uninterruptible processes observed	TID
Basic Health	SLE	Processes	0 Zombie processes observed	TID

图 40.1 SCA 工具生成的 HTML 报告

`scatool` 命令由 `sca-server-report` 包提供。默认情况下不会安装该包。此外，您需要 `sca-patterns-base` 包，以及任一产品特定的 `sca-patterns-*` 包（与您想要在其上运行 `scatool` 命令的计算机上安装的产品匹配）。

以 `root` 用户身份或者结合 `sudo` 执行 `scatool` 命令。在调用 SCA 工具时，您可以分析现有的 `supportconfig` TAR 存档，或者让该工具通过一个步骤生成并分析新的存档。该工具还提供了一个交互式控制台（带 Tab 键补齐功能），并允许用户在外部计算机上运行 `supportconfig`，然后在本地计算机上执行后续分析。

下面提供了一些示例命令：

`sudo scatool -s`

调用 `supportconfig` 并在本地计算机上生成新的 supportconfig 存档。通过应用与所安装产品匹配的 SCA 分析模式来分析存档以发现已知问题。显示基于分析结果生成的 HTML 报告的路径。通常，该报告将写入到 supportconfig 存档所在的同一个目录。

`sudo scatool -s -o /opt/sca/reports/`

与 `sudo scatool -s` 类似，唯一的差别在于，HTML 报告将写入到 `-o` 指定的路径。

`sudo scatool -a PATH_TO_TARBALL_OR_DIR`

分析指定的 supportconfig 存档文件（或者 supportconfig 存档解压缩到的指定目录）。生成的 HTML 报告保存在 supportconfig 存档或目录所在的位置。

`sudo scatool -a SLES_SERVER.COMPANY.COM`

与外部服务器 `SLES_SERVER.COMPANY.COM` 建立 SSH 连接，并在该服务器上运行 `supportconfig`。supportconfig 存档随后将复制回本地计算机，并在该计算机上进行分析。生成的 HTML 报告保存在默认的 `/var/log` 目录中。（`SLES_SERVER.COMPANY.COM` 上只创建 Supportconfig 存档）。

`sudo scatool -c`

启动 `scatool` 的交互式控制台。按 →| 两次可查看可用命令。

有关其他选项和信息，请运行 `sudo scatool -h` 或参见 `scatool` 手册页。

40.4.2 SCA 设备

如果您决定使用 SCA 设备来分析 supportconfig 存档，则需要配置一台服务器（或虚拟机）作为专用的 SCA 设备服务器。然后，可以使用 SCA 设备服务器，在企业中运行 SUSE Linux Enterprise Server 或 SUSE Linux Enterprise Desktop 的所有计算机上分析 supportconfig 存档。您只需要将 supportconfig 存档上载到设备服务器，等待它进行分析。此过程无需任何交互。在 MariaDB 数据库中，SCA 设备将会跟踪已分析的所有 supportconfig 存档。您可以直接从设备 Web 界面阅读 SCA 报告。或者，可以让设备通过电子邮件将 HTML 报告发送给任何管理用户。有关细节，请参见第 40.4.2.5.4 节 “通过电子邮件发送 SCA 报告”。

40.4.2.1 安装快速入门

要通过命令行快速安装和设置 SCA 设备，请遵照此处的说明操作。该过程适用于专家用户，主要通过单纯的安装与设置命令来完成。有关更多信息，请参考第 40.4.2.2 节 “先决条件”到第 40.4.2.3 节 “安装和基本设置”中的详细说明。

先决条件

- Web 和 LAMP 模式
- Web 和脚本编写模块（您必须注册计算机才能选择此模块）。

注意：需要 root 特权

下面过程中的所有命令必须以 `root` 身份来运行。

过程 40.3 使用匿名 FTP 进行上载的安装

设置并运行设备后，无需任何人工交互。因此，在使用 cron 作业创建和上载 supportconfig 存档时，非常适合使用这种方法来设置设备。

1. 在安装设备的计算机上，登录到控制台并执行以下命令：

```
zypper install sca-appliance-* sca-patterns-* vsftpd
systemctl enable apache2
systemctl start apache2
systemctl enable vsftpd
systemctl start vsftpd
yast ftp-server
```

2. 在 YaST FTP 服务器中，选择身份验证 › 启用上载 › 允许匿名用户上载 › 完成 › 是，以创建 /srv/ftp/upload。
3. 执行以下命令：

```
systemctl enable mysql
systemctl start mysql
mysql_secure_installation
setup-sca -f
```

mysql_secure_installation 将创建 MariaDB `root` 口令。

过程 40.4 使用 SCP/TMP 进行上载的安装

这种设置设备的方法需要在键入 SSH 口令时进行人工交互。

1. 在安装设备的计算机上，登录到控制台。

2. 执行以下命令：

```
zypper install sca-appliance-* sca-patterns-*
systemctl enable apache2
systemctl start apache2
sudo systemctl enable mysql
systemctl start mysql
mysql_secure_installation
setup-sca
```

40.4.2.2 先决条件

要运行 SCA 设备服务器，需要满足以下先决条件：

- 安装所有 `sca-appliance-*` 包。
- 安装 `sca-patterns-base` 包。此外，需要为您想要使用设备分析的 supportconfig 存档类型安装产品特定的 `sca-patterns-*`。
- Apache
- PHP
- MariaDB
- 匿名 FTP 服务器（可选）

40.4.2.3 安装和基本设置

如第 40.4.2.2 节 “先决条件”中所列，SCA 设备与其他包存在若干依赖项。因此，在安装和设置 SCA 设备服务器之前，需要做一些准备工作：

1. 对于 Apache 和 MariaDB，请安装 `Web` 和 `LAMP` 安装模式。
2. 设置 Apache 和 MariaDB，并有选择性地设置一个匿名 FTP 服务器。有关详细信息，请参见第 32 章 “Apache HTTP 服务器”和第 33 章 “使用 YaST 设置 FTP 服务器”。
3. 将 Apache 和 MariaDB 配置为在引导时启动：

```
sudo systemctl enable apache2 mysql
```

4. 启动这两个服务：

```
sudo systemctl start apache2 mysql
```

现在，您便可以根据过程 40.5 “安装和配置 SCA 设备”中所述安装和设置 SCA 设备。

过程 40.5 安装和配置 SCA 设备

安装这些包后，可以使用 `setup-sca` 脚本来对 SCA 设备使用的 MariaDB 管理与报告数据库进行基本的配置。

使用该脚本可以配置以下选项，以便将 supportconfig 存档从您的计算机上载到 SCA 设备：

- `scp`
- 匿名 FTP 服务器

1. 安装设备和 SCA 基本模式库：

```
sudo zypper install sca-appliance-* sca-patterns-base
```

2. 此外，请为您要分析的 supportconfig 存档类型安装模式包。例如，如果您的环境中安装了 SUSE Linux Enterprise Server 11 和 SUSE Linux Enterprise Server 12 服务器，请安装 `sca-patterns-sle11` 和 `sca-patterns-sle12` 这两个包。
安装所有可用模式：

```
zypper install sca-patterns-*
```

3. 要对 SCA 设备进行基本的设置，请使用 `setup-sca` 脚本。调用该命令的方式取决于您要以哪种方式将 supportconfig 存档上载到 SCA 设备服务器：

- 如果您配置了使用 `/srv/ftp/upload` 目录的匿名 FTP 服务器，请结合 `-f` 选项执行设置脚本，并遵照屏幕上的说明操作。

```
setup-sca -f
```

注意：使用其他目录的 FTP 服务器

如果 FTP 服务器使用的目录不是 `/srv/ftp/upload`，请先调整以下配置文件，使其指向正确的目录：`/etc/sca/sdagent.conf` 和 `/etc/sca/sdbroker.conf`。

- 如果您想要通过 `scp` 将 supportconfig 文件上载到 SCA 设备服务器的 `/tmp` 目录，请不带任何参数来调用该设置脚本，然后遵照屏幕上的说明操作：

```
setup-sca
```

该设置脚本将会根据它的要求运行一些检查，并配置所需的组件。它会提示您输入两个口令：您设置的 MariaDB 的 MySQL `root` 口令，以及用于登录 SCA 设备 Web 界面的 Web 用户口令。

4. 输入现有的 MariaDB `root` 口令。SCA 设备将使用该口令连接到 MariaDB。
5. 定义 Web 用户的口令。该口令将写入 `/srv/www/htdocs/sca/web-config.php`，并设置为用户 `scdiag` 的口令。以后，您可随时更改用户名和口令，请参见第 40.4.2.5.1 节 “Web 界面的口令”。

在成功完成安装和设置后，便可以开始使用 SCA 设备，请参见第 40.4.2.4 节 “使用 SCA 设备”。但是，您应该修改某些选项，例如，更改 Web 界面的口令，更改 SCA 模式更新源，启用存档模式，或者配置电子邮件通知。有关细节，请参见第 40.4.2.5 节 “自定义 SCA 设备”。

警告：数据保护

由于 SCA 设备服务器上的报告包含已分析其 supportconfig 存档的计算机的安全相关信息，因此，必须保护好 SCA 设备服务器上的数据，以防未经授权的人员访问。

40.4.2.4 使用 SCA 设备

您可以将现有的 supportconfig 存档手动上载到 SCA 设备，也可以一步即完成创建新 supportconfig 存档以及将其上载到 SCA 设备的操作。可以通过 FTP 或 SCP 来上载存档。对于这两种上载方式，您需要知道可用来访问 SCA 设备的 URL。要通过 FTP 上载，需要为 SCA 设备配置一台 FTP 服务器，请参见过程 40.5 “安装和配置 SCA 设备”。

40.4.2.4.1 将 Supportconfig 存档上载到 SCA 设备

- 要创建 supportconfig 存档并通过（匿名）FTP 上载：

```
sudo supportconfig -U “ftp://SCA-APPLIANCE.COMPANY.COM/upload”
```

- 要创建 supportconfig 存档并通过 SCP 上载：

```
sudo supportconfig -U “scp://SCA-APPLIANCE.COMPANY.COM/tmp”
```

  系统将提示您输入运行 SCA 设备的服务器的 `root` 用户口令。

- 如果要手动上载一个或多个存档，请将现有的存档文件（通常位于 `/var/log/nts_*.tbz`）复制到 SCA 设备中。对于目标，请使用设备服务器的 `/tmp` 目录，或使用 `/srv/ftp/upload` 目录（如果为 SCA 设备服务器配置了 FTP）。

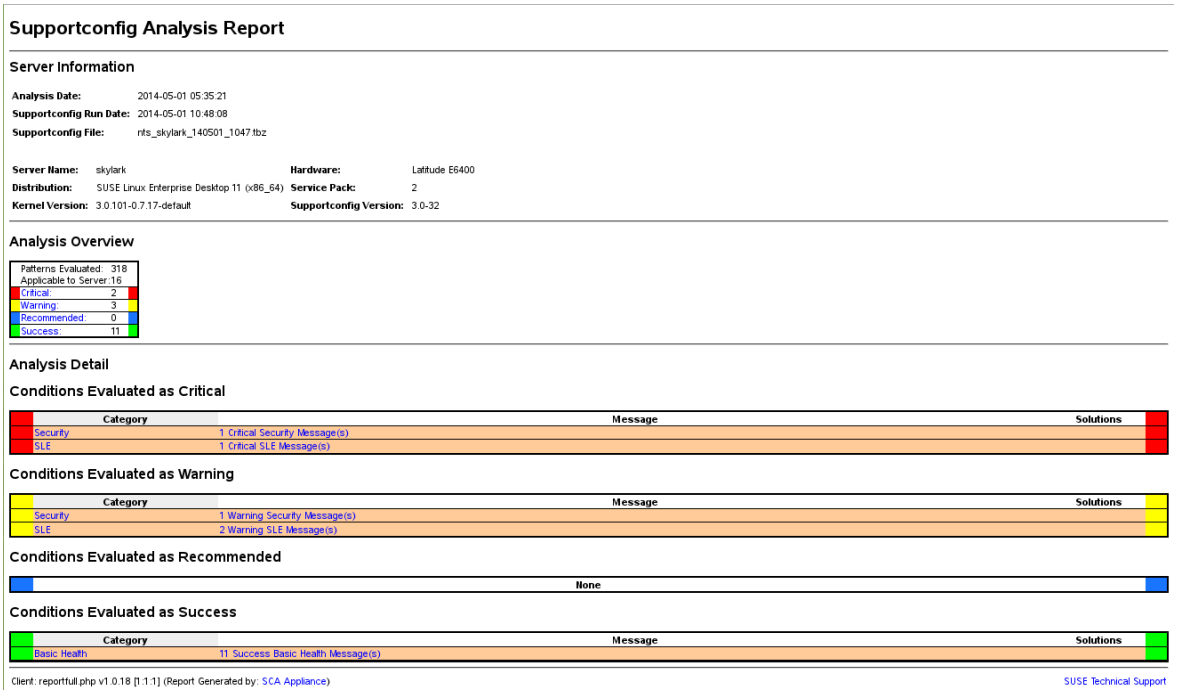
40.4.2.4.2 查看 SCA 报告

可以在装有浏览器并可以访问 SCA 设备报告索引页的任何计算机上查看 SCA 报告。

1. 启动 Web 浏览器并确保 JavaScript 和 Cookie 已启用。
2. 输入 SCA 设备的报告索引页作为 URL。

   ```
   https://sca-appliance.company.com/sca
   ```

   如果有任何疑问，请咨询系统管理员。
3. 系统将提示您输入用于登录的用户名和口令。



Supportconfig Analysis Report

Server Information

Analysis Date: 2014-05-01 05:35:21
Supportconfig Run Date: 2014-05-01 10:48:08
Supportconfig File: nts_skylark_140501_1047.tbz

Server Name: skylark
Distribution: SUSE Linux Enterprise Desktop 11 (x86_64)
Kernel Version: 3.0.101-0.7.17-default
Hardware: Latitude E6400
Service Pack: 2
Supportconfig Version: 3.0-32

Analysis Overview

Patterns Evaluated:	318
Applicable to Server:	16
Critical:	2
Warning:	3
Recommended:	0
Success:	11

Analysis Detail

Conditions Evaluated as Critical

Category	Message	Solutions
Security	1 Critical Security Message(s)	
SLE	1 Critical SLE Message(s)	

Conditions Evaluated as Warning

Category	Message	Solutions
Security	1 Warning Security Message(s)	
SLE	2 Warning SLE Message(s)	

Conditions Evaluated as Recommended

None

Conditions Evaluated as Success

Category	Message	Solutions
Basic Health	11 Success Basic Health Message(s)	

Client: reportfull.php v1.0.18 [1.1.1] (Report Generated by: SCA Appliance) SUSE Technical Support

图 40.2 SCA 设备生成的 HTML 报告

4. 登录后，单击您想要阅读的报告的日期。

5. 首先单击基本状态类别将其展开。

6. 在讯息列中单击某一个项。SUSE 知识库中的相应文章即会打开。阅读建议的解决方案，并遵循所述的说明。

7. 如果 Supportconfig 分析报告的解决方案列显示了其他项，请单击这些项。阅读建议的解决方案，并遵循所述的说明。

8. 请查看 SUSE 知识库 (http://www.suse.com/support/kb/)，以了解与 SCA 识别的问题直接相关的结果。设法解决这些问题。

9. 查看可前瞻性处理的结果，以免将来发生问题。

40.4.2.5 自定义 SCA 设备

以下几节显示了如何更改 Web 界面的口令、更改 SCA 模式更新源、启用存档模式以及配置电子邮件通知。

40.4.2.5.1 Web 界面的口令

SCA 设备 Web 界面要求使用用户名和口令登录。默认的用户名为 `scdiag`，默认的口令为 `linux`（如果未做其他的指定，请参见过程 40.5 “安装和配置 SCA 设备”）。请尽早将默认口令更改为较为安全的口令。您也可以修改用户名。

过程 40.6 更改 WEB 界面的用户名或口令

1. 在 SCA 设备服务器的系统控制台上以 `root` 用户身份登录。
2. 在编辑器中打开 `/srv/www/htdocs/sca/web-config.php`。
3. 根据需要更改 `$username` 和 `$password` 的值。
4. 保存文件并退出。

40.4.2.5.2 SCA 模式的更新

默认情况下，所有 `sca-patterns-*` 包将由一个 `root` cron 作业来定期更新，该作业将在夜间执行 `sdagent-patterns` 脚本，而该脚本又会运行 `zypper update sca-patterns-*`。定期的系统更新将会更新所有 SCA 设备包和模式包。要手动更新 SCA 设备和模式，请运行：

```
sudo zypper update sca-*
```

默认情况下，将从 SUSE Linux Enterprise 12 SP5 更新储存库安装更新。如果需要，您可以将更新源更改为某台 SMT 服务器。当 `sdagent-patterns` 运行 `zypper update sca-patterns-*` 时，将从当前配置的更新通道获取更新。如果该通道在 SMT 服务器上，将从该服务器提取包。

过程 40.7 禁用 SCA 模式的自动更新

1. 在 SCA 设备服务器的系统控制台上以 `root` 用户身份登录。
2. 在编辑器中打开 `/etc/sca/sdagent-patterns.conf`。
3. 将项

   ```
   UPDATE_FROM_PATTERN_REPO=1
   ```

更改为

```
UPDATE_FROM_PATTERN_REPO=0
```

4. 保存文件并退出。计算机无需重启动就能应用更改。

40.4.2.5.3 存档模式

系统在分析了 supportconfig 存档并将其结果储存在 MariaDB 数据库中后，会从 SCA 设备中删除所有这些存档。但是，如果要进行查错，在计算机中保留 supportconfig 存档的副本可能会有所帮助。默认情况下，存档模式处于禁用状态。

过程 40.8 在 SCA 设备中启用存档模式

1. 在 SCA 设备服务器的系统控制台上以 `root` 用户身份登录。
2. 在编辑器中打开 `/etc/sca/sdagent.conf`。
3. 将项

   ```
   ARCHIVE_MODE=0
   ```

   更改为

   ```
   ARCHIVE_MODE=1
   ```

4. 保存文件并退出。计算机无需重启动就能应用更改。

启用存档模式后，SCA 设备会将 supportconfig 文件保存到 `/var/log/archives/saved` 目录，而不是将其删除。

40.4.2.5.4 通过电子邮件发送 SCA 报告

SCA 设备可通过电子邮件发送所分析的每个 supportconfig 的 HTML 报告文件。在默认情况下禁用该功能。启用该功能后，您可以定义要将报告发送到的电子邮件地址列表，并定义会触发报告发送动作的状态讯息级别 (`STATUS_NOTIFY_LEVEL`)。

STATUS_NOTIFY_LEVEL 的可能值

$STATUS_OFF
: 停用发送 HTML 报告功能。

$STATUS_CRITICAL
: 仅发送包含“关键”状态的 SCA 报告。

$STATUS_WARNING
: 仅发送包含“警告”或“关键”状态的 SCA 报告。

$STATUS_RECOMMEND
: 仅发送包含“建议”、“警告”或“关键”状态的 SCA 报告。

$STATUS_SUCCESS
: 发送包含“成功”、“建议”、“警告”或“关键”状态的 SCA 报告。

过程 40.9 为 SCA 报告配置电子邮件通知

1. 在 SCA 设备服务器的系统控制台上以 `root` 用户身份登录。
2. 在编辑器中打开 `/etc/sca/sdagent.conf`。
3. 搜索 `STATUS_NOTIFY_LEVEL` 项。该项默认设置为 `$STATUS_OFF`（禁用电子邮件通知）。
4. 要启用电子邮件通知，请将 `$STATUS_OFF` 更改为要针对其生成电子邮件报告的状态讯息级别，例如：

   ```
   STATUS_NOTIFY_LEVEL=$STATUS_SUCCESS
   ```

   有关细节，请参见STATUS_NOTIFY_LEVEL 的可能值。
5. 定义要将报告发送到的收件人列表：
   a. 搜索 `EMAIL_REPORT='root'` 项。
   b. 将 `root` 替换为要向其发送 SCA 报告的电子邮件地址列表。各电子邮件地址必须以空格分隔。例如：

      ```
      EMAIL_REPORT='tux@my.company.com wilber@your.company.com'
      ```

6. 保存文件并退出。计算机无需重启动就能应用更改。以后生成的所有 SCA 报告都将通过电子邮件发送到指定的地址。

40.4.2.6 备份和恢复数据库

要备份和恢复储存 SCA 报告的 MariaDB 数据库，请按如下所述使用 `scadb` 命令。

过程 40.10 备份数据库

1. 在运行 SCA 设备的服务器的系统控制台上以 `root` 用户身份登录。
2. 通过执行以下命令将设备置于维护模式：

```
scadb maint
```

3. 使用以下命令启动备份进程：

```
scadb backup
```

数据将保存到 TAR 存档 `sca-backup-*sql.gz` 中。

4. 如果您正在使用模式创建数据库开发自己的模式（参见第 40.4.3 节 “开发自定义分析模式”），则还要备份以下数据：

```
sdpdb backup
```

数据将保存到 TAR 存档 `sdp-backup-*sql.gz` 中。

5. 将以下数据复制到另一台计算机或外部储存媒体：

- `sca-backup-*sql.gz`
- `sdp-backup-*sql.gz`
- `/usr/lib/sca/patterns/local`（仅当您已创建自定义模式时才需要复制该数据）

6. 使用以下命令重新激活 SCA 设备：

```
scadb reset agents
```

过程 40.11 恢复数据库

要基于备份恢复数据库，请按如下所述操作：

1. 在运行 SCA 设备的服务器的系统控制台上以 `root` 用户身份登录。
2. 将最新的 `sca-backup-*sql.gz` 和 `sdp-backup-*sql.gz` TAR 存档复制到 SCA 设备服务器。
3. 要解压缩文件，请运行：

```
gzip -d *-backup-*sql.gz
```

4. 要将数据导入数据库，请执行：

```
scadb import sca-backup-*sql
```

5. 如果您正在使用模式创建数据库创建自己的模式，则还要通过以下命令导入以下数据：

```
sdpdb import sdp-backup-*sql
```

6. 如果您正在使用自定义模式，则还要基于备份数据恢复 `/usr/lib/sca/patterns/local`。
7. 使用以下命令重新激活 SCA 设备：

```
scadb reset agents
```

8. 使用以下命令更新数据库中的模式模块：

```
sdagent-patterns -u
```

40.4.3 开发自定义分析模式

SCA 设备随附了一个完整的模式开发环境（SCA 模式数据库），可让您开发自己的自定义模式。模式可用任何编程语言编写。要使这些模式可用于 supportconfig 分析过程，需要将其保存到 `/usr/lib/sca/patterns/local` 并使其可执行。然后，SCA 设备和 SCA 工具将会针对

作为分析报告一部分的新 supportconfig 存档运行这些自定义模式。有关如何创建（和测试）自己的模式的详细说明，请参见http://www.suse.com/communities/conversations/sca-pattern-development/。

40.5 在安装过程中收集信息

在安装过程中，`supportconfig` 不可用。不过，您可以使用 `save_y2logs` 从 YaST 收集日志文件。该命令会在 `/tmp` 目录下创建一个 `.tar.xz` 存档。

如果在安装过程中很早就出现问题，您或许可以从 `linuxrc` 创建的日志文件中收集信息。`Linuxrc` 是一个在 YaST 启动之前运行的小命令。该日志文件位于 `/var/log/linuxrc.log`。

重要：安装日志文件在已安装系统中不可用

日志文件在安装过程中可用，但在已安装好的系统中却不可用。当安装程序仍在运行时，请正确保存安装日志文件。

40.6 内核模块支持

对于任何企业操作系统，一个重要的要求就是您获得的环境方面的支持级别。内核模块是硬件（“控制器”）与操作系统之间关联最密切的连接器。SUSE Linux Enterprise 中的每个内核模块都有一个 `supported` 标志，该标志可使用以下三个值：

- “yes”，相当于 `supported`
- “external”，相当于 `supported`
- “”（空白，未设置），相当于 `unsupported`

可以使用下列规则：

- 默认情况下，经过自我重新编译的内核的所有模块都会标记为 unsupported。
- SUSE 合作伙伴支持的内核模块以及使用 `SUSE SolidDriver 程序` 提供的内核模块会标记为 “external”。

- 如果未设置 `supported` 标志，装载此模块便会污染该内核。系统不支持污染的内核。不支持的内核模块包含在一个附加的 RPM 包 (`kernel-FLAVOR-extra`) 中，该包只适用于 SUSE Linux Enterprise Desktop 和 SUSE Linux Enterprise Workstation Extension。默认情况 (`FLAVOR` = `default` | `xen` |...) 下，不会装载这些内核。此外，安装程序中不会提供这些不支持的模块，并且 `kernel-FLAVOR-extra` 包也不会包含在 SUSE Linux Enterprise 媒体中。
- 不是根据与 Linux 内核许可证兼容的许可证提供的内核模块也会污染内核。有关细节，请参见 `/usr/src/linux/Documentation/sysctl/kernel.txt` 及 `/proc/sys/kernel/tainted` 的状态。

40.6.1 技术背景

- Linux 内核：在 SUSE Linux Enterprise `12 SP5` 上，`/proc/sys/kernel/unsupported` 的值默认设为 2 (`装载不支持的模块时，syslog 中不发出警告`)。在安装程序和已安装好的系统中，均会使用此默认值。有关更多信息，请参见 `/usr/src/linux/Documentation/sysctl/kernel.txt`。
- `modprobe`：用于检查模块依赖项和装载模块的 `modprobe` 实用程序会相应地检查 `supported` 标志的值。如果该值为 “yes” 或 “external”，则会装载该模块，否则不会。有关如何覆盖此行为的信息，请参见第 40.6.2 节 “使用不支持的模块”。

注意：支持

SUSE 一般不支持通过 `modprobe -r` 去除储存模块。

40.6.2 使用不支持的模块

尽管广泛可支持性非常重要，但有时会发生需要装载不支持模块的情况 (例如，要进行测试或调试，或者硬件供应商提供了热修复)。

- 要覆盖默认行为，请编辑 `/etc/modprobe.d/10-unsupported-modules.conf`，将变量 `allow_unsupported_modules` 的值更改为 `1`。如果 initrd 中需要一个不支持的模块，请记得运行 `dracut -f` 来更新 initrd。

如果只想尝试装载模块一次，可将 `--allow-unsupported-modules` 选项与 `modprobe` 结合使用。有关更多信息，请参见 `modprobe` 手册页。

- 在安装期间，可通过驱动程序更新磁盘添加不支持的模块，这样便会装载这些模块。要在引导过程中及引导之后强制装载不支持的模块，请使用内核命令行选项 `oem-modules`。安装和初始化 `suse-module-tools` 包时，系统将评估内核标志 `TAINT_NO_SUPPORT`（`/proc/sys/kernel/tainted`）。如果内核已污染，将启用 `allow_unsupported_modules`。这可以防止不支持的模块在安装的系统中装载失败。如果安装期间没有任何不支持的模块，并且未使用其他特殊的内核命令行选项（`oem-modules=1`），则默认行为仍是禁止不支持的模块。

请记住，装载和运行不支持的模块会导致 SUSE 不支持该内核和整个系统。

40.7 更多信息

- `man supportconfig` — `supportconfig` 手册页。
- `man supportconfig.conf` — supportconfig 配置文件的手册页。
- `man scatool` — `scatool` 手册页。
- `man scadb` — `scadb` 手册页。
- `man setup-sca` — `setup-sca` 手册页。
- https://mariadb.com/kb/en/ — MariaDB 文档。
- http://httpd.apache.org/docs/ 和 第 32 章 “Apache HTTP 服务器” — 有关 Apache Web 服务器的文档。
- 第 33 章 “使用 YaST 设置 FTP 服务器” — 有关如何设置 FTP 服务器的文档。
- http://www.suse.com/communities/conversations/sca-pattern-development/ — 有关如何创建（和测试）您自己的 SCA 模式的说明。
- http://www.suse.com/communities/conversations/basic-server-health-check-supportconfig/ — 使用 Supportconfig 的基本服务器状态检查。

- https://www.novell.com/communities/coolsolutions/cool_tools/create-your-own-supportconfig-plugin/ — 创建自己的 Supportconfig 插件。
- http://www.suse.com/communities/conversations/creating-a-central-supportconfig-repository/ — 创建中心 Supportconfig 储存库。

41 常见问题及其解决方案

本章将描述一系列可能发生的问题及其解决方法。即使您的情况并未精确地列在这里，也可能有足够相似的情况可提供解决您的问题的方法提示。

41.1 查找和收集信息

Linux 报告情况时是很详细的。当您的系统发生问题时，可以从几个位置查看，通常大多数是 Linux 系统的标准日志，有一些是与 SUSE Linux Enterprise Server 系统相关的日志。多数日志文件可以用 YaST（杂项 › 启动日志）查看。

使用 YaST 可以收集支持团队所需的所有系统信息。使用其他 › 支持，然后选择问题类别。当所有信息都被集合后，将其附加在您的支持请求。

将出现最常检查的日志文件的列表，并附有其典型用途说明。包含 ~ 的路径是指当前用户的用户主目录。

表 41.1 日志文件

日志文件	描述
`~/.xsession-errors`	来自当前运行的桌面应用程序的消息。
`/var/log/apparmor/`	来自 AppArmor 的日志文件，详细信息请参见《Security Guide》。
`/var/log/audit/audit.log`	来自审计的日志文件，用来跟踪对系统的文件、目录或资源的任何访问，并跟踪系统调用。有关详细信息，请参见《Security Guide》。
`/var/log/mail.*`	来自邮件系统的消息。
`/var/log/NetworkManager`	来自 NetworkManager 的日志文件通过网络连接收集问题

日志文件	描述
/var/log/samba/	包含 Samba 服务器及客户端日志消息的目录。
/var/log/warn	所有来自内核与系统日志守护程序的消息为“警告”或更高级别。
/var/log/wtmp	包含当前计算机会话的用户登录记录的二进制文件。可使用 last 查看它。
/var/log/Xorg.*.log	来自 X Window 系统的各种启动和运行时日志文件。在调试失败的 X 启动时，该日志很有用。
/var/log/YaST2/	包含 YaST 操作及其结果的目录。
/var/log/zypper.log	Zypper 的日志文件。

除了日志文件外，您的计算机还可提供关于运行中的系统的信息。请参见表 41.2: /proc 文件系统的系统信息

表 41.2 /proc 文件系统的系统信息

文件	描述
/proc/cpuinfo	包含处理器信息，包括处理器类型、制造商、型号和性能。
/proc/dma	显示当前使用的 DMA 通道。
/proc/interrupts	显示正在使用的中断和已使用的中断数量。
/proc/iomem	显示 I/O（输入/输出）内存的状态。
/proc/ioports	显示当时正在使用的 I/O 端口。
/proc/meminfo	显示内存状态。

文件	描述
/proc/modules	显示各个模块。
/proc/mounts	显示当前装入的设备。
/proc/partitions	显示所有硬盘的分区。
/proc/version	显示当前的 Linux 版本。

除了 /proc 文件系统外，Linux 内核还可通过 sysfs 模块（一个内存内的文件系统）导出信息。该模块表示了内核对象及其属性以及关系。有关 sysfs 的更多信息，请参见第 21 章 “使用 udev 进行动态内核设备管理” 中 udev 的环境。表 41.3 包含 /sys 下最常见目录的概述。

表 41.3 /sys 文件系统的系统信息

文件	描述
/sys/block	包含系统中发现的每个块设备的子目录。通常多数是磁盘类设备。
/sys/bus	包含每个物理总线类型的子目录。
/sys/class	包含按设备功能类型分组的子目录（如图形、网络、打印机等）
/sys/device	包含全局设备层次结构。

Linux 自带了几个用于系统分析和监视的工具。请参见《System Analysis and Tuning Guide》，第 2 章 “System Monitoring Utilities”以选择在系统诊断中使用的最重要的工具。

以下包含的每个情景都以一个描述问题的标题开头，后跟一两段内容，提供建议的解决方案、解决方案详细信息的参考，以及对其他可能相关的情景的交叉引用。

41.2 安装问题

安装问题是指计算机无法进行安装的情况。一种可能是完全无法进行安装，另一种是无法启动图形安装程序。本节将着重介绍您可能会遇到的一些典型问题，并提供可行的解决方案或针对此类情况的变通方案。

41.2.1 检查媒体

如果您使用 SUSE Linux Enterprise Server 安装媒体时遇到任何问题，请检查安装媒体的完整性。从该媒体引导，然后从引导菜单中选择检查安装媒体。在运行中的系统上，启动 YaST 并选择软件 › 媒体检查。要检查 SUSE Linux Enterprise Server 媒体，将它插入驱动器中，在 YaST 的媒体检查屏幕中单击启动检查。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。媒体问题可能是您在自行刻录媒体时发生的。以较低的速度 (4x) 刻录媒体有助于避免问题。

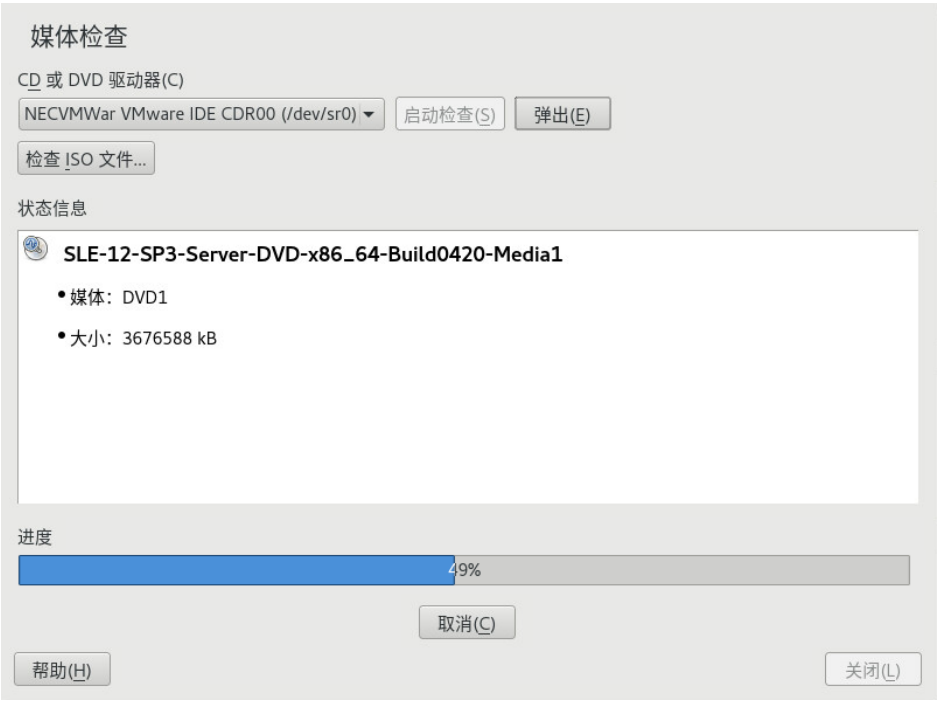


图 41.1 检查媒体

41.2.2 没有可用于引导的 DVD 驱动器

如果您的计算机没有可引导的 DVD-ROM 驱动器，或者 Linux 不支持您的驱动器，则有几种无需内置 DVD 驱动器便可安装计算机的方法：

使用外置的引导设备

如果您的 BIOS 和安装内核支持，请从外部 DVD 驱动器或 USB 储存设备引导。有关如何创建可引导 USB 储存设备的说明，请参见《部署指南》，第 6 章 “使用 YaST 进行安装”，第 6.2.2 节 “PC (AMD64/Intel 64/ARM AArch64)：系统启动”。

通过 PXE 进行网络引导

如果计算机没有 DVD 驱动器，但是提供了有效的以太网连接，则可以执行完全基于网络的安装。详情请参见《部署指南》，第 10 章 “远程安装”，第 10.1.3 节 “通过 VNC 进行远程安装 — PXE 引导和网络唤醒”和《部署指南》，第 10 章 “远程安装”，第 10.1.6 节 “通过 SSH 进行远程安装 — PXE 引导和网络唤醒”。

41.2.2.1 外置引导设备

Linux 支持多数的现有 DVD 驱动器。如果系统上没有 DVD 驱动器，仍可能用通过 USB、FireWire 或 SCSI 连接的外部 DVD 驱动器引导系统。这主要取决于 BIOS 与所使用硬件的交互。如果遇到问题，有时执行 BIOS 更新可能会有用。

从 Live CD 安装时，也可以创建用于引导的 “Live 闪存盘”。

41.2.3 从安装媒体引导失败

计算机不从安装媒体引导的一个原因可能是 BIOS 中引导顺序的设置不正确。BIOS 引导顺序必须将 DVD 驱动器设置为第一引导项。否则计算机将尝试从其他媒体引导，通常为硬盘。有关更改 BIOS 引导顺序的指南可在随主板提供的文档中找到，也可以参见以下段落。

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，计算机会执行若干项诊断硬件测试。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、F1 或 Esc。按此键，直到出现 BIOS 设置屏幕。

过程 41.1 更改 BIOS 引导顺序

1. 使用由引导例程声明的适当键输入 BIOS，然后等待 BIOS 屏幕出现。
2. 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。
3. 在所打开的屏幕中，查找名为 BOOT SEQUENCE 或 BOOT ORDER 的子项。按 Page ↑ 或 Page ↓ 键来更改设置，直到 DVD 驱动器在最前面。
4. 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。

过程 41.2 更改 SCSI BIOS（ADAPTEC 主机适配器）中的引导顺序

1. 按 Ctrl–A 打开设置。
2. 选择磁盘实用程序。现在将显示所连接的硬件组件。
   记下您 DVD 驱动器的 SCSI ID。
3. 按 Esc 退出菜单。
4. 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。
5. 输入 DVD 驱动器的 ID，然后再次按 Enter 键。
6. 按 Esc 键两次以返回到 SCSI BIOS 的开始屏幕。
7. 退出此屏幕，并确认是以引导计算机。

不论最终安装将使用何语言及键盘布局，大多数 BIOS 配置都使用美式键盘布局，如下图所示：

图 41.2 美式键盘布局

41.2.4 无法引导

某些硬件类型（主要是过旧或非常新的硬件）可能无法安装。此问题往往是由于安装内核中缺少此类硬件的支持或该内核中包含的某些功能（如 ACPI，它仍会在某些硬件上引起问题）而引起的。

如果系统无法使用第一个安装引导屏幕上的标准安装方式进行安装，请尝试使用以下方法：

1. 将第一张 DVD 留在驱动器中，然后使用 Ctrl–Alt–Del 或硬件重设置按钮来重引导计算机。
2. 在出现引导屏幕时，按 F5 键，使用键盘上的箭头键浏览至无 ACPI，然后按 Enter 键起动引导和安装过程。此选项将禁用对 ACPI 电源管理技术的支持。
3. 按《部署指南》，第 6 章 “使用 YaST 进行安装”中所述的步骤进行安装。

如果此操作失败，请按照以上步骤继续，但应选择安全设置。此选项将禁用 ACPI 和 DMA 支持。大多数硬件应使用此选项引导。

如果以上两个选项都失败，请使用引导选项提示向安装内核传递支持此硬件类型所需的任何其他参数。关于可用作引导选项的参数的更多信息，请参见 `/usr/src/linux/Documentation/kernel-parameters.txt` 中的内核文档。

提示：获取内核文档

安装 `kernel-source` 包以查看内核文档。

在为完成安装执行引导之前，还可以在引导提示下输入其他与 ACPI 相关的内核参数：

`acpi=off`

此参数禁用计算机上的整个 ACPI 子系统。如果您的计算机无法处理 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=force`

始终启用 ACPI，即使计算机使用的是 2000 年以前的 BIOS。如果除了 `acpi=off` 之外还设置了此参数，则此参数将启用 ACPI。

`acpi=noirq`

不要将 ACPI 用于 IRQ 路由。

`acpi=ht`

只运行足够的 ACPI 来启用超线程。

`acpi=strict`

降低对不严格遵循 ACPI 规格的平台的容许度。

`pci=noacpi`

禁用新 ACPI 系统的 PCI IRQ 路由。

`pnpacpi=off`

在您的 BIOS 设置包含错误的中断或端口时，此选项用于串行或并行问题。

`notsc`

禁用时戳计数器。此选项可用于解决系统上的计时问题。这是一项新功能，如果看到计算机上有衰退，尤其是时间相关的或甚至完全挂起，此选项值得一试。

`nohz=off`

禁用 nohz 功能。如果您的计算机挂起，则此选项可能有帮助。否则就没有用处。

一旦确定了正确的参数组合，YaST 会自动将其写入引导加载程序配置中以确保系统下一次能够正确引导。

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。如果内存测试返回一个错误，则通常这是硬件错误。

41.2.5 无法启动图形安装程序

在将媒体插入驱动器并重引导计算机之后，出现安装屏幕，但是在选择安装之后，图形安装程序没有启动。

有多种方法可解决此情况：

- 尝试为安装对话框另选一种屏幕分辨率。
- 选择文本方式进行安装。
- 使用图形安装程序进行远程安装（通过 VNC）。

过程 41.3 安装时更改屏幕分辨率

1. 引导以安装。
2. 按 F3 键打开一个菜单，从中选择一个较低的安装分辨率。
3. 选择安装，然后按《部署指南》，第 6 章 “使用 YaST 进行安装”中所述的步骤进行安装。

过程 41.4 用文本方式进行安装

1. 引导以安装。
2. 按 F3，然后选择文本方式。
3. 选择安装，然后按《部署指南》，第 6 章 “使用 YaST 进行安装”中所述的步骤进行安装。

过程 41.5 VNC 安装

1. 引导以安装。
2. 在引导选项提示下输入以下文本：

   ```
   vnc=1 vncpassword=SOME_PASSWORD
   ```

   将 SOME_PASSWORD 替换为用于 VNC 安装的口令。
3. 选择安装，然后按 Enter 键启动安装。

   系统未正确启动图形安装例程，而是仍以文本方式继续运行，接着暂停，显示一条消息，其中包含了可通过浏览器界面或 VNC 查看器应用程序访问安装程序的 IP 地址和端口号。

4. 如果使用浏览器来访问安装程序，请启动浏览器并输入由未来 SUSE Linux Enterprise Server 计算机上的安装例程提供的地址信息，然后按 Enter：

```
http://IP_ADDRESS_OF_MACHINE:5801
```

随后浏览器窗口中将打开一个对话框，提示您输入 VNC 口令。输入口令，然后按《部署指南》，第 6 章 “使用 YaST 进行安装”中所述的步骤进行安装。

重要：跨平台支持

通过 VNC 安装这一方法可在任意操作系统下的任意浏览器上进行，只要启用了 Java 支持即可。

看到提示时，提供您的 VNC 查看器的 IP 地址和口令。然后，将打开一个窗口，其中显示了多个安装对话框。照常进行安装。

41.2.6 只启动简陋的引导屏幕

将媒体插入了驱动器，BIOS 例程结束，但是系统未启动图形引导屏幕。而是启动了一个非常简陋的文本界面。如果计算机的图形内存不足而无法生成图形引导屏幕，则可能发生这种情况。

虽然文本引导屏幕看起来比较简陋，但是它所提供的功能与图形引导屏幕几乎是相同的。

引导选项

与图形界面不同的是，不能使用键盘的鼠标键来选择其他引导选项。文本引导屏幕上的引导菜单提供了一些可在引导提示下输入的关键字。这些关键字与图形版本中提供的选项相对应。输入您的选择，然后按 Enter 起动引导过程。

自定义引导选项

在选择引导选项之后，请在引导提示下输入相应的关键字，或者根据第 41.2.4 节 “无法引导” 中所述输入自定义引导选项。要启动安装过程，请按 Enter 键。

屏幕分辨率

使用功能键 (F1 ... F12) 确定安装的屏幕分辨率。如果需要以文本方式引导，请选择 F3。

41.2.7 日志文件

有关安装期间创建的日志文件的详细信息，请参见第 40.5 节 “在安装过程中收集信息”。

41.3 引导问题

引导问题是指系统不能正确引导时出现的情况（不能引导到预期的目标和登录屏幕）。

41.3.1 GRUB 2 引导加载程序无法装载

如果硬件运行正常，则可能是由于引导加载程序已损坏而使 Linux 无法在计算机上启动。在这种情况下，需要修复引导加载程序。为此，您需要按第 41.6.2 节 “使用救援系统”中所述启动救援系统，然后根据第 41.6.2.4 节 “修改和重新安装引导加载程序”中的说明操作。

另外，您可以按以下方式使用救援系统来修复引导加载程序。从安装媒体引导计算机。在引导屏幕中，选择更多 › 引导 Linux 系统。使用默认内核选项选择包含已安装的系统和内核的磁盘。

系统引导后，启动 YaST 并切换到系统 › 引导加载程序。确保启用了将通用引导代码写入 MRB 选项，然后按确定。如此会通过重写来修复损坏的引导加载程序，或者安装缺失的引导加载程序。

其他导致计算机无法引导的原因可能与 BIOS 相关：

BIOS 设置

检查与硬盘相关的 BIOS 设置。如果在当前的 BIOS 设置中找不到硬盘本身，GRUB 2 可能就不能启动。

BIOS 引导顺序

请检查您的系统引导顺序中是否包含硬盘。如果未启用硬盘选项，即使系统正确安装，在访问所需的硬盘时仍可能无法引导。

41.3.2 无图形登录

如果计算机能够启动，但是无法引导进入图形登录管理器，则问题可能出在默认的 systemd 目标选项或 X Window 系统的配置上。要检查当前的 systemd 默认目标，请运行命令 `sudo systemctl get-default`。如果返回的值为 not `graphical.target`，请运行命令

`sudo systemctl isolate graphical.target`。如果图形登录屏幕已启动，请登录并启动 YaST › 系统 › 服务管理器，然后将默认系统目标设置为图形界面。此后，系统应该能够引导进入图形登录屏幕。

如果即使已引导或者切换到图形目标，图形登录屏幕也不启动，则原因可能是您的桌面或 X Window 软件设置错误或者已损坏。请检验 `/var/log/Xorg.*.log` 中的日志文件，查找它尝试启动的 X 服务器发出的详细消息。如果桌面在启动期间发生错误，可能会在系统日记中记录错误讯息，您可以使用命令 `journalctl` 查询该日记（有关详细信息，请参见第 15 章 “journalctl：查询 systemd 日记”）。如果这些错误消息指出问题出在 X 服务器中的配置上，请尝试修正这些问题。如果图形系统仍无法启动，请考虑重安装图形桌面。

41.3.3 无法装入 Btrfs 根分区

如果 `btrfs` 根分区已损坏，请尝试以下选项：

- 使用 `-o recovery` 选项装入该分区。
- 如果不起使用，请在您的根分区上运行 `btrfs-zero-log`。

41.3.4 强制检查根分区

如果根分区损坏，请在引导提示符处使用参数 `forcefsck` 。这样会将选项 `-f` (force) 传递给 `fsck` 命令。

41.4 登录问题

登录问题是指计算机虽然引导到预期的欢迎屏幕或登录提示界面，却拒绝接受用户名和口令，或者虽然接受了用户名和口令，但是行为异常（无法启动图形桌面、发生错误或转到了命令行等）。

41.4.1 有效的用户名和口令组合失败

如果系统配置为使用网络身份验证或目录服务，但出于某些原因无法从为其配置的服务器上检索到结果，通常就会发生此问题。只有作为唯一本地用户的 `root` 用户仍能登录到这些计算机。以下是计算机看似能够运行但无法正确处理登录的常见原因：

- 网络出现故障。有关此问题的进一步说明，请转到第 41.5 节 “网络问题”。
- DNS 在当时不起作用（这使得 GNOME 不起作用，并使系统无法向安全服务器发出经验证的请求）。如果是这种情况，则表现为计算机对任何操作的响应都需要极其长的时间。有关该主题的详细信息，请参见第 41.5 节 “网络问题”。
- 如果将系统配置为使用 Kerberos，则系统的本地时间与 Kerberos 服务器时间之间的差异可能超过了可接受的值（通常为 300 秒）。如果 NTP（网络时间协议）未正确地起作用，或者本地 NTP 服务器不起作用，则 Kerberos 身份验证将不再工作，因为该身份验证依赖于整个网络的通用时钟同步。
- 系统的身份验证配置不正确。请对相关的 PAM 配置文件进行检查以确定是否存在指令输入错误或排序错误。有关 PAM 的其他背景信息及相关配置文件的语法，请参见《Security Guide》, 第 2 章 “Authentication with PAM”。
- 主分区是加密的。有关该主题的详细信息，请参见第 41.4.3 节 “登录至加密的主分区失败”。

在不涉及外部网络问题的所有情况下，解决方法是将系统重引导到单用户方式并修复配置，然后再次引导到操作方式并重试登录。要引导到单用户方式，请执行以下操作：

1. 重引导系统。此时将出现引导屏幕，其中显示一个提示。
2. 按 Esc 退出启动屏幕，并转到 GRUB 2 基于文本的菜单。
3. 按 B 进入 GRUB 2 编辑器。
4. 在包含内核参数的行中添加以下参数：

   ```
   systemd.unit=rescue.target
   ```

5. 按 F10。

6. 输入 `root` 的用户名与口令。
7. 进行必要的一切更改。
8. 在命令行中输入 `systemctl isolate graphical.target`，以引导进入完全的多用户和网络模式。

41.4.2 有效的用户名和口令不被接受

这是到目前为止用户最常遇到的问题，因为有许多原因可能引起该问题。登录失败可由多种原因造成，取决于您是使用本地用户管理和身份验证，还是使用网络身份验证。

本地用户管理失败可由以下原因造成：

- 用户可能输入了错误的口令。
- 用户包含桌面配置文件的主目录已损坏或被写保护。
- 验证该特定用户的 X Window 系统可能存在问题，尤其是在安装当前版本之前，该用户的主目录已被其他 Linux 分发版使用时。

要找到本地登录失败的原因，请执行如下操作：

1. 在尝试调试整个身份验证机制之前，请检查用户所记的口令是否正确。如果用户可能记错了口令，请使用“YaST 用户管理”模块更改用户的口令。注意 Caps Lock 键，如果需要请解锁。
2. 以 `root` 身份登录，并使用 `journalctl -e` 检查系统日记，找出登录过程和 PAM 的错误讯息。
3. 尝试从控制台登录（使用 Ctrl–Alt–F1）。如果成功了，则问题不在 PAM 上，因为可以在该计算机上身份验证此用户。尝试找出与 X Window 系统或 GNOME 桌面相关的任何问题。有关更多信息，请参考第 41.4.4 节 “登录成功但 GNOME 桌面发生故障”。
4. 如果用户的主目录被其他 Linux 产品所使用，请将该用户主目录中的 `Xauthority` 文件删除。使用控制台登录（通过 Ctrl–Alt–F1），然后以该用户的身份运行 `rm .Xauthority`。这样应该可以消除该用户的 X 身份验证问题。然后再次尝试图形登录。

5. 如果桌面由于配置文件损坏而无法启动，请参见第 41.4.4 节 “登录成功但 GNOME 桌面发生故障”。

下面列出了在特定的计算机上对特定用户进行的网络身份验证可能失败的常见原因：

- 用户可能输入了错误的口令。
- 用户名存在于计算机的本地身份验证文件中，但同时网络身份验证系统也提供了该用户名，从而引起冲突。
- 主目录存在，但已损坏或不可用。该目录可能处于写保护状态或位于此刻无法访问的服务器上。
- 用户无权登录到身份验证系统中的该特定主机。
- 计算机出于某种原因更改了主机名，而用户无权登录到该主机。
- 计算机无法访问包含该用户信息的身份验证服务器或目录服务器。
- 验证该特定用户的 X Window 系统可能存在问题，尤其是在安装当前办法之前，该用户的主目录已被其他 Linux 分发版使用时。

要通过网络身份验证找到登录问题的原因，请执行以下步骤：

1. 在尝试调试整个身份验证机制之前，请检查用户所记的口令是否正确。
2. 确定计算机在身份验证时要依赖的目录服务器，并确保计算机在正常运行且与其他计算机正常通讯。
3. 确定该用户的用户名和口令在其他计算机上是否有效，以确保存在该用户的身份验证数据且已正确分发。
4. 确定其他用户是否可以登录到该故障计算机。如果其他用户可以正常登录，或 `root` 可以登录的话，请登录并使用 `journalctl -e` > 文件检查系统日记。找到与登录尝试相对应的时间戳记，然后确定 PAM 是否生成了任何错误消息。
5. 尝试从控制台登录（使用 Ctrl–Alt–F1 ）。如果成功，则问题不在用户主目录中的 PAM 或目录服务器上，因为可以在该计算机上验证此用户。尝试找出与 X Window 系统或 GNOME 桌面相关的任何问题。有关更多信息，请参考第 41.4.4 节 “登录成功但 GNOME 桌面发生故障”。

6. 如果用户的主目录被其他 Linux 产品所使用，请将该用户主目录中的 `Xauthority` 文件删除。使用控制台登录（通过 Ctrl–Alt–F1 ），然后以该用户的身份运行 `rm .Xauthority`。这样应该可以消除该用户的 X 身份验证问题。然后再次尝试图形登录。
7. 如果桌面由于配置文件损坏而无法启动，请参见第 41.4.4 节 “登录成功但 GNOME 桌面发生故障”。

41.4.3 登录至加密的主分区失败

对于便携式计算机建议使用加密的主分区。如果无法登录到您的便携式计算机，原因通常很简单：您的分区无法解锁。

在引导时，需要输入通行口令来解锁加密的分区。如果不输入它，引导进程继续，但保持分区锁定。

要解锁您的加密分区，请如下操作：

1. 按 Ctrl–Alt–F1 切换到文本控制台。
2. 成为 `root` 用户。
3. 用以下步骤重启动解锁进程：

```
systemctl restart home.mount
```

4. 输入您的通行口令以解锁加密的分区。
5. 用 Alt–F7 退出文本控制台并切换回登录屏幕。
6. 如常登录。

41.4.4 登录成功但 GNOME 桌面发生故障

如果是这种情况，可能您的 GNOME 配置文件已损坏。可能出现的症状包括键盘不起作用、屏幕几何图形变形，甚至整个屏幕变成灰色。而最重要的差别在于其他用户登录时，该计算机能正常运行。那么可能只需将用户的 GNOME 配置目录移到某个新位置，以便使 GNOME 初始化一个新的桌面，这样就能很快地解决此问题。虽然用户不得不重配置 GNOME，但不会丢失任何数据。

1. 按 Ctrl–Alt–F1 切换到文本控制台。
2. 用您的用户名登录。
3. 将用户的 GNOME 配置目录移到某个临时位置：

```
mv .gconf  .gconf-ORIG-RECOVER
mv .gnome2 .gnome2-ORIG-RECOVER
```

4. 注销。
5. 再次登录，但别运行任何应用程序。
6. 通过以下命令将 `~/.gconf-ORIG-RECOVER/apps/` 目录复制回新的 `~/.gconf` 目录，这样就能恢复您的个人应用程序配置数据（包括 Evolution 电子邮件客户端数据）：

```
cp -a .gconf-ORIG-RECOVER/apps .gconf/
```

如果这引起登录问题，则尝试只恢复重要的应用程序数据并重配置其他的应用程序。

41.5 网络问题

系统的许多问题可能都与网络相关，即使初看起来不是这样。例如，系统不允许用户登录可能是某种网络问题造成的。本节介绍一个简单的核对表，您可以使用它来确定任何所遇到的网络问题的原因。

过程 41.6 如何识别网络故障

在检查计算机的网络连接时，请执行如下操作：

1. 如果使用的是以太网连接，请首先检查硬件。确保您的网络电缆已正确插入计算机和路由器（或集线器等）。以太网连接器旁边的控制灯通常应全部激活。
   如果连接失败，请检查网线在别的计算机上是否正常。如果正常，则可能是网卡引起了该问题。如果您的网络设置中有集线器或交换机，它们也可能有故障。
2. 如果使用的是无线连接，请检查是否可与其他计算机建立此无线链接。如果没有，请联系无线网络的管理员。

3. 一旦完成了对基本网络连通性的检查，请尝试找出没有响应的服务。收集设置中所需的所有网络服务器的地址信息。在相应的 YaST 模块中查找这些信息，或者询问您的系统管理员。下面列出了设置中涉及的一些典型网络服务器以及服务中断的症状。

   DNS（名称服务）

   名称服务中断或发生故障会在许多方面影响网络运行。如果本地计算机依赖于任一网络服务器进行身份验证，而这些服务器由于名称解析问题而无法找到，则用户甚至还不能登录。网络中由中断的名称服务管理的计算机将无法“看到”彼此且不能通信。

   NTP（时间服务）

   NTP 服务发生故障或完全中断可能会影响 Kerberos 身份验证和 X 服务器功能。

   NFS（文件服务）

   如果任何应用程序所需的数据储存在 NFS 装入目录中，则一旦此服务停止或配置错误，应用程序便无法启动或正常运行。最坏的情况是，如果由于 NFS 服务器发生故障而无法找到包含 `.gconf` 子目录的用户主目录，则该用户主目录所属的用户的个人桌面配置将无法启动。

   Samba（文件服务）

   如果任何应用程序需要的数据储存在有故障的 Samba 服务器上的某个目录中，它便无法启动或正常运行。

   NIS（用户管理）

   如果您的 SUSE Linux Enterprise Server 系统依赖有故障的 NIS 服务器提供用户数据，用户将无法登录此计算机。

   LDAP（用户管理）

   如果您的 SUSE Linux Enterprise Server 系统依赖有故障的 LDAP 服务器提供用户数据，用户将无法登录此计算机。

   Kerberos（身份验证）

   身份验证无法进行，登录至任何计算机都会失败。

   CUPS（网络打印）

   用户无法打印。

4. 请检查网络服务器是否正在运行并且您的网络设置是否允许您建立连接：

重要：限制

下面介绍的调试步骤只适用于简单的网络服务器/客户端设置，不涉及任何内部路由。假设服务器和客户端都是同一子网的成员，不需要额外的路由。

a. 使用 `ping IP_ADDRESS/HOSTNAME`（用服务器的主机名或 IP 地址替换）来检查各台服务器是否正在运行，且能够对网络作出响应。如果此命令成功，表示您所查找的主机在正常运行，并且网络的名称服务配置正确。
如果 ping 命令失败，同时显示消息 目标主机不可访问，则表明您的系统或期望的服务器未正确配置或已宕机。从另一台计算机运行 `ping IP address` 或 `YOUR_HOSTNAME` 命令，来检查是否可连接您的系统。如果可以从其他计算机访问您的计算机，则表明该服务器未运行或未正确配置。
如果 ping 命令失败且返回 `unknown host`，则表示名称服务未正确配置或使用的主机名不正确。要对该问题进行进一步的检查，请参见步骤 4.b。如果 ping 命令仍然失败，则可能网卡未正确配置或网络硬件存在故障。

b. 请使用 `host HOSTNAME` 来检查您尝试连接的服务器的主机名是否正确地转译为 IP 地址，反之亦然。如果此命令返回了该主机的 IP 地址，则名称服务已在正常运行。如果 `host` 命令失败，请检查您主机上所有与名称和地址解析相关的网络配置文件：

`/etc/resolv.conf`

此文件用于对当前使用的名称服务器和域进行跟踪。您可手工修改该文件，或者由 YaST 或 DHCP 自动调整。建议采用自动调整。但是，请确保此文件具有以下结构并且所有的网络地址和域名都正确无误：

```
search FULLY_QUALIFIED_DOMAIN_NAME
nameserver IPADDRESS_OF_NAMESERVER
```

此文件中可以包含多个名称服务器地址，但是其中必须至少有一个能够对您的主机提供正确的名称解析。如果需要，请使用 YaST 的“网络设置”模块（“主机名/DNS”选项卡）调整此文件。

如果您的网络连接是通过 DHCP 处理的，请在 YaST 的“网络设置”模块（“主机名/DNS”选项卡）中选择通过 DHCP 设置主机名（可针对所有接口全局设置，也可以逐个接口设置）和通过 DHCP 更新名称服务器和搜索列表，以允许 DHCP 更改主机名和名称服务信息。

`/etc/nsswitch.conf`

此文件告诉 Linux 到何处查找名称服务信息。它应显示为：

```
 ...
hosts: files dns
networks: files dns
...
```

`dns` 条目是必需的。它告诉 Linux 要使用外部名称服务器。通常这些项是 YaST 自动管理的，但最好谨慎地检查。

如果主机上的所有相关条目均正确，请让系统管理员检查 DNS 服务器配置，以确定时区信息是否正确。有关 DNS 的详细信息，请参见第 26 章 “域名系统”。如果确信主机和 DNS 服务器的 DNS 配置正确，请检查网络和网络设备的配置。

c. 如果系统无法与网络服务器建立连接，并且已排除了名称服务出现问题的可能，则请检查网卡的配置。

使用 `ip addr show NETWORK_DEVICE` 命令来检查此设备是否已正确配置。确保已正确设置带网络掩码（`/MASK`）的 `inet address`。如果 IP 地址中出现错误或网络掩码中缺少一位，将使您的网络配置无法使用。如有必要，也在服务器上执行该检查。

d. 如果名称服务和网络硬件已正确配置且正在运行，但有些外部网络连接仍然长时间超时或完全失败，请使用 `traceroute FULLY_QUALIFIED_DOMAIN_NAME` 命令（以 `root` 用户的身份执行）来跟踪这些请求所经过的网络路由。此命令将列出某一请求从您的计算机传递到其目标所经过的所有网关（跳跃）。其中列出了每个跃点的响应时间以及此跃点是否可访问。请将 traceroute 和 ping 结合使用以确定故障原因并通知管理员。

一旦确定了网络故障的原因，就可以自行解决（如果问题出在您自己的计算机上），或告诉网络系统管理员您的发现，以便其重配置服务或修复必要的系统。

41.5.1 NetworkManager 问题

如果网络连接有问题，请按过程 41.6 “如何识别网络故障” 中所述缩小范围。如果 NetworkManager 看上去有问题，请执行以下步骤，以获取相关的日志来找出 NetworkManager 失败原因的线索：

1. 以 `root` 用户身份打开外壳并登录。
2. 重启动 NetworkManager：

   ```
   systemctl restart Network.Manager
   ```

3. 以普通用户身份打开一个网页，例如 http://www.opensuse.org，看看是否可以连接。
4. 收集 `/var/log/NetworkManager` 中有关 NetworkManager 状态的任何信息。

有关 NetworkManager 的更多信息，请参考第 37 章 “使用 NetworkManager”。

41.6 数据问题

数据问题是指无论计算机是否能够正确引导，有一点是明确的，即系统上的数据损坏并且系统需要恢复。这些情况下需要对关键数据进行备份，以便您能够在系统出现故障时恢复故障前的状态。

41.6.1 管理分区映像

有时您需要从整个分区甚至硬盘来执行备份。Linux 附带了 `dd` 工具，该工具可用于创建磁盘的精确副本。与 `gzip` 一起使用可节约一些空间。

过程 41.7 备份和恢复硬盘

1. 以 `root` 用户身份启动外壳。
2. 选择源设备。通常形如 `/dev/sda`（标记为 `SOURCE`）。
3. 确定要把您的映像储存在何处（标记为 `BACKUP_PATH`）。它不能与您的源设备相同。换句话说：如果从 `/dev/sda` 备份，则映像文件不得储存在 `/dev/sda` 下。

4. 运行以下命令创建压缩映像文件：

```
dd if=/dev/SOURCE | gzip > /BACKUP_PATH/image.gz
```

5. 用以下命令恢复硬盘：

```
gzip -dc /BACKUP_PATH/image.gz | dd of=/dev/SOURCE
```

如果只需备份某个分区，请将 SOURCE 占位符替换为相应的分区。在这种情况下，映像文件可以位于同一硬盘上不同的分区中。

41.6.2 使用救援系统

有多种原因会造成系统无法正常启动和运行。系统崩溃后造成文件系统损坏、配置文件损坏或引导加载程序配置损坏是最常见的原因。

为了帮助您解决这些状况，SUSE Linux Enterprise Server 提供了一套您可以引导的救援系统。该救援系统是一个小型 Linux 系统，可以装载到一个 RAM 磁盘并以 root 文件系统的形式装入，使您可以从外部访问 Linux 分区。使用该救援系统，可以恢复或修改系统中任何一个重要的方面：

- 操作任意类型的配置文件。
- 检查文件系统中的缺陷和启动自动修复进程。
- 访问“更改 root”环境下的已安装系统。
- 检查、修改和重新安装引导加载程序配置。
- 从安装有误的设备驱动程序或不可用内核恢复。
- 使用 parted 命令调整分区大小。在 GNU Parted 网站 http://www.gnu.org/software/parted/parted.html 上可以找到有关该工具的更多信息。

该救援系统可以从各种来源和位置进行装载。最简单的选择是从原始安装媒体上引导该救援系统。

注意：在 IBM Z 上启动救援系统

在 IBM Z 上，可将安装系统用于救援目的。要启动救援系统，请根据第 41.7 节 “IBM Z：将 initrd 用作救援系统”中的说明操作。

1. 将安装媒体插入 DVD 驱动器中。
2. 重引导系统。
3. 在引导屏幕上按 F4 并选择 DVD-ROM。然后从主屏幕选择救援系统。
4. 在 `Rescue:` 提示符处输入 `root`。无需口令。

如果硬件设置不包含 DVD 驱动器，可以从网络源引导该救援系统。以下示例适用于远程引导的情形，如果使用另一引导媒体（例如 DVD），则要相应地修改 `info` 文件，并像正常安装一样进行引导。

1. 进入 PXE 引导设置的配置界面，添加下面的行：`install=PROTOCOL://INSTSOURCE` 和 `rescue=1`。但如果需要启动修复系统，请使用 `repair=1`。如同正常的安装一样，`PROTOCOL` 代表任何支持的网络协议（NFS、HTTP、FTP 等），`INSTSOURCE` 代表网络安装源的路径。
2. 如《部署指南》, 第 9 章 “准备目标系统的引导”, 第 9.7 节 “局域网唤醒”中所述，使用“网络唤醒”引导系统。
3. 在 `Rescue:` 提示符处输入 `root`。无需口令。

一旦进入该救援系统，便可通过 Alt–F1 到 Alt–F6 键来访问虚拟控制台。

`/bin` 目录中提供了一个外壳和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系统的文件和网络实用程序。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` 和 `shutdown`，以及用于维护网络的 `ip` 和 `ss`。目录 `/usr/bin` 包含 vi 编辑器、find、less 和 SSH。

要查看系统讯息，请使用命令 `dmesg`，或使用 `journalctl` 查看系统日志。

41.6.2.1 检查和操作配置文件

举一个可以通过该救援系统修复配置的例子，假设有一个被损坏的配置文件，使该系统无法正常引导。您可以通过救援系统修复该配置文件。

要操作配置文件，请执行以下步骤：

1. 用上述方法之一启动救援系统。
2. 要在救援系统中装入位于 `/dev/sda6` 下的 root 文件系统，请使用如下命令：

   ```
   mount /dev/sda6 /mnt
   ```

   系统所有目录现在均位于 `/mnt` 之下
3. 将目录切换为所装入的 root 文件系统：

   ```
   cd /mnt
   ```
4. 在 vi 编辑器中打开有问题的配置文件。调整并保存配置。
5. 从救援系统中卸载 root 文件系统：

   ```
   umount /mnt
   ```
6. 重引导计算机。

41.6.2.2 修复和检查文件系统

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 root 文件系统，系统引导可能以显示 “kernel panic” 结束。在这种情况下，唯一的方法是从外部修复系统。该系统包含的实用程序可检查并修复 `btrfs`、`ext2`、`ext3`、`ext4`、`reiserfs`、`xfs`、`dosfs` 和 `vfat` 文件系统。您可以试试命令 `fsck.` 文件系统，例如，如果需要对 `btrfs` 进行文件系统检查，请使用 `fsck.btrfs`。

41.6.2.3 访问已安装系统

如果需要从救援系统访问已安装系统，需要在更改 root 环境中执行此操作。例如，修改引导加载程序配置或执行硬件配置实用程序。

要设置基于已安装系统的更改 root 环境，请执行以下步骤：

1.

   提示：导入 LVM 卷组

   如果您使用的是 LVM 设置（有关更多一般性细节，请参见《储存管理指南》），请导入所有现有的卷组，以便能够查找和装入设备：

   ```
   rootvgimport -a
   ```

   运行 `lsblk` 以检查哪个节点对应于根分区。在本例中，该节点为 `/dev/sda2`：

   ```
   lsblk
   NAME        MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
   sda           8:0    0 149,1G  0 disk
   ├─sda1        8:1    0     2G  0 part  [SWAP]
   ├─sda2        8:2    0    20G  0 part  /
   └─sda3        8:3    0   127G  0 part
     └─cr_home 254:0    0   127G  0 crypt /home
   ```

2. 从安装的系统装入根分区：

   ```
   mount /dev/sda2 /mnt
   ```

3. 装入 `/proc`、`/dev` 和 `/sys` 分区：

   ```
   mount -t proc none /mnt/proc
   mount --rbind /dev /mnt/dev
   mount --rbind /sys /mnt/sys
   ```

4. 现在可以“更改根分区”为新的环境，并保留 `bash` 外壳：

   ```
   chroot /mnt /bin/bash
   ```

5. 最后，装入已安装系统的剩余分区：

```
mount -a
```

6. 现在可以访问已安装系统了。在重引导系统之前，请用 `umount -a` 卸载分区并用“exit” 退出`更改 root`环境。

警告：限制

尽管对已安装系统的文件和应用程序有完全访问权，但仍有一些限制。运行的内核是用救援系统引导的那个，不是用更改 root 环境引导的那个。它仅支持关键硬件，如果内核版本不完全相同，则无法从已安装系统中添加内核模块。始终用 `uname -r` 检查当前正在运行的（救援）内核版本，然后查明更改 root 环境中 `/lib/modules` 目录下是否有匹配的子目录。如果是，可以使用已安装模块；否则，需要在其他媒体（例如闪盘）上提供模块的正确版本。很多时候，救援内核版本与已安装模块不同，这样您完全无法访问声卡等。也不可能启动图形用户界面。

还应注意，在使用 “F1” 到 Alt–F6 键切换控制台时，要退出 Alt–更改 root 环境。

41.6.2.4 修改和重新安装引导加载程序

有时，系统无法引导是因为引导加载程序配置已损坏。例如，如果没有正常工作的引导加载程序，启动例程将无法将物理驱动器转化为 Linux 文件系统中的实际位置。

要检查引导加载程序配置并重新安装引导加载程序，请执行以下步骤：

1. 如第 41.6.2.3 节 “访问已安装系统” 中所述执行必要的步骤以访问已安装系统。
2. 检查系统上是否已安装 GRUB 2 引导加载程序。如果未安装，请安装 `grub2` 包并运行

```
grub2-install /dev/sda
```

3. 根据第 12 章 “引导加载程序 GRUB 2”中所述的 GRUB 2 配置原则，检查下列文件是否正确配置，并根据需要应用修复。

   - `/etc/default/grub`
   - `/boot/grub2/device.map`（选用文件，手动创建后才存在）

- `/boot/grub2/grub.cfg`（此文件是系统生成的，不要编辑）
- `/etc/sysconfig/bootloader`

4. 使用以下命令序列重新安装引导加载程序：

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

5. 卸载分区，从“更改 root”环境中注销并重引导该系统：

```
umount -a
exit
reboot
```

41.6.2.5 修复内核安装

内核更新会产生新的 bug，这样会影响系统运行。例如，系统某个硬件的驱动程序有故障，您就无法访问和使用该硬件。在这种情况下，需还原到上一个工作内核（如果在系统上可用），或从安装媒体安装原始内核。

提示：如何在更新后保留最后几个内核

为了防止内核更新出现故障后无法进行引导，请使用内核多版本功能，并告知 `libzypp` 在更新后保留哪些内核。

例如，要始终保留最后两个内核和当前正在运行的内核，请将

```
multiversion.kernels = latest,latest-1,running
```

添加到 `/etc/zypp/zypp.conf` 文件。有关详细信息，请参见《部署指南》，第 15 章“安装多个内核版本”。

类似的情况是，当您需要重新安装或更新不受 SUSE Linux Enterprise Server 支持的设备的已损坏驱动程序时。例如，当硬件供应商使用特定设备时，比如硬件 RAID 控制器，它需要一个二进制驱动程序才能被操作系统识别。供应商通常会发布含有固定版或更新版必要驱动程序的驱动程序更新磁盘 (DUD)。

这两种情况下，您都需要以救援模式访问已安装系统，并修复内核相关问题，否则系统可能无法正确引导：

1. 从 SUSE Linux Enterprise Server 安装媒体引导。
2. 如果您正在从内核更新故障中恢复，请跳过此步骤。如果需要使用驱动程序更新磁盘 (DUD)，请在出现引导菜单后按 F6 装载驱动程序更新，并选择驱动程序更新的路径或 URL，然后确认是。
3. 从引导菜单中选择救援系统，并按 Enter。如果选择使用 DUD，将要求您指定储存驱动程序更新的位置。
4. 在 `Rescue:` 提示符处输入 `root`。无需口令。
5. 手动将目标系统和“更改 root”装入新环境。有关详细信息，请参见第 41.6.2.3 节 “访问已安装系统”。
6. 如果使用的是 DUD，请安装/重新安装/更新有故障的设备驱动程序包。始终确保已安装的内核版本与您正在安装的驱动程序版本完全相同。
   如果要修复有故障的内核更新安装，可以从安装媒体使用以下过程安装原始内核。
   a. 使用 `hwinfo --cdrom` 命令确定您的 DVD 设备信息，使用 `mount /dev/sr0 /mnt` 命令装入设备。
   b. 导航到 DVD 上储存内核文件的目录，例如，`cd /mnt/suse/x86_64/`。
   c. 使用 `rpm -i` 命令安装具有您风格的必需 `kernel-*`、`kernel-*-base` 和 `kernel-*-extra` 包。
7. 更新配置文件，必要时可重初始化引导加载程序。有关详细信息，请参见第 41.6.2.4 节 “修改和重新安装引导加载程序”。
8. 从系统驱动器中删除所有可引导的媒体，然后重引导。

41.7 IBM Z：将 initrd 用作救援系统

如果升级或修改了 SUSE® Linux Enterprise Server for IBM Z 的内核，则可能会意外地将系统重引导为不一致的状态，这样会使已安装系统的标准 IPL 过程失败。在这种情况下，您可以使用安装系统来提供救援。

按《部署指南》, 第 4 章 “在 IBM Z 上安装”, 第 4.2 节 “准备安装”中所述，对 SUSE Linux Enterprise Server for IBM Z 安装系统执行 IPL。选择开始安装，然后输入所有必需的参数。安装系统装载之后，系统会询问您要使用哪个显示类型来控制安装，此时请选择 SSH。现在，您可以不输入口令直接以 root 身份通过 SSH 登录系统。

在此情况下，没有做任何磁盘配置。需要在在能进入以前配置磁盘。

过程 41.8 配置 DASD

1. 用以下的命令配置 DASD：

   ```
   dasd_configure 0.0.0150 1 0
   ```

   DASD 以 0.0.0150 连接。 1 表示激活该磁盘（此位置若为 0 则将停用该磁盘）。 0 表示磁盘“无 DIAG 模式”（ 1 使磁盘的 DAIG 访问可用）。

2. 现在，DASD 为联机（用 cat /proc/partitions 检查），并可用于后续命令。

过程 41.9 配置 ZFCP 磁盘

1. 配置 zFCP 磁盘，首先要配置 zFCP 调节器。请使用以下命令完成该操作：

   ```
   zfcp_host_configure 0.0.4000 1
   ```

   0.0.4000 是调节器的连接目标通道 1 表示激活(0 使调节器无效)。

2. 调节器被激活后，可以配置磁盘。请使用以下命令完成该操作：

   ```
   zfcp_disk_configure 0.0.4000 1234567887654321 8765432100000000  1
   ```

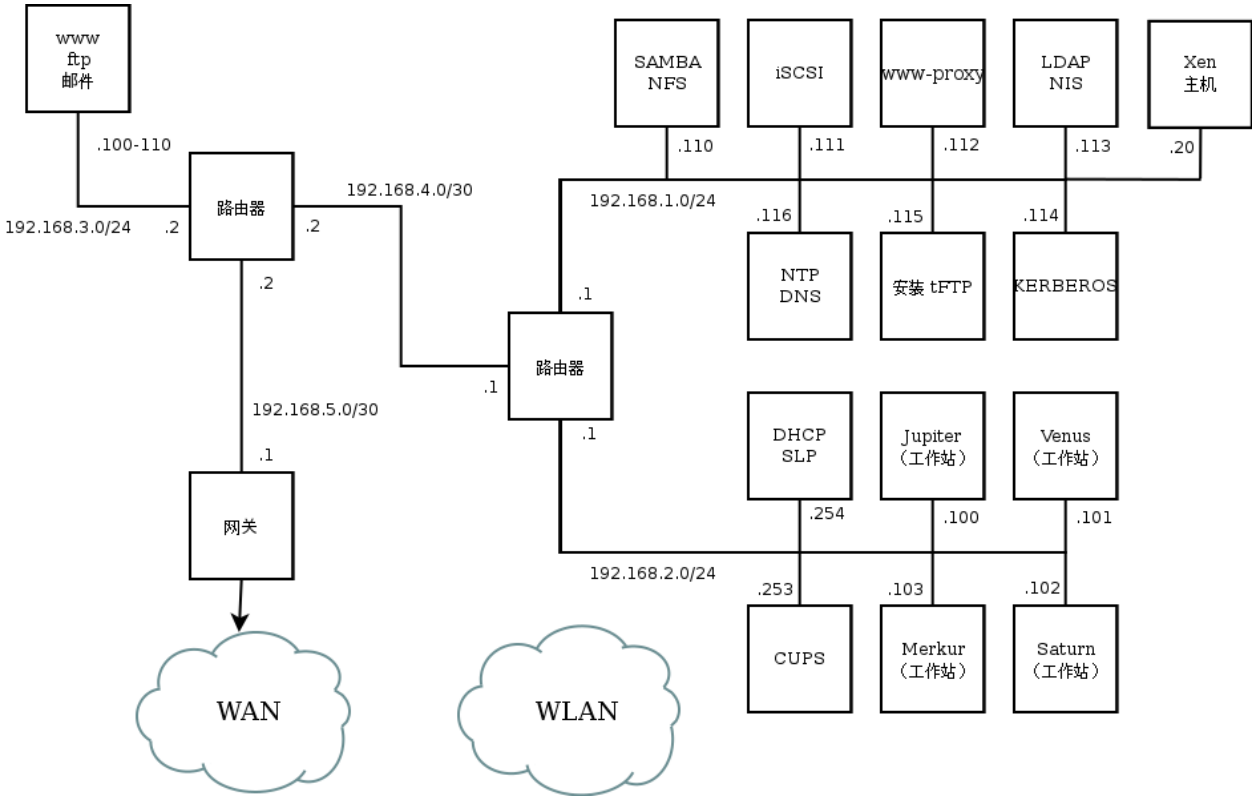
   0.0.4000 是以前用的通道 ID, 1234567887654321 为 WWPN (国际端口号码 World wide Port Number), 而 8765432100000000 是 LUN (逻辑单位号码 logical unit number). 1 意味着激活该磁盘 (这里的 0 将使该磁盘无效)。

3. 现在，zFCP 磁盘为联机（用 cat /proc/partitions 检查），并可用于后续命令。

现在，救援系统已完全设置好，您可以开始修复安装的系统。有关如何修复最常见问题的说明，请参见第 41.6.2 节 “使用救援系统”。

A 网络示例

此网络示例贯穿 SUSE® Linux Enterprise Server 文档的所有与网络相关的章节。



B GNU Licenses

This appendix contains the GNU Free Documentation License version 1.2.

GNU Free Documentation License

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA. Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or non-commercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the

name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.2
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section entitled “GNU
Free Documentation License”.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.