



SUSE LINUX

ADMINISTRATIONSHANDBUCH

9. Auflage 2004

Copyright ©

Dieses Werk ist geistiges Eigentum der SUSE LINUX AG.

Es darf als Ganzes oder in Auszügen kopiert werden, vorausgesetzt, dass sich dieser Copyrightvermerk auf jeder Kopie befindet.

Alle in diesem Buch enthaltenen Informationen wurden mit größter Sorgfalt zusammengestellt. Dennoch können fehlerhafte Angaben nicht völlig ausgeschlossen werden. Die SUSE LINUX AG, die Autoren und die Übersetzer haften nicht für eventuelle Fehler und deren Folgen.

Die in diesem Buch verwendeten Soft- und Hardwarebezeichnungen sind in vielen Fällen auch eingetragene Warenzeichen; sie werden ohne Gewährleistung der freien Verwendbarkeit benutzt. Die SUSE LINUX AG richtet sich im Wesentlichen nach den Schreibweisen der Hersteller. Die Wiedergabe von Waren- und Handelsnamen usw. in diesem Buch (auch ohne besondere Kennzeichnung) berechtigt nicht zu der Annahme, dass solche Namen (im Sinne der Warenzeichen und Markenschutz-Gesetzgebung) als frei zu betrachten sind. Hinweise und Kommentare richten Sie an documentation@suse.de.

Autoren: Stefan Behlert, Frank Bodammer, Stefan Dirsch, Olaf Donjak, Roman Drahtmüller, Torsten Duwe, Thorsten Dubiel, Thomas Fehr, Stefan Fent, Werner Fink, Kurt Garloff, Carsten Groß, Joachim Gleißner, Andreas Grünbacher, Franz Hassels, Andreas Jaeger, Klaus Kämpf, Hubert Mantel, Johannes Meixner, Lars Müller, Matthias Nagorni, Anas Nashif, Siegfried Olschner, Peter Pöml, Thomas Renninger, Heiko Rommel, Marcus Schäfer, Nicolaus Schüler, Klaus Singvogel, Hendrik Vogelsang, Klaus G. Wagner, Rebecca Walter, Christian Zoz

Redaktion: Jörg Arndt, Karl Eichwalder, Antje Faber, Berthold Gunreben, Roland Haidl, Jana Jaeger, Edith Parzefall, Ines Pozo, Thomas Rölz, Thomas Schraitle

Layout: Manuela Piotrowski, Thomas Schraitle

Satz: DocBook-XML und \LaTeX

Dieses Buch ist auf 100 % chlorfrei gebleichtem Papier gedruckt.

Inhaltsverzeichnis

I	Installation	5
1	Die Installation	7
1.1	Textbasierte Installation mit YaST	8
1.1.1	Hintergrundinfo	8
1.1.2	Der Startbildschirm	8
1.1.3	Die Grundlage: linuxrc	10
1.2	SUSE LINUX starten	15
1.2.1	Der grafische SUSE-Bildschirm	16
1.3	Besondere Installationen	17
1.3.1	Installation ohne CD-ROM-Unterstützung	17
1.3.2	Installation via Netzwerk	17
1.4	Tipps und Tricks	19
1.4.1	Bootdiskette unter DOS erstellen	19
1.4.2	Bootdiskette unter einem unix-artigen System erstellen	21
1.4.3	Booten von Diskette (SYSLINUX)	22
1.4.4	CD 2 zum Booten verwenden	23
1.4.5	Unterstützt Linux mein CD-ROM-Laufwerk?	23
1.5	ATAPI-CD-ROM bleibt beim Lesen hängen	23
1.6	SCSI-Geräten und dauerhafte Gerätedateinamen	25
1.7	Partitionieren für Fortgeschrittene	25
1.7.1	Die Größe der Swap-Partition	26

1.7.2	Einsatzgebiet des Rechners	26
1.7.3	Optimierungsmöglichkeiten	28
1.8	LVM-Konfiguration mit YaST	30
1.8.1	Logical Volume Manager (LVM)	31
1.9	Soft-RAID	38
1.9.1	Gängige RAID-Level	39
1.9.2	Soft-RAID-Konfiguration mit YaST	40
2	Update des Systems und Paketverwaltung	43
2.1	SUSE LINUX aktualisieren	44
2.1.1	Vorbereitungen	44
2.1.2	Update mit YaST	46
2.1.3	Manuell gesteuertes Update	47
2.1.4	Aktualisieren einzelner Pakete	49
2.2	Softwareänderungen von Version zu Version	49
2.2.1	Von 7.3 auf 8.0	50
2.2.2	Von 8.0 auf 8.1	51
2.2.3	Von 8.1 auf 8.2	53
2.2.4	Von 8.2 auf 9.0	54
2.2.5	Von 9.0 auf 9.1	55
2.3	RPM – Der Paket-Manager der Distribution	58
2.3.1	Prüfen der Authentizität eines Pakets	59
2.3.2	Pakete verwalten	59
2.3.3	RPM und Patches	62
2.3.4	Anfragen stellen	63
2.3.5	Quellpakete installieren und kompilieren	66
2.3.6	RPM-Pakete mit build erzeugen	68
2.3.7	Tools für RPM-Archive und die RPM-Datenbank	69

II	Konfiguration	71
3	YaST im Textmodus (ncurses)	73
3.1	Bedienung	74
3.1.1	Das YaST-Kontrollzentrum	74
3.1.2	Die YaST-Module	75
3.2	Einschränkung der Tastenkombinationen	76
3.3	Aufruf der einzelnen Module	77
3.4	Das YaST Online Update	77
3.4.1	Das YOU-Modul	77
3.4.2	Online Update per Kommandozeile	78
4	Das X Window System	81
4.1	Installation des X Window Systems optimieren	82
4.1.1	Screen-Section	84
4.1.2	Device-Section	86
4.1.3	Monitor- und Modes-Section	87
4.2	Installation und Konfiguration von Fonts	88
4.2.1	Details zu Font-Systemen	89
4.3	Konfiguration von OpenGL/3D	94
4.3.1	Hardwareunterstützung	94
4.3.2	OpenGL-Treiber	95
4.3.3	Diagnose-Tool 3Ddiag	96
4.3.4	OpenGL-Testprogramme	96
4.3.5	Troubleshooting	96
4.3.6	Installationssupport	97
4.3.7	Weiterführende Online-Dokumentation	97

5 Druckerbetrieb	99
5.1 Grundlagen des Druckens	100
5.1.1 Beispiele für Standarddruckersprachen	100
5.1.2 Ablauf des Druckauftrages	100
5.1.3 Verschiedene Drucksysteme	104
5.2 Voraussetzungen zum Drucken	105
5.2.1 Allgemeine Voraussetzungen	105
5.2.2 Bestimmung eines geeigneten Druckertreibers	106
5.2.3 Zur GDI-Drucker Problematik	107
5.3 Drucker einrichten mit YaST	109
5.3.1 Warteschlangen und Konfigurationen	109
5.3.2 Grundsätzliches zur YaST Druckerkonfiguration	110
5.3.3 Automatische Konfiguration	111
5.3.4 Manuelle Konfiguration	112
5.4 Konfiguration für Anwendungsprogramme	115
5.5 Das CUPS-Drucksystem	116
5.5.1 Namenskonvention	116
5.5.2 IPP und Server	116
5.5.3 Konfiguration des CUPS-Servers	117
5.5.4 Netzwerkdrucker	118
5.5.5 Interne Auftragsbearbeitung	119
5.5.6 Tipps & Tricks	121
5.6 CUPS Schnelleinstieg	123
5.6.1 Überblick über das CUPS Drucksystem	123
5.6.2 Der Spooler	124
5.6.3 PPD Dateien	125
5.6.4 Der Filter	126
5.6.5 Die Backends	129
5.6.6 Kommandozeilentools	131
5.6.7 Das Web-Frontend des cupsd	133
5.6.8 CUPS im Netzwerk konfigurieren	134
5.7 Drucken aus Anwendungsprogrammen	139

5.8	Kommandozeilentools für das CUPS-Drucksystem	139
5.8.1	Für lokale Warteschlangen	140
5.8.2	Warteschlangen im Netz	142
5.8.3	Störungsbehebung mit obigen Befehlen bei CUPS . . .	143
5.9	Drucken im TCP/IP-Netzwerk	144
5.9.1	Bezeichnungen	144
5.9.2	Schnellkonfiguration für einen Client-Rechner	145
5.9.3	Protokolle zum Drucken im TCP/IP-Netzwerk	147
5.9.4	Filterung beim Drucken im Netzwerk	154
5.9.5	Problemlösungen	159
5.9.6	LPD-und-IPP Print-Server	164
6	Weiterführende Hinweise zum Druckerbetrieb	167
6.1	Manuelle Konfiguration lokaler Druckerschnittstellen	168
6.1.1	Parallele Schnittstellen	168
6.1.2	USB-Anschluss	170
6.1.3	IrDA-Druckerschnittstelle	172
6.1.4	Serielle Schnittstellen	173
6.2	Manuelle Konfiguration von LPRng/lpdfilter	173
6.3	Der Druckerspooler LPRng	173
6.3.1	Drucken aus Anwendungsprogrammen	175
6.4	Kommandozeilentools für den LPRng	175
6.4.1	Für lokale Warteschlangen	175
6.4.2	Für entfernte Warteschlangen	178
6.4.3	Störungsbehebung mit obigen Befehlen beim LPRng .	179
6.5	Der Druckerfilter des LPRng/lpdfilter Drucksystems	180
6.5.1	Konfiguration des lpdfilter	182
6.5.2	Eigene Ergänzungen für den lpdfilter	182
6.5.3	Fehlersuche beim lpdfilter	189
6.6	Etwas über Ghostscript	190
6.6.1	Beispiele für die Arbeit mit Ghostscript	191
6.7	Etwas über a2ps	194

6.7.1	Direkte Druckerausgabe einer Textdatei mit a2ps . . .	195
6.8	PostScript-Umformatierung mit den psutils	195
6.8.1	psnup	196
6.8.2	pstops	196
6.8.3	psselect	198
6.8.4	Kontrolle am Bildschirm mit Ghostscript	198
6.9	Zur Kodierung von ASCII-Text	199
6.9.1	Veranschaulichung	200
7	Booten und Bootmanager	203
7.1	Der Bootvorgang auf dem PC	204
7.1.1	Master Boot Record	204
7.1.2	Bootsektoren	204
7.1.3	Booten von DOS oder Windows	205
7.2	Bootkonzepte	205
7.3	Map Files, GRUB und LILO	206
7.4	Booten mit GRUB	207
7.4.1	Das GRUB-Bootmenü	208
7.4.2	Die Datei device.map	214
7.4.3	Die Datei /etc/grub.conf	215
7.4.4	Bootpasswort setzen	216
7.4.5	Mögliche Probleme und weiterführende Informationen	217
7.5	Linux-Bootloader entfernen	218
7.5.1	MBR wiederherstellen (DOS/Win9x/ME)	218
7.5.2	MBR wiederherstellen (Windows XP)	219
7.5.3	MBR wiederherstellen (Windows 2000)	219
7.6	Für alle Fälle: Boot-CD erstellen	219
7.6.1	Boot-CD mit ISOLINUX	219

8	Mobiles Arbeiten unter Linux	223
8.1	PCMCIA	224
8.1.1	Die Hardware	224
8.1.2	Die Software	224
8.1.3	Die Konfiguration	226
8.1.4	Wenn's trotzdem nicht geht	228
8.1.5	Installation via PCMCIA	233
8.1.6	Weitere Hilfsprogramme	234
8.1.7	Kernel oder PCMCIA Paket aktualisieren	234
8.1.8	Weiterführende Informationen	235
8.2	SCPM – System Configuration Profile Management	235
8.2.1	Grundbegriffe und Grundlagen	236
8.2.2	Der YaST Profil-Manager	237
8.2.3	SCPM einrichten	237
8.2.4	Profile anlegen und verwalten	238
8.2.5	Zwischen Konfigurationsprofilen umschalten	239
8.2.6	Erweiterte Profileinstellungen	240
8.2.7	Profilauswahl beim Booten	241
8.2.8	Probleme und deren Lösung	243
8.3	IrDA – Infrared Data Association	244
8.3.1	Software	244
8.3.2	Konfiguration	245
8.3.3	Verwendung	245
8.3.4	Troubleshooting	246
8.4	Bluetooth – Geräte drahtlos verbinden	247
8.4.1	Profile	247
8.4.2	Software	247
8.4.3	Konfiguration	248
8.4.4	Systemkomponenten und nützliche Hilfsmittel	249
8.4.5	Beispiele	250
8.4.6	Troubleshooting	252
8.4.7	Weitere Informationen	253

9	Powermanagement	255
9.1	Stromsparfunktionen	256
9.2	APM	258
9.2.1	Der APM-Daemon (apmd)	259
9.2.2	Weitere Befehle	261
9.3	ACPI	261
9.3.1	Praxis	262
9.4	Pause für die Festplatte	267
9.5	Das powersave Paket	268
9.5.1	Konfiguration des powersave Pakets	269
9.5.2	Konfiguration von APM und ACPI	270
9.5.3	Zusätzliche ACPI-Features	272
9.5.4	Troubleshooting	272
9.6	Das YaST Powermanagement-Modul	275
III	System	281
10	SUSE LINUX auf AMD64 Systemen	283
10.1	64-bit SUSE LINUX für AMD64	284
10.1.1	Hardware	284
10.1.2	Software	284
10.1.3	Verwendung von 32-bit Software	285
10.1.4	Softwareentwicklung unter 64-bit	285
10.2	Weitere Informationen	285
11	Der Linux Kernel	287
11.1	Kernel-Update	288
11.2	Die Kernelquellen	289
11.3	Konfiguration des Kernels	289
11.3.1	Kommandozeilenkonfiguration	290
11.3.2	Konfiguration im Textmodus	290
11.3.3	Konfiguration unter dem X Window System	290

11.4	Kernel-Module	291
11.4.1	Erkennung der aktuellen Hardware mit hwinfo	291
11.4.2	Umgang mit Modulen	292
11.4.3	/etc/modprobe.conf	293
11.4.4	Kmod – der Kernel Module Loader	293
11.5	Einstellungen bei der Kernelkonfiguration	294
11.6	Übersetzen des Kernels	294
11.7	Kernel installieren	295
11.8	Festplatte nach der Übersetzung aufräumen	296
12	Systemmerkmale	297
12.1	Linux-Standards	298
12.1.1	Linux Standard Base (LSB)	298
12.1.2	Filesystem Hierarchy Standard (FHS)	298
12.1.3	teTeX — TeX unter SuSE Linux	298
12.1.4	Zu FTP	298
12.1.5	Zu HTTP	299
12.2	Hinweise zu speziellen Softwarepaketen	299
12.2.1	Paket bash und /etc/profile	299
12.2.2	Paket cron	300
12.2.3	Protokoll-Dateien – das Paket logrotate	300
12.2.4	Manual-Pages	302
12.2.5	Der Befehl ulimit	302
12.2.6	Der Befehl free	303
12.2.7	Die Datei /etc/resolv.conf	304
12.2.8	Einstellungen für GNU Emacs	304
12.3	Booten mit der initial ramdisk	305
12.3.1	Problemstellung	305
12.3.2	Konzept der initial ramdisk	306
12.3.3	Ablauf des Bootvorgangs mit initrd	306
12.3.4	Bootloader	307
12.3.5	Anwendung von initrd bei SUSE	308

12.3.6	Mögliche Schwierigkeit – Selbstkompilierte Kernel . .	309
12.3.7	Ausblick	310
12.4	linuxrc	310
12.4.1	Hauptmenü	311
12.4.2	Einstellungen	311
12.4.3	System-Information	311
12.4.4	Laden von Modulen	312
12.4.5	Parametereingabe	312
12.4.6	System / Installation starten	313
12.4.7	Parameter an linuxrc übergeben	315
12.5	Das SUSE Rettungssystem	316
12.5.1	Das Rettungssystem starten	317
12.5.2	Das Rettungssystem benutzen	318
12.6	Virtuelle Konsolen	321
12.7	Tastaturbelegung	321
12.8	Lokale Anpassungen – I18N/L10N	322
12.8.1	Einige Beispiele	323
12.8.2	Anpassung für Sprachunterstützung	323
13	Das Bootkonzept	325
13.1	Das init-Programm	326
13.2	Die Runlevels	327
13.3	Wechsel des Runlevels	328
13.4	Die Init-Skripten	330
13.4.1	Init-Skripten hinzufügen	332
13.5	Der YaST Runlevel-Editor	334
13.6	SuSEconfig und /etc/sysconfig	335
13.7	Der YaST Sysconfig-Editor	337

IV Netzwerk 341

14 Grundlagen der Vernetzung 343

14.1 TCP/IP – Das von Linux verwendete Protokoll	344
14.1.1 Schichtenmodell	345
14.1.2 IP-Adressen und Routing	348
14.1.3 Domain Name System – DNS	351
14.2 IPv6 – Internet der nächsten Generation	353
14.2.1 Vorteile von IPv6	354
14.2.2 Das Adresssystem von IPv6	355
14.2.3 IPv4 versus IPv6 – Wandern zwischen den Welten . . .	360
14.2.4 Weiterführende Literatur und Links zu IPv6	361
14.3 Manuelle Netzwerkkonfiguration	362
14.3.1 Konfigurationsdateien	363
14.3.2 Startup-Skripten	370
14.4 Die Einbindung ins Netzwerk	370
14.4.1 Vorbereitungen	371
14.4.2 Konfiguration mit YaST	371
14.4.3 Hotplug/PCMCIA	373
14.4.4 Konfiguration von IPv6	373
14.5 Routing unter SuSE Linux	374
14.6 DNS – Domain Name System	375
14.6.1 Nameserver BIND starten	375
14.6.2 Die Konfigurationsdatei /etc/named.conf	377
14.6.3 Konfigurationsoptionen im Abschnitt options	378
14.6.4 Der Konfigurationsabschnitt Logging	380
14.6.5 Aufbau der Zonen-Einträge	380
14.6.6 Aufbau der Zonendateien	381
14.6.7 Sichere Transaktionen	385
14.6.8 Zonendaten dynamisch aktualisieren	386
14.6.9 DNSSEC	387
14.6.10 Weitere Informationen	387

14.7	LDAP – Ein Verzeichnisdienst	387
14.7.1	LDAP versus NIS	390
14.7.2	Aufbau eines LDAP-Verzeichnisbaums	390
14.7.3	Serverkonfiguration mit slapd.conf	393
14.7.4	Handhabung von Daten im LDAP-Verzeichnis	398
14.7.5	Weitere Informationen	403
14.8	NIS – Network Information Service	404
14.8.1	NIS Master und Slave Server	404
14.8.2	Das NIS-Client-Modul in YaST	407
14.9	NFS – verteilte Dateisysteme	408
14.9.1	Importieren von Dateisystemen mit YaST	409
14.9.2	Manuelles Importieren von Dateisystemen	409
14.9.3	Exportieren von Dateisystemen mit YaST	410
14.9.4	Manuelles Exportieren von Dateisystemen	411
14.10	DHCP	413
14.10.1	Das DHCP-Protokoll	413
14.10.2	DHCP-Softwarepakete	414
14.10.3	Der DHCP-Server dhcpcd	414
14.10.4	Rechner mit fester IP-Adresse	417
14.10.5	Besonderheiten bei SUSE Linux	418
14.10.6	Weitere Informationen	419
14.11	Zeitsynchronisation mit xntp	419
14.11.1	Konfiguration im Netzwerk	420
14.11.2	Einrichten eines lokalen Zeitnormals	420
15	Der Webserver Apache	423
15.1	Grundlagen	424
15.1.1	Webserver	424
15.1.2	HTTP	424
15.1.3	URLs	424
15.1.4	Automatische Ausgabe einer Standardseite	425
15.2	HTTP-Server mit YaST einrichten	425

15.3	Apache Module	426
15.4	Neuerungen mit Apache 2	427
15.5	Threads	428
15.6	Installation	429
15.6.1	Paketauswahl in YaST	429
15.6.2	Apache aktivieren	429
15.6.3	Module für aktive Inhalte	429
15.6.4	Zusätzliche empfehlenswerte Pakete	430
15.6.5	Installation von Modulen mit apxs	430
15.7	Konfiguration	431
15.7.1	Konfiguration mit SuSEconfig	431
15.7.2	Manuelle Konfiguration	432
15.8	Apache im Einsatz	436
15.9	Aktive Inhalte	437
15.9.1	Server Side Includes: SSI	438
15.9.2	Common Gateway Interface: CGI	438
15.9.3	GET und POST	439
15.9.4	Sprachen für CGI	439
15.9.5	Aktive Inhalte mit Modulen erzeugen	440
15.9.6	mod_perl	440
15.9.7	mod_php4	442
15.9.8	mod_python	443
15.9.9	mod_ruby	443
15.10	Virtual Hosts	443
15.10.1	Namensbasierte Virtual Hosts	444
15.10.2	IP-basierte Virtual Hosts	445
15.10.3	Mehrere Instanzen von Apache	446
15.11	Sicherheit	447
15.11.1	Das Risiko gering halten	447
15.11.2	Zugriffsrechte	447
15.11.3	Immer auf dem Laufenden bleiben	448
15.12	Troubleshooting	448

15.13 Weitere Dokumentation	449
15.13.1 Apache	449
15.13.2 CGI	449
15.13.3 Sicherheit	450
15.13.4 Weitere Quellen	450
16 Datei-Synchronisation	451
16.1 Software zur Datensynchronisation	452
16.1.1 InterMezzo	452
16.1.2 unison	453
16.1.3 CVS	453
16.1.4 mailsync	454
16.2 Kriterien für die Programmauswahl	454
16.2.1 Client-Server-Modell versus Gleichberechtigung	454
16.2.2 Portabilität	454
16.2.3 Interaktiv vs. Automatisch	455
16.2.4 Geschwindigkeit	455
16.2.5 Konflikte: Auftreten und Lösung	455
16.2.6 Dateiwahl, Dateien hinzufügen	455
16.2.7 Geschichte	456
16.2.8 Datenmenge / Plattenbedarf	456
16.2.9 GUI	456
16.2.10 Anforderungen an den Benutzer	457
16.2.11 Sicherheit gegen Angriffe	457
16.2.12 Sicherheit gegen Datenverlust	457
16.3 Einführung InterMezzo	458
16.3.1 Architektur	458
16.3.2 Einrichten eines InterMezzo-Servers	459
16.3.3 Einrichten von InterMezzo-Clients	460
16.3.4 Problembehebung	460
16.4 Einführung unison	461
16.4.1 Einsatzgebiete	461

16.4.2	Voraussetzungen	461
16.4.3	Bedienung	462
16.4.4	Weiterführende Literatur	463
16.5	Einführung CVS	463
16.5.1	Einsatzgebiete	463
16.5.2	Einrichten eines CVS-Servers	464
16.5.3	Benutzung von CVS	464
16.5.4	Weiterführende Literatur	466
16.6	Einführung mailsync	467
16.6.1	Einsatzgebiet	467
16.6.2	Konfiguration und Benutzung	467
16.6.3	Mögliche Probleme	469
16.6.4	Weiterführende Literatur	470
17	Heterogene Netzwerke	471
17.1	Samba	472
17.1.1	Einführung in Samba	472
17.1.2	Installation und Konfiguration des Servers	474
17.1.3	Samba als Anmelde-Server	478
17.1.4	Installation der Clients	480
17.1.5	Optimierung	481
17.2	Netatalk	481
17.2.1	Konfiguration des Fileservers	482
17.2.2	Konfiguration des Druckservers	486
17.2.3	Starten des Servers	487
17.2.4	Weiterführende Informationen	488
17.3	NetWare-Emulation mit MARSNWE	488
17.3.1	NetWare-Emulator MARSNWE starten	488
17.3.2	Die Konfigurationsdatei /etc/nwserv.conf	489
17.3.3	Zugriff auf NetWare-Server und deren Administration	492
17.3.4	IPX-Router mit ipxrip	493

18 Internet	495
18.1 Der smpppd als Einwahlhelfer	496
18.1.1 Programmkomponenten zur Einwahl ins Internet . . .	496
18.1.2 Die Konfiguration des smpppd	496
18.1.3 kinternet und cinternet im Remote-Einsatz	497
18.2 Konfiguration eines ADSL / T-DSL Anschlusses	498
18.2.1 Standardkonfiguration	498
18.2.2 DSL Verbindung per Dial-on-Demand	499
18.3 Proxy-Server: Squid	500
18.3.1 Was ist ein Proxy-Cache?	500
18.3.2 Informationen zu Proxy-Cache	501
18.3.3 Systemanforderungen	503
18.3.4 Squid starten	504
18.3.5 Die Konfigurationsdatei /etc/squid/squid.conf	506
18.3.6 Transparente Proxy-Konfiguration	512
18.3.7 Squid und andere Programme	515
18.3.8 Weitere Informationen zu Squid	519
19 Sicherheit im Netzwerk	521
19.1 Masquerading und Firewall	522
19.1.1 Grundlagen des Masquerading	522
19.1.2 Grundlagen Firewalling	524
19.1.3 SuSEfirewall2	525
19.2 SSH – secure shell, die sichere Alternative	529
19.2.1 Das OpenSSH-Paket	530
19.2.2 Das ssh-Programm	530
19.2.3 scp – sicheres Kopieren	530
19.2.4 sftp - sicherere Dateiübertragung	531
19.2.5 Der SSH Daemon (sshd) – die Serverseite	531
19.2.6 SSH-Authentifizierungsmechanismen	533
19.2.7 X-, Authentifizierungs- und sonstige Weiterleitung . .	534
19.3 Netzwerkauthentifizierung — Kerberos	535

19.3.1	Kerberos-Terminologie	536
19.3.2	Wie funktioniert es?	538
19.3.3	Auswirkungen von Kerberos für den Benutzer	541
19.3.4	Weitere Informationen über Kerberos	542
19.4	Installation und Administration von Kerberos	543
19.4.1	Festlegung der Kerberos-Realms	543
19.4.2	Einrichtung der KDC-Hardware	544
19.4.3	Zeitsynchronisation	545
19.4.4	Konfiguration der Protokollfunktion	546
19.4.5	Installation des KDC	546
19.4.6	Konfiguration von Kerberos-Clients	549
19.4.7	Einrichtung der Fernadministration	552
19.4.8	Erstellung von Kerberos Host Principals	554
19.4.9	Aktivierung der PAM-Unterstützung für Kerberos	556
19.4.10	Konfiguration von SSH für Kerberos-Authentifizierung	556
19.4.11	Benutzung von LDAP und Kerberos	557
19.5	Sicherheit ist Vertrauenssache	561
19.5.1	Grundlagen	561
19.5.2	Lokale Sicherheit und Netzwerksicherheit	561
19.5.3	Tipps und Tricks: Allgemeine Hinweise	570
19.5.4	Zentrale Meldung von neuen Sicherheitsproblemen	573
V	Anhang	575
A	Dateisysteme unter Linux	577
B	Access Control Lists unter Linux	589
C	Manual-Page von e2fsck	603
D	Manual-Page von reiserfsck	609
E	Deutsche Übersetzung der GNU General Public License	615
	Literaturverzeichnis	627

Willkommen

Das SUSE LINUX Administrationshandbuch vermittelt Ihnen Hintergrundinformationen zur Funktionsweise Ihres SUSE LINUX Systems. Beginnend bei Grundlagen zu Dateisystemen, Kernelkonfiguration und Bootprozessen bis hin zum Aufsetzen eines Apache-Webservers und der Einrichtung sicherer Authentifizierung führt Sie dieses Buch an die Linux-Systemadministration heran.

Das SUSE LINUX Administrationshandbuch gliedert sich in fünf übergeordnete Komplexe:

Installation Details zu speziellen Installationsvarianten, zum Update, zu LVM und zu RAID...

Konfiguration Konfiguration von Bootloader und X Window System, Druckerbetrieb und mobiles Arbeiten unter Linux...

System Spezielle Merkmale eines SUSE LINUX Systems, Details zu Kernel, Bootkonzept und Init-Prozess...

Netzwerk Einbindung ins (heterogene) Netzwerk, Aufsetzen eines Apache-Webservers, Dateisynchronisation und Sicherheitsaspekte...

Anhänge Dateisysteme und Access Control Lists

Die digitalen Versionen der SUSE LINUX Handbücher finden Sie im Verzeichnis `/usr/share/doc/manuals/`.

Neuerungen im Administrationshandbuch

Folgende Änderungen zur Vorgängerversion der Dokumentation (SUSE LINUX 9.0) haben sich ergeben:

- Das Kapitel *Mobiles Arbeiten unter Linux* wurde um einen ausführlichen Abschnitt zur Verwendung von Bluetooth ergänzt (vgl. Abschnitt 8.4 auf Seite 247).
- Das Kapitel *Powermanagement* wurde um Informationen zum `power-save` Paket ergänzt (vgl. Abschnitte 9.5 auf Seite 268 und 9.6 auf Seite 275).
- Das Kapitel *Der Webserver Apache* wurde auf die Verwendung von Apache 2 hin optimiert (vgl. Kapitel 15 auf Seite 423).
- Das Kapitel *Samba* wurde auf die Verwendung von Samba 3 hin optimiert (vgl. Kapitel 17.1 auf Seite 472).
- Das Drucker-Kapitel wurde auf den Einsatz von CUPS hin optimiert (vgl. Kapitel 5 auf Seite 99).
- Das Kapitel *Das X Window System* wurde um einen ausführlichen Abschnitt zum Umgang mit Fonts unter SUSE LINUX ergänzt (vgl. Abschnitt 4.2 auf Seite 88).

Typografische Konventionen

In diesem Buch werden die folgenden typografischen Konventionen verwendet:

- YaST
Die Angabe eines Programmnamens.
- `/etc/passwd`: Die Angabe einer Datei oder eines Verzeichnisses.
- *⟨Platzhalter⟩*: Die Zeichenfolge *⟨Platzhalter⟩* ist durch den tatsächlichen Wert zu ersetzen.
- `PATH`: Eine Umgebungsvariable mit dem Namen `PATH`

- `ls`: Befehle.
- `--help`: Optionen und Parameter
- `user`: Benutzer.
- `(Alt)`: Eine zu drückende Taste.
- 'Datei': Menü-Punkte, Buttons
- "Prozess getötet": Systemmeldungen

Dank

Die Entwickler von Linux treiben in weltweiter Zusammenarbeit mit hohem freiwilligem Einsatz das Werden von Linux voran. Wir danken ihnen für ihr Engagement – ohne sie gäbe es diese Distribution nicht. Bedanken wollen wir uns außerdem auch bei Frank Zappa und Pawar.

Nicht zuletzt geht unser besonderer Dank selbstverständlich an LINUS TORVALDS!

Have a lot of fun!

Ihr SUSE Team

Teil I

Installation

Die Installation

SUSE LINUX lässt sich sehr flexibel installieren. Die Varianten reichen von einer grafischen Schnellinstallation bis zur textbasierten Variante, die zahlreiche manuelle Anpassungen zulässt.

Im Folgenden finden Sie die besonderen Installationsvarianten und Hinweise zur Verwendung unterschiedlicher Installationsquellen (CD-ROM, NFS). In diesem Kapitel finden Sie auch Tipps zu Problemen bei der Installation sowie Anleitungen zu deren Behebung. Den Abschluss bildet ein Abschnitt zur detaillierten Partitionierung.

1.1	Textbasierte Installation mit YaST	8
1.2	SUSE LINUX starten	15
1.3	Besondere Installationen	17
1.4	Tipps und Tricks	19
1.5	ATAPI-CD-ROM bleibt beim Lesen hängen	23
1.6	SCSI-Geräten und dauerhafte Gerätedateinamen	25
1.7	Partitionieren für Fortgeschrittene	25
1.8	LVM-Konfiguration mit YaST	30
1.9	Soft-RAID	38

Hinweis

Hier im Administrationshandbuch finden Sie nur besondere Installationsvarianten. Die ausführliche Beschreibung der grafischen Standardinstallation finden Sie zu Beginn des Benutzerhandbuchs.

Hinweis

1.1 Textbasierte Installation mit YaST

1.1.1 Hintergrundinfo

Zusätzlich zur Installation mit grafischer Benutzerführung kann das System mithilfe der Textmenüs von YaST installiert werden (Konsolenmodus). Alle YaST-Module stehen auch in diesem Textmodus zur Verfügung. Der Textmodus kann insbesondere dann eingesetzt werden, wenn man keine grafische Oberfläche benötigt (Serversysteme) oder wenn die Grafikkarte von dem X Window System nicht unterstützt wird. Selbstverständlich werden Sehbehinderte, die auf eine textuelle Schnittstelle angewiesen sind, auch diesen Textmodus verwenden.

1.1.2 Der Startbildschirm

Zunächst müssen Sie die Bootreihenfolge im BIOS des Rechners so einstellen, dass vom CDROM-Laufwerk gebootet wird. Legen Sie die DVD oder CD 1 in das Laufwerk und starten Sie den Rechner neu. Nach wenigen Augenblicken wird der Startbildschirm angezeigt.

Wählen Sie mit den Tasten **↑** und **↓** innerhalb von 10 Sekunden 'Manual Installation', damit *nicht* automatisch YaST gestartet wird. Geben Sie in der Zeile `boot options` Bootparameter ein, falls Ihre Hardware derartige Parameter verlangt. In der Regel sind jedoch besondere Parameter nicht erforderlich. Mit dem Parameter `textmode=1` erzwingen Sie, dass der Textmodus von YaST benutzt wird. Bei der Eingabe von Text ist zu beachten, dass in dieser frühen Bootphase noch die US-Tastaturbelegung aktiv ist.

Mit der Taste **F2** ('Video mode') legen Sie die Bildschirmauflösung für die Installation fest. Wählen Sie dort 'Text Mode', um in den reinen Textmodus zu gelangen, wenn die Graphikkarte während der Installation sonst Probleme bereitet. Drücken Sie abschließend **Return**. Nun erscheint eine Box

mit der Fortschrittsanzeige "Loading Linux kernel"; dann bootet der Kernel und linuxrc wird gestartet. Das Programm linuxrc ist menügeführt und wartet auf Eingaben des Benutzers.

Mögliche Probleme

- Diverse Boot-Schwierigkeiten können in der Regel mit Kernel-Parametern umgangen werden. Für die Fälle, bei denen DMA Schwierigkeiten bereitet, wird die Startoption 'Installation - Safe Settings' angeboten.
- Sollte Ihr CD-ROM-Laufwerk (ATAPI) beim Booten des Systems hängenbleiben, lesen Sie bitte den Abschnitt 1.5 auf Seite 23.
- Bei Schwierigkeiten mit ACPI *Advanced Configuration and Power Interface* stehen die folgenden Kernelparameter zur Verfügung:

acpi=off Dieser Parameter schaltet das komplette ACPI-System ab. Dies ist zum Beispiel sinnvoll, wenn Ihr Computer über gar keine ACPI-Unterstützung verfügt oder Sie den konkreten Verdacht haben, dass die ACPI-Implementierung Probleme bereitet.

acpi=oldboot Schaltet das ACPI-System fast komplett aus, und nur die Teile, die für das Booten nötig sind, werden verwendet.

acpi=force Schaltet ACPI ein, auch wenn Ihr Rechner ein BIOS von vor 2000 hat. Dieser Parameter überschreibt **acpi=off**.

pci=noacpi Dieser Parameter schaltet das PCI IRQ-Routing vom neuen ACPI-System aus.

Vgl. auch den SDB-Artikel: http://portal.suse.de/sdb/de/2002/09/81_acpi.html

- Wählen Sie 'Memory Test' im Bootmenü, um den Speicher zu überprüfen, wenn es beim Laden des Kernels oder im Verlauf der Installation zu „unerklärlichen“ Schwierigkeiten kommt. Linux stellt hohe Anforderungen an die Hardware. Der Speicher und dessen Timing müssen einwandfrei eingestellt sein! Mehr Info unter http://portal.suse.de/sdb/de/2000/11/thallma_memtest86.html

Lassen Sie den Speichertest am besten über Nacht laufen.

1.1.3 Die Grundlage: linuxrc

Mit dem Programm linuxrc können Sie Einstellungen zur Installation vornehmen sowie notwendige Treiber als Kernelmodule laden. Am Ende wird linuxrc YaST starten, und die eigentliche Installation der Systemsoftware und der Programme kann beginnen.

Mit \uparrow und \downarrow wählen Sie einen Menüpunkt und mit \leftarrow und \rightarrow wählen Sie ein Kommando aus, etwa 'Ok' oder 'Abbruch'. Mit (Return) wird das Kommando ausgeführt.

Eine genaue Beschreibung von linuxrc finden Sie in Abschnitt 12.4 auf Seite 310.

Einstellungen

Das Programm linuxrc beginnt automatisch mit der Sprach- und Tastaturauswahl.

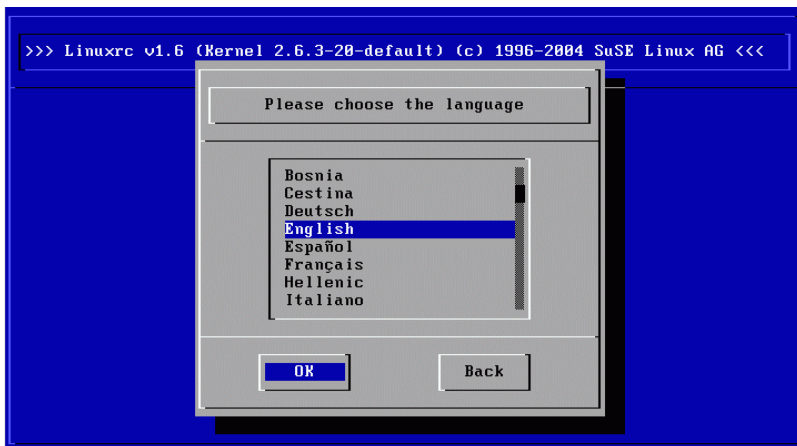


Abbildung 1.1: Auswahl der Sprache

- Wählen Sie die Sprache für die Installation aus (zum Beispiel 'Deutsch') und bestätigen Sie mit (Return) .
- Wählen Sie dann die Tastaturbelegung (zum Beispiel 'Deutsch').

Mögliche Probleme

- linuxrc bietet die gewünschte Tastaturbelegung nicht an. In einem solchen Fall wählen Sie zunächst eine alternative Belegung (Notnagel: 'English (US)'); nach der Installation kann später auf die genaue Belegung mit YaST umgeschaltet werden.

Hauptmenü von linuxrc

Jetzt sind wir im Hauptmenü von linuxrc (Abbildung 1.2).

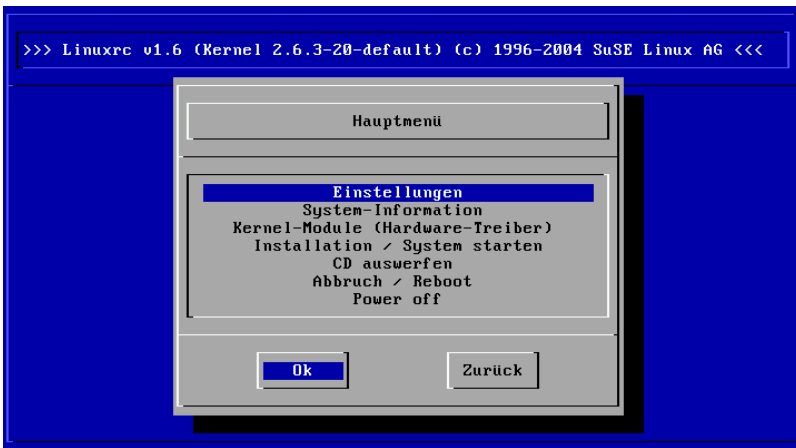


Abbildung 1.2: Hauptmenü von linuxrc

Hier gibt es folgende Optionen:

‘Einstellungen’ Hier können Sie Sprache, Bildschirm oder Tastatur anpassen.

‘System-Information’ Hier gibt es Informationen über die Hardware, soweit diese vom Kernel erkannt wurde oder von bereits geladenen Modulen angesprochen wird.

‘Kernel-Module (Hardware-Treiber)’

Hier müssen Sie eventuell die zur Hardware passenden Module laden. Zudem finden Sie hier zusätzlich zu ladende Dateisysteme wie zum Beispiel ReiserFS. Im Regelfall müssen Sie diesen Menüpunkt

nicht aufrufen, wenn Sie sowohl Festplatte(n) als auch das CD-ROM-Laufwerk (ATAPI) an einem (E)IDE-Controller angeschlossen haben, da die (E)IDE-Unterstützung bereits fest in den Kernel eingebaut ist. Details zur Modulauswahl finden Sie im nächsten Abschnitt.

‘Installation/System starten’ Hier wird zur eigentlichen Installation übergegangen.

‘Abbruch/Reboot’ Falls Sie sich nochmal alles anders überlegen...

‘Power off’ Um das System anzuhalten und auszuschalten.

Einbindung der Hardware über Module

Wählen Sie das Laden der Kernelmodule über den Menüpunkt ‘Kernel-Module’ dann, wenn Sie Unterstützung für spezielle Systemmerkmale benötigen. Traditionell ist dies für SCSI, Netzwerkkarten oder PCMCIA der Fall oder wenn das CD-ROM-Laufwerk, von dem installiert werden soll, *kein* ATAPI-Laufwerk ist; mittlerweile sind aber auch andere Komponenten als Modul ausgelagert (zum Beispiel IDE) bzw. neu hinzugekommen (zum Beispiel USB, FireWire oder Dateisysteme).

Wie Sie Module laden, lesen Sie in Abschnitt 12.4 auf Seite 310. Im folgenden Untermenü wählen Sie aus, wofür Sie Module laden wollen bzw. müssen. Es kommen in Frage:

Ein SCSI-Modul wenn Sie eine SCSI-Festplatte oder SCSI-CD-ROM-Laufwerk haben.

Ein CD-ROM-Modul falls Ihr CD-ROM-Laufwerk *nicht* am (E)IDE-Controller oder *nicht* am SCSI-Controller hängt. Dies trifft vor allem für ältere CD-ROM-Laufwerke zu, die über einen proprietären Controller am Rechner angeschlossen sind.

Ein Netzwerk-Modul falls Sie über NFS oder FTP installieren wollen; vgl. Abschnitt *Installation via Netzwerk*.

Ein oder mehrere Dateisysteme zum Beispiel ReiserFS oder ext3.

Hinweis

Wenn Sie Support für Ihr Installationsmedium (proprietäres CD-ROM-Laufwerk, Parallelport-CD-ROM-Laufwerk, Netzwerkkarte, PCMCIA) unter den Standard-Modulen vermissen, können Sie eventuell auf die zusätzlichen Treiber einer Modul-Diskette zurückgreifen; zum Erstellen einer solchen Diskette vgl. 1.4 auf Seite 19. Gehen Sie bis ans Ende der Liste und wählen Sie dort den Punkt 'Weitere Module'; die Modul-Diskette wird von linuxrc in diesem Fall angefordert.

Hinweis

Installation starten

Da im Regelfall 'Installation/System starten' bereits ausgewählt ist, brauchen Sie nur noch **(Return)** zu drücken, um zur eigentlichen Installation zu gelangen.

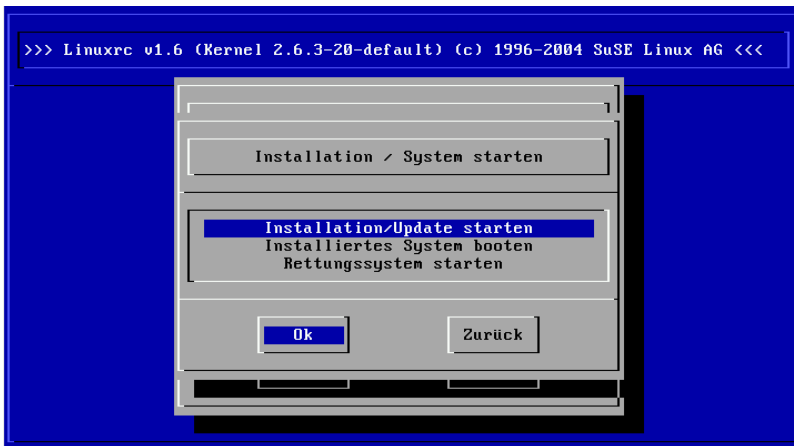


Abbildung 1.3: Installationsmenü von linuxrc

Hier stehen folgende Punkte zur Auswahl:

'Installation/Update starten' Das, was Sie vermutlich gleich machen werden.

'Installiertes System booten' Dieser Punkt wird eventuell später einmal benötigt, falls es zu Problemen mit dem Bootloader kommen sollte.

‘Rettungssystem starten’ Hier starten Sie ein Rettungssystem, das Ihnen bei größeren Problemen mit dem installierten System helfen kann.

‘CD auswerfen’ CD aus dem Laufwerk auswerfen.

Zur Installation drücken Sie nun **(Return)** für den Menüpunkt ‘Installation/Update starten’. Dann muss das Quellmedium ausgewählt werden; in der Regel reicht es aus, den Cursor an der Vorauswahl stehen zu lassen: ‘CD-ROM’.



Abbildung 1.4: Quellmedium in linuxrc auswählen

Drücken Sie nun **(Return)**. Es wird die Installationsumgebung direkt von der CD 1 bzw. DVD gestartet. Sobald dieser Vorgang abgeschlossen ist, startet YaST mit der Textoberfläche (ncurses). Die Installation beginnt.

Mögliche Probleme

- Der verwendete SCSI-Adapter wird nicht erkannt:
 - ▷ Versuchen Sie, das Modul eines kompatiblen Treibers zu laden.
 - ▷ Verwenden Sie einen Kernel, der den entsprechenden SCSI-Treiber fest hinzugebunden hat. Ein derartiger Kernel muss selbst erstellt werden.
- Das verwendete ATAPI-CD-ROM-Laufwerk bleibt beim Lesen hängen: siehe Abschnitt 1.5 auf Seite 23.

- Unter Umständen kann es zu Problemen beim Laden der Daten in die RAM-Disk kommen, sodass YaST nicht geladen werden kann. Meistens führt der folgende Weg zu einem brauchbaren Ergebnis: Wählen Sie im linuxrc-Hauptmenü 'Einstellungen' -> 'Debug (Experte)'; dort stellen Sie 'Erzwingen Rootimage' (*Force root image*) auf 'nein'. Gehen Sie zurück ins Hauptmenü und beginnen Sie die Installation erneut.

1.2 SUSE LINUX starten

Nach der Installation bleibt die Frage zu klären, wie Sie Linux im täglichen Betrieb starten wollen. In der folgenden Übersicht werden die verschiedenen Alternativen für einen Linux-Start vorgestellt. Welche dieser Startmethoden für Sie die beste ist, hängt vor allem vom Verwendungszweck ab.

Bootdiskette Sie starten Linux über die *Bootdiskette* (Startdiskette). Diese Möglichkeit funktioniert immer und die Bootdiskette kann mit YaST erzeugt werden; vgl. [1] Kapitel *YaST – Konfiguration*, Abschnitt *Erstellen einer Boot-, Rettungs- oder Moduldiskette*.

Die Bootdiskette ist auch eine gute Zwischenlösung, falls Sie beim Einrichten der anderen Möglichkeiten nicht sofort zurechtkommen oder falls Sie die Entscheidung über den endgültigen Bootmechanismus verschieben wollen. Auch wenn Sie den Bootloader eines anderen Betriebssystems nicht überschreiben wollen, ist die Bootdiskette eine brauchbare Lösung.

Linux Bootloader Die technisch sauberste und universellste Lösung ist die Verwendung eines Linux Bootmanagers wie GRUB (GRand Unified Bootloader) oder LILO (LIinux LOader), die vor dem Booten die Auswahl zwischen verschiedenen Betriebssystemen zulassen. Der Bootloader kann entweder bereits während der Installation eingerichtet oder später zum Beispiel über YaST konfiguriert werden.

Achtung

Es gibt BIOS-Varianten, die die Struktur des Bootsektors (MBR) überprüfen und nach einer GRUB- oder LILO-Installation fälschlich eine Virus-Warnung ausgeben. Diese Schwierigkeit lässt sich leicht beheben, indem Sie im BIOS die 'Virus Protection' ausschalten, falls diese Option vorhanden ist. Später können Sie sie wieder einschalten. Dieses Feature ist allerdings überflüssig, falls Sie ausschließlich Linux als Betriebssystem verwenden.

Achtung

Eine eingehende Diskussion verschiedener Bootmethoden, insbesondere aber von GRUB und LILO finden Sie in Kapitel 7 auf Seite 203 ff.

1.2.1 Der grafische SUSE-Bildschirm

Auf Konsole 1 der grafische SUSE-Bildschirm dargestellt, wenn als Kernel-Parameter die Option „vga=<wert>“ aktiv ist. Bei der Installation mit YaST wird diese Option automatisch in Abhängigkeit von der gewählten Auflösung und verwendeten Grafikkarte eingetragen.

SUSE-Bildschirm deaktivieren

Prinzipiell haben Sie drei Möglichkeiten:

- Den SUSE-Bildschirm bei Bedarf deaktivieren. Tippen Sie auf der Kommandozeile ein: `echo 0 >/proc/splash`.
So lässt sich der grafische Bildschirm ausschalten. Durch folgenden Befehl lässt er sich wieder einschalten: `echo 0x0f01 >/proc/splash`.
- Den SUSE-Bildschirm standardmäßig deaktivieren:
Fügen Sie der Bootloader-Konfiguration einen Kernelparameter `splash=0` hinzu. Im Kapitel 7 auf Seite 203 finden Sie nähere Informationen dazu. Falls Sie ohnehin lieber den Textmodus wünschen, der bei früheren Versionen Standard war, setzen Sie alternativ `vga=normal`.
- Den SUSE-Bildschirm für immer deaktivieren:
Kompilieren Sie einen neuen Kernel und deaktivieren Sie die Option ‘Use splash screen instead of boot logo’ im Menu ‘frame-buffer support’.

Hinweis

Wenn Sie im Kernel den Framebuffer-Support deaktiviert haben, ist der Splash-Screen automatisch auch deaktiviert. Wenn Sie einen eigenen Kernel kompilieren, kann SUSE dafür keinen Support garantieren!

Hinweis

1.3 Besondere Installationen

1.3.1 Installation ohne CD-ROM-Unterstützung

Was tun, wenn eine Standard-Installation via CD-ROM-Laufwerk nicht möglich ist? Ihr CD-ROM-Laufwerk könnte zum Beispiel nicht unterstützt werden, weil es sich um ein älteres proprietäres Laufwerk handelt. Oder Sie haben bei Ihrem Zweitrechner (zum Beispiel ein Notebook) eventuell gar kein CD-ROM-Laufwerk, dafür aber einen Ethernet-Adapter.

SUSE LINUX bietet die Möglichkeit, auf einem solchen Rechner ohne CD-ROM-Unterstützung über eine Netz-Verbindung zu installieren: Zumeist kommen in solchen Fällen NFS oder FTP via Ethernet zum Einsatz.

1.3.2 Installation via Netzwerk

Für diesen Weg kann kein Installationssupport in Anspruch genommen werden. Nur erfahrene Computer-Benutzer sollten ihn beschreiten. Um SUSE LINUX über eine Quelle im Netzwerk zu installieren, sind zwei Schritte notwendig:

1. Die zur Installation notwendigen Daten (CDs, DVD) auf einem Rechner verfügbar machen, der später als Installationsquelle agiert.
2. Booten des zu installierenden Systems über Diskette oder CD und Konfiguration des Netzwerkes.

Anlegen einer Netzwerk-Installationsquelle

Legen Sie die Netzwerkfreigabe an, indem Sie die Installations-CDs in einzelne Verzeichnisse kopieren und diese dann auf einem System mit NFS-Server-Funktionalität bereitstellen. Zum Beispiel können Sie auf einem existierenden SUSE LINUX-Rechner jede CD mit folgendem Befehl kopieren: `cp -a /mnt/cdrom /suse-share/`. Benennen Sie anschließend das Verzeichnis um (zum Beispiel nach CD1): `mv /suse-share/cdrom /suse-share/CD1`.

Wiederholen Sie diesen Vorgang für die restlichen CDs. Abschließend geben Sie das `/suse-share` Verzeichnis über NFS frei; vgl. Abschnitt 14.9 auf Seite 408.

Booten zur Installation über Netzwerk

Legen Sie nun das Bootmedium in das Laufwerk ein. Wie eine Bootdiskette erstellt werden kann, ist in den Abschnitten 1.4.1 auf der nächsten Seite und 1.4.2 auf Seite 21 beschrieben. Nach kurzer Zeit erscheint das Bootmenü. Wählen Sie hier 'Manual Installation'. Zusätzliche Kernelparameter können Sie ebenfalls eingeben. Bestätigen Sie die Auswahl mit Enter. Der Kernel wird geladen und Sie werden aufgefordert, die erste Moduldiskette einzulegen.

Kurze Zeit später erscheint `linuxrc` und fordert Sie auf, einige Parameter einzugeben:

1. Wählen Sie die Sprache und ggf. die Tastaturbelegung in `linuxrc`.
2. Wählen Sie 'Kernel-Module (Hardware-Treiber)'.
3. Laden Sie für Ihr System eventuell notwendige IDE-, RAID- oder SCSI-Treiber.
4. Wählen Sie 'Netzwerktreiber laden' und laden Sie den für Sie notwendigen Netzwerktreiber (zum Beispiel `eepr0100`).
5. Wählen Sie 'Dateisystemtreiber laden' und laden Sie die notwendigen Treiber (zum Beispiel `reiserfs`).
6. Wählen Sie 'Zurück' und anschließend 'Installation / System starten'.
7. Wählen Sie 'Installation / Update starten'.
8. Wählen Sie 'Netzwerk' und dann als Netzwerkprotokoll zum Beispiel NFS.
9. Wählen Sie die Netzwerkkarte, die Sie nutzen wollen.
10. Geben Sie die IP-Adressen und die weiteren Netzwerkinformationen ein.
11. Geben Sie die IP-Adresse des NFS-Servers an, der die Installationsdaten bereitstellt.
12. Geben Sie den Pfad zur NFS-Freigabe an (zum Beispiel `/suse-share/CD1`).

`linuxrc` wird nun die Installationsumgebung von der Netzwerkquelle laden und abschließend YaST starten. Beenden Sie die Installation wie in [1], Kapitel *Installation* beschrieben.

Mögliche Probleme

- Die Installation bricht ab, bevor sie überhaupt erst richtig begonnen hat: Das Installationsverzeichnis des anderen Rechners wurde nicht mit `exec`-Rechten exportiert. Holen Sie das bitte nach.
- Der Server erkennt den Rechner nicht, auf dem SUSE LINUX installiert werden soll. Tragen Sie den Namen und die IP-Adresse des neu zu installierenden Rechners in der Datei `/etc/hosts` des Servers ein.

1.4 Tipps und Tricks

1.4.1 Bootdiskette unter DOS erstellen

Sie brauchen formatierte 3,5 Zoll-HD-Disketten und ein 3,5 Zoll-Disketten-Laufwerk, das bootfähig sein muss.

Auf der CD 1 im Verzeichnis `boot` sind einige Disketten-abbilder (Images) enthalten. Solch ein Image kann mit geeigneten Hilfsprogrammen auf eine Diskette kopiert werden; die Diskette ist dann eine Bootdiskette.

Die Disketten-Images beinhalten außerdem noch den Loader `Syslinux` und das Programm `linuxrc`. `Syslinux` erlaubt es, während des Bootvorganges den gewünschten Kernel auszuwählen und bei Bedarf Parameter über die verwendete Hardware zu übergeben. Das Programm `linuxrc` unterstützt Sie beim Laden der Kernelmodule für Ihre spezielle Hardware und startet schließlich die Installation.

Bootdiskette mit `rawwritewin` erzeugen

Unter Windows steht Ihnen das grafische Programm `rawwritewin` zur Verfügung. Unter Windows finden Sie dieses Programm auf CD1 im Verzeichnis `dosutils/rawwritewin/`.

Nach dem Start müssen Sie das Image File angeben. Die Image files liegen ebenfalls auf der CD1 im Verzeichnis `boot`. Minimal benötigen Sie die Images `bootdisk` und `modules1`. Um diese im Dateibrowser anzuzeigen müssen Sie den Dateityp auf „all files“ ändern.

Legen Sie danach eine Diskette in Ihr Diskettenlaufwerk und klicken Sie auf „write“.

Um mehrere Disketten zu beschreiben wiederholen Sie einfach diese Prozedur.

Bootdiskette mit rawrite erzeugen

Es kommt das DOS-Programm `rawrite.exe` (CD 1, Verzeichnis `dosutils\rawrite` zum Erstellen der SUSE Boot- und Modul-Disketten zum Einsatz. Sie benötigen dazu einen Rechner mit einem DOS (zum Beispiel FreeDOS) oder Windows.

Im Folgenden werden die Schritte beschrieben, falls Sie mit Windows arbeiten:

1. Legen Sie die CD 1 von SUSE LINUX ein.
2. Öffnen Sie ein DOS-Fenster (im Startmenü unter 'Zubehör' -> 'MS-DOS-Eingabeaufforderung').
3. Starten Sie das Programm `rawrite.exe` mit der richtigen Pfadangabe für das CD-Laufwerk. Im Beispiel befinden Sie sich auf der Festplatte C: im Verzeichnis `Windows` und Ihr CD-Laufwerk hat den Buchstaben D:

```
C:\Windows: d:\dosutils\rawrite\rawrite
```

4. Nach dem Start fragt das Programm nach Quelle *source* und Ziel *destination* der zu kopierenden Datei. Das ist hier die zum CD-Satz gehörige Bootdiskette, deren Image sich auf CD 1 unter `boot/` befindet. Der Dateiname heißt einfach `bootdisk`. Vergessen Sie auch hier nicht die Pfadangabe für Ihr CD-Laufwerk.

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

```
Enter source file name: d:\boot\bootdisk
Enter destination drive: a:
```

Sobald Sie das Ziellaufwerk `a:` eingegeben haben, fordert Sie `rawrite` auf, eine formatierte Diskette einzulegen und auf **(Enter)** zu drücken. Im weiteren Verlauf wird dann der Fortschritt der Kopieraktion angezeigt. Abbruch ist mit der Tastenkombination **(Strg)-C** möglich.

Auf diese Art und Weise können Sie auch die anderen Diskettenimages `modules1`, `modules2`, `modules3` und `modules4` erstellen. Diese werden benötigt, wenn Sie USB- oder SCSI-Geräte bzw. eine Netzwerk- oder PCMCIA-Karte haben und diese während der Installation bereits ansprechen wollen. Eine Moduldiskette kann auch benötigt werden, wenn Sie ein spezielles Dateisystem bereits während der Installation verwenden wollen.

1.4.2 Bootdiskette unter einem unix-artigen System erstellen

Voraussetzung

Sie können auf ein unix-artiges oder ein Linux-System mit einem funktionstüchtigen CD-ROM-Laufwerk zurückgreifen. Sie brauchen eine geprüfte Diskette (formatiert).

Gehen Sie folgendermaßen vor, um Bootdisketten zu erstellen:

1. Falls Sie die Disketten noch formatieren müssen:

```
fdformat /dev/fd0u1440
```

2. Mounten Sie die CD 1; zum Beispiel nach /media/cdrom:

```
mount -t iso9660 /dev/cdrom /media/cdrom
```

3. Wechseln Sie in das Verzeichnis boot auf der CD:

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

4. Erstellen Sie die Bootdiskette mit

```
dd if=/media/cdrom/boot/bootdisk of=/dev/fd0 bs=8k
```

In der LIESMICH- bzw. der README-Datei im boot-Verzeichnis erfahren Sie Details zu den Diskettenimages; diese Dateien können Sie mit `more` oder `less` lesen.

Auf diese Art und Weise können Sie auch die anderen Diskettenimages `modules1`, `modules2`, `modules3` und `modules4` erstellen. Diese werden benötigt, wenn Sie USB- oder SCSI-Geräte bzw. eine Netzwerk- oder PCMCIA-Karte haben und diese während der Installation bereits ansprechen wollen. Eine Moduldiskette kann auch benötigt werden, wenn Sie ein spezielles Dateisystem bereits während der Installation verwenden wollen.

Etwas komplexer wird die Angelegenheit, wenn Sie zum Beispiel einen selbstkompilierten Kernel während der Installation verwenden wollen. Schreiben Sie in diesem Fall zunächst das Standard-Image (`bootdisk`) auf die Diskette und überschreiben Sie dann den eigentlichen Kernel (`linux`) mit dem eigenen Kernel (vgl. Abschnitt 11.6 auf Seite 294):

```
dd if=/media/cdrom/boot/bootdisk of=/dev/fd0 bs=8k
mount -t msdos /dev/fd0 /mnt
cp /usr/src/linux/arch/i386/boot/vmlinuz /mnt/linux
umount /mnt
```

1.4.3 Booten von Diskette (SYSLINUX)

Die Bootdiskette kann immer dann zum Einsatz kommen, wenn besondere Anforderungen zum Zeitpunkt der Installation vorliegen (zum Beispiel CD-ROM-Laufwerk nicht verfügbar). Zum Erstellen der Bootdisk vgl. 1.4.1 auf Seite 19 bzw. 1.4.2 auf der vorherigen Seite.

Der Bootvorgang wird von dem Bootloader SYSLINUX (`syslinux`) eingeleitet. SYSLINUX ist so konfiguriert, dass in geringem Umfang eine Hardwareerkennung beim Booten durchgeführt wird. Im Wesentlichen handelt es sich um die folgenden Schritte:

- Prüfen, ob das BIOS einen Framebuffer gemäß VESA 2.0 unterstützt und den Kernel entsprechend booten.
- Monitordaten (DDC-Info) auslesen.
- Den 1. Block von der 1. Festplatte (MBR) lesen, um später bei der Bootloader-Konfiguration die Zuordnung von BIOS-ID zu Linux-Gerätenamen *Devices* festzulegen. Dabei wird versucht, den Block über die lba32-Funktionen des BIOS zu lesen, um zu sehen, ob das BIOS diese Funktionen unterstützt.

Hinweis

Wenn beim Start von SYSLINUX (Umschalt) bzw. (Shift) gedrückt ist, werden all diese Schritte übersprungen. Für die Fehlersuche: Man kann in `syslinux.cfg` die Zeile

`verbose 1`

einfügen; dann teilt der Bootloader mit, welche Aktion jeweils an der Reihe ist.

Hinweis

Mögliche Probleme

Falls der Rechner nicht von Diskette booten will, müssen Sie zuvor möglicherweise die Bootreihenfolge im BIOS des Rechners auf A, C, CDROM umstellen.

1.4.4 CD 2 zum Booten verwenden

Zusätzlich zur CD 1 ist auch die zweite CD bootfähig. Während CD 1 über ein bootfähiges ISO-Image arbeitet, wird CD 2 über ein 2.88 MB großes Diskimage gebootet. Verwenden Sie die CD 2 immer dann, wenn Sie genau wissen, dass Sie von CD booten können, das jedoch mit CD 1 nicht funktioniert (Fallback-Lösung).

1.4.5 Unterstützt Linux mein CD-ROM-Laufwerk?

Generell kann man sagen, dass die meisten CD-ROM-Laufwerke unterstützt werden.

- Bei ATAPI-Laufwerken sollte es keine Probleme geben.
- Bei SCSI-CD-ROM-Laufwerken kommt es nur darauf an, ob der SCSI-Controller unterstützt wird, an dem das CD-ROM-Laufwerk angeschlossen ist. In der Komponenten-Datenbank CDB sind die unterstützten SCSI-Controller aufgeführt. Wenn Ihr SCSI-Controller nicht unterstützt wird und am Controller auch die Festplatte hängt, haben Sie sowieso ein Problem ...
- Auch viele herstellerspezifische CD-ROM-Laufwerke funktionieren mit Linux. In dieser Gruppe kann es gleichwohl zu Problemen kommen. Falls Ihr Laufwerk nicht explizit erwähnt ist, können Sie es immer noch mit einem ähnlichen Typ des gleichen Herstellers versuchen.
- USB CD-ROM-Laufwerke werden ebenfalls unterstützt. Sollte das BIOS Ihres Rechners das Booten von USB-Geräten noch nicht unterstützen, müssen Sie die Installation über Bootdisketten starten. Näheres hierzu finden Sie unter 1.4.3 auf der vorherigen Seite. Stellen Sie vor dem Booten von Diskette sicher, dass alle notwendigen USB-Geräte bereits angeschlossen und eingeschaltet sind.

1.5 ATAPI-CD-ROM bleibt beim Lesen hängen

Wenn das ATAPI CD-ROM-Laufwerk nicht erkannt wird oder es beim Lesen hängen bleibt, ist häufig die Hardware nicht korrekt eingerichtet. Nor-

malerweise sollten die einzelnen Geräte am (E)IDE-Bus fortlaufend angeschlossen sein, das heisst das erste Gerät ist Master am ersten Controller, das zweite Slave. Das dritte Gerät schließlich ist Master am zweiten Controller und das vierte dort wieder Slave.

Oft befindet sich in einem Rechner neben der Festplatte nur das CD-ROM-Laufwerk, das als Master am zweiten Controller hängt. Linux kommt in manchen Fällen mit dieser Lücke nicht selbstständig zurecht. Meistens kann dem Kernel durch Angabe eines entsprechenden Parameters aber auf die Sprünge geholfen werden (`hdc=cdrom`).

Gelegentlich ist für ein Laufwerk nur ein falscher Jumper (Brücke) gesetzt; das heißt, es ist als Slave konfiguriert, obwohl es als Master am zweiten Controller angeschlossen ist oder umgekehrt. Im Zweifelsfall sollten diese Einstellungen überprüft und gegebenenfalls korrigiert werden.

Außerdem gibt es noch eine Reihe fehlerhafter EIDE-Chipsätze. Diese sind mittlerweile zum größten Teil bekannt; der Kernel enthält Code, um derartige Probleme zu umgehen. Für diese Fälle existiert ein spezieller Kernel (vgl. das README in `/boot` der Installations-CD-ROM).

Sollte das Booten nicht auf Anhieb funktionieren, so versuchen Sie bitte die nachfolgenden Kernelparameter:

`hdx=cdrom` x steht hier für a, b, c, d etc. und ist folgendermaßen zu lesen:

- a – Master am 1. IDE-Controller
- b – Slave am 1. IDE-Controller
- c – Master am 2. IDE-Controller

Ein Beispiel für einzugebender Parameter ist `hdb=cdrom`. Mit diesem Parameter geben Sie dem Kernel das CD-ROM-Laufwerk an, falls dieser es nicht findet und Sie ein ATAPI-CD-ROM-Laufwerk haben.

`idx=noautotune` x steht für 0, 1, 2, 3 etc. und ist folgendermaßen zu lesen:

- 0 – 1. IDE-Controller
- 1 – 2. IDE-Controller

Ein Beispiel für einzugebender Parameter ist hier `ide0=noautotune`. Dieser Parameter hilft in der Regel bei (E)IDE-Festplatten.

1.6 SCSI-Geräten und dauerhafte Gerätedateinamen

SCSI-Geräte wie z.B. Festplattenpartitionen bekommen beim Booten Gerätedateinamen mehr oder weniger dynamisch zugewiesen. Dies ist solange kein Problem, wie sich an der Zahl oder an der Konfiguration der Geräte nichts ändert. Wenn aber eine weitere SCSI-Festplatte hinzukommt und diese vor der alten Festplatte vom Kernel erkannt wird, dann erhält die alte Platte neue Namen und die Einträge in der Mounttabelle `/etc/fstab` passen nicht mehr.

Um diese Schwierigkeit zu vermeiden, sollte man `boot.scsidev` einsetzen. `boot.scsidev` nimmt die Einrichtung der SCSI-Geräte beim Booten vor und trägt dauerhafte Gerätenamen unter `/dev/scsi/` ein, die man dann in der `/etc/fstab` verwenden kann.

Im Expertenmodus des Runlevel-Editors ist `boot.scsidev` für die Stufe B einzuschalten, dann werden die notwendigen Links in `/etc/init.d/boot.d` angelegt, um die Namen während des Bootens zu erzeugen.

1.7 Partitionieren für Fortgeschrittene

Im Kapitel zur Standardinstallation (siehe [1]) wird auf Möglichkeiten der Partitionierung des Systems eingegangen. Dieser Abschnitt soll nun detaillierte Informationen bereitstellen, mit denen Sie sich ein optimales Partitionierungsschema anlegen können. Dies ist insbesondere für diejenigen interessant, die ihr System optimal konfigurieren möchten, sowohl in puncto Sicherheit, als auch was Geschwindigkeit betrifft, und die dafür bereit sind, unter Umständen das bestehende System komplett neu aufzusetzen.

Ein grundlegendes Verständnis der Funktionsweise eines UNIX-Dateisystemes wird vorausgesetzt. Die Begriffe Mountpoint sowie physikalische, erweiterte und logische Partition sollten Ihnen nicht fremd sein.

Stellen Sie als ersten Schritt folgende Informationen zusammen:

- Einsatzgebiet dieses Rechners (File-Server, Application-Server, Compute-Server, Einzelplatzrechner)?
- Wie viele Leute werden an diesem Rechner arbeiten (simultane Logins)?
- Wie viele Festplatten hat der Rechner, wie groß sind diese und welches System verwenden sie (EIDE-, SCSI- oder RAID-Controller)?

1.7.1 Die Größe der Swap-Partition

Oft werden Sie lesen: Mindestens doppelt so viel Swap wie Hauptspeicher. Diese Formulierung stammt noch aus einer Zeit, in der 8 MB RAM im Rechner nicht wenig war. Der Rechner soll also über ungefähr 30 bis 40 MB virtuellen Speicher, also Ram plus Swap verfügen. Mit modernen Applikationen müssen auch diese Werte nach oben hin korrigiert werden. Als durchschnittlicher Benutzer ist man auf absehbare Zeit mit 256 MB virtuellem Speicher auf der sicheren Seite. Auf keinen Fall sollten Sie überhaupt keinen Swap-Speicher anlegen.

1.7.2 Einsatzgebiet des Rechners

Einsatz als Einzelrechner

Der häufigste Anwendungsfall für einen Linux-Rechner ist der Einsatz als Einzelplatzrechner. Damit Sie sich an konkreten Werten orientieren können, haben wir ein paar Beispielkonfigurationen zusammengestellt, die Sie je nach Bedarf bei sich zu Hause oder in der Firma übernehmen können. In Tabelle 1.1 sehen Sie einen kleinen Überblick der verschiedenen Installationsvolumina für ein Linux-System.

Tabelle 1.1: Beispiele für Größen von Installationen

Installation	Benötigter Plattenplatz
sehr klein	180 MB bis 400 MB
klein	400 MB bis 1500 MB
mittel	1500 MB bis 4 GB
groß	mehr als 4 GB

Beispiel: Standard-Arbeitsplatzrechner (klein)

Sie haben eine ca. 500 MB große Festplatte übrig und möchten auf diese Linux installieren: eine 64 MB große Swap-Partition und den Rest für / (Root-Partition).

Beispiel: Standard-Arbeitsplatzrechner (Durchschnitt)

Sie haben 2 GB für Linux frei. Kleine Boot-Partition /boot (5-10 MB bzw. 1 Zylinder), 128 MB für Swap, 800 MB für / und den Rest für eine separate /home-Partition.

Beispiel: Standard-Arbeitsplatzrechner (Luxus)

Falls Ihnen 2 GB oder mehr auf mehreren Platten zur Verfügung stehen, gibt es keine pauschale Partitionierung. Lesen Sie hierzu bitte Abschnitt 1.7.3 auf der nächsten Seite.

Einsatz als Fileserver

Hier kommt es *wirklich* auf Festplattenperformance an. SCSI-Geräten sollte unbedingt der Vorzug gegeben werden. Achten Sie auch auf Leistungsfähigkeit der Platten und des verwendeten Controllers.

Ein Fileserver bietet die Möglichkeit, Daten zentral zu halten. Hierbei kann es sich um Benutzerverzeichnisse, eine Datenbank oder sonstige Archive handeln. Der Vorteil ist eine wesentlich einfachere Administration. Falls der Fileserver ein größeres Netz bedienen soll (ab 20 Usern), wird die Optimierung des Plattenzugriffs essentiell. Angenommen, Sie möchten einen Linux-Fileserver aufbauen, der 25 Benutzern Heimatverzeichnisse (Home) zur Verfügung stellen soll: Sie wissen, jeder Benutzer wird maximal 100-150 MB für seine persönlichen Daten in Anspruch nehmen. Falls nicht jeder dieser Benutzer stets in seinem Home kompiliert, reicht hierfür eine 4-GB-Partition, welche einfach unter `/home/` gemountet wird.

Haben Sie 50 Benutzer, so wäre rein rechnerisch eine 8-GB-Partition notwendig. Besser ist es in diesem Fall jedoch, `/home/` auf zwei 4-GB-Festplatten aufzuteilen, da diese sich dann die Last (und Zugriffszeit!) teilen.

Hinweis

Den Cache eines Webbrowsers sollten die Benutzer unbedingt auf lokalen Festplatten halten!

Hinweis

Einsatz als Compute-Server

Ein Compute-Server ist in der Regel ein leistungsstarker Rechner, der berechnungsintensive Aufgaben im Netz übernimmt. Solch eine Maschine verfügt typischerweise über einen etwas größeren Hauptspeicher (ab 512 MB RAM). Der einzige Punkt, an dem für einen schnellen Plattendurchsatz gesorgt werden muss, sind etwaige Swap-Partitionen. Auch hier gilt: mehrere Swap-Partitionen auf mehrere Platten verteilen.

1.7.3 Optimierungsmöglichkeiten

Die Platten sind zumeist der begrenzende Faktor. Um diesen Flaschenhals zu umgehen, gibt es drei Möglichkeiten, die am Besten zusammen eingesetzt werden sollten:

- Verteilen Sie die Last gleichmäßig auf mehrere Platten.
- Setzen Sie ein optimiertes Dateisystem ein (zum Beispiel `reiserfs`).
- Statten Sie den Fileserver mit genügend Speicher aus (256 MB Minimum).

Parallelisierung durch mehrere Platten

Die erstgenannte Methode bedarf einer tiefergehenden Erklärung. Die Gesamtzeit, die vergeht, bis angeforderte Daten bereitgestellt werden, setzt sich (in etwa) aus folgenden Teilen zusammen:

1. Zeit, bis die Anforderung beim Plattencontroller ist.
2. Zeit, bis der Plattencontroller diese Anforderung an die Festplatte schickt.
3. Zeit, bis die Festplatte ihren Kopf positioniert.
4. Zeit, bis sich das Medium zum richtigen Sektor gedreht hat.
5. Zeit für die Übertragung.

Punkt 1 ist abhängig von der Anbindung über das Netzwerk und muss dort geregelt werden. Punkt 2 ist eine relativ vernachlässigbare Zeit, die vom Plattencontroller selbst abhängt. Punkte 3 und 4 sind die Hauptbereiche. Gemessen wird die Positionierung in ms. Verglichen mit den in ns gemessenen Zugriffszeiten im Hauptspeicher ist das ein Faktor von 1 Million! Punkt 4 ist von der Drehzahl der Platte abhängig. Auch diese Zeit wird meist mehrere ms betragen. Punkt 5 von der Drehzahl und der Anzahl der Köpfe, ebenso wie von der aktuellen Position des Kopfes (innen oder außen).

Für die optimale Performance sollte man also bei Punkt 3 angreifen. Hier kommt bei SCSI-Geräten das Feature disconnect ins Spiel. Mit diesem Feature passiert in etwa folgendes:

Der Controller sendet an das angeschlossene Gerät (in diesem Fall die Festplatte) den Befehl Gehe zu Track *x*, Sektor *y*. Nun muss sich die träge Mechanik der Platte in Bewegung setzen. Wenn die Platte intelligent ist (also disconnect beherrscht) und der Treiber für den Controller dieses Feature auch beherrscht, schickt der Controller der Platte unmittelbar daraufhin einen disconnect-Befehl und die Platte trennt sich vom SCSI-Bus ab. Ab jetzt können andere SCSI-Geräte ihre Transfers erledigen. Nach einer Weile (je nach Strategie bzw. Last auf dem SCSI-Bus) wird wieder die Verbindung zur Platte aktiviert. Idealerweise hat diese bereits den geforderten Track erreicht.

In einem Multitasking-Multiuser Betriebssystem wie Linux kann man hier natürlich gut optimieren. Sehen wir uns einen Ausschnitt einer Ausgabe des Befehls `df` an (vgl. Ausgabe 1.1).

Beispiel 1.1: Beispielausgabe df-Befehl

```
Filesystem Size Used Avail Use% Mounted on
/dev/sda5 1.8G 1.6G 201M 89% /
/dev/sda1 23M 3.9M 17M 18% /boot
/dev/sdb1 2.9G 2.1G 677M 76% /usr
/dev/sdc1 1.9G 958M 941M 51% /usr/lib
shmfs 185M 0 184M 0% /dev/shm
```

Was bringt uns diese Parallelisierung? Angenommen wir geben in `/usr/src` als Benutzer `root` ein:

```
tar xzf package.tar.gz -C /usr/lib
```

Das soll also `package.tar.gz` nach `/usr/lib/package` installieren. Hierzu werden von der Shell `tar` und `gzip` aufgerufen (befinden sich in `/bin` und somit auf `/dev/sda`), dann wird `package.tar.gz` von `/usr/src` gelesen (befindet sich auf `/dev/sdb`). Als Letztes werden die extrahierten Daten nach `/usr/lib` geschrieben (liegt unter `/dev/sdc`). Sowohl Positionierung, als auch Lesen/Schreiben der platteninternen Puffer können nun quasi parallel ausgeführt werden.

Das ist ein Beispiel von vielen. Als Faustregel gilt, dass bei Vorhandensein entsprechend vieler (gleich schneller) Platten `/usr` und `/usr/lib` auf verschiedenen Platten lagern sollten. Hierbei sollte `/usr/lib` ca. 70 Prozent der Kapazität von `/usr` haben. Das Rootverzeichnis `/` sollte sich bei der Verlagerung auf zwei Platten wegen der Zugriffshäufigkeit auf der Platte mit `/usr/lib` befinden.

Ab einer gewissen Menge an SCSI-Platten (ca. 4 bis 5) sollte man sich jedoch ernsthaft mit einer RAID-Lösung in Software oder gleich besser mit der Anschaffung eines RAID-Controllers beschäftigen. Dadurch werden dann Operationen auf den Platten nicht nur quasiparallel, sondern echt parallel ausgeführt. Fehlertoleranz ist ein weiteres angenehmes Nebenprodukt.

Plattendurchsatz und die Größe des Hauptspeichers

Wir weisen an vielen Stellen darauf hin, dass die Größe des Hauptspeichers unter Linux oft wichtiger ist als die Geschwindigkeit des Prozessors. Ein Grund – wenn nicht sogar der Hauptgrund – ist die Eigenschaft von Linux, dynamische Puffer mit Festplattendaten anzulegen. Hierbei arbeitet Linux mit allerlei Tricks wie read ahead (holt vorsorglich Sektoren im Voraus) und delayed write (spart sich Schreibzugriffe, um sie dann auf einmal auszuführen). Letzteres ist der Grund, warum man einen Linux-Rechner nicht einfach ausschalten darf. Beide Punkte sind dafür verantwortlich, dass sich der Hauptspeicher mit der Zeit scheinbar immer füllt und dass Linux so schnell ist; vgl. auch Abschnitt 12.2.6 auf Seite 303.

1.8 LVM-Konfiguration mit YaST

Mit diesem professionellen Partitioniertool haben Sie die Möglichkeit, bestehende Partitionen zu bearbeiten, zu löschen oder neue Partitionen anzulegen. Von hier aus gelangen Sie zur Soft-RAID- und LVM-Konfiguration.

Hinweis

Hintergrundinformationen und Tipps zum Partitionieren finden Sie im Abschnitt 1.7 auf Seite 25.

Hinweis

Im Normalfall werden die Partitionen während der Installation festgelegt. Wenn Sie eine zweite Festplatte einbauen wollen, können Sie diese auch im bestehenden Linux-System integrieren. Hierzu ist die neue Festplatte zunächst zu partitionieren, dann müssen die Partitionen gemountet und in die `/etc/fstab` eingetragen werden. Gegebenenfalls ist es nötig, einige Daten umzukopieren, um eine zu kleine `/opt/-`Partition von der alten Festplatte auf die neue zu verschieben.

Wenn Sie die Festplatte, mit der Sie gerade arbeiten, umpartitionieren wollen, ist Vorsicht geboten – grundsätzlich ist dies möglich, danach muss das

System aber sofort neu gebootet werden. Unbedenklicher ist es, von der CD zu booten und dann die Umpartitionierung vorzunehmen. Hinter dem Button 'Experten...' im Partitionierer befindet sich ein Popup-Menü mit folgenden Befehlen:

Partitionstabelle neu einlesen Dient dazu, die Partitionierung neu von der Platte einzulesen. Dies benötigen Sie zum Beispiel, wenn Sie die Partitionierung auf der Textkonsole manuell vorgenommen haben.

Mountpunkte von bestehender /etc/fstab übernehmen

Dies ist nur während der Installation relevant. Das Einlesen der alten `fstab` nützt, wenn Sie Ihr System nicht updaten, sondern neu installieren. Dann brauchen Sie die Mountpunkte nicht per Hand eingeben.

Partitionstabelle und Disk-Label löschen

Hiermit überschreiben Sie den alten Partitiontable komplett. Das kann zum Beispiel hilfreich sein, falls Sie Probleme mit ungewöhnlichen Plattenlabels haben sollten. Mit dieser Methode gehen allerdings alle Daten auf der Festplatte verloren.

1.8.1 Logical Volume Manager (LVM)

Ab Kernel Version 2.6 steht Ihnen LVM in der Version 2 zur Verfügung. Dieser ist rückwärtskompatibel zum bisherigen LVM und kann alte Volume-Groups weiterverwalten. Wenn Sie neue Volume-Groups anlegen, müssen Sie entscheiden, ob Sie das neue Format oder die rückwärtskompatible Version verwenden möchten. LVM2 benötigt keine Kernel-Patches mehr, und verwendet den `device-mapper`, der in Kernel 2.6 integriert ist. Beginnend mit diesem Kernel kann LVM nur noch in der Version 2 verwendet werden. In diesem Kapitel ist mit LVM daher immer LVM in der Version 2 gemeint.

Alternativ zu LVM2 können Sie auch EVMS (Enterprise Volume Management System) verwenden. Dieses bietet eine einheitliche Schnittstelle zu Logical Volumes, aber auch zu Raid Volumes. EVMS setzt wie LVM2 auf den Device Mapper in Kernel 2.6 auf.

Der Logical Volume Manager (LVM) ermöglicht Ihnen eine flexible Verteilung des Festplattenplatzes auf die verschiedenen Filesysteme. Da die Partitionen in einem laufenden System nur mit relativ großem Aufwand geändert werden können, wurde der LVM entwickelt: Er stellt einen virtuellen Pool (Volume Group – kurz VG) an Speicherplatz zur Verfügung, aus

dem logische Volumes (LV) nach Bedarf erzeugt werden. Das Betriebssystem greift dann auf diese, statt auf die physikalischen Partitionen zu.

Besonderheiten:

- Mehrere Festplatten/Partitionen können zu einer großen logischen Partition zusammengefügt werden.
- Neigt sich bei einem LV (zum Beispiel `/usr/`) der freie Platz dem Ende zu, können Sie diese bei geeigneter Konfiguration vergrößern.
- Mit dem LVM können Sie sogar im laufenden System Festplatten oder LVs ergänzen. Voraussetzung ist allerdings hot-swap fähige Hardware, die für solche Eingriffe geeignet ist.
- Mehrere Festplatten können im RAID 0 (striping) Modus mit entsprechend verbesserter Performance verwendet werden.
- Das „snapshot“-Feature ermöglicht vor allem bei Servern konsistente Backups während dem laufenden System.

Der Einsatz von LVM lohnt bereits bei umfangreich genutzten Home-PCs oder kleinen Servern. Wenn Sie einen wachsenden Datenbestand haben wie zum Beispiel bei Datenbanken, MP3-Archiven oder Benutzerverzeichnissen etc., dann bietet sich der Logical Volume Manager an. Dann ist es zum Beispiel möglich, Filesysteme zu haben, die größer sind als eine physikalische Festplatte. Ein weiterer Vorteil des LVM ist, dass bis zu 256 LVs angelegt werden können. Beachten Sie jedoch bitte, dass sich die Arbeit mit dem LVM sehr von der mit konventionellen Partitionen unterscheidet.

Anleitung und weiterführende Informationen zur Konfiguration des „Logical Volume Manager“ (LVM) finden Sie im offiziellen LVM-Howto <http://tldp.org/HOWTO/LVM-HOWTO/>.

Konfiguration des LVM mit YaST

Die LVM-Konfiguration von YaST wird vorbereitet, indem Sie während der Installation eine LVM-Partition anlegen. Dazu müssen Sie im Vorschlagsbildschirm auf 'Partitionierung' klicken, im folgenden Fenster dann auf 'Verwerfen' oder 'Ändern'. Danach müssen Sie eine Partition für LVM anlegen. Dazu wählen Sie im Partitionierer 'Anlegen' -> 'Nicht formatieren' und dort den Punkt '0x8e Linux LVM'. Die weitere Partitionierung mit LVM können Sie direkt im Anschluss oder auch später im installierten System vornehmen, indem Sie im Partitionierer die LVM-Partition markieren und dann auf 'LVM...' klicken.

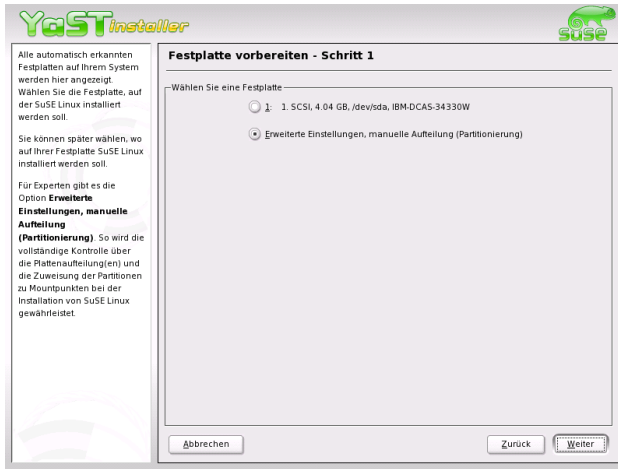


Abbildung 1.5: YaST: LVM während der Installation aktivieren

LVM – Partitionierer

Nachdem Sie unter Partitionieren 'LVM...' gewählt haben, kommen Sie in einen Dialog, in dem Sie die Partitionierung Ihrer Festplatten ändern können. Hier können Sie bestehende Partitionen löschen, existierende Partitionen ändern und neue anlegen. Eine Partition, die für LVM verwendet werden soll, muss die Partitionskennung 8E haben. Diese Partitionen sind mit dem Text „Linux LVM“ in der Partitionsliste des Fensters versehen (s. letzter Abschnitt).

Hinweis

Umpartitionieren von Logical Volumes

Am Anfang der PVs werden Informationen über das Volume in die Partition geschrieben. So „weiß“ eine PV, zu welcher Volume Group das gehört. Wenn Sie neu partitionieren möchten, ist es empfehlenswert, den Anfang dieser Volumes zu löschen. Bei einer Volume Group „system“ und einem Physical Volume „/dev/sda2“ geht das zum Beispiel mit dem Befehl `dd if=/dev/zero of=/dev/sda2 bs=512 count=1`.

Hinweis

Es ist nicht nötig, alle Partitionen, die für LVM vorgesehen sind, einzeln auf die Partitionskennung 8E zu setzen. YaST setzt die Partitionskennung einer

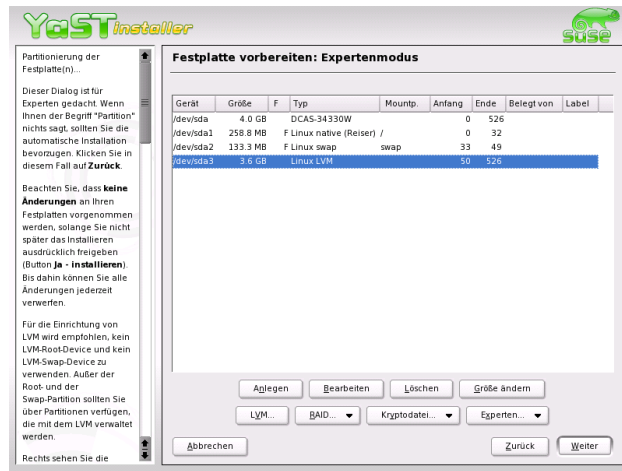


Abbildung 1.6: YaST: LVM-Partitionierer

Partition, die einer LVM Volume Group zugeordnet wird, automatisch auf 8E, wenn dies nötig ist. Wenn auf Ihren Platten unpartitionierte Bereiche vorhanden sind, sollten Sie in diesem Dialog für alle diese Bereiche LVM-Partitionen anlegen. Diese Partitionen sollten Sie sofort auf die Partitionsbezeichnung 8E setzen. Diese müssen nicht formatiert werden, und es kann für sie kein Mountpunkt eingetragen werden.

Falls auf Ihrem System bereits eine gültige LVM-Konfiguration existiert, wird diese bei Beginn der LVM-Konfiguration automatisch aktiviert. Ist diese Aktivierung erfolgt, kann die Partitionierung aller Platten, die eine Partition enthalten, die zu einer aktivierten Volume Group gehört, nicht mehr verändert werden. Der Linux-Kernel weigert sich, die veränderte Partitionierung einer Festplatte einzulesen, solange auch nur eine Partition dieser Platte benutzt wird.

Eine Umpartitionierung von Platten, die nicht zu einer LVM Volume Group gehören, ist natürlich problemlos möglich. Falls Sie bereits eine gültige LVM-Konfiguration auf Ihrem System haben, ist ein Umpartitionieren normalerweise nicht erforderlich. In dieser Maske müssen Sie nun alle mountpoints konfigurieren, die nicht auf LVM Logical Volumes liegen. Zumindest das Root-Filesystem muss in YaST auf einer normalen Partition liegen. Wählen Sie diese Partition aus der Liste aus und legen Sie sie mit dem Button 'Bearbeiten' als Root-Filesystem fest.

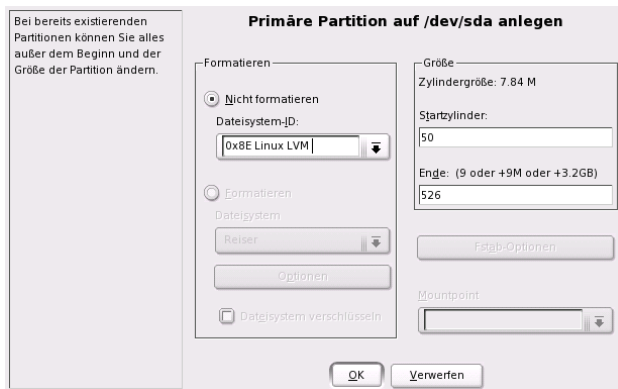


Abbildung 1.7: YaST: LVM-Partition anlegen

Wir empfehlen aufgrund der größeren Flexibilität von LVM, alle weiteren Filesysteme auf LVM Logical Volumes zu legen. Nach Festlegen der Root-Partition können sie diesen Dialog verlassen.

LVM – Einrichtung der Physical Volumes

Im Dialog 'LVM' werden die LVM Volume Groups (oft mit „VG“ abgekürzt) verwaltet. Wenn auf Ihrem System noch keine Volume Group existiert, werden Sie in einem Popup-Fenster aufgefordert, eine anzulegen. Als Name für die Volume Group auf der sich die Dateien des SUSE LINUX Systems befinden, wird `system` vorgeschlagen.

Die so genannte Physical Extent Size (oft abgekürzt mit PE-Size) bestimmt die maximale Größe eines Physical und Logical Volumes in dieser Volume Group. Dieser Wert wird normalerweise auf 4 Megabyte festgelegt. Dies lässt eine Maximalgröße für ein Physical und Logical Volume von 256 Gigabyte zu. Sie sollten die Physical Extent Size also nur dann erhöhen (zum Beispiel auf 8, 16 oder 32 Megabyte), wenn Sie größere Logical Volumes als 256 Gigabyte benötigen.

In dem folgenden Dialog sind alle Partitionen aufgelistet, die entweder den Type „Linux LVM“ oder „Linux native“ haben. Es werden also keine Swap- und DOS-Partitionen angezeigt. Wenn eine Partition bereits einer Volume Group zugeordnet ist, wird der Name der Volume Group in der Liste angezeigt, nicht zugeordnete Partitionen enthalten die Kennung „--“.

Volume-Gruppe anlegen

Nun muss eine Volume-Gruppe angelegt werden.
 Sie müssen dafür keine Änderungen vornehmen.
 Falls Sie jedoch Experte sind, ändern Sie je nach Bedarf
 die Standardwerte:

Name der Volume-Gruppe:

Größe (Physical Extent Size)

☐ Altes LVM1-kompatibles Metadatenformat verwenden

Abbildung 1.8: YaST: Volume Group anlegen

Die gegenwärtig bearbeitete Volume Group kann in der Auswahlbox links oben geändert werden. Mit den Buttons rechts oben ist es möglich, zusätzliche Volume Groups anzulegen und bestehende VGs zu löschen. Es können allerdings nur solche Volume Groups gelöscht werden, denen keine Partitionen mehr zugeordnet sind. Für ein normal installiertes SUSE LINUX System ist es nicht nötig, mehr als eine Volume Group anzulegen. Eine Partition, die einer Volume Group zugeordnet ist, wird auch Physical Volume (oft mit PV abgekürzt) genannt.

Um eine bisher nicht zugeordnete Partition der angewählten Volume Group hinzuzufügen, wählen Sie zuerst die Partition an und aktivieren dann den Button 'Volume hinzufügen' unterhalb der Auswahlliste. Daraufhin wird der Name der Volume Group bei der angewählten Partition eingetragen. Sie sollten alle Partitionen, die Sie für LVM vorgesehen haben, einer Volume Group zuordnen, sonst bleibt der Platz auf der Partition ungenutzt. Bevor Sie den Dialog verlassen können, muss jeder Volume Group mindestens eine Physical Volume zugeordnet sein.

Logical Volumes

Im diesem Dialog werden die Logical Volumes (oft einfach mit „LV“ abgekürzt) verwaltet.

Logical Volumes sind jeweils einer Volume Group zugeordnet und haben eine bestimmte Größe. Wenn Sie beim Anlegen der Logical Volumes ein Striping Array anlegen möchten, sollten Sie das LV mit den meisten Stripes als erstes anlegen. Ein Striping LV mit n Stripes kann nur dann korrekt angelegt werden, wenn sich der Plattenplatz, der vom LV benötigt wird, noch

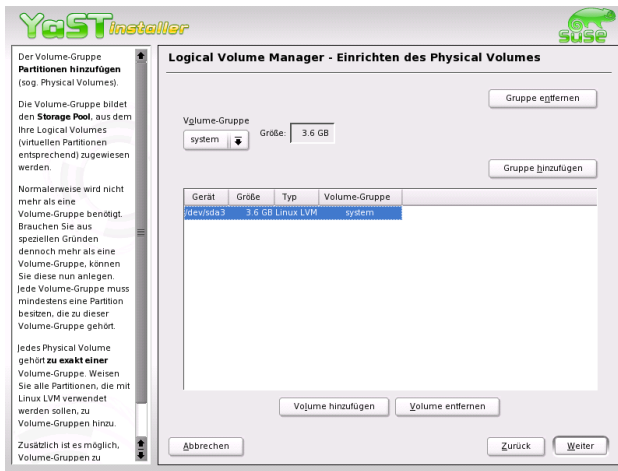


Abbildung 1.9: YaST: Übersicht über die Partitionen

gleichmäßig auf n Physical Volumes verteilen lässt. Wenn nur zwei PVs zur Verfügung stehen, ist ein LV mit drei Stripes natürlich nicht möglich.

Normalerweise wird auf einem Logical Volume ein Filesystem (zum Beispiel reiserfs, ext2) angelegt und ihm ein Mountpunkt zugeordnet. Unter diesem Mountpunkt sind dann im installierten System die Dateien zu finden, die auf diesem Logical Volume gespeichert sind. In der Liste sind alle normalen Linux-Partitionen, denen ein Mountpunkt zugeordnet ist, alle Swap-Partitionen und alle bereits existierenden Logical Volumes eingetragen.

Achtung

Der Einsatz des LVM ist ggf. mit erhöhten Risiken wie zum Beispiel Datenverlust verbunden. Mögliche Gefahren sind Programmabstürze, Stromausfälle oder fehlerhafte Kommandos.

Sichern Sie bitte Ihre Daten bevor Sie LVM einsetzen oder Volumes umkonfigurieren – arbeiten Sie also nie ohne Backup!

Achtung

Wenn Sie bereits vorher auf Ihrem System LVM konfiguriert hatten, sind die existierenden Logical Volumes hier eingetragen. Sie müssen diesen Logical Volumes allerdings noch den passenden Mountpunkt zuordnen. Wenn Sie erstmalig auf einem System LVM konfigurieren, existieren

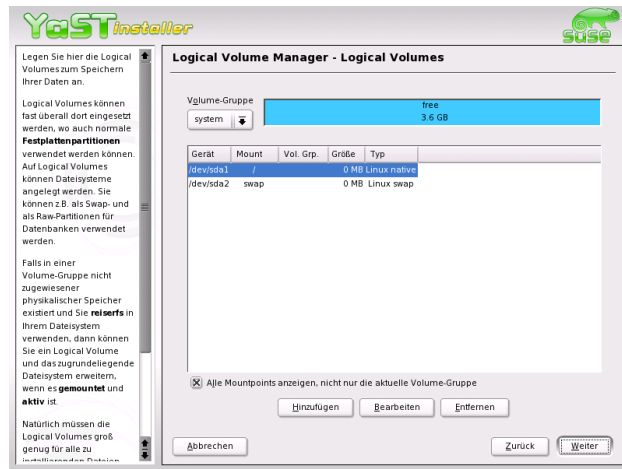


Abbildung 1.10: YaST: Verwaltung der Logical Volumes

in dieser Maske noch keine Logical Volumes und Sie müssen für jeden Mountpunkt ein Logical Volume erzeugen (mit dem Button 'Hinzufügen'), die Größe, den Filesystem-Typ (zum Beispiel reiserfs oder ext2) und den Mountpunkt (zum Beispiel `/var/`, `/usr/`, `/home/`) festlegen.

Wenn Sie mehrere Volume Groups angelegt haben, können Sie in der Auswahlliste links oben zwischen den einzelnen Volume Groups wechseln. Die angelegten Logical Volumes liegen jeweils in der links oben angezeigten Volume Group. Haben Sie alle Logical Volumes so angelegt, wie sie benötigt werden, ist die LVM-Konfiguration beendet. Sie können den Dialog verlassen und mit der Software-Auswahl fortfahren, falls Sie sich im Installations-Prozess befinden.

1.9 Soft-RAID

Der Sinn von RAID (engl. *Redundant Array of Independent Disks*) ist, mehrere Festplattenpartitionen zu einer großen *virtuellen* Festplatte zu vereinen, um die Performance und die Datensicherheit zu optimieren. Dabei geht das eine jedoch auf Kosten des anderen. Der so genannte *RAID-Level* definiert den Zusammenschluss und die gemeinsame Ansteuerung der Festplatten, die von einem RAID-Controller vorgenommen wird.

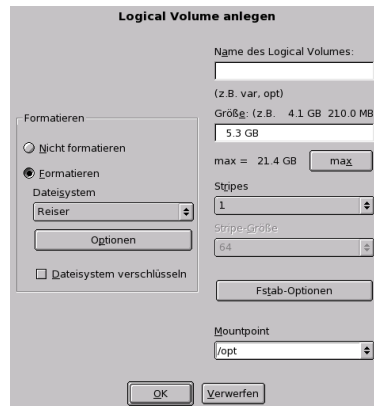


Abbildung 1.11: YaST: Logical Volumes anlegen

Ein RAID-Controller verwendet meist das SCSI-Protokoll, da es gegenüber dem IDE-Protokoll mehr Festplatten besser ansteuern kann und besser für eine parallele Abarbeitung der Befehle geeignet ist.

Statt eines RAID-Controllers, der unter Umständen sehr teuer sein kann, ist auch Soft-RAID in der Lage, diese Aufgaben zu übernehmen. SUSE LINUX bietet Ihnen die Möglichkeit, mit Hilfe von YaST mehrere Festplatten zu einem Soft-RAID-System zu vereinen – eine sehr günstige Alternative zu Hardware-RAID.

1.9.1 Gängige RAID-Level

RAID 0 Dieser Level verbessert die Performance Ihres Datenzugriffs. Im Grunde ist dies gar kein RAID, da es keine Datensicherung gibt, doch die Bezeichnung *RAID 0* hat sich für diese Art von System eingebürgert. Bei RAID 0 schließt man mindestens zwei Festplatten zusammen. Die Performance ist sehr gut – jedoch ist das RAID-System zerstört und Ihre Daten sind verloren, wenn auch nur eine von noch so vielen Festplatten ausfällt.

RAID 1 Dieser Level bietet eine zufrieden stellende Sicherheit für die Daten, weil diese 1:1 auf eine andere Festplatte kopiert werden. Dies nennt man *Festplattenspiegelung* – ist eine Platte zerstört, liegt eine Kopie deren Inhalts auf einer anderen. Es dürfen alle bis auf eine

der Festplatten fehlerhaft sein, ohne Daten verloren zu haben. Die Schreibperformance leidet durch den Kopiervorgang ein wenig bei einer Verwendung von RAID 1 (10-20% langsamer), dafür geht der Lesezugriff deutlich schneller im Vergleich zu einer einzelnen normalen physikalischen Festplatte, weil die Daten doppelt vorhanden sind und somit parallel ausgelesen werden können.

RAID 5 RAID 5 ist ein optimierter Kompromiss aus den beiden anderen Levels was Performance und Redundanz betrifft. Der Festplattenplatz entspricht der Anzahl der eingesetzten Platten minus einer. Die Daten werden wie bei RAID 0 über die Festplatten verteilt. Für die Sicherheit sorgen die *Paritätsblöcke*, die bei RAID 5 auf einer der Partitionen angelegt werden. Diese werden mit XOR miteinander verknüpft – somit lässt sich beim Ausfall einer Partition durch den dazugehörigen Paritätsblock der Inhalt nach XOR rekonstruieren. Bei RAID 5 ist zu beachten, dass nicht mehr als eine Festplatte gleichzeitig ausfallen darf. Fällt eine aus, muss sie schnellstmöglich ausgetauscht werden, da sonst Datenverlust droht.

1.9.2 Soft-RAID-Konfiguration mit YaST

Zur Soft-RAID-Konfiguration gelangen Sie entweder über ein eigenes 'RAID'-Modul unter 'System' oder über das Partitionierungs-Modul unter 'Hardware'.

1. Schritt: Partitionieren Zunächst sehen Sie unter 'Experten-Einstellungen' im Partitionierungs-Tool Ihre Partitionen aufgelistet. Wenn Sie bereits Soft-RAID-Partitionen angelegt haben, erscheinen diese hier. Andernfalls müssen Sie neue anlegen. Bei RAID 0 und RAID 1 benötigen Sie mindestens zwei Partitionen – bei RAID 1 sind das im Normalfall genau zwei. Für eine Verwendung von RAID 5 hingegen sind mindestens drei Partitionen nötig. Es ist zu empfehlen, nur Partitionen gleicher Größe zu nehmen.

Die einzelnen Partitionen eines RAIDs sollten auf verschiedenen Festplatten liegen, damit das Risiko eines Datenverlustes durch den Defekt einer Festplatte bei RAID 1 und 5 verhindert wird bzw. die Performance bei RAID 0 optimiert wird.

2. Schritt: RAID anlegen Wenn Sie auf 'RAID' klicken, erscheint der Dialog, in dem Sie den RAID-Level 0, 1 oder 5 auswählen. In der nächsten Maske haben Sie die Möglichkeit, die Partitionen dem neuen

RAID zuzuordnen. Hinter 'Experten-Optionen' finden Sie Einstellmöglichkeiten für die *chunk-size* – hier können Sie Fein-Tuning für die Performance vornehmen. Die Aktivierung der Checkbox 'Persistent superblock' sorgt dafür, dass RAID-Partitionen gleich beim Booten als solche erkannt werden.

Nach Beendigung der Konfiguration sehen Sie auf der Experten-Seite im Partitionierungs-Modul dann das Device `/dev/md0` (etc.) als *RAID* gekennzeichnet.

Troubleshooting Ob eine RAID-Partition zerstört ist, können Sie dem Inhalt der Datei `/proc/mdstats` entnehmen. Grundsätzliche Vorgehensweise in einem Fehlerfall ist es, Ihr Linux-System herunterzufahren und die defekte Festplatte durch eine neue gleichartig partitionierte zu ersetzen. Dann starten Sie Ihr System neu und verwenden den Befehl `raidhotadd /dev/mdX /dev/sdX`. Damit wird die neue Festplatte automatisch in das RAID-System integriert und vollautomatisch rekonstruiert.

Eine Anleitung zur Konfiguration von Soft-RAID und weitere Details hierzu finden Sie im angegebenen Howto:

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

oder in der Linux-RAID-Mailingliste zum Beispiel über:

- <http://www.mail-archive.com/linux-raid@vger.rutgers.edu>

Dort finden Sie auch Hilfe, falls wider Erwarten komplexe Probleme auftreten sollten.

Update des Systems und Paketverwaltung

SUSE LINUX bietet die Möglichkeit, ein bestehendes System ohne Neuinstallation zu aktualisieren. Dabei muss unterschieden werden zwischen der *Aktualisierung einzelner Softwarepakete* und einem *Update des gesamten Systems*.

Einzelne Pakete können auch von Hand mit dem Paketmanager rpm installiert werden.

2.1	SUSE LINUX aktualisieren	44
2.2	Softwareänderungen von Version zu Version	49
2.3	RPM – Der Paket-Manager der Distribution	58

2.1 SUSE LINUX aktualisieren

Es ist ein bekanntes Phänomen, dass Software von Version zu Version wächst. Deshalb empfiehlt es sich *vor* dem Update mit `df` nachzuschauen, wie sehr die einzelnen Partitionen bereits ausgelastet sind. Wenn Sie den Eindruck haben, es könnte knapp werden, dann führen Sie bitte vor dem Update ein Datenbackup durch und partitionieren Sie das System neu. Es kann kein genereller Tipp gegeben werden, wie viel Platz jeweils im Einzelnen benötigt wird – der Platzbedarf ist abhängig von der Art der bestehenden Partitionierung, von der ausgewählten Software und von der Versionsnummer des bestehenden Systems auf die aktuelle SUSE LINUX Distribution.

Hinweis

Es ist empfehlenswert, auf der CD die Datei `LIESMICH README` bzw. unter DOS/Windows die Datei `LIESMICH.DOS README.DOS` zu lesen; dort notieren wir zusätzliche Änderungen, die *nach* der Drucklegung des Handbuchs erfolgt sind!

Hinweis

2.1.1 Vorbereitungen

Vor Beginn eines Updates sollten sicherheitshalber die alten Konfigurationsdateien auf ein separates Medium (Streamer, Wechselplatte, ZIP-Laufwerk, CD-ROM etc.) kopiert werden. In erster Linie handelt es sich um die Dateien, die in `/etc` gespeichert sind; weiterhin sind die Konfigurationsdateien unter `/var/lib` zu kontrollieren. Zudem kann es nichts schaden, die aktuellen Benutzerdaten unter `/home` (die HOME-Verzeichnisse) auf ein Backup-Medium zu schreiben. Das Sichern der Daten ist als Systemadministrator `root` durchzuführen; nur `root` hat die Rechte, alle lokalen Dateien zu lesen. Bevor Sie den Update-Vorgang einleiten, notieren Sie sich die Rootpartition; mit dem Kommando `df /` können Sie den Gerätenamen der Rootpartition herausfinden; in dem Fall der Ausgabe 2.1 ist `/dev/hda2` die zu notierende Root-Partition.

Beispiel 2.1: Überblick mit `df -h`

Dateisystem	Größe	Benut	Verf	Ben%	montiert auf
<code>/dev/hda1</code>	1,9G	189M	1.7G	10%	<code>/dos</code>
<code>/dev/hda2</code>	8,9G	7,1G	1,4G	84%	<code>/</code>
<code>/dev/hda5</code>	9,5G	8,3G	829M	92%	<code>/home</code>

Die Ausgabe zeigt, dass die Partition `/dev/hda2` unter `/` in das Dateisystem eingehängt (gemountet) ist.

Mögliche Probleme

PostgreSQL Vor einem PostgreSQL-Update (`postgres`) empfiehlt es sich in der Regel, die Datenbanken zu dumpen; vgl. die Manualpage von `pg_dump`. Dies ist natürlich nur dann erforderlich, wenn Sie PostgreSQL vor dem Update tatsächlich *benutzt* haben.

Promise-Controller Die Festplatten-Controller der Firma Promise finden sich inzwischen auf hochwertigen Mainboards in verschiedenen Rechnern. Manchmal als reine IDE-Controller (für UDMA 100), manchmal als IDE-RAID-Controller. Seit SUSE LINUX 8.0 werden diese Controller direkt vom Kernel unterstützt und als normale Controller für IDE-Festplatten behandelt. Erst das zusätzlich Kernel-Modul `pdraid` ermöglicht die RAID-Funktionalität.

In manchen Fällen kann es beim Update passieren, dass Festplatten am Promise-Controller vor den Festplatten am normalen IDE-Controller erkannt werden. Das System wird dann nach einem Kernel-Update nicht mehr booten und sich typischerweise mit `Kernel panic: VFS: unable to mount root fs` verabschieden. In diesem Fall muss beim Booten der Kernel-Parameter `ide=reverse` angegeben werden, um die Reihenfolge der Plattenkennung umzudrehen; vgl. Abschnitt 1.1.2 auf Seite 8. Dieser Parameter muss für dauerhafte Verwendung mit YaST in die Bootkonfiguration eingetragen werden; vgl. das Kapitel *Benutzerdefinierte Installation, Booten (Bootloader-Installation)* im Handbuch [1].

Achtung

Nur die im BIOS eingeschalteten Controller werden gefunden. Insbesondere hat das nachträgliche Ein- oder Ausschalten von Controllern im BIOS direkte Auswirkungen auf die Devicenamen. Bei unbedachtem Vorgehen ist unter Umständen das Booten nicht mehr möglich!

Achtung

Technische Erklärung Die Reihenfolge der Controller hängt vom Mainboard ab, jeder Hersteller hat seine eigene Strategie, Zusatzcontroller zu verdrahten. Mit dem Befehl `lspci` wird diese Reihenfolge sichtbar. Wenn der Promise-Controller vor dem normalen IDE-Controller

aufgeführt wird, ist der Kernel-Parameter `ide=reverse` nach einem Update notwendig. Mit dem alten Kernel (ohne direkte Promise Unterstützung) wurde der Controller ignoriert und der normale IDE-Controller zuerst erkannt. Die erste Platte war dann `/dev/hda`. Mit dem neuen Kernel wird der Promise-Controller direkt gefunden und seine (bis zu vier) Platten als `/dev/hda`, `/dev/hdb`, `/dev/hdc` und `/dev/hdd` gemeldet. Die bisherige `/dev/hda`-Platte wird plötzlich zu `/dev/hde` und daher beim Bootvorgang nicht mehr gefunden.

2.1.2 Update mit YaST

Nach den in Abschnitt 2.1.1 auf Seite 44 genannten Vorarbeiten leiten Sie den Bootvorgang ein.

1. Starten Sie das System wie zur Installation (vgl. Benutzerhandbuch) und wählen Sie dann in YaST — nach Festlegung der Sprache — *nicht* ‘Neuinstallation’, sondern ‘Update des bestehenden Systems’.
2. YaST wird ermitteln, ob mehr als eine Rootpartition vorhanden ist; falls nein, geht es weiter mit dem Systembackup. Falls mehrere Partitionen vorhanden sind, müssen Sie die richtige Partition auswählen und mit ‘Weiter’ bestätigen (beim Beispiel in Abschnitt 2.1.1 auf Seite 44 hatten Sie `/dev/hda2` notiert).

YaST wird alte `fstab` einlesen, die sich auf dieser Partition befindet, um dann die dort eingetragenen Dateisysteme zu analysieren und schließlich zu mounten.

3. Danach besteht die Möglichkeit, eine Sicherungskopie der Systemdateien während des Updates erstellen zu lassen. Diese Option verlangt, dass der Update-Vorgang, sollte aber gewählt werden, wenn Sie kein aktuelles Systembackup haben.
4. Entweder kann im folgenden Dialog festgelegt werden, dass nur die bereits installierte Software erneuert wird oder dass dem System wichtige neue Softwarekomponenten hinzugesellt werden (Upgrade-Modus). Es ist empfehlenswert, die vorgegebene Zusammenstellung zu akzeptieren (zum Beispiel ‘Standard-System’). Etwaige Unstimmigkeiten können Sie mit YaST später beseitigen.

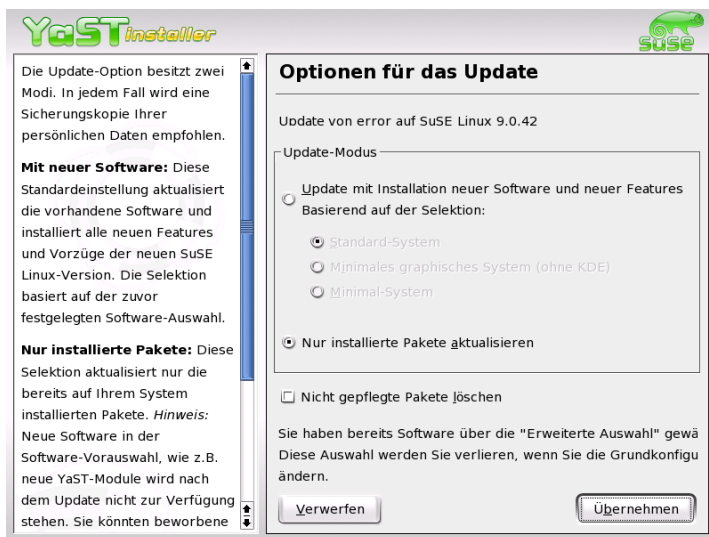


Abbildung 2.1: Update der Software

2.1.3 Manuell gesteuertes Update

Das Basissystem erneuern

Da beim Aktualisieren des Grundsystems die zentralen Bestandteile des Systems (wie zum Beispiel Bibliotheken) ausgetauscht werden müssen, kann diese Aufgabe nicht im normalen Betrieb, das heisst aus dem bereits laufenden Linuxsystem heraus, erledigt werden.

Sie müssen also die Update-Umgebung starten. Dies geschieht im Normalfall mit der CD bzw. DVD oder mit der selbst erstellten Diskette zum Booten (Bootdisk). Wenn Sie manuelle Eingriffe während des Updates vornehmen oder das gesamte Update mit der ncurses-ui von YaST (Textmodus) durchführen wollen, sind im Wesentlichen die Schritte notwendig, die bereits in Abschnitt 1.1 auf Seite 8 ff. ausführlich beschrieben sind:

1. Direkt im Anschluss an das Booten des Kernels von der Bootdisk oder der CD bzw. DVD wird automatisch linuxrc gestartet.
2. Im linuxrc sind im Hauptmenü unter dem Menüpunkt 'Einstellungen' Sprache und Tastatur festzulegen und jeweils mit 'Ok' zu bestätigen.

3. Über den Menüpunkt 'Kernel-Module' müssen ggf. die notwendigen Hardware- und Software-Treiber geladen werden; zum genauen Vorgehen vgl. Abschnitt 1.1.3 auf Seite 10 und die linuxrc-Beschreibung 12.4.4 auf Seite 312.
4. Es kann über die Menüpunkte 'Installation / System starten' -> 'Installation/Update starten' zur Auswahl des Quellmediums übergegangen werden (vgl. 12.4.6 auf Seite 313).
5. Von linuxrc wird die Installationsumgebung geladen und es wird YaST gestartet.

Im Eingangsmenü von YaST wählen Sie — nach Kontrolle der Sprache und Überprüfung der Hardware durch YaST — den Punkt 'Update des bestehenden Systems'.

Im Anschluss versucht YaST, die Root-Partition herauszufinden und bietet das Ergebnis zur Auswahl bzw. Bestätigung an; in der angezeigten Liste geben Sie Ihre Root-Partition an, wie oben notiert (Beispiel: `/dev/hda2`). So beauftragen Sie YaST, die alte `fstab` einzulesen, die sich auf dieser Partition befindet; YaST wird die dort eingetragenen Dateisysteme analysieren und dann mounten.

Danach besteht die Möglichkeit, eine Sicherungskopie der Systemdateien während des Updates erstellen zu lassen.

Entweder kann im folgenden Dialog festgelegt werden, dass nur die bereits installierte Software erneuert wird oder dass dem System wichtige neue Softwarekomponenten hinzugesellt werden (Upgrade-Modus). Es ist empfehlenswert, die vorgegebene Zusammenstellung zu akzeptieren (zum Beispiel 'Standard-System'). Etwaige Unstimmigkeiten können Sie mit YaST beseitigen.

Warndialog: 'Ja', damit das Übertragen der neuen Software von dem Quellmedium auf die Festplatte des Systems geschehen kann. Es folgt die Überprüfung der RPM-Datenbank.

Anschließend werden zunächst die zentralen Bestandteile des Systems aktualisiert, wobei YaST automatisch Sicherungen von Dateien anlegt, die seit der letzten Installation während des Betriebs verändert wurden; weiterhin werden alte Konfigurationsdateien ggf. mit der Endung `.rpmorig` bzw. `.rpmsave` gesichert; der Vorgang der Installation bzw. des Updates wird in `/var/adm/inst-log/installation-*` protokolliert und ist jederzeit nachlesbar.

Update des restlichen Systems

Ist das Basissystem aktualisiert, gelangen Sie in einen speziellen Update-Modus von YaST. Dort können Sie nach Ihren Wünschen den Rest des Systems updaten.

Nachdem diese Aufgabe erledigt ist, müssen Sie den Vorgang wie eine Erstinstallation abschließen. Unter anderem sollten Sie einen neuen Kernel auswählen; YaST wird diese Option anbieten.

Mögliche Probleme

Falls sich nach dem Update bestimmte Shell-Umgebungen nicht mehr so verhalten wie gewohnt, kontrollieren Sie bitte unbedingt, ob die aktuellen Punkt-Dateien im Homeverzeichnis noch zum System passen. Ist dies nicht der Fall, übernehmen Sie bitte die aktuellen Versionen von `/etc/skel`; zum Beispiel: `cp /etc/skel/.profile /.profile`.

2.1.4 Aktualisieren einzelner Pakete

Unabhängig von einem Gesamt-Update können Sie jederzeit einzelne Pakete aktualisieren; dabei müssen Sie *selbst* freilich darauf achten, dass das System konsistent bleibt: Update-Empfehlungen finden Sie unter <http://www.suse.de/de/support/download/updates/> aufgelistet.

In der Paketauswahl von YaST können Sie nach Herzenslust schalten und walten. Wählen Sie ein Paket zum Update aus, das für den Betrieb des Systems eine zentrale Rolle spielt, werden Sie von YaST gewarnt. Derartige Pakete sollten im speziellen Update-Modus aktualisiert werden. Beispielsweise enthalten etliche Pakete *shared libraries*, die möglicherweise zum Zeitpunkt des Updates von laufenden Prozessen verwendet werden. Ein Update im laufenden System würde daher dazu führen, dass diese Programme nicht mehr korrekt funktionieren können.

2.2 Softwareänderungen von Version zu Version

In den folgenden Abschnitten wird aufgelistet, welche Details sich von Version zu Version geändert haben. In dieser Übersicht erscheint beispielsweise, ob grundlegende Einstellungen neu vorgenommen oder ob Konfigura-

tionsdateien an andere Stellen verschoben wurden oder ob bekannte Programme erkennbar modifiziert wurden. Es werden die Dinge genannt, die den Benutzer bzw. den Administrator bei der täglichen Arbeit unmittelbar berühren. Die Liste ist keineswegs erschöpfend und vollständig.

Probleme und Besonderheiten der jeweiligen Version werden bei Bekanntwerden auf dem WWW-Server veröffentlicht; vgl. die unten angegebenen Links. Wichtige Updates einzelner Pakete sind über <http://www.suse.de/de/support/download/updates/> zugänglich.

2.2.1 Von 7.3 auf 8.0

Probleme und Besonderheiten: <http://sdb.suse.de/sdb/de/html/bugs80.html>.

- Bootdisketten werden nur noch in Form von Diskettenimages (bisher Verzeichnis `disks`, jetzt `boot`) mitgeliefert. Eine Bootdiskette benötigen Sie nur, wenn Sie nicht von CD booten können; je nach Hardware oder Installationsvorhaben sind zusätzlich Disketten von den Images `modules1`, `modules2` etc. zu erstellen; zum Vorgehen vgl. 1.4 auf Seite 19 bzw. 1.4.2 auf Seite 21.
- YaST2 ersetzt nunmehr vollständig YaST1, auch im Text- bzw. Konsolenmodus. Wenn im Text von YaST die Rede ist, ist immer die neue Version gemeint.
- Einige BIOSse benötigen den Kernelparameter `realmode-power-off`; dieser hieß bis Kernelversion 2.4.12 `real-mode-poweroff`.
- Die `START`-Variablen der `rc.config` zum Starten von Diensten sind nicht mehr erforderlich. Alle Dienste werden gestartet, wenn die entsprechenden Links in den Runlevel-Verzeichnissen vorhanden sind; die Links werden mit `insserv` angelegt.
- Systemdienste werden über Variablen-Einträge in den Dateien in `/etc/sysconfig` konfiguriert; beim Update werden die Einstellungen aus den Dateien in `/etc/rc.config.d` übernommen.
- `/etc/init.d/boot` in mehrere Skripte aufgeteilt und, wenn sinnvoll, in andere Pakete verschoben (vgl. `kbd`, `isapnp`, `lvm` usw.); vgl. 13.4 auf Seite 330.
- Im Bereich Netzwerk wurde eine Reihe von Änderungen vorgenommen; vgl. dazu Abschnitt 14.4 auf Seite 370.

- Zur Verwaltung der Protokoll-Dateien *logfiles* wird das *logrotate* verwendet; */etc/logfiles* ist nicht mehr erforderlich ; vgl. Abschnitt 12.2.3 auf Seite 300.
- Das Login für *root* per *telnet* oder *rlogin* kann durch Einstellungen in den Dateien in */etc/pam.d* erlaubt werden; das Setzen von *ROOT_LOGIN_REMOTE* auf *yes* wird wegen Sicherheitsaspekten nicht mehr zugelassen.
- *PASSWD_USE_CRACKLIB* kann mit YaST aktiviert werden.
- Wenn NIS-Dateien für *autofs* über NIS verteilt werden sollen, ist das NIS-Client-Modul von YaST zur Konfiguration zu verwenden; aktivieren Sie dort 'Automounter starten'. Dadurch ist die Variable *USE_NIS_FOR_AUTOFS* obsolet.
- *locate* zum schnellen Finden von Dateien gehört nicht mehr zum Standardumfang der installierten Software. Bei Bedarf bitte nachinstallieren (*find-locate*) – dann wird auch wie früher automatisch ca. 15 Minuten nach dem Einschalten der *updatedb*-Prozess gestartet!
- Für *pine* ist Maus-Support aktiviert. Das bedeutet, dass man *Pine* in einem *xterm* (o. Ä.) auch mit der Maus bedienen kann, wenn man auf die Menüpunkte klickt. Das bedeutet allerdings auch weiterhin, dass Cut & Paste nur bei gedrückter Shift-Taste funktioniert, wenn der Mouse-Support aktiv ist. Bei einer Neuinstallation ist dies deaktiviert. Beim Update ist jedoch nicht auszuschließen, dass diese Funktion aktiv ist (wenn eine älter *~/ .pinerc* vorhanden ist). In diesem Fall kann man in der *Pine*-Konfiguration die Option *enable-mouse-in-xterm* deaktivieren und alles ist wieder gut.

2.2.2 Von 8.0 auf 8.1

Probleme und Besonderheiten: <http://sdb.suse.de/sdb/de/html/bugs81.html>.

- Änderungen bei Benutzer- und Gruppennamen des Systems: Um Übereinstimmung mit *UnitedLinux* zu erreichen, wurden einige Einträge in */etc/passwd* bzw. */etc/group* angepasst.
 - ▷ Geänderte Benutzer: *ftp* nun in Gruppe *ftp* (nicht mehr in *daemon*).

- ▷ Umbenannte Gruppen: `www` (war `wwwadmin`); `games` (war `game`).
 - ▷ Neue Gruppen: `ftp` (mit GID 50); `floppy` (mit GID 19); `cdrom` (mit GID 20); `console` (mit GID 21); `utmp` (mit GID 22).
- Änderungen im Zusammenhang mit dem FHS (vgl. Abschnitt 12.1.2 auf Seite 298):
 - ▷ Eine Beispiel-Umgebung für HTTPD (Apache) wird unter `/srv/www` angelegt (war `/usr/local/httpd`).
 - ▷ Eine Beispiel-Umgebung für FTP wird unter `/srv/ftp` angelegt (war `/usr/local/ftp`). Hierzu ist das Paket `ftplib` benötigt.
- Um einen gezielten Zugriff auf gesuchte Software zu ermöglichen, sind die einzelnen Pakete nicht mehr in wenigen unübersichtlichen Serien untergebracht, sondern in eingängigen RPM-Gruppen. Das hat zur Konsequenz, dass es auf den CDs keinen kryptischen Verzeichnisse unter `suse` mehr gibt, sondern nur noch wenige nach Architekturen benannte Verzeichnisse wie zum Beispiel `ppc`, `i586` oder `noarch`.
- Bei einer Neuinstallation werden nunmehr die folgenden Programme eingerichtet bzw. nicht mehr automatisch installiert:
 - ▷ Der Bootloader GRUB, der entschieden mehr Möglichkeiten als LILO bietet. LILO bleibt jedoch erhalten, wenn ein *Update* des Systems durchgeführt wird.
 - ▷ Der Mailer `postfix` anstelle von `sendmail`.
 - ▷ Anstelle von `majordomo` wird die moderne Mailinglistensoftware `mailman` installiert.
 - ▷ `hardened_suse` bitte von Hand bei Bedarf auswählen und die aktuelle Dokumentation dazu lesen!
- Aufgeteilte Pakete: `rpm` in `rpm` und `rpm-devel`; `popt` in `popt` und `popt-devel`; `libz` in `zlib` und `zlib-devel`.
`yast2-trans-*` nun nach Sprachen aufgeteilt: `yast2-trans-cs` (tschechisch), `yast2-trans-de` (deutsch), `yast2-trans-es` (spanisch) etc.; bei der Installation werden nicht mehr alle Sprachen installiert, um Plattenplatz zu sparen. Bei Bedarf die notwendigen Pakete für die YaST-Sprachunterstützung bitte nachinstallieren!
- Umbenannte Pakete: `bzip` in `bzip2`.

- Nicht mehr mitgelieferte Pakete: `openldap`, bitte nun `openldap2` verwenden; `su1`, bitte nun auf `sudo` umsteigen.

2.2.3 Von 8.1 auf 8.2

Probleme und Besonderheiten: <http://sdb.suse.de/sdb/de/html/bugs82.html>.

- 3D-Support für nVidia-basierte Grafikkarten (Änderungen):
Die RPM-`NVIDIA_GLX/NVIDIA_kernel` (einschließlich das `switch2nvidia_glx`-Skript) sind nicht mehr enthalten. Bitte laden Sie sich den nVidia-Installer für Linux IA32 von der nVidia-Webseite (<http://www.nvidia.com>) herunter, installieren den Treiber mit diesem, und verwenden dann `SxX2` bzw. `YaST`, um 3D-Support zu aktivieren.
- Bei einer Neuinstallation wird der `xinetd` anstelle des `inetd` installiert und mit sicheren Vorgaben konfiguriert; vgl. das Verzeichnis `/etc/xinetd.d`. Bei einem Systemupdate bleibt jedoch der `inetd` erhalten.
- PostgreSQL liegt nun in Version 7.3 vor. Beim Umstieg von einer Version 7.2.x ist ein `dump/restore` mit `pg_dump` erforderlich. Wenn Ihre Applikation die Systemkataloge abfragt, dann sind weitere Anpassungen notwendig, da mit Version 7.3 Schemas eingeführt wurden. Zusätzliche Informationen finden Sie unter: http://www.ca.postgresql.org/docs/momjian/upgrade_tips_7.3
- Die Version 4 von `stunnel` unterstützt keine Optionen an der Kommandozeile mehr. Es wird jedoch das Skript `/usr/sbin/stunnel3_wrapper` mitgeliefert, das in der Lage ist, die Kommandozeilenoptionen in eine für `stunnel` geeignete Konfigurationsdatei zu konvertieren und diese beim Aufruf zu verwenden (anstelle von `OPTIONS` setzen Sie bitte Ihre Optionen ein):

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

Die erzeugte Konfigurationsdatei wird auch auf die Standardausgabe ausgegeben, sodass Sie diese Angaben leicht verwenden können, um eine permanente Konfigurationsdatei für die Zukunft zu erzeugen.

- `openjade` (`openjade`) ist nun die DSSSL-Engine, die anstelle von `jade` (`jade_dsl`) zum Einsatz kommt, wenn `db2x.sh`

(docbook-toys) aufgerufen wird. Aus Gründen der Kompatibilität stehen die einzelnen Programme auch ohne das Präfix `o` zur Verfügung.

Falls eigene Anwendungen von dem Verzeichnis `jade_dsl` und den dort bislang installierten Dateien abhängig sind, müssen entweder die eigenen Anwendungen auf das neue Verzeichnis `/usr/share/sgml/openjade` angepasst oder es kann als `root` ein Link angelegt werden:

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

Um einen Konflikt mit dem `r2sz` zu vermeiden, heißt das Kommandozeilentool `sx` weiterhin `s2x` bzw. `sgml2xml` oder `osx`.

2.2.4 Von 8.2 auf 9.0

Probleme und Besonderheiten: <http://sdb.suse.de/sdb/de/html/bugs90.html>.

- Die regelmäßigen Wartungsdienste in `/etc/cron.daily`, `/etc/cron.weekly` und `/etc/cron.monthly` werden um 4:00 Uhr ausgeführt, Diese Zeiten gelten nur für Neuinstallationen; nach einem Update ist `/etc/crontab` gegebenenfalls anzupassen.
- Der RPM-Paketmanager steht nun in Version 4 zur Verfügung. Die Funktionalität zum Paketebauen ist nunmehr in das eigenständige Programm `rpmbuild` überführt worden; `rpm` wird weiterhin zum Installieren, Aktualisieren und zu Datenbankabfragen verwendet; vgl. Abschnitt 2.3 auf Seite 58.
- Im Bereich *Drucken* es gibt das Paket `foomatic-filters`. Der Inhalt wurde aus dem `cups-drivers` abgesplittet, da sich gezeigt hat, dass man damit auch dann drucken kann, wenn CUPS nicht installiert ist. So kann man Konfigurationen mit YaST einstellen, die vom Drucksystem (CUPS, LPRng) unabhängig sind. Als Konfigurationsdatei enthält dies Paket die Datei `/etc/foomatic/filter.conf`.
- Auch bei dem Einsatz von LPRng/lpfilter werden nun die Pakete `foomatic-filters` und `cups-drivers` benötigt.
- Die XML-Ressourcen der mitgelieferten Softwarepakete werden über Einträge in `/etc/xml/suse-catalog.xml` zugänglich gemacht. Diese Datei darf nicht mit `xmlcatalog` bearbeitet werden

werden, weil sonst gliedernde Kommentare verschwinden, die benötigt werden, um ein ordnungsgemäßes Update zu gewährleisten. `/etc/xml/suse-catalog.xml` wird über ein `nextCatalog-Statement` in `/etc/xml/catalog` zugänglich gemacht, sodass XML-Tools wie `xmllint` oder `xsltproc` die lokalen Ressourcen automatisch finden können.

2.2.5 Von 9.0 auf 9.1

Probleme und Besonderheiten: <http://portal.suse.de/sdb/de/2004/02/bugs91.html>.

- SUSE LINUX wurde komplett auf die Kernelversion 2.6 umgestellt; die Vorgängerversion 2.4 sollte nicht mehr verwendet werden, da die mitgelieferten Programme mit Kernel 2.4 möglicherweise nicht funktionieren. Weiterhin sind folgende Einzelheiten zu beachten:
 - ▷ Das Laden der Module werden nun über die Datei `/etc/modprobe.conf` konfiguriert; die Datei `/etc/modules.conf` ist obsolet. YaST wird die Datei versuchen zu konvertieren (vgl. auch das Skript `/sbin/generate-modprobe.conf`).
 - ▷ Module haben nun das Suffix `.ko`.
 - ▷ Das Modul `ide-scsi` wird beim Brennen von CDs nicht mehr benötigt.
 - ▷ Bei den Optionen der ALSA-Soundmodule ist das Prefix `snd_` entfernt worden.
 - ▷ `sysfs` ergänzt nun `/proc`-Dateisystem.
 - ▷ Das Powermanagement (speziell ACPI) wurde verbessert und kann nun über ein YaST-Modul eingestellt werden.
- Zu den Änderungen bei den Eingabegeräten (*Input Devices*) vgl. den oben genannten Portalartikel.
- Programme, die gegen NGPT (*Next Generation POSIX Threading*) gelinkt sind, laufen nicht mit `glibc 2.3.x`. Alle davon betroffenen Programme, die nicht mit SUSE LINUX mitgeliefert werden, müssen entweder mit `linuxthreads` oder `NPTL` (*Native POSIX Thread Library*) neu kompiliert werden. Bei der Portierung ist `NPTL` zu bevorzugen, da das der in die Zukunft weisende Standard ist.

Bei Schwierigkeiten mit NPTL kann auf die älteren linuxthreads-Implementierung durch das Setzen der folgenden Umgebungsvariablen ausgewichen werden (dabei muss *<kernel-version>* durch die Versionsnummer des entsprechenden Kernels ersetzt werden):

```
LD_ASSUME_KERNEL=kernel-version
```

Dabei sind folgende Versionsnummern möglich:

2.2.5 (i386, s390): linuxthreads ohne Floating Stacks

2.4.1 (AMD64, IPF, s390x, i686): linuxthread mit Floating Stacks

Hinweise zum Kernel und linuxthreads *mit* Floating Stacks:

Programme, die `errno`, `h_errno` und `_res` verwenden, müssen die einschlägigen Header-Dateien (`errno.h`, `netdb.h` und `resolv.h`) mit `#include` einbinden. C++-Programme mit Multithread-Unterstützung, die *Thread Cancellation* verwenden, müssen mit der Umgebungsvariablen `LD_ASSUME_KERNEL=2.4.1` dazu gebracht werden, die Bibliothek linuxthreads zu verwenden.

- NPTL (*Native POSIX Thread Library*) ist bei SUSE LINUX 9.1 als Thread-Paket dabei. NPTL wurde binärkompatibel zu der älteren Bibliothek linuxthreads entwickelt. An den Stellen jedoch, an denen linuxthreads gegen den POSIX-Standard verstößt, erfordert NPTL Anpassungen; im Einzelnen sind zu nennen: Signal-Behandlung; `getpid` liefert in allen Threads denselben Wert zurück; Threads-Handlers, die mit `pthread_atfork` registriert sind, laufen nicht, wenn `vfork` verwendet wird.
- Als Kodierung für das System ist nun UTF-8 voreingestellt. Bei einer Standardinstallation wird also eine Locale mit `.UTF-8` als Kodierungsangabe (*Encoding*) festgelegt (z.B. `de_DE.UTF-8`).
- Shell-Tools aus dem `coreutils` wie `tail`, `chown`, `head`, `sort` etc. folgen in der Vorgabeeinstellung nun dem POSIX-Standard von 2001 (*Single UNIX Specification, version 3 == IEEE Std 1003.1-2001 == ISO/IEC 9945:2002*) und nicht mehr dem Standard von 1992. Das alte Verhalten kann man mit einer Umgebungsvariablen erzwingen:

```
_POSIX2_VERSION=199209
```

Der neue Wert ist 200112 und wird als Vorgabe für `_POSIX2_VERSION` angenommen. Den SUS-Standard kann man hier nachlesen (frei, aber eine Registrierung ist erforderlich):

<http://www.unix.org>

Hier eine kurze Gegenüberstellung:

Tabelle 2.1: Gegenüberstellung POSIX 1992/POSIX 2001

POSIX 1992	POSIX 2001
chown tux.users	chown tux:users
tail +3	tail -n +3
head -1	head -n 1
sort +3	sort -k +3
nice -10	nice -n 10
split -10	split -l 10

Hinweis

Software von Drittanbietern folgt möglicherweise noch nicht dem neuen Standard; in einem solchen Fall ist es ratsam, die Umgebungsvariable wie oben beschrieben zu setzen:
`_POSIX2_VERSION=199209`.

Hinweis

- `/etc/gshadow` wurde aufgegeben und entfernt, da die Datei überflüssig ist; die Gründe dafür sind:
 - ▷ Seitens der glibc gibt es keine Unterstützung.
 - ▷ Es gibt keine offizielle Schnittstelle für diese Datei; sogar in der shadow-Suite gibt es keine solche Schnittstelle.
 - ▷ Die meisten Tools, die das Gruppenpasswort überprüfen, unterstützen die Datei nicht und ignorieren sie aus den eben genannten beiden Gründen.
- Mit dem Update von samba 2.x auf samba 3.x steht die winbind-Authentifikation nicht mehr zur Verfügung; die anderen Methoden sind weiterhin möglich. Aus diesem Grund wurden die folgenden Programme entfernt:

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

Vgl. auch: <http://www.squid-cache.org/Doc/FAQ/FAQ-23.html#ss23.5>

- Der FHS (siehe 12.1.2 auf Seite 298) sieht nun vor, dass XML-Ressourcen (DTDs, Stylesheets etc.) unter `/usr/share/xml` installiert werden. Aus diesem Grund sind einige Verzeichnisse nun nicht mehr unter `/usr/share/sgml` zu finden. Bei Problemen müssen entweder die eigenen Skripte oder Makefiles angepaßt bzw. die offiziellen Kataloge (insbesondere `/etc/xml/catalog` bzw. `/etc/sgml/catalog`) verwendet werden.

2.3 RPM – Der Paket-Manager der Distribution

Bei SUSE LINUX kommt RPM (*RPM Package Manager*) mit den Hauptprogrammen `rpm` und `rpmbuild` als Management für die Softwarepakete zum Einsatz. Damit steht den Benutzern, den Systemadministratoren und nicht zuletzt dem Pakete-Macher die mächtige RPM-Datenbank zur Verfügung, über die jederzeit detaillierte Informationen zur installierten Software abgefragt werden können.

Im Wesentlichen kann `rpm` in fünf Modi agieren: Softwarepakete installieren bzw. de-installieren oder updaten, die RPM-Datenbank neu erstellen, Anfragen an die RPM-Datenbank bzw. an einzelne RPM-Archive richten, Pakete auf Integrität überprüfen und Pakete signieren. `rpmbuild` dient dazu, installierbare Pakete aus den unangetasteten Quellen (*pristine sources*) herzustellen.

Installierbare RPM-Archive sind in einem speziellen binären Format gepackt; die Archive bestehen aus den zu installierenden (Programm-)Dateien und aus verschiedenen Meta-Informationen, die während der Installation von `rpm` benutzt werden, um das jeweilige Softwarepaket zu konfigurieren, oder die zu Dokumentationszwecken in der RPM-Datenbank abgelegt werden. RPM-Archive haben die Dateinamen-Endung `.rpm`.

Mit `rpm` lassen sich LSB-konforme Pakete verwalten; zu LSB vgl. Abschnitt 12.1.1 auf Seite 298.

Hinweis

Bei etlichen Paketen sind die für die Software-Entwicklung notwendigen Komponenten (Bibliotheken, Header- und Include-Dateien etc.) in eigene Pakete ausgelagert. Diese Entwicklungspakete werden nur benötigt, wenn Sie Software *selbst* übersetzen (kompilieren) wollen – beispielsweise neuere GNOME-Pakete. Solche Pakete sind in der Regel an dem Namenszusatz `-devel` zu erkennen: `alsa-devel`, `gimp-devel`, `kdelibs-devel` etc.

Hinweis

2.3.1 Prüfen der Authentizität eines Pakets

RPM-Pakete von SUSE LINUX sind mit GnuPG signiert:

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

Mit folgendem Befehl kann man die Signatur eines RPM-Pakets überprüfen und so feststellen, ob es wirklich von SUSE oder einer anderen vertrauenswürdigen Stelle stammt:

```
rpm --checksig apache-1.3.12.rpm
```

Insbesondere bei Updatepaketen aus dem Internet ist diese Vorsichtsmaßnahme zu empfehlen. Unser öffentlicher Paketsignierschlüssel ist standardmäßig in `/root/.gnupg/` hinterlegt. Seit Version 8.1 liegt der Schlüssel zusätzlich im Verzeichnis `/usr/lib/rpm/gnupg/`, damit auch normale Benutzer die Signatur von RPM-Paketen prüfen können.

2.3.2 Pakete verwalten: Installieren, Updaten und Deinstallieren

Im Normalfall ist das Installieren eines RPM-Archivs schnell erledigt:

```
rpm -i <paket>.rpm
```

Mit diesem Standardbefehl wird ein Paket aber nur dann installiert, wenn die Abhängigkeiten erfüllt sind und wenn es zu keinen Konflikten kommen kann; `rpm` fordert per Fehlermeldung die Pakete an, die zum Erfüllen der Abhängigkeiten notwendig sind. Die Datenbank wacht im Hintergrund darüber, dass es zu keinen Konflikten kommt: Eine Datei darf in der Regel nur zu einem Paket gehören. Mit verschiedenen Optionen kann man sich über diese Regel hinwegsetzen. Wer dies tut, der sollte aber genau wissen, was er tut, da er damit eventuell die Updatefähigkeit des Systems aufs Spiel setzt.

Interessant sind auch die Optionen `-U` bzw. `--upgrade` und `-F` bzw. `--freshen`, um ein Paket zu aktualisieren.

```
rpm -F <paket>.rpm
```

Dadurch wird eine ältere Version des gleichen Pakets gelöscht und die neue Version installiert. Der Unterschied zwischen den beiden Versionen liegt darin, dass bei `-U` auch Pakete installiert werden, die bisher nicht im System verfügbar waren, während die Option `-F` nur dann ein Paket erneuert, wenn es bereits zuvor installiert war. Gleichzeitig versucht `rpm`, sorgfältig mit den *Konfigurationsdateien* umzugehen, wobei – etwas vereinfacht – die folgende Strategie zum Tragen kommt:

- Falls eine Konfigurationsdatei vom Systemadministrator nicht verändert wurde, wird von `rpm` die neue Version der entsprechenden Datei installiert. Es sind keine Eingriffe seitens des Administrators notwendig.
- Falls eine Konfigurationsdatei vom Administrator zu irgendeinem Zeitpunkt vor dem Update geändert wurde, wird `rpm` die geänderte Datei dann – und nur dann – mit der Erweiterung `.rpmorig` oder `.rpmsave` sichern und die neue Version aus dem RPM-Paket installieren, falls sich zwischen ursprünglicher Datei und der Datei aus dem Update-Paket etwas geändert hat. In diesem Fall ist es sehr wahrscheinlich, dass Sie die frisch installierte Konfigurationsdatei anhand der Kopie (`.rpmorig` oder `.rpmsave`) auf Ihre Systembedingungen hin abstimmen müssen.
- `.rpmnew`-Dateien werden immer dann auftauchen, wenn es die Konfigurationsdatei bereits gibt *und* wenn in der `.spec`-Datei die `noreplace`-Kennung gesetzt wurde.

Im Anschluss an ein Update sollten nach dem Abgleich alle `.rpmorig`, `.rpmsave`- bzw. `.rpmnew`-Dateien entfernt werden, um bei folgenden Updates nicht zu stören. Die Erweiterung `.rpmorig` wird gewählt, wenn die Datei der RPM-Datenbank noch nicht bekannt war, sonst kommt `.rpmsave` zum Zuge; mit anderen Worten: `.rpmorig` entsteht beim Update von Fremdformat auf RPM und `.rpmsave` beim Update von RPM-alt auf RPM-neu. Bei `.rpmnew` kann keine Aussage gemacht werden, ob vom Systemadministrator eine Änderung an der Konfigurationsdatei vorgenommen wurde oder ob nicht. Eine Liste dieser Dateien finden Sie unter `/var/adm/rpmconfigcheck`.

Beachten Sie, dass einige Konfigurationsdateien (zum Beispiel `/etc/httpd/httpd.conf`) mit Absicht nicht überschrieben werden, um den sofortigen Weiterbetrieb mit den eigenen Einstellungen zu ermöglichen. Die Option `-U` ist also mehr als ein Äquivalent für die Abfolge `-e` (Deinstallieren/Löschen) und `-i` (Installieren). Wann immer möglich, dann ist der Option `-U` der Vorzug zu geben.

Hinweis

Nach jedem Update müssen Sie die von `rpm` angelegten Sicherungskopien mit der Erweiterung `.rpmorig` oder `.rpmsave` kontrollieren, das sind Ihre alten Konfigurationsdateien. Falls erforderlich, übernehmen Sie bitte Ihre Anpassungen aus den Sicherungskopien in die neuen Konfigurationsdateien und löschen Sie dann die alten Dateien mit der Erweiterung `.rpmorig` bzw. `.rpmsave`.

Hinweis

YaST mit der Option `-i` ist in der Lage, alle Paketabhängigkeiten aufzulösen und eine entsprechende Installation durchzuführen:

```
yast -i <paket>
```

Wenn ein Paket entfernt werden soll, geht man ähnlich vor:

```
rpm -e <paket>
```

`rpm` wird ein Paket aber nur dann entfernen, wenn keine Abhängigkeiten mehr bestehen. So ist es zum Beispiel theoretisch nicht möglich, `Tcl/Tk` zu löschen, solange noch irgendein anderes Programm `Tcl/Tk` benötigt – auch darüber wacht RPM mithilfe der Datenbank. Falls in einem Ausnahmefall eine derartige Löschoption nicht möglich sein sollte, obwohl keine Abhängigkeiten mehr bestehen, kann es hilfreich sein, die RPM-Datenbank mittels der Option `--rebuilddb` neu aufzubauen; vgl. unten die Anmerkungen zur RPM-Datenbank.

2.3.3 RPM und Patches

Um die Betriebssicherheit eines Systems zu gewährleisten, ist es notwendig, von Zeit zu Zeit Pakete in das System einzuspielen, die es auf einen neuen Stand bringen. Bisher konnte ein Fehler in einem Paket nur dadurch behoben werden, dass man das komplette Paket ersetzt hat. Bei großen Paketen mit kleinen Fehlern können so schnell große Datenmengen zusammen kommen. Seit der Version 8.1 gibt es bei SUSE daher ein neues Feature in RPM, das es ermöglicht, Patches zu Paketen einzuspielen.

Die interessantesten Informationen zu einem Patch-RPM sollen am Beispiel `pine` aufgezeigt werden:

- Passt das Patch-RPM zu meinem System?

Um dies zu prüfen, sollten Sie zunächst die installierte Version des Paketes abfragen. Im Fall von `pine` geht das mit dem Befehl

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

Als Nächstes wird das Patch-RPM untersucht, ob es zu genau dieser Version von `pine` passt:

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

Dieser Patch passt zu drei verschiedenen Versionen von `pine`. Auch die in unserem Fall installierte Version ist dabei enthalten, so dass der Patch eingespielt werden kann.

- Welche Dateien werden durch den Patch ersetzt?

Die von einem Patch betroffenen Dateien können leicht aus dem Patch-RPM ausgelesen werden. Der Parameter `-P` von `rpm` dient dazu, spezielle patch-relevanten Möglichkeiten auszuwählen. Demnach bekommt man die Liste der Dateien mit

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

oder, wenn der Patch bereits installiert ist, mit

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

- Wie spielt man ein Patch-RPM in das System ein?

Patch-RPMs werden wie normale RPMs verwendet. Der einzige Unterschied liegt darin, dass für sie ein passendes RPM bereits eingespielt sein muss.

- Welche Patches sind im System eingespielt und auf welchen Paketversionen haben sie aufgesetzt?

Eine Liste aller Patches, die im System eingespielt sind bekommen Sie mit dem Befehl `rpm -qPa`. Wenn, wie in unserem Beispiel, in einem neuen System erst ein Patch eingespielt ist, sieht das so aus:

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

Wenn Sie nach einiger Zeit wissen möchten, welche Paketversion denn zunächst eingespielt war, so ist dies ebenfalls noch in der RPM-Datenbank vorhanden. Sie bekommen diese Information für `pine` mit dem Kommando:

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

Weitere Informationen, auch zum Patch-Feature von RPM, finden Sie in dem Manualpages von `rpm` und `rpmbuild`.

2.3.4 Anfragen stellen

Mit der Option `-q query` leitet man Anfragen ein. Damit ist es möglich die RPM-Archive selbst zu durchleuchten (Option `-p` (*PaketDatei*)) als auch die RPM-Datenbank zu befragen. Die Art der angezeigten Information kann man mit den zusätzlichen Optionen auswählen; vgl. Tabelle 2.2.

Tabelle 2.2: Die wichtigsten Abfrageoptionen (-q [-p] paket)

<code>-i</code>	Paket-Informationen anzeigen
<code>-l</code>	Dateiliste des Pakets anzeigen

<code>-f <DATEI></code>	Anfrage nach Paket, das die Datei <code><DATEI></code> besitzt; <code><DATEI></code> muss mit vollem Pfad angegeben werden!
<code>-s</code>	Status der Dateien anzeigen (impliziert <code>-l</code>)
<code>-d</code>	Nur Dokumentationsdateien auflisten (impliziert <code>-l</code>)
<code>-c</code>	Nur Konfigurationsdateien auflisten (impliziert <code>-l</code>)
<code>--dump</code>	Alle überprüfbaren Infos zu jeder Datei anzeigen (mit <code>-l</code> , <code>-c</code> oder <code>-d</code> benutzen!)
<code>--provides</code>	Fähigkeiten des Pakets auflisten, die ein anderes Paket mit <code>--requires</code> anfordern kann
<code>--requires, -R</code>	Paket-Abhängigkeiten ausgeben
<code>--scripts</code>	Die diversen (De-)Installations-Skripten ausgeben

Der folgende Befehl gibt die Information in Ausgabe 2.2 aus:

```
rpm -q -i wget
```

Beispiel 2.2: rpm -q -i wget

```
Name       : wget                      Relocations: (not relocateable)
Version    : 1.8.2                    Vendor: SuSE Linux AG, Nuernberg, Germany
Release    : 301                      Build Date: Di 23 Sep 2003 20:26:38 CEST
Install date: Mi 08 Okt 2003 11:46:31 CEST Build Host: levi.suse.de
Group: Productivity/Networking/Web/Utilities Source RPM: wget-1.8.2-301.src.rpm
Size       : 1333235                  License: GPL
Signature  : DSA/SHA1, Di 23 Sep 2003 22:13:12 CEST, Key ID a84edae89c800aca
Packager   : http://www.suse.de/feedback
URL        : http://wget.sunsite.dk/
Summary    : A tool for mirroring FTP and HTTP servers
Description:
Wget enables you to retrieve WWW documents or FTP files from a server.
This can be done in script files or via the command line.
[...]
```

Die Option `-f` führt nur dann zum Ziel, wenn man den kompletten Dateinamen, einschließlich des Pfades, kennt. Sie können beliebig viele zu suchende Dateinamen angeben, zum Beispiel:

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

führt zu dem Ergebnis:

```
rpm-3.0.3-3
wget-1.5.3-55
```

Kennt man nur einen Teil des Dateinamens, so muss man sich mit einem Shell-Skript behelfen (vgl. Datei 2.3); der gesuchte Dateiname ist als Parameter beim Aufruf des Skripts zu übergeben.

Beispiel 2.3: Paket-Suchskript

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

Mit dem Befehl kann man sich gezielt die Auflistung der Informationen (Updates, Konfiguration, Änderungen etc.) zu einem bestimmten Paket anzeigen lassen; hier beispielsweise zu dem Paket rpm:

```
rpm -q --changelog rpm
```

Es werden allerdings nur die letzten 5 Einträge in der RPM-Datenbank angezeigt; im Paket selbst sind alle Einträge (der letzten 2 Jahre) enthalten. Diese Abfrage funktioniert, wenn CD 1 unter /cdrom eingehangen ist:

```
rpm -qp --changelog /cdrom/suse/i586/rpm-3*.rpm
```

Anhand der installierten Datenbank lassen sich auch Überprüfungen durchführen. Eingeleitet werden diese Vorgänge mit der Option `-v` (gleichbedeutend mit `-y` oder `--verify`). Damit veranlasst man rpm, all die Dateien anzuzeigen, die sich im Vergleich zur ursprünglichen Version, wie sie im Paket enthalten war, geändert haben. rpm stellt dem eigentlichen Dateinamen bis zu acht Zeichen voran, die auf folgende Änderungen hinweisen:

Tabelle 2.3: Die Überprüfungen

5	MD5-Prüfsumme
S	Dateigröße

L	Symbolischer Link
T	Modification Time
D	major und minor Gerätenummer <i>device number</i>
U	Benutzer <i>user</i>
G	Gruppe <i>group</i>
M	Modus (einschl. Rechte und Typus)

Bei Konfigurationsdateien wird zusätzlich ein *c* ausgegeben. Beispiel, falls etwas an */etc/wgetrc* aus dem *wget* geändert wurde:

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

Die Dateien der RPM-Datenbank liegen unter */var/lib/rpm/*.

Bei einer */usr/*-Partition von 1 GB kann die Datenbank durchaus 30 MB Plattenplatz beanspruchen; insbesondere nach einem kompletten Update. Falls die Datenbank über Gebühr groß erscheint, ist es meist hilfreich, mit der Option *--rebuilddb* eine neue Datenbank auf Basis der existierenden zu erstellen. Es ist sinnvoll, vor einem solchen Rebuild eine Kopie der existierenden Datenbank aufzubewahren.

Weiterhin legt das *cron*-Skript *cron.daily* täglich gepackte Kopien der Datenbank unter */var/adm/backup/rpmdb* an, deren Anzahl durch die Variable *MAX_RPMDB_BACKUPS* (Standard: 5) in der */etc/sysconfig/backup* vorgegeben wird; es ist mit bis zu 3 MB pro Backup bei einem 1 GB großen */usr* Verzeichnis rechnen.

2.3.5 Quellpakete installieren und kompilieren

Alle Quellpakete haben die Erweiterung *.src.rpm* hinter dem eigentlichen Paketnamen; diese Dateien sind die „Source-RPMs“.

Hinweis

Diese Pakete können mit YaST – wie jedes andere Paket – installiert werden, allerdings werden Quellpakete nie als installiert (*[i]*) markiert wie die regulären anderen Pakete. Dies liegt daran, dass die Quellpakete nicht in die RPM-Datenbank aufgenommen werden; in der RPM-Datenbank nämlich erscheint nur *installierte* Betriebssoftware.

Hinweis

Die Arbeitsverzeichnisse für `rpm` bzw. `rpmbuild` unter `/usr/src/packages` müssen vorhanden sein (falls keine eigenen Einstellungen wie etwa via `/etc/rpmrc` vorgenommen wurden):

SOURCES für die originalen Quellen (`.tar.gz`-Dateien etc.) und für die distributionsspezifischen Anpassungen (`.dif`-Dateien).

SPECS für die `.spec`-Dateien, die in der Art eines Meta-Makefiles den build-Prozess steuern.

BUILD unterhalb dieses Verzeichnisses werden die Quellen entpackt, gepatcht und kompiliert.

RPMS hier werden die fertigen Binary-Pakete abgelegt.

SRPMS und hier die Source-RPMs.

Wenn Sie ein Quellpaket mit YaST installieren, dann werden die für den build-Prozess notwendigen Komponenten unter `/usr/src/packages` installiert: die Quellen und die Anpassungen unter **SOURCES** und die dazugehörige `.spec`-Datei unter **SPECS**.

Hinweis

Bitte machen Sie keine RPM-Experimente mit wichtigen System-Komponenten (`glibc`, `rpm`, `sysvinit` etc.), Sie setzen damit die Funktionstüchtigkeit Ihres Systems aufs Spiel.

Hinweis

Im Folgenden wird das Paket `wget.src.rpm` betrachtet. Nachdem das Quellpaket `wget.src.rpm` mit YaST installiert wurde, gibt es die Dateien:

```
/usr/src/packages/SPECS/wget.spec
/usr/src/packages/SOURCES/wget-1.4.5.dif
/usr/src/packages/SOURCES/wget-1.4.5.tar.gz
```

Mit `rpmbuild -b X /usr/src/packages/SPECS/wget.spec` wird der Kompiliervorgang angestoßen; dabei kann `X` für verschiedene Stufen stehen (vgl. die `--help`-Ausgabe bzw. die RPM-Dokumentation); hier nur eine kurze Erläuterung:

-bp Quellen im Verzeichnis `/usr/src/packages/BUILD` präparieren: entpacken und patchen

- bc** wie -bp, jedoch zusätzlich noch kompilieren
- bi** wie -bc, jedoch zusätzlich noch installieren; Achtung, wenn ein Paket nicht das BuildRoot-Feature unterstützt, ist es möglich, dass Sie sich während dieses Installationsvorgangs wichtige Konfigurationsdateien überschreiben!
- bb** wie -bi, jedoch zusätzlich noch das sog. Binary-RPM herstellen; bei Erfolg liegt es in `/usr/src/packages/RPMS`.
- ba** wie -bb, jedoch zusätzlich noch das sog. Source-RPM herstellen; bei Erfolg liegt es in `/usr/src/packages/SRPMS`.
- short-circuit** Mit dieser Option lassen sich einzelne Schritte überspringen.

Das erzeugte Binary-RPM ist schließlich mit `rpm -i` oder besser mit `rpm -U` zu installieren.

2.3.6 RPM-Pakete mit build erzeugen

Bei vielen Paketen besteht die Gefahr, dass während ihrer Erstellung ungewollt Dateien in das laufende System kopiert werden. Um dies zu vermeiden, können Sie das `build` verwenden, das eine definierte Umgebung herstellt, in der das Paket gebaut wird. Zum Aufbau dieser chroot-Umgebung muss dem `build` Skript ein kompletter Paketbaum zur Verfügung stehen. Dieser kann auf Festplatte, über NFS oder auch von DVD bereitgestellt werden. Dem Skript teilt man die entsprechende Stelle mit dem Befehl `build --rpms <Pfad>` mit. Im Unterschied zu `rpm` möchte der Befehl `build` das SPEC-File im gleichen Verzeichnis haben, wie die eigentlichen Quellen. Wenn Sie wie im obigen Beispiel `wget` neu übersetzen möchten, und die DVD unter `/media/dvd` in das System eingehängt ist, verwenden Sie als Benutzer `root` folgende Befehle:

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

Daraufhin wird unter `/var/tmp/build-root` eine minimale Umgebung aufgebaut, in der das Paket gebaut wird. Die entstandenen Pakete liegen danach in `/var/tmp/build-root/usr/src/packages/RPMS`

Das `build` Skript stellt noch einige weitere Optionen zur Verfügung. So kann man eigene RPMs bevorzugt verwenden lassen, die Initialisierung

der Build-Umgebung auslassen oder den `rpm`-Befehl auf eine der bereits beschriebenen Stufen beschränken. Sie erhalten mehr Informationen mit dem Befehl `build --help` und in der `lrefman[1]build`.

2.3.7 Tools für RPM-Archive und die RPM-Datenbank

Der Midnight Commander (`mc`) kann den Inhalt eines RPM-Archivs anzeigen bzw. Teile daraus zu kopieren. Er bildet ein solches Archiv als ein virtuelles Dateisystem ab, so dass alle gewohnten Menüpunkte des Midnight Commander – wenn sinnvoll – zur Verfügung stehen: Die Kopfzeilen-Informationen der Datei `HEADER` kann man sich mit `(F3)` ansehen; mit den Cursor-Tasten und `(Enter)` lässt sich durch die Struktur des Archivs browsen, um bei Bedarf mit `(F5)` Komponenten herauszukopieren. – Übrigens, mittlerweile gibt es auch für den Emacs ein `rpm.el`, ein Frontend für `rpm`.

KDE enthält das Tool `kpackage`, bei GNOME finden Sie `gnorpm`.

Mit `Alien` (`alien`) ist es möglich, die Paketformate der verschiedenen Distributionen zu konvertieren. So kann man versuchen, alte TGZ-Archive vor dem Installieren nach RPM umzuwandeln, damit *während* der Installation die RPM-Datenbank mit den Paket-Informationen versorgt wird. Aber Achtung: `alien` ist ein Perl-Skript und befindet sich nach Angaben der Programm-Autoren noch in einem Alpha-Stadium – wenngleich es bereits eine hohe Versionsnummer erreicht hat.

Teil II

Konfiguration

YaST im Textmodus (ncurses)

Dieses Kapitel richtet sich v. a. an Systemadministratoren und Experten, auf deren Rechner kein X-Server läuft und die auf das textbasierte Installationswerkzeug angewiesen sind.

Sie erhalten in diesem Kapitel grundlegende Informationen zum Aufruf und zur Bedienung von YaST im Textmodus (ncurses). Zudem erfahren Sie, wie Sie Ihr System automatisch online aktualisieren können und so immer auf dem neuesten Stand halten.

3.1	Bedienung	74
3.2	Einschränkung der Tastenkombinationen	76
3.3	Aufruf der einzelnen Module	77
3.4	Das YaST Online Update	77

3.1 Bedienung

Mit den Tasten **Tab**, **Alt-Tab**, **Leertaste**, verschiedenen Pfeiltasten (**↑** und **↓**) und **Enter** sowie mit den Shortcuts lässt sich im Prinzip das ganze Programm bedienen.

3.1.1 Das YaST-Kontrollzentrum

Wenn Sie YaST im Textmodus starten, erscheint zuerst das YaST-Kontrollzentrum (s. Abb. 3.1).



Abbildung 3.1: Das Hauptfenster von YaST-ncurses

Sie sehen hier drei Bereiche: In der linken Fensterhälfte, von einem breiten weißen Rahmen umgeben, sind die Kategorien dargestellt, denen die einzelnen Module untergeordnet sind. Die aktive Kategorie ist durch farbige Hinterlegung gekennzeichnet. In der rechten Hälfte sehen Sie, von einem dünnen weißen Rahmen umgeben, einen Überblick über die Module, die in der aktiven Kategorie enthalten sind. Im unteren Fensterbereich liegen die Buttons für 'Hilfe' und 'Verlassen'.

Nach dem ersten Start des YaST-Kontrollzentrums ist automatisch die Kategorie 'Software' selektiert. Die Kategorie wechseln Sie mit den Tasten **↓** und **↑**. Zum Start eines Moduls aus der selektierten Kategorie betätigen Sie die Taste **→**. Die Modulauswahl erscheint jetzt mit breiter Umrandung. Selektieren Sie das gewünschte Modul über die Tasten **↓** und **↑**. Durch andauerndes Drücken der Pfeiltasten „scrollen“ Sie durch die Übersicht der verfügbaren Module. Sobald ein Modul selektiert wurde, erscheint der

Modultitel farblich hinterlegt. Gleichzeitig wird im unteren Fensterbereich eine kurze Modulbeschreibung eingeblendet.

Über die **(Enter)** Taste starten Sie das gewünschte Modul. Verschiedene Buttons oder Auswahlfelder im Modul enthalten einen andersfarbigen (bei Standardeinstellungen gelben) Buchstaben. Mit der Kombination **(Alt)-(gelberBuchstabe)** können Sie den jeweiligen Button ohne umständliche **(Tab)**-Navigation direkt anwählen.

Das YaST-Kontrollzentrum verlassen Sie, indem Sie den Button 'Verlassen' betätigen oder indem Sie den Unterpunkt 'Verlassen' in der Kategorieübersicht selektieren und **(Enter)** drücken.

3.1.2 Die YaST-Module

Bei der folgenden Beschreibung der Bedienelemente innerhalb der YaST-Module wird davon ausgegangen, dass sämtliche Funktionstasten und **(Alt)**-Tastenkombinationen funktionieren und nicht systemweit anders belegt wurden. Zu möglichen Ausnahmen lesen Sie bitte Abschnitt 3.2 auf der nächsten Seite.

Navigation zwischen Buttons/Auswahllisten

Mit **(Tab)** und **(Alt)-(Tab)** oder **(Shift)-(Tab)** navigieren Sie jeweils zwischen den Buttons und/oder den Rahmen von Auswahllisten hin und her.

Navigation in Auswahllisten In einem aktivierten Rahmen, in dem sich eine Auswahlliste befindet, springen Sie immer mit den Pfeiltasten (**(↑)** und **(↓)**) zwischen den einzelnen Elementen, zum Beispiel zwischen den einzelnen Modulen einer Modulgruppe im Kontrollzentrum. Sollten einzelne Einträge innerhalb eines Rahmens über dessen Breite herausragen, „scrollen“ Sie mit **(Shift)-(→)** bzw. **(Shift)-(←)** horizontal nach rechts und links (alternativ funktioniert auch **(Strg)-(e)** bzw. **(Strg)-(a)**). Diese Kombination funktioniert auch dort, wo ein bloßes **(→)** oder **(←)** wie im Kontrollzentrum einen Wechsel des aktiven Rahmens bzw. der aktuellen Auswahlliste zur Folge hätte.

Buttons, Radiobuttons und Checkboxes

Die Auswahl von Buttons mit einer leeren eckigen Klammer (Checkbox) oder leerer runder Klammer (Radiobuttons) erfolgt mit **(Leerfaste)** oder **(Enter)**. Alternativ lassen sich Radiobuttons und Checkboxes wie normale Buttons gezielt über **(Alt)-(gelberBuchstabe)** anwählen. In diesem Fall entfällt die separate Bestätigung mit **(Enter)**. Per Tab-Navigation ist ein separates **(Enter)** notwendig, damit die ausgewählte

Aktion ausgeführt oder der entsprechende Menüpunkt aktiv wird (vgl. Abb. 3.2).

Die Funktionstasten Die F-Tasten (F1) bis (F12) sind ebenfalls mit Funktionen belegt. Sie dienen zur schnellen Ansprache der verschiedenen Buttons, die zur Verfügung stehen. Welche F-Tasten mit Funktionen belegt sind, hängt davon ab, in welchem Modul Sie sich im YaST befinden, da in verschiedenen Modulen verschiedene Buttons angeboten sind (z.B. Details, Infos, Hinzufügen, Löschen ...). Für Freunde des alten YaST1 liegen z.B. die Buttons 'OK', 'Weiter' und 'Beenden' auf der Taste (F10). In der Hilfe zu YaST, die Sie mit (F1) erhalten, erfahren Sie die Funktionen hinter den einzelnen F-Tasten.

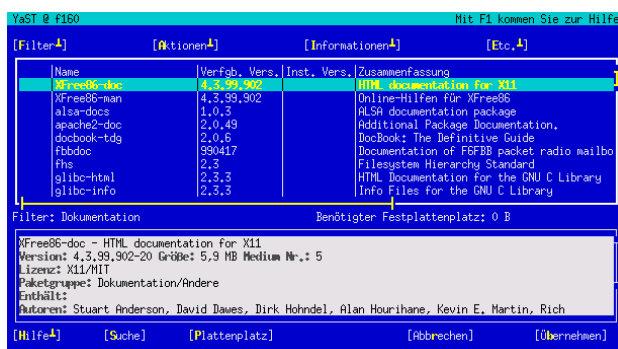


Abbildung 3.2: Das Modul zur Softwareinstallation

3.2 Einschränkung der Tastenkombinationen

Sollten auf Ihrem System bei laufendem X-Server systemweite (Alt)-Tastenkombinationen bestehen, kann es sein, dass die (Alt)-Kombinationen im YaST nicht funktionieren. Des Weiteren können Tasten wie (Alt) oder (Shift) durch Einstellungen des benutzten Terminals vorbelegt sein.

Ersatz von (Alt) durch (Esc): Alt-Shortcuts können mit (Esc) anstatt (Alt) durchgeführt werden, zum Beispiel ersetzt (Esc)-(h) die Tastenkombination (Alt)-(h).

Ersatz von Vor- und Zurückspringen mittels **(Strg)-f und **(Strg)-b** :**

Falls **(Alt)-** und **(Shift)-**Kombinationen durch den Windowmanager oder das Terminal vorbelegt sind, können Sie hier alternativ die Kombinationen **(Strg)-f** (vorwärts) und **(Strg)-b** (zurück) verwenden.

Einschränkung von Funktionstasten:

Auch die F-Tasten sind mit Funktionen belegt. Auch hier können bestimmte F-Tasten durch die Wahl des Terminals vorbelegt sein und daher nicht für YaST zur Verfügung stehen. Auf einer reinen Textkonsole sollten allerdings die **(Alt)-**Tastenkombinationen und die F-Tasten stets in vollem Umfang verfügbar sein.

3.3 Aufruf der einzelnen Module

Zur Zeitersparnis lässt sich jedes der YaST-Module auch einzeln aufrufen. Gestartet werden die Module einfach mit dem Aufruf: `yast modulname`

Das Netzwerkmodul wird zum Beispiel über `yast lan` gestartet. Eine Liste aller Modulnamen, die auf Ihrem System zur Verfügung stehen, erhalten Sie entweder mit dem Aufruf `yast -l` oder über `yast --list`.

3.4 Das YaST Online Update

3.4.1 Das YOU-Modul

Das YaST Online Update (YOU) lässt sich wie jedes andere YaST-Modul als `root` von der Kommandozeile aus aufrufen:

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

`yast online_update` ruft das entsprechende Modul auf. Durch die optionale Angabe von `url` weisen Sie YOU einen Server (lokal oder im Internet) zu, von dem alle Informationen und Patches bezogen werden sollen. Wird diese Angabe nicht beim initialen Aufruf gemacht, wählen Sie den Server/das Verzeichnis über die YaST-Maske aus. Die Bedienung der Maske selbst ist analog der im *Benutzerhandbuch* beschriebenen Bedienung des grafischen YaST-Moduls. Wie bei seinem grafischen Pendant lässt sich auch hier ein per Cronjob automatisiertes Update über den Button 'Vollautomatisches Update konfigurieren' einrichten.

3.4.2 Online Update per Kommandozeile

Über das Kommandozeilentool `online_update` können Sie Ihr System vollautomatisch, z.B. aus Skripten heraus aktualisieren.

Im konkreten Fall möchten Sie, dass Ihr System regelmäßig zu einer bestimmten Zeit nach Updates auf einem bestimmten Server sucht, die Patches und Patchinformationen herunterlädt, aber noch nicht installiert. Zu einem späteren Zeitpunkt möchten Sie die Menge der Patches sichten und die zu installierenden Patches auswählen:

- Setzen Sie einen Cronjob auf, der folgendes Kommando ausführt:

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

`-u` leitet die Basis-URL des Verzeichnisbaums ein, aus dem die Patches bezogen werden sollen. Es werden die Protokolle `http`, `ftp`, `smb`, `nfs`, `cd`, `dvd` und `dir` unterstützt. Mit `-g` laden Sie die Patches zwar herunter in ein lokales Verzeichnis, installieren Sie aber noch nicht. Optional können Sie die Menge der Patches nach einer der drei Typangaben `security` (sicherheitsrelevante Updates), `recommended` (empfehlenswerte Updates) und `optional` (optionale Updates) filtern. Ohne Filterangabe würde `online_update` alle verfügbaren neuen Patches des Typs `security` und `recommended` herunterladen.

- Die heruntergeladenen Pakete können Sie anschließend entweder sofort installieren, oder die einzelnen Patches näher untersuchen. Die Patches speichert `online_update` im Pfad `/var/lib/YaST2/you/mnt/` ab. Rufen Sie, um die Patches abschließend zu installieren, folgenden Befehl auf:

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

Der Parameter `-u` übergibt die (lokale) URL, unter der die zu installierenden Patches zu finden sind. `-i` startet den Installationsvorgang.

- Möchten Sie die heruntergeladenen Patches vor der Installation sichten und evtl. einzelne verwerfen, rufen Sie die YOU-Maske auf:

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

YOU startet und nimmt als Quelle der Patches statt eines entfernten Verzeichnisses im Internet das lokale Verzeichnis mit den bereits heruntergeladenen Patches. Anschließend selektieren Sie die gewünschten Patches wie bei jeder normalen Installation mittels des Paket-Managers.

Weitere Informationen zu `online_update` erhalten Sie als Ausgabe des Kommandos `online_update -h`.

Das X Window System

Das X Window System (X11) ist der Quasi-Standard für grafische Benutzeroberflächen unter Unix. X11 ist zudem netzwerkbasiert, sodass Anwendungen, die auf einem Rechner gestartet wurden ihre Ausgabe auf einem anderen Rechner darstellen können, wenn beide miteinander vernetzt sind. Die Art des Netzes (LAN oder Internet) spielt hierbei keine Rolle.

Wir stellen Ihnen in diesem Kapitel Optimierungsmöglichkeiten für Ihre X Window System-Umgebung vor, geben Ihnen Hintergrundinformationen zum Umgang mit Fonts unter SUSE LINUX und gehen auf die OpenGL/3D-Konfiguration ein. Die YaST Modulbeschreibungen zur Konfiguration von Monitor, Grafikkarte, Maus und Tastatur finden Sie im *Benutzerhandbuch*.

4.1	Installation des X Window Systems optimieren . . .	82
4.2	Installation und Konfiguration von Fonts	88
4.3	Konfiguration von OpenGL/3D	94

X11 entstand als Gemeinschaftsproduktion von DECTM (Digital Equipment CorporationTM) und dem Projekt Athena am MITTM (Massachusetts Institute of TechnologyTM). Die erste Version (X11R1) wurde im September 1987 freigegeben. Seit Release 6 hat das X Consortium, Inc.TM, ab 1996 The Open GroupTM die Entwicklung des X Window System übernommen.

XFreeTM ist eine frei verfügbare Implementierung von X-Servern für PC-Unix-Systeme (vgl. <http://www.XFree86.org>). XFree wurde und wird auch weiterhin – verstreut über die ganze Welt – von Programmierern entwickelt, die sich 1992 zum XFree-Team zusammengeschlossen haben. Daraus entstand die 1994 gegründete Firma The XFree86 Project, Inc.TM, deren Ziel es ist, XFreeTM einer breiten Öffentlichkeit zur Verfügung zu stellen und sowohl forschend als auch entwickelnd an der Zukunft des X Window System mitzuarbeiten.

Um die zur Verfügung stehende Hardware (Maus, Grafikkarte, Monitor, Tastatur) optimal nutzen zu können, besteht die Möglichkeit, die Konfiguration manuell zu optimieren. Im Folgenden wird auf einige Aspekte der Optimierung eingegangen. Detaillierte Informationen zur Konfiguration des X Window System finden sich in verschiedenen Dateien im Verzeichnis `/usr/share/doc/packages/xf86` sowie natürlich in der Manpage `man XF86Config`.

Achtung

Bei der Konfiguration des X Window Systems sollte besonders sorgsam vorgegangen werden! Auf keinen Fall sollte X11 gestartet werden, bevor die Konfiguration abgeschlossen wurde. Ein falsch eingestelltes System kann zu irreparablen Schäden an der Hardware führen; besonders gefährdet sind Festfrequenz-Monitore. Die Autoren dieses Buches und die SUSE LINUX AG lehnen jede Verantwortung für eventuell entstehende Schäden ab. Der vorliegende Text wurde mit größtmöglicher Sorgfalt erstellt. Dennoch kann nicht garantiert werden, dass die hier vorgestellten Methoden korrekt sind und Ihrer Hardware keinen Schaden zufügen.

Achtung

4.1 Installation des X Window Systems optimieren

Im Folgenden soll der Aufbau der Konfigurationsdatei `/etc/X11/XF86Config` vorgestellt werden. Diese Datei ist in Abschnitte

(engl. *Sections*) aufgeteilt, die jeweils mit dem Schlüsselwort `Section` "bezeichner" eingeleitet werden und mit `EndSection` beendet werden. Es folgt ein grober Abriss der wichtigsten Abschnitte.

Die Programme `SaX2` und `xf86config` erstellen die Datei `XF86 Config`, standardmäßig in `/etc/X11`. Dies ist die primäre Konfigurationsdatei für das X Window System. Hier finden sich die gemachten Angaben zu Maus, Monitor und Grafikkarte.

`XF86Config` setzt sich aus mehreren Abschnitten zusammen (den sog. *Sections*), die sich mit jeweils einem Aspekt der Konfiguration beschäftigen. Eine *Section* hat stets die Form:

```
Section Abschnittsbezeichnung
eintrag 1
eintrag 2
eintrag n
EndSection
```

Es existieren folgende Typen von *Sections*:

Tabelle 4.1: Abschnitte (sog. *sections*) in `/etc/X11/XF86Config`

Typ	Bedeutung
<code>Files</code>	Dieser Abschnitt beschreibt die verwendeten Pfade für Zeichensätze und die RGB-Farbtabelle.
<code>ServerFlags</code>	Hier werden allgemeine Schalter angegeben.
<code>InputDevice</code>	Über diesen Abschnitt werden die Eingabegeräte konfiguriert. Es werden sowohl Tastaturen und Mäuse als auch spezielle Eingabegeräte (Touchtablett, Joysticks usw.) über diesen Abschnitt konfiguriert. Wichtige Bezeichner sind hier <code>Driver</code> und die Optionen, die <code>Protocol</code> und <code>Device</code> festlegen.
<code>Monitor</code>	Beschreibt den verwendeten Monitor. Elemente dieses Abschnittes sind ein Name, auf den später bei der Definition des Screens verwiesen wird, sowie die Beschreibung der Bandbreite (<code>Bandwidth</code>) und der zulässigen Synchronisationsfrequenzen (<code>HorizSync</code> und <code>VertRefresh</code>). Die Angaben erfolgen in MHz, kHz bzw. Hz. Grundsätzlich lehnt der Server jede Modeline ab, die nicht der Spezifikation des Monitors entspricht. Damit soll verhindert werden, dass durch Experimente an den Modelines versehentlich zu hohe Frequenzen an den Monitor geschickt werden.

Modes	Hier werden die Darstellungsparameter der einzelnen Bildschirmauflösungen festgelegt. Diese Parameter können von SaX2 aufgrund der vom Benutzer vorgegebenen Werte berechnet werden und müssen im Regelfall nicht verändert werden. Manuell eingreifen können Sie an dieser Stelle aber beispielsweise, wenn Sie einen Festfrequenzbildschirm anschließen möchten. Eine genaue Erläuterung der einzelnen Parameter würde den Rahmen dieses Buches sprengen, Sie finden allerdings eine detaillierte Erläuterung der Bedeutung der einzelnen Zahlenwerte in der HOWTO Datei <code>/usr/share/doc/howto/en/XFree86-Video-Timings-HOWTO.gz</code> .
Device	Dieser Abschnitt definiert eine bestimmte Grafikkarte. Diese wird durch den angegebenen Namen referenziert.
Screen	Diese Section schließlich fügt einen Monitor und ein Device zusammen und es ergeben sich daraus die notwendigen Angaben für XFree. Der Unterabschnitt <code>Display</code> erlaubt die Angabe der virtuellen Bildschirmgröße (<code>Virtual</code>), des <code>ViewPort</code> und der verwendeten Modes mit diesem Screen.
ServerLayout	Dieser Abschnitt legt das Layout einer Single- oder Multiheadkonfiguration fest. Hier werden die Eingabegeräte <code>InputDevice</code> und die Anzeigegeräte <code>Screen</code> zu einem Ganzen zusammengefasst.

Näher betrachtet werden die Sections `Monitor`, `Device` und `Screen`. In der Manualpage von `XFree86` und der Manualpage von `XF86Config` finden sich weitere Informationen zu den verbleibenden Sections.

In `XF86Config` können mehrere `Monitor`- und `Device`-Abschnitte vorkommen. Auch mehrere `Screen`-Abschnitte sind möglich; welcher davon verwendet wird, hängt dann vom nachfolgenden Abschnitt `ServerLayout` ab.

4.1.1 Screen-Section

Zunächst soll die `Screen`-Section näher betrachtet werden. Diese bringt eine `Monitor`- mit einer `Device`-Section zusammen und bestimmt, welche Auflösungen mit welcher Farbtiefe bereitgestellt werden sollen.

Eine Screen-Section kann beispielsweise wie in Datei 4.1 aussehen.

Beispiel 4.1: Die Screen-Section der Datei */etc/X11/XF86Config*

```
Section "Screen"
    DefaultDepth 16
    SubSection "Display"
        Depth 16
        Modes "1152x864" "1024x768" "800x600"
        Virtual 1152x864
    EndSubSection
    SubSection "Display"
        Depth 24
        Modes "1280x1024"
    EndSubSection
    SubSection "Display"
        Depth 32
        Modes "640x480"
    EndSubSection
    SubSection "Display"
        Depth 8
        Modes "1280x1024"
    EndSubSection
    Device "Device[0]"
    Identifier "Screen[0]"
    Monitor "Monitor[0]"
EndSection
```

Die Zeile `Identifier` (hier `Screen[0]`) gibt diesem Abschnitt eine eindeutige Bezeichnung, durch die er dann im darauf folgenden Abschnitt `ServerLayout` eindeutig referenziert werden kann. Über die Zeilen `Device` und `Monitor` werden dem `Screen` eindeutig die schon weiter oben in der Datei definierte Grafikkarte und der Monitor zugeordnet. Dies sind nichts weiter als Verweise auf die `Device`- und `Monitor`-Sections mit den entsprechenden Namen bzw. Identifiern. Auf diese Sections wird weiter unten noch näher eingegangen.

Mittels der `DefaultDepth`-Angabe kann ausgewählt werden, in welcher Farbtiefe der Server startet, wenn er ohne eine explizite Angabe der Farbtiefe gestartet wird. Es folgt für jede Farbtiefe eine `Display`-Subsection. Die Farbtiefe, für die die Subsection gilt, wird durch das Schlüsselwort `Depth` festgelegt. Mögliche Werte für `Depth` sind 8, 15, 16 und 24. Nicht alle X-Server-Module unterstützen jeden dieser Werte.

Nach der Farbtiefe wird mit `Modes` eine Liste von Auflösungen festgelegt. Diese Liste wird vom X-Server von links nach rechts durchlaufen. Für jede Auflösung wird in der `Modes`-Section in Abhängigkeit von der `Monitor`-Section eine passende `Modeline` gesucht, die vom Monitor und der Grafikkarte dargestellt werden kann.

Die erste in diesem Sinne passende Auflösung ist die, in der der X-Server startet (der sog. `Default-Mode`). Mit den Tasten `(Strg)-(Alt)-(Grau +)` kann in der Liste nach rechts, mit `(Strg)-(Alt)-(Grau -)` nach Links gewandert werden. So kann die Bildschirmauflösung zur Laufzeit des X Window Systems variiert werden.

Die letzte Zeile der Subsection `Display` mit `Depth 16` bezieht sich auf die Größe des virtuellen Bildschirms. Die maximal mögliche Größe des virtuellen Bildschirms hängt vom Speicherausbau der Videokarte und der gewünschten Farbtiefe ab, nicht aber von der maximalen Auflösung des Monitors. Da moderne Grafikkarten sehr viel Grafikspeicher anbieten, können Sie sehr große virtuelle Desktops anlegen. Beachten Sie dann aber bitte, dass Sie evtl. keine 3D-Funktionalität mehr nutzen können, wenn Sie praktisch den gesamten Grafikspeicher mit einem virtuellen Desktop füllen. Hat die Karte zum Beispiel 16 MB Video-RAM, so kann, bei 8 Bit Farbtiefe, der virtuelle Bildschirm bis zu 4096x4096(!) Pixel groß sein. Speziell bei den beschleunigten Servern empfiehlt es sich jedoch nachdrücklich, nicht den gesamten Speicher der Videokarte für den virtuellen Bildschirm zu verwenden, da der nicht verwendete Speicherbereich auf der Videokarte von diesen Servern für verschiedene Caches für Zeichensätze und Grafikbereiche verwendet wird.

4.1.2 Device-Section

Eine `Device`-Section beschreibt eine bestimmte Grafikkarte. Es können beliebig viele `Device`-Sections in `XF86Config` enthalten sein, solange sich ihr Name, der mit dem Schlüsselwort `Identifier` angegeben wird, unterscheidet. In der Regel werden – falls Sie mehrere Grafikkarten eingebaut haben – die Sections einfach durchnummeriert, die erste wird dann mit `Device[0]`, die zweite mit `Device[1]` bezeichnet usw.. In der folgenden Datei sehen Sie den Ausschnitt aus der `Device` Section eines Computers, in dem eine Matrox Millennium PCI Grafikkarte eingebaut ist:

```
Section "Device"
    BoardName      "MGA2064W"
    BusID          "0:19:0"
    Driver          "mga"
```

```
Identifier      "Device[0]"
VendorName      "Matrox"
Option          "sw_cursor"
EndSection
```

Wenn Sie SaX2 zur Konfiguration verwenden, dann dürfte die Device-Section ungefähr so wie oben abgebildet aussehen. Insbesondere Driver und BusID sind natürlich von der in Ihrem Computer eingebauten Hardware abhängig und werden von SaX2 automatisch bestimmt. Die BusID bestimmt den PCI- bzw. AGP-Steckplatz, in den die Grafikkarte eingesteckt ist. Diese stimmt mit der vom Kommando `lspci` ausgegebenen ID überein. Beachten Sie, dass der X-Server die Angaben in dezimaler, das Programm `lspci` hingegen in hexadezimaler Schreibweise ausgibt!

Über den Parameter `Driver` legen Sie den zu verwendenden Treiber für diese Grafikkarte fest. Im Falle der Matrox Millennium heißt das Treibermodul `mga`. Diese werden vom X-Server über den im Abschnitt `Files` definierten `ModulePath` im Unterverzeichnis `drivers` gesucht. In einer Standardinstallation ist dies das Verzeichnis `/usr/X11R6/lib/modules/drivers`. Hierzu wird an den Namen einfach `_drv.o` angehängt, im Falle des `mga` Treibers wird als die Treiberdatei `mga_drv.o` geladen.

Über zusätzliche Optionen kann das Verhalten des X-Servers bzw. des Treibers beeinflusst werden. In der Device Section ist hier exemplarisch die Option `sw_cursor` gesetzt worden. Dies deaktiviert den Hardwaremauscursor und stellt den Mauszeiger in Software dar. Je nach Treibermodul stehen ihnen verschiedene Optionen zur Verfügung, diese sind in den Beschreibungsdateien zu den Treibermodulen im Verzeichnis `/usr/X11R6/lib/X11/doc` zu finden. Allgemein gültige Optionen finden Sie auch in den Manpages (`man XF86Config` und `man XFree86`).

4.1.3 Monitor- und Modes-Section

Die Monitor-Sections und die Modes Section beschreiben, analog zu den Device-Sections, jeweils einen Monitor. Die Konfigurationsdatei `/etc/X11/XF86Config` kann wieder beliebig viele, unterschiedlich benannte Monitor-Sections enthalten. In der `ServerLayout`-Section wird dann festgelegt, welche Monitor-Section ausschlaggebend ist.

Für die Monitordefinition gilt, noch mehr als für die Beschreibung der Grafikkarte, dass das Erstellen einer Monitor-Section und insbesondere der Modes-Section nur von erfahrenen Benutzern gemacht werden sollte. Der

wesentliche Bestandteil der Modes-Section sind die sog. Modelines, in denen Horizontal- und Vertikal-Timings für die jeweilige Auflösung angegeben werden. In der Monitor-Section werden die Eigenschaften des Monitors, insbesondere die zulässigen Ablenkfrequenzen, festgehalten.

Achtung

Ohne ein grundlegendes Verständnis der Funktionsweise von Monitor und Grafikkarte sollte an den Modelines nichts verändert werden, da dies unter Umständen zur Zerstörung des Monitors führen kann!

Achtung

Diejenigen, die sich (zu)trauen, eigene Monitorbeschreibungen zu entwickeln, sollten mit der Dokumentation im Verzeichnis `/usr/X11/lib/X11/doc` vertraut sein. Besonders zu erwähnen ist [20], wo die Funktion der Hardware und das Erstellen von Modelines detailliert beschrieben wird. Eine deutsche Einführung in dieses Thema findet sich im XFree-Kapitel in [21].

Glücklicherweise ist mittlerweile die manuelle Erstellung von Modelines oder Monitordefinitionen fast nie mehr nötig. Wenn Sie einen modernen Multisync-Monitor verwenden, können die zulässigen Frequenzbereiche und optimalen Auflösungen in der Regel, wie im SaX2 Konfigurationsabschnitt erwähnt, direkt via DDC vom X-Server aus dem Monitor gelesen werden. Sollte dies nicht möglich sein, können Sie auch einen der eingebauten VESA-Modi des X-Servers verwenden. Diese sollten auf praktisch allen Grafikkarten/Monitorkombinationen einwandfrei funktionieren.

4.2 Installation und Konfiguration von Fonts

Das Installieren zusätzlicher Fonts unter SUSE LINUX ist sehr einfach. Es genügt die Fonts in ein beliebiges Verzeichnis zu kopieren, das sich im X11 Font-Pfad (siehe Abschnitt 4.2.1 auf Seite 93) befindet und, damit die Fonts auch über das neue Xft-Fontrendering-System benutzbar sind, auch ein Unterverzeichnis der in `/etc/fonts/fonts.conf` konfigurierten Verzeichnisse ist (siehe Abschnitt 4.2.1 auf der nächsten Seite).

Sie können die Fontdateien manuell als `root` in solch ein geeignetes Verzeichnis kopieren, zum Beispiel nach `/usr/X11R6/lib/X11/fonts/truetype/`, oder auch den KDE Fontinstaller im KDE Kontrollzentrum dazu benutzen. Das Ergebnis ist identisch.

Anstelle die Fonts tatsächlich zu kopieren, können Sie natürlich auch symbolische Links anlegen, wenn Sie zum Beispiel lizenzierte Fonts auf einer gemounteten Windows Partition haben und diese nutzen möchten. Anschließend rufen Sie `SuSEconfig --module fonts` auf.

`SuSEconfig --module fonts` ruft das Skript `/usr/sbin/fonts-config` auf, das die Konfiguration der Fonts übernimmt. Für Details was dieses Skript tut, lesen Sie bitte die zugehörige Manual-Page (`man fonts-config`).

Es spielt keine Rolle, welche Typen von Fonts installiert werden sollen, die Prozedur ist die gleiche für Bitmap-Fonts, TrueType/OpenType-Fonts und Type1-(PostScript)-Fonts. Alle diese Fontarten können in jedes beliebige Verzeichnis installiert werden. Lediglich CID-keyed Fonts sind ein Spezialfall, siehe Abschnitt 4.2.1 auf Seite 94.

4.2.1 Details zu Font-Systemen

XFree enthält zwei völlig verschiedene Font-Systeme, das alte *X11 Core-Font-System* und das völlig neu entworfene *Xft/fontconfig* System. Im Folgenden wird auf beide Systeme kurz eingegangen.

Xft

Beim Entwurf von Xft wurde von Anfang an darauf geachtet, dass es skalierbare Fonts, inclusive Antialiasing, gut unterstützt. Bei Benutzung von Xft werden die Fonts im Gegensatz zum X11 Core-Font-System von dem Programm gerendert, welches die Fonts benutzt und nicht vom X-Server. Dadurch bekommt das jeweilige Programm Zugriff auf die Fontdateien selbst und volle Kontrolle über Details, wie die Glyphen genau gerendert werden. Zum einen wird dadurch die korrekte Darstellung von Text in manchen Sprachen erst möglich, zum anderen ist der direkte Zugriff auf die Fontdateien sehr hilfreich, um Fonts zum Drucken zu einzubetten (engl. *to embed*) und so zu erreichen, dass der Ausdruck tatsächlich so aussieht wie die Bildschirmausgabe.

Die beiden Desktopumgebungen KDE und Gnome, Mozilla und viele andere Applikationen benutzen unter SUSE LINUX bereits standardmäßig Xft. Xft wird also bereits von erheblich mehr Applikationen benutzt als das alte X11 Core-Font-System.

Xft benutzt die Fontconfig-Bibliothek um Fonts zu finden und um die Art und Weise wie sie gerendert werden zu beeinflussen. Das Verhalten von fontconfig wird durch eine systemweite Konfigurationsdatei

/etc/fonts/fonts.conf und eine benutzerspezifische Konfigurationsdatei ~/.fonts.conf gesteuert. Jede dieser fontconfig Konfigurationsdateien muss mit

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
```

beginnen und mit

```
</fontconfig>
```

enden. Um Verzeichnisse, in denen nach Fonts gesucht wird hinzuzufügen, können Sie Zeilen wie die folgende

```
<dir>/usr/local/share/fonts/</dir>
```

hinzufügen. Das ist aber selten nötig. Das benutzerspezifische Verzeichnis ~/.fonts/ ist bereits per Default in /etc/fonts/fonts.conf eingetragen. Wenn ein Benutzer also für sich persönlich zusätzliche Fonts installieren möchte, genügt es, diese nach ~/.fonts zu kopieren.

Sie können auch Regeln einfügen, um das Aussehen der Fonts zu beeinflussen, zum Beispiel

```
<match target="font">
  <edit name="antialias" mode="assign">
    <bool>>false</bool>
  </edit>
</match>
```

um das Antialiasing für alle Fonts auszuschalten, oder

```
<match target="font">
  <test name="family">
    <string>Luxi Mono</string>
    <string>Luxi Sans</string>
  </test>
  <edit name="antialias" mode="assign">
    <bool>>false</bool>
  </edit>
</match>
```

wenn Sie es nur für bestimmte Fonts ausschalten möchten.

Die meisten Applikationen benutzen standardmäßig die Fontnamen `sans-serif` (oder das äquivalente `sans`), `serif` oder `monospace`. Dies sind keine wirklich existierenden Fonts sondern nur Aliases, die abhängig von der eingestellten Sprache auf einen geeigneten Font aufgelöst werden. Jeder Benutzer kann sich leicht Regeln zu seiner `~/.fonts.conf` hinzufügen um zu erreichen, dass diese Aliases auf seine Lieblingsfonts aufgelöst werden:

```
<alias>
  <family>sans-serif</family>
  <prefer>
    <family>FreeSans</family>
  </prefer>
</alias>
<alias>
  <family>serif</family>
  <prefer>
    <family>FreeSerif</family>
  </prefer>
</alias>
<alias>
  <family>monospace</family>
  <prefer>
    <family>FreeMono</family>
  </prefer>
</alias>
```

Weil fast alle Applikationen diese Aliase standardmäßig verwenden, wirkt das fast für das ganze System. Sie bekommen so mit sehr geringem Aufwand Ihre Lieblingsfonts fast überall, ohne in jedem Program einzeln die Fonteneinstellungen ändern zu müssen.

Um festzustellen, welche Fonts überhaupt installiert und verfügbar sind, gibt es das Kommando `fc-list`.

`fc-list ""` gibt zum Beispiel eine Liste aller Fonts aus. Möchten Sie wissen, welche skalierbaren Fonts (`:outline=true`) verfügbar sind, die alle für Hebräisch benötigen Glyphen enthalten (`:lang=he`), und sich für alle diese Fonts den Fontnamen (`family`), den Stil (`style`), den Fettheitsgrad (`weight`) und den Dateinamen, der den Font enthält ausgeben lassen, können Sie zum Beispiel folgendes Kommando benutzen:

```
fc-list ":lang=he:outline=true" family style weight file
```

Die Ausgabe dieses Kommandos könnte zum Beispiel so aussehen:

```
/usr/X11R6/lib/X11/fonts/truetype/FreeSansBold.ttf: FreeSans:style=Bold:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeMonoBoldOblique.ttf: FreeMono:style=BoldOblique:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeSerif.ttf: FreeSerif:style=Medium:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSerifBoldItalic.ttf: FreeSerif:style=BoldItalic:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeSansOblique.ttf: FreeSans:style=Oblique:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSerifItalic.ttf: FreeSerif:style=Italic:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeMonoOblique.ttf: FreeMono:style=Oblique:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeMono.ttf: FreeMono:style=Medium:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSans.ttf: FreeSans:style=Medium:weight=80
/usr/X11R6/lib/X11/fonts/truetype/FreeSerifBold.ttf: FreeSerif:style=Bold:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeSansBoldOblique.ttf: FreeSans:style=BoldOblique:weight=200
/usr/X11R6/lib/X11/fonts/truetype/FreeMonoBold.ttf: FreeMono:style=Bold:weight=200
```

Die wichtigsten Parameter, die mit `fc-list` abgefragt und ausgegeben werden können, sind:

Tabelle 4.2: Mögliche Parameter von `fc-list`

Parameter	Bedeutung und mögliche Werte
family	Der Name der Fontfamilie, zum Beispiel FreeSans
foundry	Der Fonthersteller, zum Beispiel urw
style	Der Fontstil, zum Beispiel Medium, Regular, Bold, Italic, Heavy, ...
lang	Die Sprache(n), die der Font unterstützt. Zum Beispiel de für Deutsch, ja für Japanisch, zh-TW für traditionelles Chinesisch, zh-CN für vereinfachtes Chinesisch ...
weight	Der <i>Fettheitsgrad</i> , zum Beispiel 80 für nicht fett, 200 für fett.
slant	Der <i>Kursivitätsgrad</i> , meist 0 für nicht kursiv, 100 für kursiv.
file	Der Dateiname unter dem der Font gespeichert ist.
outline	true wenn es sich um einen Outline-Font handelt, sonst false.
scalable	true wenn es sich um einen skalierbaren Font handelt, sonst false.
bitmap	true wenn es sich um einen Bitmap-Font handelt, sonst false.
pixelsize	Die Größe des Fonts in Pixel. Im Zusammenhang mit <code>fc-list</code> nur sinnvoll für Bitmap-Fonts.

X11 Core-Fonts

Heutzutage unterstützt auch das X11 Core-Font-System nicht nur Bitmap-Fonts, sondern auch skalierbare Fonts wie Type1-Fonts, TrueType/OpenType-Fonts und auch CID-keyed Fonts. Auch Unicode-Fonts werden bereits seit längerer Zeit unterstützt.

Ursprünglich wurde wurde das X11 Core-Font-System 1987 für X11R1 entwickelt um monochrome Bitmap-Fonts zu verarbeiten und man merkt bis heute, dass alle oben erwähnten Erweiterungen nachträglich hinzugefügt wurden.

Zum Beispiel werden skalierbare Fonts nur ohne Antialiasing und Subpixel-Rendering unterstützt und das Laden großer, skalierbarer Fonts mit Glyphen für viele Sprachen kann sehr langsam sein. Auch die Benutzung von Unicode Fonts kann langsam sein und verschwendet erheblich mehr Speicher als notwendig.

Es gibt viele andere grundlegende Schwächen des X11 Core-Font-Systems und man kann wahrscheinlich sagen, dass es hoffnungslos veraltet und nicht mehr sinnvoll erweiterbar ist. Es muss aus Gründen der Rückwärtskompatibilität verfügbar bleiben, aber soweit wie möglich sollte man das modernere Xft/fontconfig System verwenden.

Beachten Sie bitte, dass nur Verzeichnisse vom X-Server beachtet werden, die

- im Abschnitt `Files` der Datei `/etc/X11/XF86Config` als `FontPath` eingetragen sind.
- eine gültige `font.dir` Datei besitzen (wird von `SuSEconfig` generiert).
- nicht zur Laufzeit des X-Servers mit Hilfe des Kommandos `xset -fp` abgemeldet wurden.
- bzw. zur Laufzeit des X-Servers mit Hilfe des Kommandes `xset +fp` eingebunden wurden.

Wenn der X-Server bereits läuft, können neu installierte Fonts in bereits eingebundenen Verzeichnisse mit dem Kommando `xset fp rehash` zur Verfügung gestellt werden. Dieses Kommando wird von `SuSEconfig --module fonts` bereits aufgerufen.

Da das Kommando `xset` Zugriff auf den laufenden X-Server benötigt, kann das allerdings nur funktionieren, wenn `SuSEconfig --module`

`fonts` aus einer Shell gestartet wurde, die Zugriff auf den laufenden X-Server hat. Am einfachsten erreichen Sie dies, indem Sie sich durch Eingeben des Kommandos `sux` und anschließende Eingabe des Root-Passwortes in einem Terminal zu `root` machen, `sux` übergibt die Zugriffsrechte des Benutzers, der den X-Server gestartet hat, an die Rootshell.

Zum Testen, ob die Fonts richtig installiert wurden und tatsächlich über das X11 Core-Font-System verfügbar sind, können Sie das Kommando `xlsfonts` verwenden, das alle verfügbaren Fonts auflistet.

SUSE LINUX verwendet standardmäßig UTF-8 Locales, daher sollten Sie in der Regel Unicode-Fonts verwenden, die Sie daran erkennen, dass der von `xlsfonts` gelistete Fontname mit `iso10646-1` endet. Alle verfügbaren Unicode-Fonts können Sie sich also mit `xlsfonts | grep iso10646-1` anzeigen lassen.

Fast alle unter SUSE LINUX verfügbaren Unicode-Fonts enthalten mindestens alle nötigen Glyphen für die europäischen Sprachen, für die früher die Encodings `iso-8859-*` verwendet wurden.

CID-keyed Fonts

Im Gegensatz zu den anderen Fonttypen ist es bei CID-keyed Fonts nicht egal, in welches Verzeichnis sie installiert werden. Sie sollten auf jeden Fall nach `/usr/share/ghostscript/Resource/CIDFont/` installiert werden. Für Xft/fontconfig spielt das zwar keine Rolle, aber Ghostscript und das X11 Core-Font-System erfordern dies.

Hinweis

Weitere Informationen zum Thema Fonts unter X11 erhalten Sie unter <http://www.xfree86.org/current/fonts.html>.

Hinweis

4.3 Konfiguration von OpenGL/3D

Als 3D-Schnittstelle steht unter Linux die OpenGL-Schnittstelle zur Verfügung. Direct3D von Microsoft ist unter Linux nicht verfügbar.

4.3.1 Hardwareunterstützung

SUSE LINUX beinhaltet für die 3D-Hardwareunterstützung diverse OpenGL-Treiber. Eine Übersicht finden Sie in der Tabelle 4.3 auf der nächsten Seite.

Tabelle 4.3: Unterstützte 3D-Hardware

OpenGL Treiber	Unterstützte Hardware
nVidia	nVidia Chips: alle außer Riva 128(ZX)
DRI	3Dfx Voodoo Banshee, 3Dfx Voodoo-3/4/5, Intel i810/i815/i830M, Intel 845G/852GM/855GM/865G, Matrox G200/G400/G450/G550, ATI Rage 128(Pro)/Radeon

Bei einer Neuinstallation mit YaST kann bereits während der Installation die 3D-Unterstützung aktiviert werden, wenn eine entsprechende Unterstützung von YaST erkannt wird. Bei Grafikchips von nVidia muss vorher noch der nvidia-Treiber eingespielt werden. Wählen Sie dazu bitte während der Installation den nVidia-Treiber Patch in YOU (YaST Online Update) an. Aus Lizenzgründen können wir den nVidia-Treiber leider nicht mitliefern.

Sollte ein Update eingespielt worden sein, muss der 3D-Hardwaresupport anderweitig eingerichtet werden. Die Vorgehensweise hängt dabei vom zu verwendenden OpenGL-Treiber ab und wird im folgenden Abschnitt genauer erklärt.

4.3.2 OpenGL-Treiber

nVidia und DRI

Diese OpenGL-Treiber können sehr komfortabel mit SaX2 eingerichtet werden. Beachten Sie bitte, dass bei nVidia-Karten vorher noch der nVidia-Treiber eingespielt werden muss (s.o.). Mit dem Kommando `3Ddiag` können Sie überprüfen, ob die Konfiguration für nVidia bzw. DRI korrekt ist.

Aus Sicherheitsgründen dürfen nur die Benutzer der Gruppe `video` auf die 3D-Hardware zugreifen. Stellen Sie deshalb sicher, dass alle Benutzer, die auf der Maschine lokal arbeiten, in der Gruppe `video` eingetragen sind. Ansonsten wird für OpenGL-Programme der sehr langsame *Software Rendering Fallback* des OpenGL-Treibers verwendet. Mit dem Kommando `id` können Sie überprüfen, ob der aktuelle Benutzer der Gruppe `video` angehört. Ist dies nicht der Fall, kann er mittels YaST zu dieser Gruppe hinzugefügt werden.

4.3.3 Diagnose-Tool 3Ddiag

Um die 3D-Konfiguration unter SUSE LINUX überprüfen zu können, steht das Diagnosetool 3Ddiag zur Verfügung. Beachten Sie bitte, dass es sich dabei um ein Kommandozeilentool handelt, das Sie in einem Terminal aufrufen müssen.

Das Programm überprüft dabei beispielsweise die XFree-Konfiguration, ob die entsprechenden Pakete für 3D-Support installiert sind und ob die korrekte OpenGL-Bibliothek sowie GLX Extension verwendet wird. Befolgen Sie bitte die Anweisungen von 3Ddiag, wenn es zu "failed" Meldungen kommt. Im Erfolgsfall werden ausschließlich "done" Meldungen auf dem Bildschirm ausgegeben.

Mit 3Ddiag -h lassen sich zulässige Optionen für 3Ddiag ermitteln.

4.3.4 OpenGL-Testprogramme

Als OpenGL-Testprogramme eignen sich neben glxgears Spiele wie tuxracer und armagetron (gleichnamige Pakete). Bei aktiviertem 3D-Support sollten sich diese auf einem halbwegs aktuellen Rechner flüssig spielen lassen. Ohne 3D-Support ist dies nicht möglich bzw. nicht zumutbar (Diashow-Effekt). Eine zuverlässige Aussage darüber, ob 3D aktiviert ist, liefert die Ausgabe von `glxinfo`. `direct rendering` muss hier auf `Yes` stehen.

4.3.5 Troubleshooting

Sollte sich der OpenGL 3D-Test als negativ herausstellen (kein flüssiges Spielen möglich), sollte erst mit 3Ddiag überprüft werden, ob keine Fehlkonfiguration vorliegt (failed Meldungen) und diese ggf. behoben werden. Hilft auch das nicht oder lagen keine failed Meldungen vor, hilft oft nur noch ein Blick in die Logdateien von XFree86. Oft findet man hier in `/var/log/XFree86.0.log` von XFree86 die Zeile `DRI is disabled`. Dafür kann es mehrere Ursachen geben, die sich jedoch nur mit genauem Studium der Logdatei finden lassen, womit der Laie in aller Regel überfordert ist.

In diesen Fällen liegt in der Regel kein Konfigurationsfehler vor, da dieser bereits von 3Ddiag erkannt worden wäre. Somit bleibt ohnehin nur der Software Rendering Fallback des DRI Treibers, der jedoch keinerlei 3D-Hardware-Support bietet. Man sollte ebenfalls auf die Verwendung von 3D-Support verzichten, wenn sich OpenGL Darstellungsfehler oder gar

Stabilitätsprobleme ergeben. Verwenden Sie SaX2 um den 3D-Support zu deaktivieren.

4.3.6 Installationssupport

Abgesehen von Software Rendering Fallback des DRI Treibers befinden sich unter Linux alle OpenGL-Treiber im Entwicklungsstadium und sind deshalb zum Teil noch als experimentell anzusehen. Wir haben uns dennoch entschlossen, die Treiber auf der Distribution mitzuliefern, da die Nachfrage nach 3D-Hardwarebeschleunigung unter Linux sehr groß ist. Aufgrund des z.T. experimentellen Stadiums der OpenGL-Treiber können wir im Rahmen des Installationssupports jedoch nicht auf das Einrichten von 3D-Hardwarebeschleunigung eingehen und bei diesbezüglichen Problemen nicht weiterhelfen. Das grundlegende Einrichten der grafischen Benutzeroberfläche X11 beinhaltet also keinesfalls auch das Einrichten von 3D-Hardwarebeschleunigung. Wir hoffen jedoch, dass dieses Kapitel viele Fragen zu diesem Thema beantwortet. Bei Problemen mit dem 3D-Hardwaresupport empfehlen wir Ihnen, im Zweifelsfall auf 3D-Support zu verzichten.

4.3.7 Weiterführende Online-Dokumentation

- DRI: `/usr/X11R6/lib/X11/doc/README.DRI` (XFree86-doc)

Druckerbetrieb

In diesem Kapitel wird Standardwissen zum Druckerbetrieb geliefert. Es dient insbesondere auch dazu, geeignete Problemlösungen für den Druckerbetrieb in Netzwerken zu finden.

5.1	Grundlagen des Druckens	100
5.2	Voraussetzungen zum Drucken	105
5.3	Drucker einrichten mit YaST	109
5.4	Konfiguration für Anwendungsprogramme	115
5.5	Das CUPS-Drucksystem	116
5.6	CUPS Schnelleinstieg	123
5.7	Drucken aus Anwendungsprogrammen	139
5.8	Kommandozeilentools für das CUPS-Drucksystem	139
5.9	Drucken im TCP/IP-Netzwerk	144

5.1 Grundlagen des Druckens

Unter Linux werden Drucker über *Druckerwarteschlangen* angesprochen. Die zu druckenden Daten werden dabei in einer Druckerwarteschlange zwischengespeichert und durch den Druckerspooler nacheinander zum Drucker geschickt.

Meist liegen die zu druckenden Daten nicht in der Form vor, die direkt an den Drucker geschickt werden könnte. Eine Grafik beispielsweise muss normalerweise vorher in ein Format umgewandelt werden, das der Drucker direkt ausgeben kann. Die Umwandlung in die *Druckersprache* erfolgt durch den *Druckerfilter*, der vom Druckerspooler zwischengeschaltet wird, um die zu druckenden Daten ggf. so umzuwandeln, dass sie der Drucker direkt ausgeben kann.

5.1.1 Beispiele für Standarddruckersprachen

ASCII-Text Die meisten Drucker können wenigstens ASCII-Text direkt ausgeben. Die wenigen Ausnahmen, die keinen ASCII-Text direkt drucken können, werden über eine der folgenden Standarddruckersprachen angesprochen.

PostScript *PostScript* ist die Standardsprache unter Unix/Linux, in der Druckausgaben erstellt werden, die dann auf PostScript-Druckern direkt ausgegeben werden können. Diese Drucker sind relativ teuer, da PostScript eine mächtige aber komplexe Sprache ist, die im PostScript-Drucker einen hohen Rechenaufwand erfordert, wenn es zu Papier gebracht werden soll. Außerdem entstehen durch das Lizenzieren zusätzliche Kosten.

PCL3, PCL4, PCL5e, PCL6, ESC/P , ESC/P2 und ESC/P-Raster

Wenn kein PostScript-Drucker angeschlossen ist, verwendet der Druckerfilter das Programm Ghostscript, um die Daten in eine dieser anderen Standarddruckersprachen umzuwandeln. Dabei wird ein möglichst gut zu dem jeweiligen Druckermodell passender Ghostscript-Treiber verwendet, um modellspezifische Besonderheiten (z. B. Farbeinstellungen) berücksichtigen zu können.

5.1.2 Ablauf des Druckauftrages

1. Der Anwender oder ein Anwendungsprogramm erzeugt einen neuen Druckauftrag.

2. Die zu druckenden Daten werden in der Druckerwarteschlange zwischengespeichert, von wo sie der Druckerspooler an den Druckerfilter weiterleitet.
3. Der Druckerfilter macht nun normalerweise Folgendes:
 - (a) Der Typ der zu druckenden Daten wird bestimmt.
 - (b) Wenn die zu druckenden Daten nicht PostScript sind, werden sie zuerst in die Standardsprache PostScript umgewandelt. Insbesondere ASCII-Text wird normalerweise auch nach PostScript umgewandelt.
 - (c) Die PostScript-Daten werden, falls erforderlich, in eine andere Druckersprache umgewandelt.
 - Wenn ein PostScript-Drucker angeschlossen ist, werden die PostScript-Daten direkt an den Drucker (oder an eine andere Warteschlange) geschickt. Gegebenenfalls werden aber zusätzlich die Bash-Funktionen `duplex` und `tray`, die in `/usr/lib/lpddfilter/global/functions` definiert sind, aufgerufen, um Duplexdruck oder Papierschachtauswahl über PostScript-Kommandos zu ermöglichen — vorausgesetzt der PostScript-Drucker kann diese Kommandos entsprechend verarbeiten.
 - Wenn kein PostScript-Drucker angeschlossen ist, wird Ghostscript mit einem zur Druckersprache des jeweiligen Druckermodells passenden Ghostscript-Treiber verwendet, um die druckerspezifischen Daten zu erzeugen, die dann an den Drucker (oder an eine andere Warteschlange) geschickt werden.

Die druckerspezifischen Parameter für den Ghostscript-Aufruf sind an einer folgenden Stellen gespeichert:

- In der Datei `/etc/printcap` direkt in der `cm`-Zeile.
- Direkt in der Datei `/etc/lpddfilter/warteschlange/upp` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen).
- Indirekt in der Datei `/etc/lpddfilter/warteschlange/ppd` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen). Dies ist der Fall, wenn der `lpddfilter` mit `YaST` konfiguriert wurde, denn dann erfolgt die eigentliche Umwandlung in druckerspezifische Daten auf dieselbe Art

wie beim CUPS-Drucksystem mit dem Filter foomatic-rip, der den druckerspezifischen Ghostscript-Aufruf aus den Daten in derselben Foomatic PPD-Datei erzeugt, die auch für das CUPS-Drucksystem verwendet würde.

Die Ausgabe von Ghostscript kann ggf. nochmals umformatiert werden, sofern ein passendes Skript unter `/etc/lpddfilter/warteschlange/post` (hier ist warteschlange durch den tatsächlichen Namen der Warteschlange zu ersetzen) existiert.

4. Nachdem der Druckauftrag komplett an den Drucker geschickt wurde, löscht der Druckerspooler den Druckauftrag aus der Druckerwarteschlange.

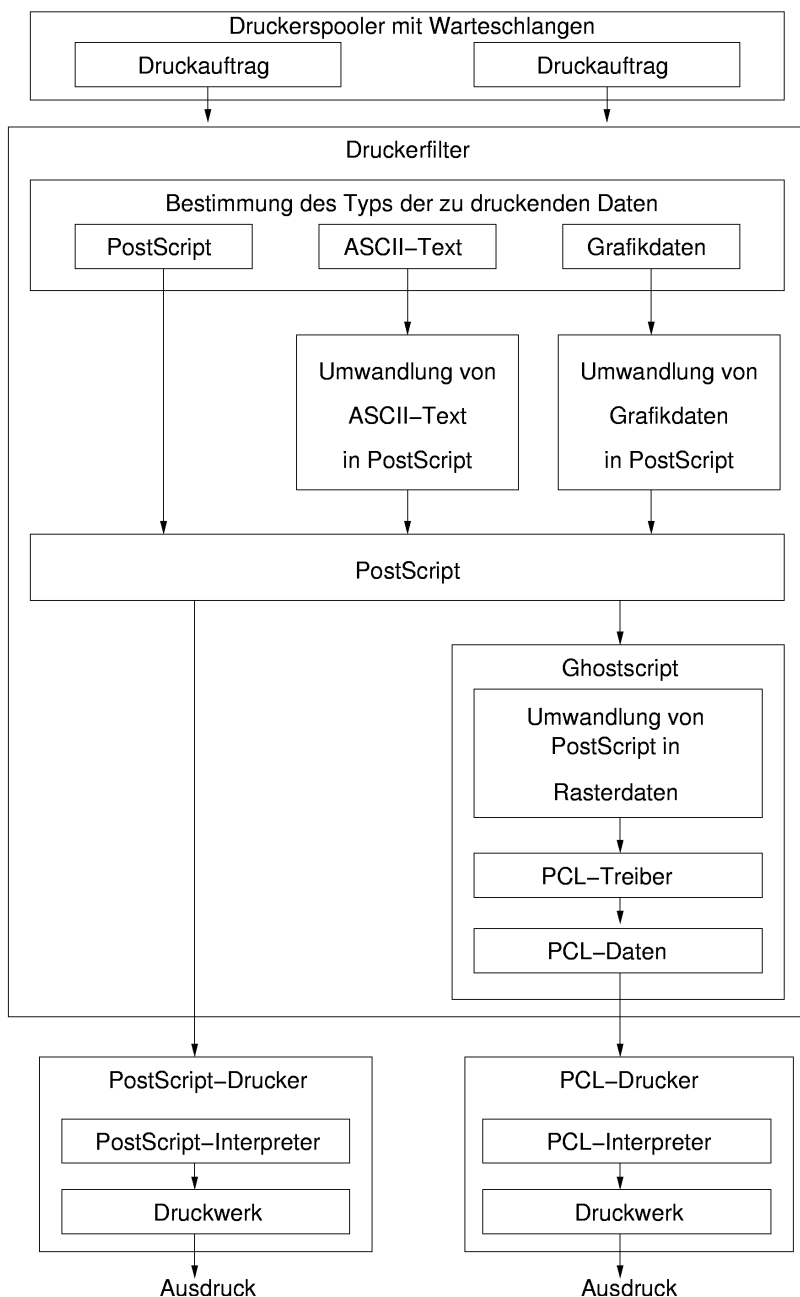


Abbildung 5.1: Überblick über den Ablauf beim Drucken

5.1.3 Verschiedene Drucksysteme

Bei SUSE LINUX werden zwei verschiedene Drucksysteme unterstützt:

LPRng/lpdfilter Das ist ein traditionelles Drucksystem bestehend aus dem Druckerspooler LPRng und dem Druckerfilter lpdfilter. Beim traditionellen Drucksystem wird die gesamte Konfiguration einer Warteschlange vom Systemverwalter festgelegt und der Benutzer kann nur zwischen verschiedenen Warteschlangen wählen. Um für einen Drucker zwischen verschiedenen Konfigurationen wählen zu können, müssen für denselben Drucker verschiedene Warteschlangen mit verschiedenen Konfigurationen eingerichtet werden. Bei einfachen Schwarzweiß-Druckern (z. B. den meisten Laserdruckern) ist eine Standardkonfiguration ausreichend, aber bei modernen Farb-Tintenstrahldruckern werden Konfigurationen für Schwarzweißdruck, Farbdruck und ggf. für hochauflösenden Farbdruck und Photodruck benötigt. Durch die festgelegten Konfigurationen ist einerseits automatisch sichergestellt, dass nur die vom Systemverwalter eingerichteten Konfigurationen benutzt werden können. Andererseits ist damit seitens des Drucksystems jegliche individuelle Einstellmöglichkeit durch den Benutzer verhindert, so dass der Systemverwalter entsprechend viele Warteschlangen einrichten muss, wenn die vielen Einstellmöglichkeiten, die moderne Drucker bieten, zur Verfügung stehen sollen.

CUPS Beim Drucksystem CUPS hat der Benutzer die Möglichkeit, für jeden Ausdruck druckerspezifische Einstellungen individuell festzulegen. Hier ist nicht die gesamte Konfiguration einer Warteschlange durch den Systemverwalter festgelegt, sondern die Möglichkeiten für druckerspezifische Einstellungen sind pro Warteschlange in einer PPD-Datei *PostScript Printer Description* hinterlegt; diese können so dem Benutzer in einem Druckdialog angeboten werden. Standardmäßig sind in der PPD-Datei alle Möglichkeiten hinterlegt, die der Drucker bietet. Durch Verändern der PPD-Datei kann der Systemverwalter die Möglichkeiten ggf. einschränken.

Da beide Drucksysteme in Konflikt miteinander stehen, ist es unter normalen Bedingungen nicht möglich, beide Drucksysteme *gleichzeitig* installiert zu haben, mit YaST2 kann jedoch zwischen beiden Drucksystemen hin und her gewechselt werden – siehe *Benutzerhandbuch* im Abschnitt *YaST — Konfigurationen, Drucker*.

5.2 Voraussetzungen zum Drucken

5.2.1 Allgemeine Voraussetzungen

- Der Drucker wird von SUSE LINUX unterstützt. Siehe dazu folgende Informationsquellen:

SUSE Druckerdatenbank <http://cdb.suse.de> bzw. <http://hardwaredb.suse.de/>

Linuxprinting.org Druckerdatenbank

<http://www.linuxprinting.org/> (The Database
<http://www.linuxprinting.org/database.html>
 bzw. die Auflistung http://www.linuxprinting.org/prINTER_list.cgi)

Ghostscript <http://www.cs.wisc.edu/~ghost/>

SUSE LINUX Ghostscript-Treiber

`file:/usr/share/doc/packages/ghostscript/catalog.devices` Hier sind die Ghostscript-Treiber aufgelistet, die bei der jeweiligen Version von SUSE LINUX tatsächlich dabei sind. Das ist wichtig, denn manchmal ist im WWW ein Ghostscript-Treiber genannt, der bei SUSE LINUX nicht vorhanden ist. Bei SUSE LINUX wird aus lizenzrechtlichen Gründen GNU Ghostscript mitgeliefert. In der Regel gibt es aber auch einen GNU Ghostscript-Treiber, mit dem der Drucker funktioniert.

- Der Drucker ist grundsätzlich ansprechbar; siehe dazu den Abschnitt 5.3.4 auf Seite 112.
- Sie sollten einen SUSE Originalkernel von den CD-ROMs verwenden; also insbesondere *keinen* selbst-kompilierten Kernel. Bei Problemen sollten Sie einen SUSE-Originalkernel installieren und mit diesem neu booten.
- Empfehlenswert ist die Installation des 'Standard-System' mit YaST2, damit alle notwendigen Pakete vorhanden sind. Wenn Sie bei der Installation des Standard-Systems keines der vorselektierten Pakete wieder deselektiert haben, ist es in Ordnung. Wenn nicht, installieren Sie wenigstens das 'Standard-System'. Die Auswahl 'Minimal-System' reichen zum Drucken nicht aus.

5.2.2 Bestimmung eines geeigneten Druckertreibers

Für PostScript-Drucker wird kein spezieller Druckertreiber benötigt. Siehe dazu den Abschnitt 5.1.2 auf Seite 100. Für Nicht-PostScript-Drucker erzeugt ein Ghostscript-Treiber die druckerspezifischen Daten. Daher ist der Ghostscript-Treiber die entscheidende Stelle, in der die Art des Ausdrucks für Nicht-PostScript-Drucker festgelegt wird. Die Wahl des Ghostscript-Treibers und ggf. spezielle treiberspezifische Einstellungen bestimmen das Druckbild. Bei den in Abschnitt 5.2.1 auf der vorherigen Seite genannten Listen sind auch Ghostscript-Treiber zu einzelnen Druckermodellen angegeben.

Wenn Sie für Ihren Drucker keinen Ghostscript-Treiber finden, dann fragen Sie ggf. beim Hersteller des Druckers nach, welche Druckersprache Ihr Modell kann. Einige Hersteller stellen sogar selbst spezielle Ghostscript Treiber für ihre Drucker zur Verfügung. Wenn der Hersteller keine Linux-relevanten Informationen zu Ihrem Druckermodell liefern kann, so kann er möglicherweise dennoch die Auswahl des Druckertreibers mit folgenden Informationen erleichtern:

- Stellen Sie fest, ob Ihr Drucker zu einem Modell kompatibel ist, das unter Linux läuft, und wählen Sie dann den Ghostscript-Treiber für das kompatible Modell. Kompatibel unter Linux bedeutet, dass Ihr Drucker mit denselben binären Steuersequenzen wie das kompatible Modell korrekt drucken kann – d. h. die Drucker verstehen dieselbe Druckersprache direkt und nicht etwa nur durch einen passenden Treiber (für ein anderes Betriebssystem) emuliert.

Sie können nicht immer aus ähnlichen Druckerbezeichnungen auf Kompatibilität schließen. Dies liegt daran, dass ähnlich bezeichnete Drucker manchmal nicht dieselbe Druckersprache verstehen.

- Welche Standarddruckersprache der Drucker versteht, kann am sichersten der Hersteller mitteilen. Auch bei den technischen Daten im Druckerhandbuch ist oft die Druckersprache angegeben.

PCL5e oder PCL6 Drucker die *PCL5e* oder *PCL6* direkt verstehen, sollten mit dem Ghostscript-Treiber *ljet4* bis zu 600x600 dpi funktionieren. Oft wird PCL5e nur als PCL5 bezeichnet.

PCL4 oder PCL5 Drucker die *PCL4* oder *PCL5* direkt verstehen, sollten mit einem der Ghostscript-Treiber *laserjet*, *ljetplus*, *ljet2p* oder *ljet3* funktionieren, sind aber auf 300x300 dpi beschränkt.

PCL3 Drucker die *PCL3* direkt verstehen, sollten mit einem der Ghostscript-Treiber *deskjet*, *hpdj*, *pcl3*, *cdjmono*, *cdj500* oder *cdj550* funktionieren.

ESC/P2, ESC/P oder ESC/P Raster

Drucker die *ESC/P2*, *ESC/P* oder *ESC/P* Raster direkt verstehen, sollten mit dem Ghostscript-Treiber *stcolor* oder mit dem Ghostscript-Treiber *uniprint* unter Verwendung einer passenden Parameterdatei *.upp* (z. B. *stcany.upp*) funktionieren.

5.2.3 Zur GDI-Drucker Problematik

Da die Druckertreiber für Linux normalerweise nicht vom Hersteller der Hardware entwickelt werden, ist es erforderlich, dass der Drucker über eine der allgemein bekannten Druckersprachen *PostScript*, *PCL* und *ESC/P* angesprochen werden kann. Normale Drucker verstehen zumindest eine der bekannten Druckersprachen. Verzichtet aber der Hersteller darauf und baut einen Drucker, der nur mit speziellen eigenen Steuersequenzen angesprochen werden kann, so hat man einen *GDI-Drucker*, der nur unter der Betriebssystemversion läuft, für die der Hersteller einen Treiber mitliefert. Da solche Drucker nach keiner bekannten Norm angesprochen werden, sind derartige Geräte abnormal und daher nicht bzw. nur unter Schwierigkeiten für Linux verwendbar.

GDI ist eine von Microsoft entwickelte Programmierschnittstelle zur grafischen Darstellung. Das Problem ist nicht die Programmierschnittstelle, sondern liegt darin, dass die sog. *GDI-Drucker* *nur* über die proprietäre Druckersprache des jeweiligen Druckersmodells angesprochen werden können. Eigentlich wäre die Bezeichnung Drucker, der *nur* über eine proprietäre Druckersprache angesprochen werden kann, korrekter.

Es gibt aber Drucker, die zusätzlich zum *GDI*-Modus eine Standard-druckersprache verstehen, wozu der Drucker passend einzustellen oder umzuschalten ist. Wenn Sie neben Linux noch ein anderes Betriebssystem verwenden, dann kann der Druckertreiber des anderen Betriebssystems den Drucker evtl. in den *GDI*-Modus umgeschaltet haben, so dass der Drucker danach nicht mehr unter Linux funktioniert. Entweder Sie schalten den Drucker unter dem anderen Betriebssystem wieder in einen Standardmodus zurück, oder Sie verwenden auch unter dem anderen Betriebssystem den Drucker nur in dem Standardmodus, wodurch dann aber oft nur noch eingeschränkte Druckmöglichkeiten (z. B. eine geringere Auflösung) zur Verfügung stehen.

Von besonderer Art sind solche Drucker, die nur rudimentäre Teile einer Standarddruckersprache verstehen — etwa nur die Befehle, die zur Ausgabe von Rastergrafik-Daten benötigt werden. Solche Drucker können manchmal ganz normal verwendet werden, da die Ghostscript-Treiber normalerweise nur Befehle zur Ausgabe von Rastergrafik-Daten verwenden. Eventuell kann dann kein ASCII-Text direkt auf dem Drucker ausgegeben werden, aber standardmäßig ist ja immer Ghostscript zwischengeschaltet. Problematisch sind solche Drucker, wenn sie dazu erst durch spezielle Steuersequenzen passend umgeschaltet werden müssen. Hier kann kein normaler Ghostscript-Treiber verwendet werden, sondern es braucht einen speziell angepassten Treiber, der diese Umschaltung vornimmt.

Für einige GDI-Drucker gibt es herstellereigener Treiber. Der Nachteil herstellereigener Linux-Treiber *für GDI Drucker* ist, dass nicht garantiert werden kann, dass diese mit verschiedenen (zukünftigen) Linux-Versionen funktionieren werden.

Drucker, die eine veröffentlichte Standarddruckersprache verstehen, sind dagegen weder von einem speziellen Betriebssystem, noch von einer speziellen Betriebssystemversion abhängig und herstellereigene Linux-Treiber für solche Drucker liefern oft die besten Druckergebnisse.

Bei SUSE LINUX werden folgende GDI-Drucker direkt durch die Druckerkonfiguration mit YaST2 unterstützt, aber da GDI-Drucker immer problematisch sind, kann es evtl. bei einzelnen Modellen nicht funktionieren bzw. es gibt deutliche Einschränkungen wie z. B. nur Schwarzweißdruck in geringer Auflösung. Bitte beachten Sie, dass wir nicht für die Verlässlichkeit der folgenden Angaben garantieren können, da wir GDI-Druckertreiber nicht selbst testen.

- Brother HL (720,730,820,1020,1040), MFC 4650,6550MC,9050 und dazu kompatible Modelle.
- HP DeskJet (710,712,720,722,820,1000) und dazu kompatible Modelle.
- Lexmark 1000,1020,1100,2030,2050,2070,3200,5000,5700,7000,7200, Z(11,42,43,51,52) und dazu kompatible Modelle.
- Oki Okipage 4w,4w+,6w,8w,8wLite,8z,400w und dazu kompatible Modelle.
- Samsung ML-(200,210,1000,1010,1020,1200,1210,1220,4500,5080,6040) und dazu kompatible Modelle.

Zumindest folgende GDI-Drucker sind unseres Wissens *nicht* durch SUSE LINUX unterstützt, aber diese Liste ist sicher längst nicht vollständig:

- Brother DCP-1000, MP-21C, WL-660
- Canon BJC (5000,5100,8000,8500), LBP 460,600,660,800, MultiPASS L6000
- Epson AcuLaser C1000, EPL 5500W,5700L,5800L
- HP LaserJet 1000,3100,3150
- Lexmark Z(12,22,23,31,32,33,82), Winwriter 100,150c,200
- Minolta PagePro 6L,1100L,18L, Color PagePro L, Magicolor 6100DeskLaser, Magicolor 2 DeskLaser Plus/Duplex
- Nec SuperScript 610plus,660,660plus
- Oki Okijet 2010
- Samsung ML 85G,5050G, QL 85G
- Sharp AJ 2100, AL 1000,800,840,F880,121

5.3 Drucker einrichten mit YaST

5.3.1 Warteschlangen und Konfigurationen

Normalerweise werden mehrere Druckerwarteschlangen aus folgenden Gründen benötigt:

- Verschiedene Drucker müssen über verschiedene Warteschlangen angesprochen werden.
- Pro Warteschlange kann der Druckerfilter individuell konfiguriert werden. Also werden verschiedene Warteschlangen für denselben Drucker verwendet, um verschiedene Konfigurationen zur Verfügung zu stellen. Bei CUPS ist das nicht notwendig, da hier der Benutzer selbst die entsprechenden Einstellungen festlegen kann; siehe dazu den Abschnitt 5.1.3 auf Seite 104.

Bei reinen Schwarzweiß-Druckern (z. B. den meisten Laserdruckern) ist eine Standardkonfiguration ausreichend, aber bei Farb-Tintenstrahldruckern werden normalerweise mindestens zwei Konfigurationen benötigt:

- Eine Standardkonfiguration, mit der der Drucker schnellen und kostengünstigen Schwarzweißdruck liefert.
- Eine color-Konfiguration bzw. Warteschlange für Farbdruk.

5.3.2 Grundsätzliches zur YaST Druckerkonfiguration

Die YaST Druckerkonfiguration kann nicht nur über die Menüs, sondern auch als Benutzer `root` direkt von der Kommandozeile mit `yast2 printer` aufgerufen werden.

Ein Hin- und Herwechseln zwischen CUPS und LPRng/lpfilter ist mit der Druckerkonfiguration von YaST in einem Sondermenü über den Button 'Erweitert' möglich. Beim Wechseln geht aber eine bestehende Konfiguration verloren, das heißt eine bestehende CUPS Konfiguration wird nicht in eine Konfiguration für LPRng/lpfilter umgewandelt und auch nicht umgekehrt.

Zwischen folgenden Drucksystemen kann mit der YaST Druckerkonfiguration gewählt bzw. gewechselt werden:

CUPS als Server (Vorgabe bei der Standardinstallation)

Wenn ein Drucker lokal angeschlossen ist, muss CUPS als Server laufen. Wird keine lokale Warteschlange mit YaST konfiguriert, dann wird der CUPS-Daemon `cupsd` nicht automatisch gestartet. Soll der `cupsd` dennoch laufen, so ist der Dienst 'cups' zu aktivieren (normalerweise für die Runlevel 3 und 5) – siehe den Abschnitt 5.9.2 auf Seite 145. Die folgende Pakete werden für dieses Drucksystem installiert:

- `cups-libs`
- `cups-client`
- `cups`
- `footmatic-filters`
- `cups-drivers`
- `cups-drivers-stp`

CUPS ausschließlich als Client Wenn es im lokalen Netzwerk einen CUPS-Netzwerk-Server gibt (siehe den Abschnitt 5.9.1 auf Seite 144) und man nur über dessen Warteschlangen drucken möchte, genügt es, wenn CUPS ausschließlich als Client läuft – siehe den Abschnitt 5.9.2 auf Seite 145. Folgende Pakete sind hierfür ausreichend:

- `cups-libs`
- `cups-client`

LPRng Wählen Sie dies, wenn das LPRng/lpfilter Drucksystem verwendet werden soll oder wenn es im Netzwerk nur einen LPD-Server gibt (siehe den Abschnitt 5.9.1 auf Seite 144) und man über dessen Warteschlangen drucken möchte – siehe den Abschnitt 5.9.2 auf Seite 145. Folgende Pakete werden hierfür installiert:

- `lprng`
- `lpdfilter`
- `footmatic-filters`
- `cups-drivers`

Das `cups-client` und das `lprng` schließen sich gegenseitig aus und dürfen nicht gemeinsam installiert sein. Das `cups-libs` muss immer installiert sein, denn etliche Pakete (z. B. Ghostscript, KDE, Samba, Wine und die YaST Druckerkonfiguration) benötigen die CUPS-Bibliotheken.

Für ein komplettes Drucksystem werden normalerweise noch etliche weitere Pakete benötigt, die aber alle beim ‘Standard-System’ automatisch installiert werden:

- `ghostscript-library`
- `ghostscript-fonts-std`
- `ghostscript-x11`
- `libgimpprint`

Die YaST Druckerkonfiguration zeigt an, welche Konfigurationen angelegt werden konnten, ohne dass ein Fehler dabei aufgetreten ist.

Da das tatsächliche Anlegen der Konfigurationen erst beim endgültigen Beenden der YaST Druckerkonfiguration geschieht, sollte man für eine Kontrolle die YaST Druckerkonfiguration erneut starten.

5.3.3 Automatische Konfiguration

Je nachdem, inwieweit YaST die Hardware automatisch erkennt und inwieweit zu dem jeweiligen Druckermodell Informationen in der YaST-Druckerdatenbank vorhanden sind, kann YaST die benötigten Daten automatisch ermitteln oder eine sinnvolle Vorauswahl anbieten. Andernfalls muss der Anwender die nötigen Informationen in den Dialogen liefern. YaST ermöglicht eine automatische Konfiguration des Druckers, wenn folgende Bedingungen erfüllt sind:

- Via automatischer Hardwareerkennung kann die parallele Schnittstelle bzw. die USB Schnittstelle automatisch korrekt eingerichtet und der daran angeschlossene Drucker automatisch erkannt werden.
- In der Druckerdatenbank findet sich die Identifikation des Druckermodells, die YaST bei der automatischen Hardwareerkennung erhalten hat. Da diese Identifikation von der Modellbezeichnung verschieden sein kann, ist es möglich, dass das Modell nur manuell ausgewählt werden kann.

Für jede Konfiguration sollte immer mit dem YaST-Testdruck ausprobiert werden, ob sie tatsächlich funktioniert, denn in vielen Fällen müssen Konfigurationsdaten ohne explizite Unterstützung durch den Hersteller des Druckers verwendet werden. Daher kann die Funktion nicht für alle Eintragungen garantiert werden.

Zudem liefert die YaST-Testseite wichtige Informationen zur jeweiligen Konfiguration.

5.3.4 Manuelle Konfiguration

Wenn eine der Bedingungen für die automatische Konfiguration nicht erfüllt ist oder wenn eine spezielle individuelle Konfiguration gewünscht wird, muss diese manuell erfolgen. Folgende Werte müssen konfiguriert werden:

Hardwareanschluss (Schnittstelle)

- Erkennt YaST das Druckermodell automatisch, ist davon auszugehen, dass der Druckeranschluss auf Hardware-Ebene funktioniert und es müssen hier keine Einstellungen konfiguriert werden.
- Erkennt YaST das Druckermodell nicht automatisch, deutet dies darauf hin, dass der Druckeranschluss auf Hardware-Ebene nicht ohne manuelle Konfiguration funktioniert. Bei der manuellen Konfiguration muss die Schnittstelle ausgewählt werden. `/dev/lp0` ist die erste parallele Schnittstelle. `/dev/usb/lp0` ist die Schnittstelle für einen USB-Drucker. Hierbei sollte unbedingt der entsprechende Test in YaST gemacht werden, um zu prüfen, ob der Drucker über die ausgewählte Schnittstelle überhaupt ansprechbar ist.

Am sichersten funktioniert es, wenn der Drucker direkt an der ersten parallelen Schnittstelle angeschlossen ist und im BIOS für die parallele Schnittstelle folgende Einstellungen gesetzt sind:

- ▷ IO-Adresse 378 (hexadezimal)
- ▷ Interrupt ist nicht relevant
- ▷ Modus Normal, SPP oder Output-Only
- ▷ DMA wird nicht verwendet

Ist trotz dieser BIOS-Einstellungen der Drucker nicht über die erste parallele Schnittstelle ansprechbar, muss die IO-Adresse entsprechend der BIOS-Einstellung explizit in der Form 0x378 bei den detaillierten Einstellungen zur parallelen Schnittstelle eingetragen werden. Sind zwei parallele Schnittstellen vorhanden, die auf die IO-Adressen 378 und 278 (hexadezimal) eingestellt sind, dann sind diese in der Form 0x378 , 0x278 einzutragen.

Name der Warteschlange Da der Warteschlangenname beim Drucken oft eingegeben werden muss, sollten nur kurze Namen aus Kleinbuchstaben und evtl. Zahlen verwendet werden.

Ghostscript-Treiber bzw. Druckersprache (Druckermodell)

Ghostscript-Treiber und Druckersprache sind durch das jeweilige Druckermodell vorgegeben und werden durch die Wahl einer zum Druckermodell passenden vordefinierten Konfiguration festgelegt, die bei Bedarf in einer gesonderten Maske individuell angepasst werden kann – d. h. durch die Wahl von Hersteller und Modell wird eigentlich die Druckersprache bzw. ein zum Drucker passender Ghostscript-Treiber mit passend vordefinierten Treibereinstellungen ausgewählt.

Da der Ghostscript-Treiber die druckerspezifischen Daten für Nicht-PostScript-Drucker erzeugt, ist die Konfiguration des Ghostscript-Treibers die entscheidende Stelle, an der die Art des Ausdrucks festgelegt wird. Zuerst die Wahl des Ghostscript-Treibers und dann passende treiberspezifische Einstellungen bestimmen das Druckbild. Hier werden die Unterschiede im Druckbild zwischen verschiedenen Konfigurationen für denselben Drucker festgelegt.

Hat YaST das Druckermodell automatisch erkannt bzw. findet sich das Modell in der Druckerdatenbank, gibt es eine sinnvolle Vorausswahl geeigneter Ghostscript-Treiber. In diesem Fall bietet YaST zu meist mehrere vordefinierte Konfigurationen an – z. B.

- Schwarzweißdruck
- Farbdruck 300 dpi
- Photodruck 600 dpi

Eine vordefinierte Konfiguration beinhaltet einen geeigneten Ghostscript-Treiber und ggf. passende treiberspezifische Einstellungen für die jeweilige Art des Ausdrucks.

Wenn es treiberspezifische Einstellungen gibt, können diese individuell in einer gesonderten Maske verändert werden. Klicken Sie auf einen Wert und wenn es dazu eine Unterauswahl gibt, sind die entsprechenden Menü-Einträge eingerückt. Nicht alle auswählbaren Kombinationen einzelner Treibereinstellungen funktionieren mit jedem Druckermodell – insbesondere in Kombination mit einer hohen Auflösung.

Ein Test durch Drucken der YaST Testseite ist unerlässlich. Wenn beim Drucken der Testseite Unsinn (z. B. viele fast leere Seiten) gedruckt wird, können Sie normalerweise den Druck sofort am Drucker stoppen, indem Sie alles Papier entnehmen und erst dann den Testdruck abbrechen. Allerdings gibt es Fälle, bei denen danach kein weiterer Ausdruck mehr möglich ist. Es ist daher problemloser, den Testdruck abzubrechen und das Ende des Ausdrucks abzuwarten.

Ist das Druckermodell nicht in der Druckerdatenbank eingetragen, so gibt es eine Auswahl an generischen Ghostscript-Treibern für die Standarddruckersprachen. Diese finden sich unter einem generischen Hersteller.

Sonstige spezielle Einstellungen Bei diesen speziellen Einstellungen sollten im Zweifelsfall die Voreinstellungen belassen werden.

Beim *CUPS* Drucksystem gibt es hier folgende spezielle Einstellungen:

- Zugriffsbeschränkungen für bestimmte Benutzer.
- Status der Warteschlange: ob der Ausdruck erfolgen soll oder nicht und ob die Warteschlange Druckaufträge annehmen soll oder nicht.
- Bannerseiten bzw. Deckblätter: ob und wenn ja welche Bannerseite vor dem eigentlichen Ausdruck gedruckt werden soll und ob und wenn ja welche Bannerseite nach dem eigentlichen Ausdruck gedruckt werden soll.

Beim *LPRng/lpdfilter* Drucksystem gibt es hier folgende spezielle hardwareunabhängige Einstellungen:

- Das Seitenlayout kann hier für den Ausdruck von ASCII-Texten festgelegt werden, nicht aber für Grafiken und Dokumente, die mit speziellen Anwendungsprogrammen erzeugt wurden.
- Für spezielle Fälle kann die Warteschlange als sog. *ascii*-Warteschlange eingerichtet werden. Bei einer *ascii*-Warteschlange wird der Druckerfilter gezwungen, die Ausgabe als ASCII-Text vorzunehmen. Das ist nötig, um bei ASCII-Textdateien, die der Druckerfilter nicht als ASCII-Text erkennt, die ASCII-Text-Ausgabe zu erzwingen (z. B. um PostScript-Quelltexte zu drucken).
- Die länderspezifische Kodierung betrifft die Darstellung von länderspezifischen Sonderzeichen beim Ausdruck von ASCII-Texten und von einfachem Text in HTML-Seiten aus Netscape.

5.4 Konfiguration für Anwendungsprogramme

Anwendungsprogramme verwenden die bestehenden Warteschlangen wie beim Drucken auf der Kommandozeile. Daher werden in den Anwendungsprogrammen nicht der Drucker, sondern die existierenden Warteschlangen konfiguriert.

Auf der Kommandozeile druckt man mit einem der folgenden Befehle:

```
lpr -P color <Dateiname>
lp -d color <Dateiname>
```

Dabei ist *<Dateiname>* durch den Namen der zu druckenden Datei zu ersetzen. Durch die Option *-P* bzw. *-d* kann die Warteschlange explizit bestimmt werden. Mit *-P color* bzw. *-d color* wird beispielsweise die Warteschlange *color* verwendet.

Oft bieten Anwendungsprogramme die existierenden Warteschlangen in einem Druckmenü an, so dass keine weitere Konfiguration notwendig ist. Wenn nicht, dann kann (bzw. muss) im Anwendungsprogramm ein Druckbefehl eingegeben (bzw. konfiguriert) werden. Das ist normalerweise der obige Druckbefehl ohne die Angabe *<Dateiname>*, also z.B. *lpr -P color* bzw. *lp -d color*.

5.5 Das CUPS-Drucksystem

Die folgenden Ausführungen geben einen leichten Einstieg in das CUPS Drucksystem. Wollen Sie sich als erfahrener Linux-Benutzer nur mit den wichtigsten Fakten zu CUPS beschäftigen, lesen Sie bitte Abschnitt 5.6 auf Seite 123.

5.5.1 Namenskonvention

Mit *Client* oder *Clientprogramm* bezeichnet man ein Programm, das gestartet wird, um Druckaufträge an den Drucker-Daemon zu schicken. Ein *Drucker-Daemon* ist ein lokaler Dienst, um Druckaufträge entgegenzunehmen und weiterzuschicken oder selbst zu verarbeiten. Ein *Server* ist ein Daemon, der einen oder mehrere Drucker mit den Druckdaten beliefern kann. Jeder Server hat gleichzeitig die Funktionalität eines Daemons. Meist wird weder von Benutzern noch von CUPS-Entwicklern sonderlich zwischen den Begriffen *Server* und *Daemon* unterschieden.

5.5.2 IPP und Server

Druckaufträge werden mit CUPS basierten Programmen, wie *lpr*, *kprinter* oder *xpp*, und mit Hilfe des *Internet Printing Protocols*, kurz IPP, verschickt. IPP ist in den Internet-Standards RFC-2910 und RFC-2911 definiert (siehe <http://www.rfc-editor.org/rfc.html>). Das IPP ist dem Webprotokoll HTTP ähnlich: gleiche Header, aber unterschiedliche Nutzdaten. Es wird ein anderer, eigener Port 631 zur Kommunikation verwendet, der bei der IANA *Internet Authority for Number Allocation* registriert wurde.

Die Daten werden an den konfigurierten CUPS-Daemon verschickt, der im Normalfall auch der lokale Server ist. Andere Daemons können beispielsweise mit Hilfe der Shell-Variable `CUPS_SERVER` direkt angesprochen werden.

Mit Hilfe der Broadcast-Funktion des CUPS-Daemons können die von ihm lokal verwalteten Drucker dem Netzwerk bekannt gemacht werden (UDP Port 631) und erscheinen dann als Warteschlange an allen Daemons, die diese Broadcast-Pakete empfangen und auswerten dürfen (konfigurierbar). Dies ist vorteilhaft für Firmennetze, weil man so kurz nach dem Start des Rechners alle vorhandene Drucker *sehen* kann, ohne selbst Hand an die Konfiguration legen zu müssen. Gefährlich ist diese Option, wenn

der Rechner mit dem Internet verbunden ist. Bei der Konfiguration der Broadcast-Funktion ist darauf zu achten, dass nur in das lokale Netzwerk gebroadcastet wird, dass der Zugriff nur für das lokale Netzwerk erlaubt ist und dass die öffentliche IP-Adresse für die Internetverbindung nicht im Adressbereich des lokalen Netzwerks liegt. Andernfalls könnten andere Benutzer desselben Internet Service Providers die freigegebenen Drucker *sehen* und auch benutzen. Außerdem erzeugen die Broadcasts Netzwerk-Traffic, was zusätzliche Kosten bedeuten kann. Man sollte daher immer sicherstellen, dass solche Broadcast-Pakete nicht vom lokalen Drucker ins Internet geschickt werden, z. B. mit Hilfe der paketfilternden SuSE-Firewall (SuSEfirewall2). Um die Broadcasts zu empfangen, muss nichts zusätzlich konfiguriert werden. Nur beim Versenden muss dazu eine Broadcast-Adresse angegeben werden (diese ist z. B. über YaST2 zu konfigurieren).

IPP wird zur Kommunikation zwischen lokalem und remote CUPS-Daemon verwendet (also einem *CUPS-Server*). Netzwerk-Drucker neuerer Art unterstützen mittlerweile auch IPP. Nähere Informationen dazu findet man auf den Webseiten der Hersteller oder im Handbuch des Druckers. Windows 2000 (und neuer) bietet ebenfalls eine IPP-Unterstützung. Doch leider gab es Probleme mit deren Implementierungsformat. Eventuell sind diese mittlerweile behoben oder können per Service Pack behoben werden.

5.5.3 Konfiguration des CUPS-Servers

Es gibt viele Arten, um unter CUPS Drucker einzurichten und den Daemon zu konfigurieren: mit Kommandozeilentools, YaST2, KDE Control Center, Webinterface, usw. In den folgenden Abschnitten wird nur auf Kommandozeilentools und YaST2 eingegangen. Deshalb im Vorfeld der Hinweis, dass dies nicht die einzigen Möglichkeiten sind.

Achtung

Das Webinterface birgt die Gefahr, dass man das Root-Passwort kompromittiert, weil man über das Netzwerk das Root-Passwort im Klartext verschickt, wenn man den Rechnernamen in der URL angibt. Deshalb sollten Sie immer nur `http://localhost:631/` verwenden, und auf keinen Fall andere Adressen.

Achtung

Aus diesem Grund wurde auch der Administrations-Zugriff auf den CUPS-Daemon dahingehend eingeschränkt, dass er nur dann konfiguriert werden kann, wenn er unter localhost angesprochen wird (was identisch

zur IP-Adresse 127.0.0.1 ist.) Andernfalls sieht man eine entsprechende Fehlermeldung.

Um lokale Drucker zu administrieren, ist es notwendig, dass ein CUPS-Daemon auf dem lokalen Rechner läuft. Dazu installiert man das cups und die von SuSE generierten PPD-Dateien in den Paketen cups-drivers und cups-drivers-stp. Dann startet man den Server (als root) mit dem Kommando: `/etc/rc.d/cups restart`. Bei der YaST2-Konfiguration geschieht diese Installation und das Starten implizit durch Auswahl von CUPS als Drucksystem und Installation eines Druckers.

PPD steht für PostScript Printer Description und ist ein Standard, um Druckeroptionen mit PostScript Kommandos zu beschreiben. CUPS benötigt diese zur Drucker-Installation. SUSE LINUX liefert zu vielen Druckern PPD-Dateien mit. Aber auch die Hersteller bieten auf ihren Webseiten und Installations-CDs PPD-Dateien für PostScript-Drucker an (meist im Bereich Installation unter Windows NT).

Der lokale Daemon kann auch mit der Absicht gestartet werden, dass man alle Drucker aller Broadcasting-Server lokal zur Verfügung haben will, obwohl man keine lokalen Drucker hat, d. h. zur Druckerauswahl unter KDE und OpenOffice soll möglichst wenig Aufwand notwendig sein.

Broadcasting konfiguriert man entweder mit YaST2, oder man kann in der Datei `/etc/cups/cupsd.conf` die Variable Browsing auf On (default) und die Variable BrowseAddress auf einen geeigneten Wert (beispielsweise 192.168.255.255) setzen. Damit auch Druckaufträge angenommen werden, ist zumindest der `<Location /printers>` oder besser der `<Location />` der Empfang zu erlauben. Dazu ist `Allow From xyz-host.mydomain` zu ergänzen – siehe file: `/usr/share/doc/packages/cups/sam.html`. Mit dem Befehl `/etc/rc.d/cups reload` (als root) übernimmt nach dem Bearbeiten der Datei der Daemon diese neue Konfiguration.

5.5.4 Netzwerkdrucker

Unter Netzwerkdrucker versteht man meist Drucker, die ein Printserver-Netzwerkinterface eingebaut haben (wie das HP mit dem JetDirect-Interface anbietet) oder Drucker, die an einer Printserver-Box oder Router-Box mit Printserverfunktionalität angeschlossen wurden. Nicht gemeint sind damit Windows-Rechner, die einen Drucker als Share zur Verfügung stellen. Doch kann man diese auch unter CUPS leicht auf ähnliche Art und Weise ansprechen.

Netzwerkdrucker unterstützen meist das LPD-Protokoll (auf Port 515). Man kann dies mit folgendem Befehl überprüfen:

```
netcat -z <rechnername>.<domain> 515 && echo ok || echo failed
```

Wenn dieser Dienst verfügbar ist, kann man ihn mit der Device-URI (CUPS Terminologie) `lpd://Server/Queue` konfigurieren. Näheres zu den Device-URIs kann man in `file:/usr/share/doc/packages/cups/sam.html` nachlesen.

Es ist normalerweise besser, wenn man solche Drucker über den eingebauten Port 9100 (HP, Kyocera, u. v. a. m.) oder Port 35 (QMS) anspricht, d. h. ohne vorgeschaltetes LPD-Protokoll. Die Device-URI lautet dann:

```
socket://Server:Port/
```

Zum Drucken auf Windows-Druckern, muss das `samba-client` installiert und Samba richtig konfiguriert sein, d. h. die richtige Workgroup muss gesetzt sein, etc. Die Device-URI für Windows-Rechner kann verschiedene Ausprägungen haben. Die häufigste Form dürfte wohl sein: `smb://user:password@host/printer`. Für alle anderen Arten siehe `file:/usr/share/doc/packages/cups/sam.html` und die Manualpage von `smbpool`.

Ist der Netzwerkdrucker konfiguriert und besitzt man ein kleineres Netzwerk mit mehreren (Linux-)PCs, ist es geschickt, wenn man diesen Netzwerkdrucker nicht mehrfach an allen Clients konfigurieren muss. Deshalb sollte man in diesem Fall die Broadcast-Funktionalität des Daemons einschalten (s. o.). Auch ein Umstellen der Konfigurationen, wie Standardpapiergröße auf `Letter`, muss nicht an jedem einzelnen Client, sondern nur einmal am Server vorgenommen werden (siehe Abschnitt 5.8.1 auf Seite 141). Diese Konfigurationen werden zwar lokal gespeichert aber durch die CUPS-Tools, bzw. bedingt durch das IPP-Protokoll, auf den Clients dargestellt.

5.5.5 Interne Auftragsbearbeitung

Konvertierung nach PostScript

Im Prinzip kann jeder Dateityp an einen CUPS-Daemon geschickt werden. Die wenigsten Probleme hat man jedoch bei PostScript-Dateien. Eine Konvertierung durch CUPS nach PostScript erfolgt, nachdem der Dateityp anhand von `/etc/cups/mime.types` identifiziert und dann das entsprechend in `/etc/cups/mime.convs` angegebene Tool aufgerufen wird. Diese Konvertierung passiert am Server und nicht am Client. Man wollte damit erreichen, dass auf einen Drucker spezialisierte Konvertierungen nur an dem dafür vorgesehenen Server durchgeführt werden können.

Accounting

Nach dieser PostScript-Konvertierung wird die Seitenzahl des Druckjobs ermittelt. Dazu startet CUPS das (eigene) Tool `pstops (/usr/lib/cups/filter/pstops)`. Die Seitenzahl des Druckjobs wird nach `/var/log/cups/page_log` geschrieben. Die Einträge einer Zeile bedeuten:

- Druckername (beispielsweise `lp`),
- Username (beispielsweise `root`),
- Job-Nummer,
- Datumsangabe in eckigen Klammern [],
- fortlaufende Seite des Jobs,
- Anzahl der Kopien.

Weitere umwandelnde Filter

Außerdem können noch andere Filter aktiv werden, so die entsprechenden Optionen für den Druck gewählt wurden. Besonders interessant sind:

psselect wenn nur bestimmte Seiten des Dokumentes ausgedruckt werden sollen,

ps-n-up falls mehrere Dokumentseiten auf ein Blatt gedruckt werden sollen.

Diese Filter können nicht konfiguriert werden. Die Aktivierung der Optionen ist in `file:/usr/share/doc/packages/cups/sum.html` beschrieben.

Druckerspezifische Umwandlung

Im nächsten Schritt wird der Filter gestartet, der notwendig ist, um druckerspezifische Daten zu erzeugen. Diese Filter finden sich unter `/usr/lib/cups/filter/`. Welcher Filter der richtige ist, ist in der PPD-Datei im Eintrag `*cupsFilter` festgelegt. Fehlt dieser Eintrag, wird von einem PostScript-fähigen Drucker ausgegangen. Alle geräteabhängigen Druckoptionen, wie Auflösung und Papiergröße, werden in diesem Filter verarbeitet.

Es ist nicht trivial, eigene druckerspezifische Filter zu schreiben; vgl. dazu den SDB-Artikel *Selbsterstellte Filter zum Ausdruck mit CUPS* (Stichworte: `cups + filter`).

Ausgabe an das druckende Gerät

Schließlich wird das Backend aufgerufen. Dabei handelt es sich um einen speziellen Filter, der Druckdaten auf einem Gerät oder auf einem Netzwerkdrucker ausgibt (siehe `/usr/share/doc/packages/cups/overview.html`). Das Backend ermöglicht die Kommunikation mit dem Gerät oder dem Netzwerkdrucker (hängt von der in der Installation angegebenen Device-URI ab). Ein Backend kann beispielsweise `usb` sein, dann würde das Programm `/usr/lib/cups/backend/usb` aufgerufen. Darin wird das USB-Device im Dateisystem geöffnet (und gelockt), vor-initialisiert, und vom Filter kommende Daten weitergeschickt. Am Ende wird das Device initialisiert und im System frei gegeben.

Derzeit gibt es die Backends: `parallel`, `seriell`, `usb`, `ipp`, `lpd`, `http`, `socket` (aus dem CUPS-Paket), sowie `canon` und `epson` (aus `cups-drivers-stp`), und `smb` (aus `samba-client`).

Filterlos

Will man die Ausgabe ohne einen Filter ausdrucken, so kann man beim `lpr`-Kommando die Option `-l` oder beim `lp`-Kommando `-oraw` angeben. Normalerweise funktioniert dann der Ausdruck nicht, weil keine druckerspezifische Umwandlung (siehe oben) erfolgt, oder andere, wichtige Filter nicht zum Einsatz kommen. Bei anderen CUPS-Tools lauten die Optionen ähnlich.

5.5.6 Tipps & Tricks

OpenOffice

CUPS wird beim Drucken aus OpenOffice direkt unterstützt, man muss nicht mehr, wie bei StarOffice 5.2, die Drucker einzeln einrichten. OpenOffice erkennt jetzt, ob ein CUPS-Daemon läuft, und fragt diesen selbstständig nach vorhandenen Druckern und Optionen. Eine zusätzliche OpenOffice Konfiguration sollte in Zukunft unnötig sein.

Windows

Drucker an einem Windows-Rechner können mit der Device-URI `smb://server/printer` angesprochen werden – siehe oben. Im umgekehrten Fall, also wenn man von Windows auf einen CUPS-Server drucken möchte, müssen in der Samba-Konfigurationsdatei `/etc/samba/smb.conf` die Einträge `printing = CUPS` und `printcap name = CUPS` gesetzt werden, wie es bei SUSE LINUX voreingestellt ist. Nach Änderungen in

`/etc/samba/smb.conf` muss der Samba-Server neu gestartet werden – siehe dazu `file:/usr/share/doc/packages/cups/sam.html`

Raw-Drucker einrichten

Ein Raw-Drucker kann dadurch eingerichtet werden, dass man die PPD-Datei bei der Installation weglässt, d. h. Filterung und Accounting werden nicht durchgeführt. Dazu müssen die Daten im druckereigenen Datenformat geschickt werden.

Eigene Drucker-Optionen

Konfigurationsoptionen (z. B. standardmäßig eine andere Auflösung) können pro Benutzer geändert und gespeichert werden. Die Speicherung erfolgt in der Datei `~/lpoptions`. Wird ein solcher umkonfigurierter Drucker am Server entfernt, ist er weiterhin in den diversen Tools wie `kprinter` oder `xpp` sichtbar. Auch dann, wenn er nicht mehr existiert, kann man ihn noch selektieren, was zu Problemen führt. Erfahrene Benutzer können dann die störenden Zeilen problemlos aus `~/lpoptions` mit einem Editor herauslöschen. Siehe dazu auch den Supportdatenbank-Artikel *Einstellungen zum Ausdruck mit CUPS* sowie den Abschnitt 5.8.1 auf Seite 141.

Kompatibilität zu LPR

CUPS kann auch Druckjobs von LPR-Systemen empfangen. Die nötige Konfiguration in `/etc/xinetd.d/cups-lpd` kann entweder mit YaST2 erledigt werden oder ist manuell zu machen.

Fehlersuche bei CUPS

In der Konfigurationsdatei `/etc/cups/cupsd.conf` findet sich standardmäßig folgender Abschnitt:

```
# LogLevel: controls the number of messages logged to the ErrorLog file
# and can be one of the following:
#
# debug2      Log everything.
# debug       Log almost everything.
# info        Log all requests and state changes.
# warn        Log errors and warnings.
# error       Log only errors.
# none        Log nothing.
#
LogLevel info
```

Zur Fehlersuche bei CUPS setzt man `LogLevel debug` und lässt den `cupsd` mit `rc cups restart` die geänderte Konfigurationsdatei neu einlesen. Ab dann finden sich ausführliche Meldungen in `/var/log/cups/error_log`, die zur Erkennung der Ursache von Problemen dienen.

Mit folgendem Befehl kann man vor einem Test eine Marke ausgeben:

```
echo "LABEL $(date)" | tee -a /var/log/cups/error_log
```

Diese Marke wird auch genau so in `/var/log/cups/error_log` eingetragen, um dort die Meldungen nach dem Test leichter auffindbar zu machen.

5.6 CUPS Schnelleinstieg

Dieser Abschnitt richtet sich an erfahrene Linux-Benutzer. Alle wichtigen Punkte zu CUPS werden kurz und bündig dargestellt.

5.6.1 Überblick über das CUPS Drucksystem

Ein genereller Überblick findet sich in folgendem Text der CUPS Dokumentation: "An Overview of the Common UNIX Printing System", die via <http://localhost:631/overview.html> oder via <file:///usr/share/doc/packages/cups/overview.html> zur Verfügung steht. Die CUPS Home-Page ist: <http://www.cups.org/>.

Verarbeitung eines Druckjobs

Sowohl auf einem Client- als auch auf Server-System erzeugt ein Kommandozeilentool oder ein Anwendungsprogramm einen Druckauftrag und übergibt ihn an den Spooler: Anwendungsprogramme rufen entweder ein Kommandozeilentool auf (z. B. Mozilla) oder verwenden die CUPS Bibliotheksfunktionen direkt (z. B. `kprinter`). Ein Druckauftrag besteht aus Informationen für den Spooler und den zu druckenden Daten und optionalen Informationen für den Filter.

In der Kommandozeile kann dies beispielsweise so aussehen: `lp -d queue -t title -o option1=value1 -o option2=value2 file1 file2`. `lp` ist der Befehl zum Senden, die Optionen `-d queue -t title` liefern Informationen an den Spooler, `-o option1=value1 -o`

`option2=value2` sind Filter-Optionen und `file1 file2` die zu druckenden Daten.

Die folgenden Schritte passieren nur auf einem CUPS Server. Ein Rechner, auf dem eine Warteschlange angelegt ist, ist ein Server.

Der Spooler (d. h. cupsd) speichert den Druckjob im Spool-Verzeichnis:

- Speichern der Information für den Spooler und den Filter in `/var/spool/cups/c<job-number>`
- Speichern der Daten aus den zu druckenden Dateien in `/var/spool/cups/d<job-number>-<file-number>`

Die zu druckenden Daten werden gefiltert und die druckerspezifischen Daten an den Drucker gesendet:

- Starten des Filtersystems: Dazu gehört das Bestimmen der benötigten Filter, um die druckerspezifischen Daten zu erzeugen, und der Aufbau der sog. „Filterkette“ bzw. „Filter-Pipe“. Außerdem werden die Programme der Filterkette mit passend gesetzten Parametern gestartet.
- Starten des sog. *Backends*, um die druckerspezifischen Daten von der *Filter-Pipe* an den Drucker weiterzuleiten.

Der Druckjob wird beendet, wenn das Backend fertig ist. Die betreffenden Dateien werden aus dem Spool-Verzeichnis gelöscht.

5.6.2 Der Spooler

Der Hauptzweck des Spooler-Systems ist, Daten vom Sender zum Empfänger zu bewegen. Er nimmt Daten entgegen (aber nur von zulässigen Sendern) und speichert sie bis der Empfänger bereit ist, sie entgegenzunehmen. Dann sendet er die Daten an zulässige Empfänger (und schaltet gegebenenfalls vorher passende Filter dazwischen). Darüber hinaus hält der Spooler Informationen bereit, was mit den Daten passiert ist (z. B. für `lpstat -W completed -o`).

Details zu `/usr/sbin/cupsd`

`cupsd` ist der Server für das IPP Protokoll. Das IPP Protokoll kann als Erweiterung des HTTP Protokolls angesehen werden. Details zum IPP Protokoll finden sich in RFC-2910 und RFC-2911. `cupsd` lauscht am TCP Port 631 auf IPP Aufträge wie z. B. `lp -d <queue> <file>` oder `lpstat -t`. Er lauscht am TCP Port 631 auf HTTP Aufträge wie z. B. `http://localhost:631/printers/`. `cupsd` verwendet UDP Port 631 um sog. *IPP Browsing* Informationen zu senden und zu empfangen. Man kann z. B. `netcat -u -l -p 631` verwenden, um solche Informationen aufzuschnappen (sofern UDP Port 631 noch nicht vom `cupsd` belegt ist). Die Konfiguration für `cupsd` liegt unter `/etc/cups/cupsd.conf`.

Details zu `/usr/lib/cups/daemon/cups-lpd`

Der `cups-lpd` ist der Server für das LPD Protokoll (RFC 1179). Er nimmt Druckjobs an, die via LPD Protokoll am TCP Port 515 eintreffen. Entweder wird `xinetd` oder `inetd` als „Wrapper“ für `cups-lpd` verwendet. Seine Konfiguration findet sich unter `/etc/xinetd.d/cups-lpd` oder in der „`cups-lpd`“ Zeile von `/etc/inetd.conf`.

Weitere Informationen

Das *CUPS Software Administrators Manual* ist verfügbar via `http://localhost:631/sam.html` oder via `file:///usr/share/doc/packages/cups/sam.html`

5.6.3 PPD Dateien

Ein PPD Datei enthält die druckerspezifischen Optionen zusammen mit den zugehörigen PostScript Code Schnipseln, die an den PostScript Interpreter gesendet werden müssen, um eine bestimmte Option zu aktivieren.

Abhängig davon, welche druckerspezifischen Optionen für einen bestimmten Druckjob gesetzt wurden (z. B. `-o PageSize=A4`), liest das Filter-System die passenden PostScript Code Schnipsel (die sog. *PostScript Invocation Values*) aus der PPD Datei und setzt sie in den PostScript Datenstrom ein. Genaue Informationen finden sich in der *Adobe PostScript Printer Description File Format Specification, Version 4.3*.

Details zu `/etc/cups/ppd`

Dieses Verzeichnis enthält die PPD Dateien, die cupsd tatsächlich verwendet. Die Einträge (insbesondere die `"*Default..."` Einträge) in einer PPD Datei in `/etc/cups/ppd/` können von den Einträgen in der ursprünglichen PPD Datei, die beim Anlegen der Warteschlange angegeben wurde, abweichen.

Details zu `/usr/share/cups/model`

Dieses Verzeichnis enthält die ursprünglichen PPD Dateien aus den Paketen cups (CUPS PPD Dateien), cups-drivers (Foomatic PPD Dateien) und cups-drivers-stp (Gimp-Print PPD Dateien) und ggf. zusätzliche PPD Dateien (z. B. PPD Dateien von Druckerherstellern).

Zusätzliche PPD Dateien (z. B. PPD Dateien von Druckerherstellern für PostScript Drucker) können in dieses Verzeichnis kopiert werden, um sie in der YaST Druckerkonfiguration verfügbar zu machen - siehe den Supportdatenbank Artikel *Drucker einrichten ab SuSE Linux 8.2*: http://portal.suse.com/sdb/de/2003/03/jsmeix_print-einrichten-82.html

Die [LinuxPrinting.org](http://www.linuxprinting.org/)/Foomatic Druckerdatenbank

Die [LinuxPrinting.org](http://www.linuxprinting.org/)/Foomatic Druckerdatenbank besteht aus XML Dateien aus denen die Foomatic PPD Dateien erzeugt werden. Die [LinuxPrinting.org](http://www.linuxprinting.org/)/Foomatic Druckerdatenbank ist hier verfügbar: <http://www.linuxprinting.org/database.html>

5.6.4 Der Filter

Der Hauptzweck des Filtersystems ist, die ursprünglichen Daten des Druckjobs (ASCII, PostScript, PDF) in druckerspezifische Daten (PostScript, PCL, ESC/P) umzuwandeln. Normalerweise erfolgt die Filterung in folgenden Schritten:

1. Bestimmung des MIME-Typs der ursprünglichen Daten gemäß `/etc/cups/mime.types`. Wenn es nicht `„application/postscript“` ist, dann Umwandlung nach PostScript gemäß `/etc/cups/mime.convs`. Beispielsweise `„text/plain“` wird mit `/usr/lib/cups/filter/texttops` nach PostScript umgewandelt.
2. Einfügen der „PostScript Invocation Values“ in den PostScript Datenstrom gemäß der folgenden Zeile in `/etc/cups/mime.convs`:

Tabelle 5.1: *PostScript Invocation Values*

input MIME type	output MIME type	costs	filter
application/postscript	application/vnd.cups-postscript	66	pstops

3. Wenn ein nicht-PostScript Drucker mit einer Foomatic PPD Datei verwendet wird, wird das PostScript in druckerspezifische Daten konvertiert, gemäss der folgenden Zeile in jeder Foomatic (Version 3.x) PPD Datei:

Tabelle 5.2: *PostScript-Konvertierung in Foomatic PPD-Datei*

main keyword	input MIME type	costs	filter
*cupsFilter:	"application/vnd.cups-postscript	0	foomatic-rip"

Wenn ein nicht-PostScript Drucker mit einer Foomatic PPD Datei verwendet wird, ist `foomatic-rip` zusammen mit Ghostscript der PostScript Interpreter. Um PostScript in druckerspezifische Daten umzuwandeln, macht `foomatic-rip` die folgenden Schritte:

- (a) Aufbau einer Ghostscript Kommandozeile mit den notwendigen Ghostscript Parametern gemäss der für den jeweiligen Druckauftrag eingestellten druckerspezifischen Optionen (d.h. gemäss der jeweiligen „PostScript Invocation Values“ im PostScript Datenstrom).
- (b) In einigen Fällen wird ein „Postfilter“ dem Ghostscript Kommando nachgeschaltet (via „Pipe“):
 - Für PCL Drucker:
Gewisse Optionen werden durch Änderungen im PCL Datenstrom realisiert (z. B. Papierschachtwahl via `perl -e`).
 - Für manche GDI Drucker:
Umwandeln der Ghostscript-Ausgabe in druckerspezifische Daten (z. B. für HP PPA Drucker mit `pnm2ppa`).
- (c) Ausführen der Ghostscript Kommandozeile (bzw. „Pipe“).

Wenn für einen nicht-PostScript Drucker eine nicht-Foomatic PPD Datei (z. B. eine Gimp-Print PPD Datei von `/usr/share/cups/model/stp/`) verwendet wurde, können die „*cupsFilter“ Einträge in der PPD Datei wie folgt aussehen:

```
*cupsFilter: "application/vnd.cups-raster 100 rastertoprinter"  
*cupsFilter: "application/vnd.cups-command 33 commandtoepson"
```

Mit den Einträgen in `/etc/cups/mime.convs` ergibt sich ein anderer Ablauf der Filterung.

Weitere Informationen zur Filterung finden sich im Supportdatenbank Artikel *Selbst erstellte Filter zum Ausdruck mit CUPS*: http://portal.suse.com/sdb/de/2003/05/jsmeix_print-cups-filters.html und im *CUPS Software Programmers Manual*, das via <http://localhost:631/spm.html> oder <file:///usr/share/doc/packages/cups/spm.html> zur Verfügung steht.

Details zu `/usr/lib/cups/filter`

Dieses Verzeichnis enthält die verschiedenen Filterprogramme, die das CUPS Filtersystem verwendet, wenn eine PPD Datei beim Einrichten der betreffenden Warteschlange angegeben wurde.

`/usr/lib/cups/filter/*tops`

Programme, um die ursprünglichen Daten (z. B. „text/plain“) nach PostScript umzuwandeln.

`/usr/lib/cups/filter/pstops`

fügt die „PostScript Invocation Values“ ein und formatiert ggf. das PostScript um (z. B. je zwei Seiten verkleinert auf ein Blatt drucken).

`/usr/lib/cups/filter/foomatic-rip`

wandelt PostScript in druckerspezifische Daten (z. B. PCL or ESC/P) um.

Details zu `/etc/cups/interfaces`

Dieses Verzeichnis enthält den Filter, der von CUPS verwendet wird, wenn beim Einrichten der betreffenden Warteschlange keine PPD Datei, sondern ein „System V style Interface Script“ angegeben wurde.

Ein „System V style Interface Script“ ist ein einziges Filterprogramm oder Filterscript, das die druckerspezifischen Daten für alle Datentypen produzieren muss, die bei den ursprünglichen zu druckenden Daten vorkommen können.

Weitere Informationen bzgl. „System V style Interface Script“ finden sich im Supportdatenbank Artikel *Selbst erstellte Filter zum Ausdruck mit CUPS*: http://portal.suse.com/sdb/de/2003/05/jsmeix_print-cups-filters.html

Details zu raw

Wenn beim Einrichten der betreffenden Warteschlange weder eine PPD Datei noch ein „System V style Interface Script“ angegeben wurde, erfolgt überhaupt keine Filterung. Die ursprünglichen zu druckenden Daten werden wie sie sind (also „roh“) direkt vom Backend an den Empfänger (normalerweise an den Drucker) geschickt. Es kann weder der Zeilenumbruch konvertiert werden (z. B. LF -> CR+LF) noch kann am Ende ein Formfeed angefügt werden. Für derartiges kann ein „System V style Interface Script“ verwendet werden.

5.6.5 Die Backends

Normalerweise bekommt das Backend die druckerspezifischen Daten vom Filter und sendet sie weiter an den Drucker oder einen sonstigen Empfänger. Die Unterschiede zwischen Backend und Filter sind:

- Genau ein Backend, aber normalerweise mehrere Filter (eine Filterkette) werden aktiviert, um einen Druckauftrag zu verarbeiten (Ausnahme: „System V style Interface Script“).
- Das Backend ist immer das letzte Programm in der Verarbeitungskette, das ausgeführt wird um einen Druckauftrag zu verarbeiten.

Für das Drucksystem ist der Druckauftrag dann komplett abgearbeitet, wenn das Backend fertig ist. Das Backend ist fertig, wenn die Datenübertragung zum Empfänger beendet ist. Wenn danach die weitere Verarbeitung beim Empfänger scheitert (z. B. wenn der Drucker die druckerspezifischen Daten nicht zu Papier bringen kann), merkt das Drucksystem davon nichts mehr.

Wenn die Datenübertragung zum Empfänger endgültig scheitert (normalerweise macht ein Backend mehrere Versuche), meldet das Backend einen Fehler an das Drucksystem (genauer an den cupsd). Das Backend entscheidet, ob und wieviele Versuche sinnvoll sind, bis es die Datenübertragung als unmöglich meldet. Da weitere Versuche somit sinnlos sind, wird das Ausdrucken für die betroffene Warteschlange vom cupsd abgeschaltet (disable). Nachdem die Ursache des Problems behoben wurde, muss der Systemverwalter mit `/usr/bin/enable` das Ausdrucken wieder aktivieren.

Weitere Informationen zu Backends finden sich im *CUPS Software Programmers Manual*: <http://localhost:631/spm.html> oder <file:///usr/share/doc/packages/cups/spm.html>

Details zu `/usr/lib/cups/backend`

Dieses Verzeichnis enthält die verschiedenen Backends. Je nachdem wie der Drucker erreichbar ist von dem Rechner, auf dem das CUPS System läuft, bzw. je nach Typ des Empfängers muss das passende Backend verwendet werden.

Das Ziel, an das ein Backend die Daten sendet, kann jeder "URI" (Uniform Resource Identifier) sein, für den ein passendes Backend existiert. Der erste Teil des „DeviceURI“-Eintrags in `/etc/cups/printers.conf` legt das Backend fest, der Rest dient dem Backend als Parameter.

Tabelle 5.3: Backend Definition

backend	DeviceURI syntax (example DeviceURI)
parallel	parallel:/dev/lp (parallel:/dev/lp0)
usb (traditional)	usb:/dev/usb/lp* (usb:/dev/usb/lp0)
usb (new)	usb://<make>/<model>?serial=<number> (usb://HP/DeskJet%20990C?serial=1234)
ipp	ipp://<server.domain>/printers/<queue> (ipp://host.domain/printers/ps)
lpd	lpd://<server.domain>/<queue> (lpd://192.168.101.202/LPT1)
socket	socket://<server.domain>:<port> (socket://192.168.101.202:9100)
smb	see the man page of smbpool (smb://user:password@workgroup/server/share)

Falls `user` und `password` für das `smb` Backend benötigt werden, müssen Eigentümer, Gruppe und Zugriffsrechte für die Datei `/etc/cups/printers.conf` hinreichend restriktiv gesetzt sein. Die Angaben `user` und `password` werden nicht mit dem Kommando `lpstat -v` angezeigt.

Ein Backend kann auch direkt aufgerufen werden. Beispielsweise liefern dabei die Backends „parallel“ und „usb“ die IEEE-1284 Identifikation von angeschlossenen Druckern:

```
root@host# /usr/lib/cups/backend/parallel
direct parallel:/dev/lp0 "Hewlett-Packard HP LaserJet 1220" "Parallel Port #1"

root@host# /usr/lib/cups/backend/usb
direct usb://HP/DeskJet%20990C?serial=1234 "HP DeskJet 990C" "USB Printer #1"
```

```
direct usb://EPSON/Stylus%20COLOR%20900?serial=987 "EPSON Stylus COLOR 900"
"USB Printer #2"
```

Beim Starten ruft cupsd alle Backends in `/usr/lib/cups/backend/` nacheinander einmal auf. Dadurch ermittelt cupsd, welche Backends auf dem jeweiligen System einsatzfähig sind. Die einsatzfähigen Backends werden mit `lpinfo -v` angezeigt.

5.6.6 Kommandozeilentools

Nehmen Sie keine manuellen Änderungen in den Konfigurationen in `/etc/cups/` vor, wenn es dafür auch geeignete Kommandozeilentools gibt. Die Konfigurationsdateien werden nicht für jeden Druckjob neu eingelesen. Stattdessen hält der cupsd viele Informationen nur im Hauptspeicher und schreibt ggf. Informationen in die Konfigurationsdateien zurück, wenn cupsd beendet wird. Die einzige Ausnahme hiervon stellt `/etc/cups/cupsd.conf` dar.

Nach Änderungen in dieser Konfigurationsdatei muss cupsd neu gestartet werden, damit er mit der geänderten Konfiguration arbeitet. Siehe im *CUPS Software Administrators Manual* den Abschnitt *Restarting the CUPS Server*: <http://localhost:631/sam.html#RESTARTING> oder `file:///usr/share/doc/packages/cups/sam.html#RESTARTING`

Kopieren Sie niemals Konfigurationsdateien von anderen Systemen in Ihr System, es sei denn, Sie wissen genau, was Sie tun. Verwenden Sie stattdessen Kommandozeilentools. Um z. B. dieselben Warteschlangen auf mehreren Maschinen anzulegen (etwa für einen Backup-Server), kopieren Sie nicht `/etc/cups/printers.conf`, sondern machen Sie ein Skript aus den passenden Kommandozeilen (normalerweise eine Reihe von `lpadmin` Aufrufen) und lassen dieses Skript auf den verschiedenen Maschinen laufen. Dadurch bekommen Sie ggf. die jeweils passenden Fehlermeldungen auf der jeweiligen Maschine (z. B. wenn auf einer Maschine eine PPD Datei nicht vorhanden ist oder wenn ein Backend nicht verfügbar oder nicht einsatzfähig ist). Ausserdem haben Sie durch das Skript ein Protokoll der Einstellungen und Sie können mit dem Skript diese Einstellungen immer wieder herstellen.

In vielen Fällen ist die Reihenfolge der Optionen wichtig. Lesen Sie die Manual-Pages. Beispielsweise sind `lpadmin -E -p <queue>`, `lpadmin -p <queue> -E` und `lpadmin -E -p <queue> -E` alle verschieden.

Warteschlangen anlegen oder ändern

So legen Sie eine Warteschlange per Kommandozeile an:

```
root@host# lpadmin -p ps -v parallel:/dev/lp0 -D "PS" -L "2.3" \
-P /usr/share/cups/model/Postscript.ppd.gz -E
```

Kontrollieren Sie, was angelegt wurde:

```
user@host$ lpstat -l -a ps -o ps -p ps -v ps
```

Alternativ nutzen Sie die Anzeige von `/etc/cups/printers.conf`:

```
<Printer ps>
Info PS
Location 2.3
DeviceURI parallel:/dev/lp0
State Idle
Accepting Yes
JobSheets none none
QuotaPeriod 0
PageLimit 0
KLimit 0
</Printer>
```

oder die Abfrage via Web-Frontend: `http://localhost:631/printers/ps`

So ändern Sie eine Warteschlange (z. B. Beschreibung und Standort):

```
root@host# lpadmin -p ps -D "PostScript printer" -L "2.floor: room3"
```

Zeigen Sie die druckerspezifischen Optionen und deren Voreinstellungen in `/etc/cups/ppd/ps.ppd` an:

```
user@host$ lpoptions -p ps -l
Resolution/Output Resolution: 150dpi *300dpi 600dpi 1200dpi 2400dpi
PageSize/Media Size: Letter Legal Executive *A4 A5
...
```

Die Ausgabe hat folgende Syntax:

```
main-keyword/translation-string: option-keyword option-keyword \
option-keyword ...
```

Die Voreinstellung ist durch einen * vor dem „Option-Keyword“ gekennzeichnet.

Ändern Sie die druckerspezifischen Voreinstellungen in `/etc/cups/ppd/ps.ppd`:


```
root@host# lpadmin -p ps -o Resolution=600dpi -o PageSize=Letter
```

Die Syntax ist:

```
lpadmin -p queue -o main-keyword1=option-keyword1 -o \
main-keyword2=option-keyword2 ...
```

Verwenden Sie nicht `lpoptions` für diesen Zweck. Siehe den Supportdatenbank Artikel *Einstellungen zum Ausdruck mit CUPS*: http://portal.suse.com/sdb/de/2002/11/jsmeix_print-cups-options.html

Normaler Benutzer können mit `lpoptions` ihre persönlichen Voreinstellungen in `~/.lpoptions` speichern. Siehe dazu denselben Supportdatenbank Artikel.

Verwenden Sie `accept` bzw. `reject` um Druckaufträge für eine Warteschlange zu akzeptieren bzw. zurückzuweisen und `/usr/bin/enable` (`enable` ist ein Bash-Builtin) bzw. `disable` um das Ausdrucken aus einer Warteschlange zu starten bzw. zu stoppen (z. B. damit während Wartungsarbeiten am Drucker keine Druckaufträge verloren gehen).

Zum Löschen einer Warteschlange verwenden Sie den folgenden Befehl:

```
root@host# lpadmin -x ps
```

Kommandozeilentools für den täglichen Gebrauch:

Vermeiden Sie die Kommandos im BSD Stil `lpr`, `lpq` und `lprm`, denn hier sind nur einige generische Optionen unterstützt.

Verwenden Sie stattdessen die Kommandos im System V Stil `lp`, `lpstat` und `cancel`.

Verwenden Sie `lpoptions` für die druckerspezifischen Optionen.

Die allgemein verfügbaren Optionen beim Ausdruck sind im *CUPS Software Users Manual* im Abschnitt *Standard Printer Options* beschrieben: http://localhost:631/sum.html#STANDARD_OPTIONS oder file:///usr/share/doc/packages/cups/sum.html#STANDARD_OPTIONS

5.6.7 Das Web-Frontend des cupsd

Jeder `cupsd` im Netzwerk hat ein HTTP Web-Frontend. Die URL für einen lokal laufenden `cupsd` ist `http://localhost:631/` und `http://host.domain:631/` ist die URL für einen entfernten `cupsd` auf dem

Rechner `host.domain`. Voraussetzung für den Zugriff ist, dass der jeweilige Rechner und der jeweilige `cupsd` den Zugriff erlauben.

Für den täglichen Gebrauch ist das Web-Frontend oft die beste Informationsquelle zu Warteschlangen und Druckaufträgen. Beispielsweise liefert `http://localhost:631/printers/` eine Übersicht aller Warteschlangen eines lokalen `cupsd`. Oder z. B. die abgearbeiteten Druckaufträge der Warteschlange `ps` auf dem Rechner `host.domain` werden mit einer URL in der Art `http://host.domain:631/printers/ps?which_jobs=completed` angezeigt.

Das Web-Frontend ist die beste Art, die zum jeweiligen `cupsd` (bzw. zu dessen Version) passende Dokumentation zu bekommen: `http://localhost:631/` für die Dokumentation zum lokalen `cupsd` oder eine URL in der Art `http://host.domain:631/` bzw. `http://host.domain:631/documentation.html` für die Dokumentation zu einem entfernten `cupsd` auf dem Rechner `host.domain`.

Auch die PPD Dateien unter `/etc/cups/ppd/` sind via Web-Frontend verfügbar. Beispielsweise die PPD Datei der Warteschlange "ps" auf dem lokalen Rechner (`/etc/cups/ppd/ps.ppd`) mit der URL `http://localhost:631/printers/ps.ppd` und eine URL in der Art `http://host.domain:631/printers/queue.ppd` für die PPD Datei der Warteschlange `queue` auf dem Rechner `host.domain`.

5.6.8 CUPS im Netzwerk konfigurieren

CUPS Netzwerk Server Auf dem CUPS Netzwerk Server erfolgen Spooling und Filterung. Der `cupsd` eines CUPS Netzwerk Servers sendet Informationen über seine Warteschlangen an eine beliebige Liste von IP Adressen (Rechneradressen und/oder Broadcastadressen). Die Voreinstellung ist eine leere Liste. Das Senden wird in einem vorgegebenen Zeitintervall wiederholt. Die Voreinstellung ist 30 Sekunden.

Clients Clients sind Systeme, die lediglich Druckaufträge an den Server senden. Auf jedem Client sollte ein lokaler `cupsd` laufen, denn per Voreinstellung lauscht ein `cupsd` auf Informationen, die von Servern kommen. Es gibt eine Liste von Servern, von denen Informationen angenommen werden. Per Voreinstellung wird von jeglichen Servern Information angenommen. Die Information über eine bestimmte Warteschlange wird auf dem Client gelöscht, wenn in einem vorgegebenen Zeitintervall keine neue Information über die Warteschlange eintrifft. Die Voreinstellung ist 300 Sekunden.

Auf diese Weise sind die Warteschlangen des Servers direkt auf dem Client verfügbar und Benutzer auf den Clients können die im Netzwerk auf verschiedenen Server verteilten Warteschlangen durchstöbern (engl. *browse*). Dieser Vorgang wird daher „Browsing“ genannt.

Per Voreinstellung ist Browsing aktiviert. Jegliche eintreffende Browsing-Information wird akzeptiert, aber keine Browsing-Information wird versendet - siehe oben.

Konfiguration der CUPS Netzwerk Server

Konfigurieren Sie auf dem Server die Warteschlangen für die Drucker, die zu dem Server gehören. Erlauben Sie den Zugriff auf die Warteschlangen für die Client-Rechner. Aktivieren Sie das Senden von Browsing-Information an die Client-Rechner.

Hinweis

Begrenzte Browsing-Informationen

Es ist nicht möglich, Browsing-Informationen nur für einen Teil der Warteschlangen eines Servers zu senden.

Hinweis

Es ist nicht möglich, nur einen grossen Server für eine grosse Firma zu haben und Browsing-Information für einen Teil der Warteschlangen an die Clients der einen Abteilung oder des einen Gebäudes zu senden und Browsing-Information für einen anderen Teil der Warteschlangen an andere Clients zu senden. Dazu sind mehrere Server (einer pro Abteilung oder Gebäude) notwendig.

Sub-Netze erleichtern die Konfiguration, denn dann genügt es, die Browsing-Information an eine feste Broadcast-Adresse zu senden, statt an eine ständig zu pflegende Liste von einzelnen Rechneradressen.

Der Zugriff auf die Warteschlangen eines Servers ist unabhängig davon, an welche Rechner der Server Browsing-Information sendet. Ein Server kann allen Clients im Netzwerk den Zugriff auf seine Warteschlangen erlauben und nur an einen Teil der Clients Browsing-Information senden. Ein Server sollte aber keine Browsing-Information an Clients senden, die keinen Zugriff auf seine Warteschlangen haben.

Warteschlangen für die zu einem Server gehörigen Drucker konfigurieren

Die Drucker, die zu einem Server gehören, sind genau diejenigen, für die die Filterung auf dem Server erfolgt. Konfigurieren Sie die Drucker so, dass

das Drucken für Benutzer auf dem Server korrekt funktioniert.

Fügen Sie in `/etc/cups/cupsd.conf` einen Eintrag der Art `Allow From <IP of the server>` im Abschnitt `<Location />...</Location>` hinzu, damit die Warteschlangen nicht nur via `localhost`, sondern auch über den eigentlichen Rechnernamen des Servers zugreifbar sind.

Sollten hierbei Probleme auftreten, gehen Sie folgendermaßen vor:

1. Setzen Sie `LogLevel debug` in `/etc/cups/cupsd.conf`.
2. Stoppen Sie den `cupsd`.
3. Bewegen Sie `/var/log/cups/error_log*` weg (oder löschen Sie es) damit Sie nicht in zu grossen Logdateien suchen müssen.
4. Starten Sie den `cupsd`.
5. Versuchen Sie erneut, was zu dem Problem geführt hat.
6. Nun finden sich viele Meldungen in `/var/log/cups/error_log*`, die zur Ursachenermittlung nützlich sind.

Erlauben Sie den Zugriff auf die Warteschlangen für die Client-Rechner:

- Das geht mit YaST oder manuell wie folgt: Fügen Sie `Allow From <IP of the client>` oder `Allow From <network IP . *>` Einträge im `<Location />...</Location>` Abschnitt in `/etc/cups/cupsd.conf` hinzu. Verwenden Sie hier IP-Adressen statt Namen.

Aktivieren Sie das Senden von Browsing-Information an die Client-Rechner:

- Das geht mit YaST oder manuell wie folgt: Fügen Sie `BrowseAddress <host or broadcast IP address>` Einträge in `/etc/cups/cupsd.conf` hinzu.
- Denken Sie an den Abschnitt *Restarting the CUPS Server* im *CUPS Software Administrators Manual*: <http://localhost:631/sam.html#RESTARTING> bzw. `file:///usr/share/doc/packages/cups/sam.html#RESTARTING`

Der letzte Schritt ist in gewisser Weise optional. Wird er weggelassen, bekommen die Client-Rechner nicht automatisch Informationen über die Warteschlangen auf dem Server. Dennoch kann wegen dem Schritt davor auf die Warteschlangen des Servers von den Client-Rechnern aus zugegriffen werden. Gerade in grossen Netzwerken kann es sinnvoll sein, den Zugriff für alle Clients freizugeben, aber Browsing-Information nur an einen Teil der Clients (z. B. nur an die Clients in der gleichen Abteilung oder im gleichen Gebäude wie der Server) zu senden.

Konfiguration der Client-Rechner

Gehen Sie wie folgt vor:

- Aktivieren von `/etc/init.d/cups`, so dass der `cupsd` beim Booten des Client-Rechners gestartet wird. Dazu entweder den YaST Runlevel Editor oder `insserv` verwenden.
- Starten Sie `cupsd`.

Im Normalfall ist es empfohlen, sonst nichts weiter zu konfigurieren. Weder lokale Warteschlangen auf den Clients oder Voreinstellungen für `cupsd` auf den Clients sollten zusätzlich verändert werden müssen.

Spezialfälle

Um optional bestimmte Server im Netz nach deren Browsing-Informationen abzufragen, gehen Sie wie folgt vor:

- Fügen Sie `BrowseAllow <IP of the desired server>` und `BrowseDeny <IP of the unwanted server>` Einträge in `/etc/cups/cupsd.conf` hinzu. Verwenden Sie IP-Adressen statt Namen. In bestimmten Fällen kann es nützlich sein, den `BrowseOrder` Eintrag anzupassen.

Wenn es Server gibt, die keine Browsing-Information senden, aber man kann dennoch auf deren Warteschlangen zugreifen (z. B. Server anderer Abteilungen oder Gebäude), dann kann es sinnvoll sein die Browsing-Information aktiv vom Server abzufragen (engl. *polling*):

1. Fügen Sie für jeden Server einen `BrowsePoll <IP of the server>:631` Eintrag in `/etc/cups/cupsd.conf` hinzu. Verwenden Sie IP-Adressen statt Namen. Der betreffende Port ist 631.

2. Nach dem erneuten Starten des cupsd wird für jeden BrowsePoll Eintrag ein cups-pollld laufen.

Wenn Browsing generell unerwünscht ist, setzen Sie `Browsing Off` in `/etc/cups/cupsd.conf`. Das bedeutet nicht, dass nun nicht mehr auf die Warteschlangen auf den Servern zugegriffen werden kann. Man kann weiterhin z. B. die Kommandozeilentools benutzen, nur ist nun der Server explizit anzugeben (normalerweise mit der Option `-h` - siehe die Manual-Pages).

Wenn Browsing generell unerwünscht ist, gibt es keinen Grund auf dem Client einen cupsd laufen zu haben. In diesem Fall sollte eine „Client-only“ Konfiguration gemacht werden. Das geht mit YaST oder manuell wie folgt:

1. Stoppen und Deaktivieren des cupsd (YaST Runlevel Editor oder `insserv`).
2. Den passenden `ServerName` (*IP of the desired server*) Eintrag in `/etc/cups/client.conf` machen.

So ein Eintrag sollte nicht zusammen mit einem lokal laufenden cupsd vorhanden sein. Es ist maximal ein solcher Eintrag möglich. Daher sollte hier der bevorzugt verwendete Server eingetragen werden. Um auf einen anderen Server zuzugreifen, ist dieser bei den Kommandozeilentools explizit anzugeben (Option `-h`) oder die Umgebungsvariable `CUPS_SERVER` ist passend zu setzen.

Manche Anwendungsprogramme ignorieren den `ServerName` Eintrag. Dann könnte es helfen `CUPS_SERVER` zu setzen, oder im Anwendungsprogramm ist der Server explizit anzugeben (z. B. mit `-h` im Druckbefehl).

Das Minimum, um CUPS Server anzusprechen, ist die Pakete `cups-libs` und `cups-client` installiert zu haben. Dann können Server mit den Kommandozeilentools angesprochen werden.

Das unterste Minimum, um CUPS Server anzusprechen, ist nur das Paket `cups-libs` installiert zu haben. Dann können nur noch Programme (wie z. B. `kprinter`) verwendet werden, die die CUPS Bibliotheken direkt verwenden.

5.7 Drucken aus Anwendungsprogrammen

Anwendungsprogramme verwenden die bestehenden Warteschlangen wie beim Drucken auf der Kommandozeile. Daher werden in den Anwendungsprogrammen nicht der Drucker, sondern die existierenden Warteschlangen konfiguriert.

Auf der Kommandozeile druckt man mit dem Befehl `lp -d color <Dateiname>`. Dabei ist `<Dateiname>` durch den Namen der zu druckenden Datei zu ersetzen. Durch die Option `-d` kann die Warteschlange explizit bestimmt werden. Mit `-d color` wird beispielsweise die Warteschlange `color` verwendet.

Das Paket `cups-client` enthält Kommandozeilentools zum Drucken mit CUPS wie z. B. den `lp`-Befehl, sodass obiges auch für CUPS funktioniert (siehe Abschnitt 5.8). Der Druckdialog in KDE-Programmen ist dazu aber auf 'Druck über ein externes Programm' umzustellen, weil sonst kein Druckbefehl eingegeben werden kann – siehe den Abschnitt 5.9.2 auf Seite 145.

Zusätzlich gibt es grafische Druckerdialogprogramme wie `xpp` oder das KDE-Programm `kprinter`, die es ermöglichen, nicht nur die Warteschlange zu wählen, sondern auch CUPS-Standardoptionen und druckerspezifische Optionen aus der PPD-Datei über graphische Auswahlmenüs einzustellen. Um `kprinter` in verschiedenen Anwendungsprogrammen als einheitlichen Druckdialog zu bekommen, geben Sie in der Druckmaske der Anwendungsprogramme als Druckbefehl `kprinter` oder `kprinter --stdin` ein. Welcher Druckbefehl zu nehmen ist, hängt vom Anwendungsprogramm ab. Dadurch erscheint nach der Druckmaske des Anwendungsprogramms der `kprinter`-Druckerdialog, in dem Sie die Warteschlange und die weiteren Optionen einstellen. Bei dieser Methode ist darauf zu achten, dass sich die Einstellungen in der Druckmaske des Anwendungsprogramms und in `kprinter` nicht widersprechen. Einstellungen möglichst nur in `kprinter` vornehmen!

5.8 Kommandozeilentools für das CUPS-Drucksystem

Die Kommandozeilentools und deren Manual-Pages für das CUPS-Drucksystem befinden sich im `cups-client` und Dokumentation dazu

befindet sich im cups unter `/usr/share/doc/packages/cups/` insbesondere das CUPS Software Users Manual unter `file:/usr/share/doc/packages/cups/sum.html` und das CUPS Software Administrators Manual unter `file:/usr/share/doc/packages/cups/sam.html` die bei lokal laufendem cupsd auch unter `http://localhost:631/documentation.html` erreichbar sind.

Bei CUPS-Kommandozeilentools ist gelegentlich die Reihenfolge der Optionen wichtig. Im Zweifelsfall ist die jeweilige Manual-Page zu beachten.

5.8.1 Für lokale Warteschlangen

Druckaufträge erzeugen

Normalerweise druckt man auf System V Art mit `lp -d <warteschlange> <datei>` oder auf Berkeley Art mit `lpr -P<warteschlange> <datei>`.

Weitere Informationen in der Manualpage von `lpr` und der Manualpage von `lp` und im Abschnitt Using the Printing System unter `file:/usr/share/doc/packages/cups/sum.html#USING_SYSTEM` im *CUPS Software Users Manual*.

Mit dem zusätzlichen Parameter `-o` können weitreichende Optionen bzgl. der Art des Ausdrucks festgelegt werden. Weitere Informationen in der Manualpage von `lpr` und der Manualpage von `lp` und im Abschnitt Standard Printer Options unter `file:/usr/share/doc/packages/cups/sum.html#STANDARD_OPTIONS` im *CUPS Software Users Manual*.

Status anzeigen

Auf System V Art mit `lpstat -o <warteschlange> -p <warteschlange>` oder auf Berkeley Art mit `lpq -P<warteschlange>` wird der Status einer Warteschlange angezeigt.

Ohne Angabe einer Warteschlange werden alle Warteschlangen angezeigt, wobei `lpstat -o` alle aktiven Druckaufträge in der Form `<warteschlange>-<jobnummer>` anzeigt.

Mit `lpstat -l -o <warteschlange> -p <warteschlange>` wird mehr Information angezeigt und mit `lpstat -t` bzw. `lpstat -l -t` wird die maximal verfügbare Information angezeigt.

Weitere Informationen in der Manualpage von `lpq` und der Manualpage von `lpstat` und im Abschnitt Using the Printing System unter `file:/usr/share/doc/packages/cups/sum.html#USING_SYSTEM` im *CUPS Software Users Manual*.

Druckaufträge löschen

Auf System V Art `cancel <warteschlange>-<jobnummer>` oder auf Berkeley Art `lprm -P<warteschlange> <jobnummer>` löscht den Druckauftrag mit der angegebenen Jobnummer aus der angegebenen Warteschlange. Weitere Informationen in der Manualpage von `lprm` und der Manualpage von `cancel` und im Abschnitt Using the Printing System unter `file:/usr/share/doc/packages/cups/sum.html#USING_SYSTEM` im *CUPS Software Users Manual*.

Einstellung der Warteschlangen

Im *CUPS Software Users Manual* sind im Abschnitt Standard Printer Options unter `file:/usr/share/doc/packages/cups/sum.html#STANDARD_OPTIONS` hardwareunabhängige Standard-Optionen für die Art des Ausdrucks beschrieben und im Abschnitt Saving Printer Options and Defaults unter `file:/usr/share/doc/packages/cups/sum.html#SAVING_OPTIONS` ist beschrieben, wie Optionseinstellungen gespeichert werden können.

Druckerspezifische Optionen für die Art des Ausdrucks sind in der PPD-Datei, die zu der entsprechenden Warteschlange gehört, festgelegt und werden mit dem Befehl `lpoptions -p <warteschlange> -l` in folgender Form angezeigt:

```
Option/Text: Wert Wert Wert ...
```

Dabei kennzeichnet ein `*` vor einem Optionswert die aktuelle Einstellung. Beispiel:

```
PageSize/Page Size: A3 *A4 A5 Legal Letter
Resolution/Resolution: 150 *300 600
```

Hier ist die Option `PageSize` auf `A4` eingestellt und die Auflösung auf den Wert `300`.

Mit `lpoptions -p <warteschlange> -o option=wert` kann ein anderer Wert eingestellt werden.

So kann in obigem Beispiel mit folgendem Befehl die Papiergröße für die entsprechende Warteschlange auf `Letter` umgestellt werden:

```
lpoptions -p <warteschlange> -o PageSize=Letter
```

Gibt ein normaler Benutzer diesen `lpoptions`-Befehl, werden die Einstellungen nur für diesen Benutzer in der Datei `~/ .lpoptions` gespeichert.

Gibt der Systemverwalter `root` den `lpoptions`-Befehl, werden die Einstellungen als Voreinstellung für alle Benutzer auf dem lokalen Rechner in der Datei `/etc/cups/lpoptions` gespeichert. Die PPD-Datei wird hierbei nicht verändert.

Nur wenn die Standardeinstellungen in der PPD-Datei einer Warteschlange verändert werden, gilt das für jegliche Benutzer im gesamten Netzwerk, die an dieser Warteschlange drucken. Die Standardeinstellungen in der PPD-Datei einer Warteschlange kann der Systemverwalter ändern, so dass in obigem Beispiel die voreingestellte Papiergröße für die entsprechende Warteschlange für alle Benutzer im Netzwerk auf `Letter` umgestellt wird:

```
lpadmin -p <warteschlange> -o PageSize=Letter
```

Siehe dazu auch den Supportdatenbank-Artikel *Einstellungen zum Ausdruck mit CUPS*.

5.8.2 Warteschlangen im Netz

Es werden der `<print-server>` durch den Namen oder die IP-Adresse des Print-Servers ersetzt und `<warteschlange>` muss eine Warteschlange auf dem Print-Server sein.

Hier sind nur die grundlegenden Kommandos angegeben. Zu weiteren Möglichkeiten und zu Informationsquellen siehe den Abschnitt 5.8.1 auf Seite 140.

Druckaufträge erzeugen

Auf System V Art wird mit `lpr -d <warteschlange> -h <print-server> <datei>` ein Druckauftrag für die angegebene Warteschlange auf dem angegebenen Print-Server erzeugt. Voraussetzung ist, dass der Print-Server so konfiguriert wurde, dass man auf dessen Warteschlangen drucken darf. Standardmäßig ist dies bei CUPS nicht der Fall, aber mit der YaST2-Druckerkonfiguration in einem erweiterten Menüweig bei den CUPS Server Einstellungen passend konfiguriert werden kann.

Status anzeigen

Auf System V Art wird mit `lprstat -h <print-server> -o <warteschlange> -p <warteschlange>` der Status einer Warteschlange auf dem Print-Server angezeigt.

Druckaufträge löschen

Der System V Art Befehl `cancel -h <print-server> <warteschlange>-<jobnummer>` löscht den Druckauftrag mit der angegebenen Jobnummer aus der angegebenen Warteschlange auf dem Print-Server.

5.8.3 Störungsbehebung mit obigen Befehlen bei CUPS

Druckaufträge bleiben in den Warteschlangen erhalten, wenn Sie während eines Druckvorgangs den Rechner herunterfahren und dann Linux neu starten; einen eventuell fehlerhaften Druckauftrag müssen Sie mit den oben vorgestellten Befehlen aus der Warteschlange entfernen.

Kommt es zum Beispiel zu einer Störung in der Kommunikation zwischen Rechner und Drucker, kann der Drucker mit den gesendeten Daten nichts Sinnvolles anfangen. Es werden lediglich Unmengen Papier mit sinnlosen Zeichen bedruckt.

1. Entnehmen Sie zuerst alles Papier bei Tintenstrahl Druckern bzw. öffnen Sie die Papierschächte bei Laserdruckern, damit das Drucken abgebrochen wird.
2. Da der Druckauftrag erst dann aus der Warteschlange entfernt wird, nachdem er komplett an den Drucker geschickt wurde, wird er meist noch in der Warteschlange stehen. Prüfen Sie mit `lpstat -o` (bzw. mit `lpstat -h <print-server> -o`) aus welcher Warteschlange gerade gedruckt wird und löschen Sie mit `cancel <warteschlange>-<jobnummer>` (bzw. mit `cancel -h <print-server> <warteschlange>-<jobnummer>`) den Druckauftrag.
3. Eventuell werden noch einige Daten an den Drucker übertragen, obwohl der Druckauftrag aus der Warteschlange gelöscht ist. Mit dem Befehl `fuser -k /dev/lp0` für einen Drucker am Parallelport bzw. `fuser -k /dev/usb/lp0` für einen USB-Drucker können alle Prozesse beendet werden, die noch auf den Drucker zugreifen.
4. Setzen Sie den Drucker komplett zurück, indem Sie ihn einige Zeit vom Stromnetz trennen. Danach legen Sie das Papier wieder ein und schalten den Drucker an.

Wenn die Datenübertragung zum Drucker nicht möglich ist oder auf bestimmte Weise gestört wird (z.B. wenn sie längerfristig unterbrochen

wird), dann beendet sich das sogenannte CUPS-Backend, das die eigentliche Datenübertragung zum Drucker vornimmt, mit einem Fehlercode. Unter welchen genauen Umständen das passiert, hängt vom jeweiligen Backend ab (z.B. Backend für den Parallelport, für USB, für LPD-Server, für IPP-Server oder für direkte Datenübertragung via TCP-Socket). Der CUPS Server (cupsd) deaktiviert in so einem Fall das weitere Ausdrucken über die betroffene Warteschlange und die Warteschlange wird als *disabled* oder *stopped* angezeigt. Nachdem die Ursache der Störung behoben ist, muss der Systemadministrator mit dem Befehl `/usr/bin/enable <warteschlange>` (bzw. mit `/usr/bin/enable -h <print-server>` `<warteschlange>`) den Druck wieder starten.

5.9 Drucken im TCP/IP-Netzwerk

Für den LPRng Druckerspooler finden sich ausführliche Informationen im *LPRng-Howto* unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html`. Für CUPS siehe das *CUPS Software Administrators Manual* unter `file:/usr/share/doc/packages/cups/sam.html`.

5.9.1 Bezeichnungen

Print-Server Als *Print-Server* wird hier nur ein vollständiger Rechner mit genügend Rechenleistung und Speicherkapazität bezeichnet.

Printserver-Box bzw. Netzwerkdrucker

- Bei einer Printserver-Box handelt es sich um ein kleines Gerät mit TCP/IP-Netzwerkanschluss und lokaler Anschlussmöglichkeit für einen Drucker. Es gibt auch *Router-Boxen*, die über eine lokale Anschlussmöglichkeit für einen Drucker verfügen und wie eine Printserver-Box zu behandeln sind.
- Ein Netzwerkdrucker hat einen eigenen TCP/IP-Netzwerkanschluss. Das ist letztlich ein Drucker mit eingebauter Printserver-Box. Netzwerkdrucker und Printserver-Boxen sind also gleich zu behandeln.

Es besteht ein erheblicher Unterschied zwischen einem Netzwerkdrucker bzw. einer Printserver-Box und einem echten Print-Server. Es gibt auch große Drucker, bei denen zum Drucken im Netzwerk ein

kompletter Rechner als Print-Server mitgeliefert wird. Aber hier wird beim Drucken nicht der eigentliche Drucker, sondern nur der mitgelieferte Print-Server angesprochen.

LPD-Server Ein *LPD-Server* ist ein Print-Server, der über das LPD-Protokoll ansprechbar ist. Das ist der Fall, wenn auf dem Print-Server das *LPRng/lpdfilter* Drucksystem (genaugenommen der *lpd*) läuft oder wenn das *CUPS* Drucksystem läuft und dieses so konfiguriert wurde, dass der Rechner auch über das LPD-Protokoll ansprechbar ist (genaugenommen über den *cups-lpd*).

IPP-Server bzw. CUPS-Server Ein *IPP-Server* bzw. *CUPS-Server* ist ein Print-Server, der über das IPP-Protokoll ansprechbar ist. Das ist der Fall, wenn auf dem Print-Server das *CUPS* Drucksystem (genaugenommen der *cupsd*) läuft.

CUPS-Netzwerk-Server Als *CUPS-Netzwerk-Server* wird hier ein *CUPS-Server* bezeichnet, der speziell so konfiguriert wurde, dass er seine Warteschlangen per UDP-Broadcast (via UDP-Port 631) anderen Rechnern im Netzwerk mitteilt.

5.9.2 Schnellkonfiguration für einen Client-Rechner

Ein Client-Rechner im Netzwerk verfügt normalerweise über keinen lokal angeschlossenen Drucker, Ausdrücke werden vom Client-Rechner an einen Print-Server geschickt. Wenn Sie einen Print-Server haben und am Client-Rechner ist zusätzlich ein Drucker lokal angeschlossen, dann machen Sie neben der Client-Konfiguration auch die Konfiguration für einen lokal angeschlossenen Drucker. Sie sollten das Drucksystem auf dem Client-Rechner passend zum Drucksystem auf dem Print-Server wählen.

Client-Konfiguration für einen LPD-Server

Wenn es im Netzwerk keinen *CUPS-Netzwerk-Server* gibt, sondern nur einen *LPD-Server*, dann sollten Sie auf dem Client-Rechner das *LPRng/lpdfilter* Drucksystem verwenden. Eine weitere Konfiguration des Client-Rechners ist dann nicht erforderlich, denn beim *LPRng-Spooler* können direkt mit dem *lpr*-Befehl auch entfernte Warteschlangen angesprochen werden.

Voraussetzung ist, dass der *LPD-Server* so konfiguriert wurde, dass der Client-Rechner auf dessen Warteschlangen drucken darf. Zum Druck aus

Anwendungsprogrammen ist im Anwendungsprogramm als Druckbefehl `lpr -P<warteschlange>@<print-server>` einzutragen.

Manche Anwendungsprogramme sind auf CUPS voreingestellt und müssen für LPRng umgestellt werden. Insbesondere KDE und das KDE-Druckprogramm kprinter sind auf 'Druck über ein externes Programm' umzustellen, weil sonst obiger Druckbefehl nicht eingegeben werden kann.

Client-Konfiguration für einen CUPS-Netzwerk-Server

Wenn der Print-Server ein CUPS-Netzwerk-Server ist, dann können Sie mit der YaST2-Druckerkonfiguration auf 'Ändern' und dann 'Erweitert' klicken und zwischen folgenden Möglichkeiten wählen:

CUPS als Server (Vorgabe bei der Standardinstallation)

Wenn kein Drucker lokal angeschlossen ist, wurde keine lokale Warteschlange mit YaST2 konfiguriert. In diesem Fall wird der cupsd nicht automatisch gestartet. Damit der cupsd gestartet wird, ist der Dienst 'cups' zu aktivieren (normalerweise für die Runlevel 3 und 5).

Eine weitere Konfiguration auf dem Client-Rechner ist nicht notwendig, denn ein CUPS-Netzwerk-Server teilt in regelmäßigen Abständen per Broadcast allen Rechnern im Netzwerk seine Warteschlangen mit, so dass nach kurzer Wartezeit auf dem Client-Rechner die Warteschlangen des CUPS-Netzwerk-Servers automatisch zur Verfügung stehen.

Voraussetzung ist, dass der CUPS-Netzwerk-Server so konfiguriert ist, dass die Broadcast-Funktion eingeschaltet ist und eine zum Client-Rechner passende Broadcast-Adresse verwendet wird und dass der Client-Rechner berechtigt ist, auf den Warteschlangen des CUPS-Netzwerk-Servers zu drucken.

CUPS ausschließlich als Client Wenn man nur über die Warteschlangen des CUPS-Netzwerk-Servers drucken möchte, genügt es, wenn CUPS ausschließlich als Client läuft. Dazu ist bei der YaST2 *Client-only* Druckerkonfiguration nur der Name des CUPS-Netzwerk-Servers anzugeben.

Hierbei läuft auf dem Client-Rechner kein cupsd und deswegen gibt es keine Datei `/etc/printcap`. Anwendungsprogramme, die nicht auf die Verwendung von CUPS umgestellt werden können, bieten aber nur die Warteschlangen an, die in der lokalen `/etc/printcap` stehen. In diesem Fall ist es besser, wenn CUPS

als Server läuft, denn der dann lokal laufende cupsd legt automatisch eine `/etc/printcap` mit den Namen der Warteschlangen des CUPS-Netzwerk-Servers an.

5.9.3 Protokolle zum Drucken im TCP/IP-Netzwerk

Es gibt die verschiedenen Möglichkeiten, in einem TCP/IP Netzwerk zu drucken, die sich weniger nach der verwendeten Hardware, sondern nach dem jeweils verwendeten Protokoll unterscheiden. Daher wird auch bei der YaST2-Druckerkonfiguration nicht nach der Hardware, sondern nach dem Protokoll unterschieden.

Dennoch wird in der YaST2-Druckerkonfiguration zuerst ausgewählt, über welche Art "Hardware" der Druck erfolgen soll (z. B. via CUPS-Netzwerk-Server, via LPD-Netzwerk-Server oder Druck direkt auf einem Netzwerkdrucker bzw. einer Printserver-Box). Dementsprechend werden nur die dann möglichen Protokolle angeboten, wobei das Protokoll vorausgewählt ist, welches in den meisten Fällen funktionieren sollte; wenn nur ein Protokoll möglich ist, gibt es keine Auswahl. Beispiele:

- Druck via CUPS-Netzwerk-Server
 - ▷ IPP-Protokoll (einzige Möglichkeit)
- Druck via LPD-Netzwerk-Server
 - ▷ LPD-Protokoll (einzige Möglichkeit)
- Druck direkt auf einem Netzwerkdrucker bzw. einer Printserver-Box:
 - ▷ TCP-Socket
 - ▷ LPD-Protokoll
 - ▷ IPP-Protokoll

Damit Daten vom Sender zum Empfänger gemäß dem jeweiligen Protokoll übertragen werden können, müssen Sender und Empfänger das jeweilige Protokoll unterstützen. Die Software, die auf dem Sender und Empfänger läuft, muss das jeweilige Protokoll unterstützen.

Letztlich ist es egal, welche Hardware und welche Software verwendet wird, es kommt nur darauf an, dass sowohl Sender als auch Empfänger das jeweilige Protokoll unterstützen. Je nach Protokoll werden Druckaufträge oder nur rohe Daten übertragen.

Ein Druckauftrag enthält neben den zu druckenden Daten noch Zusatzinformationen — etwa von welchem Benutzer auf welchem Rechner der Druckauftrag erzeugt wurde und gegebenenfalls welche speziellen Druckoptionen gewünscht sind (z. B. welche Papiergröße beim Ausdruck verwendet werden soll und/oder ob der Ausdruck im Duplex-Modus erfolgen soll usw.).

Drucken via LPD-Protokoll

Hierbei wird vom Sender ein Druckauftrag via LPD-Protokoll an eine Warteschlange auf dem Empfänger geschickt. Gemäss LPD-Protokoll nimmt der Empfänger die Druckaufträge am Port 515 an. Es wird also auf dem Empfänger-Rechner immer ein Dienst benötigt, der die Druckaufträge am Port 515 annimmt (normalerweise heißt ein solcher Dienst `lpd`) und es wird außerdem immer eine Warteschlange benötigt, in die angenommene Druckaufträge abgelegt werden können.

Sender, die das LPD-Protokoll unterstützen:

Linux-Rechner mit LPRng-Drucksystem:

- LPRng unterstützt das Senden via LPD-Protokoll über den `lpd`. Es wird auch eine Warteschlange auf dem Sender-Rechner benötigt, aus der der `lpd` des Senders den Druckauftrag nimmt und an den `lpd` des Empfängers weiterleitet.
- Bei LPRng geht das Senden via LPD-Protokoll auch ohne lokalen `lpd`. Das `lpr`-Programm aus dem `lprng` Paket kann den Druckauftrag via LPD-Protokoll direkt an den `lpd` des Empfängers weiterleiten.

Linux-Rechner mit CUPS-Server-Drucksystem:

- CUPS unterstützt das Senden via LPD-Protokoll über den CUPS-Daemon (`cupsd`). Es wird eine Warteschlange auf dem Sender-Rechner benötigt, aus der der `cupsd` den Druckauftrag nimmt und an den `lpd` des Empfängers weiterleitet.

Linux-Rechner mit CUPS-Client-Drucksystem:

- Das Senden via LPD-Protokoll wird beim CUPS-Client-Drucksystem nicht unterstützt.

Rechner mit Fremdbetriebssystem:

- Das LPD-Protokoll ist uralt, so dass jedes Betriebssystem dieses Protokoll zumindest als Sender unterstützen sollte. Eventuell ist die Unterstützung nicht standardmäßig vorhanden, so dass geeignete Software nachzuinstallieren ist.

Empfänger, die das LPD-Protokoll unterstützen:

Linux-Rechner mit LPRng-Drucksystem:

- LPRng unterstützt das Empfangen via LPD-Protokoll über den lpd.

Linux-Rechner mit CUPS-Server-Drucksystem:

- CUPS unterstützt das Empfangen via LPD-Protokoll über den cups-lpd. Der cups-lpd ist via inetd bzw. xinetd zu aktivieren.

Linux-Rechner mit CUPS-Client-Drucksystem:

- Das Empfangen via LPD-Protokoll ist beim CUPS-Client-Drucksystem nicht unterstützt.

Print-Server und Printserver-Boxen/Netzwerkdrucker:

- Das LPD Protokoll ist uralt, so dass jeder normale Print-Server und jede normale Printserver-Box bzw. jeder normale Netzwerkdrucker dieses Protokoll unterstützen sollte.
- Bei Printserver-Boxen bzw. Netzwerkdruckern ist der Name der Warteschlange von Modell zu Modell verschieden bzw. es gibt mehrere Warteschlangen mit unterschiedlichem Verhalten.

Drucken via IPP-Protokoll

Hierbei wird vom Sender ein Druckauftrag via IPP-Protokoll an eine Warteschlange auf dem Empfänger geschickt. Gemäß IPP-Protokoll nimmt der Empfänger die Druckaufträge am Port 631 an. Es wird also auf dem Empfänger-Rechner immer ein Dienst benötigt, der die Druckaufträge am

Port 631 annimmt (bei CUPS heisst dieser Dienst cupsd) und es wird außerdem immer eine Warteschlange benötigt, in die angenommene Druckaufträge abgelegt werden können.

Sender, die das IPP-Protokoll unterstützen:

Linux-Rechner mit LPRng-Drucksystem:

- LPRng unterstützt das IPP Protokoll nicht.

Linux-Rechner mit CUPS-Server- oder CUPS-Client-Drucksystem:

- CUPS unterstützt das Senden via IPP Protokoll auch ohne lokalen cupsd. Die Programme lpr oder lp aus dem cups-client Paket oder das Programm xpp oder das KDE-Programm kprinter können den Druckauftrag via IPP Protokoll direkt an den Empfänger weiterleiten.

Rechner mit Fremdbetriebssystem:

- Das IPP-Protokoll ist relativ neu, so dass die Unterstützung von jeweiligen Fall abhängt.

Empfänger, die das IPP-Protokoll unterstützen:

Linux-Rechner mit LPRng-Drucksystem:

- LPRng unterstützt das IPP Protokoll nicht.

Linux-Rechner mit CUPS-Server-Drucksystem:

- CUPS unterstützt das Empfangen via IPP-Protokoll über den cupsd. Es wird eine Warteschlange auf dem Empfänger-Rechner benötigt, in die der cups-lpd den Druckauftrag, den er vom Sender bekommen hat, ablegen kann.

Linux-Rechner mit CUPS-Client-Drucksystem:

- Das Empfangen via IPP Protokoll ist beim CUPS-Client-Drucksystem nicht unterstützt.

Print-Server und Printserver-Boxen/Netzwerkdrucker:

- Das IPP Protokoll ist relativ neu, so dass die Unterstützung von jeweiligen Fall abhängt.

Drucken direkt via TCP-Socket

Hierbei wird kein Druckauftrag an eine entfernte Warteschlange geschickt, denn es gibt hier kein Protokoll (LPD oder IPP), welches mit Druckaufträgen und Warteschlangen umgehen kann. Stattdessen werden hier rohe Daten direkt via TCP-Socket an einen entfernten TCP-Port geschickt. Üblicherweise wird das verwendet, um druckerspezifische Daten zu Printserver-Boxen und Netzwerkdruckern zu übertragen. In vielen Fällen wird dazu der TCP-Port 9100 verwendet.

Sender, die das Drucken direkt via TCP-Socket unterstützen:

Linux-Rechner mit LPRng-Drucksystem:

- LPRng unterstützt das Senden direkt via TCP-Socket über den lpd. Es wird eine Warteschlange auf dem Sender-Rechner benötigt, aus der der lpd des Senders den Druckauftrag nimmt und die zu druckenden Daten an den TCP-Port des Empfängers schickt.
- Bei LPRng geht es auch ohne lokalen lpd. Das lpr-Programm aus dem lprng Paket kann bei Verwendung der Option -Y die zu druckenden Daten direkt via TCP-Socket an den TCP-Port des Empfängers schicken. Siehe die man-Page zu lpr.

Linux-Rechner mit CUPS-Server-Drucksystem:

- CUPS unterstützt das Senden direkt via TCP-Socket über den cupsd. Es wird eine Warteschlange auf dem Sender-Rechner benötigt, aus der der cupsd den Druckauftrag nimmt und die zu druckenden Daten an den TCP-Port des Empfängers schickt.

Linux-Rechner mit CUPS-Client-Drucksystem:

- Das Senden direkt via TCP-Socket ist beim CUPS-Client-Drucksystem nicht unterstützt.
- Dennoch kann man immer mit etwa folgendem Kommando Daten an einen Port auf einem Rechner schicken:
`cat <filename> | netcat -w 1 <host> <port>`

Empfänger, die das Drucken direkt via TCP-Socket unterstützen:

Rechner mit LPRng- oder CUPS-Server- oder CUPS-Client-Drucksystem:

- Zum Empfangen direkt via TCP-Socket ist kein Drucksystem nötig, und keines der Drucksysteme unterstützt es direkt, denn es ist im allgemeinen nicht sinnvoll, rohe Daten zu schicken, wenn es ein Drucksystem gibt, was richtige Druckaufträge und ein dazu passendes Protokoll (LPD oder IPP) unterstützt.
- Dennoch kann man z. B. beim CUPS-Drucksystem auch Daten via Port 9100 Daten annehmen und an eine Warteschlange weiterleiten, indem man in `/etc/inetd.conf` einträgt:
`9100 stream tcp nowait lp /usr/bin/lp lp -d <queue>`
 Wenn keine Filterung erfolgen soll, ist `-o raw` anzufügen.
- Man kann auch das Verhalten einer Printserver-Box emulieren, die via Port 9100 Daten annimmt und direkt an den Drucker weiterleitet, indem man in `/etc/inetd.conf` eine Zeile der Art einträgt:
`9100 stream tcp nowait lp /bin/dd dd of=/dev/lp0`

Printserver-Boxen bzw. Netzwerkdrucker:

- Die Unterstützung hängt vom jeweiligen Fall ab.
- Insbesondere welcher Port der richtige ist, ist von Modell zu Modell verschieden. Bei HP Netzwerkdruckern bzw. bei HP JetDirect Printserver-Boxen ist das standardmäßig der Port 9100 bzw. bei JetDirect Printserver-Boxen mit zwei oder drei lokalen Druckeranschlüssen die Ports 9100, 9101 und 9102. Diese Ports werden auch von vielen anderen Printserver-Boxen verwendet. Konsultieren Sie das Handbuch der Printserver-Box und fragen Sie im Zweifelsfall den Hersteller der Printserver-Box bzw. des Netzwerkdruckers,

unter welchem Port der Drucker direkt angesprochen werden kann. Informationen dazu finden sich im LPRng-Howto unter `file:///usr/share/doc/packages/lprng/LPRng-HOWTO.html` und dort insbesondere unter `file:///usr/share/doc/packages/lprng/LPRng-HOWTO.html#SECNETWORK`, `file:///usr/share/doc/packages/lprng/LPRng-HOWTO.html#SOCKETAPI` und `file:///usr/share/doc/packages/lprng/LPRng-HOWTO.html#AEN4858`

Beispiele

Fall 1: Mehrere Workstations, ein Print-Server und ein oder mehrere Printserver-Boxen bzw. Netzwerkdrucker:

Print-Server mit LPRng-Drucksystem:

- Die Workstations sollten auch das LPRng-Drucksystem verwenden.
- Für jeden an einer Printserver-Box angeschlossenen Drucker bzw. für jeden Netzwerkdrucker gibt es auf dem Print-Server eine eigene Warteschlange.
- Die Workstations übertragen die Druckaufträge via LPD Protokoll in die zum Drucker gehörende Warteschlange auf dem Print-Server.
- Je nachdem, welche Printserver-Box bzw. welcher Netzwerkdrucker welches Protokoll unterstützt, verwendet der Print-Server das LPD Protokoll oder die direkt Datenübertragung via TCP-Socket, um die zu druckenden Daten an die Printserver-Box bzw. den Netzwerkdrucker zu schicken.

Print-Server mit CUPS-Server-Drucksystem:

- Die Workstations sollten auch das CUPS-Drucksystem verwenden. Das CUPS-Client-Drucksystem ist in diesem Fall völlig ausreichend.
- Für jeden an einer Printserver-Box angeschlossenen Drucker bzw. für jeden Netzwerkdrucker gibt es auf dem Print-Server eine eigene Warteschlange.
- Die Workstations übertragen die Druckaufträge via IPP Protokoll in die zum Drucker gehörende Warteschlange auf dem Print-Server.

- Je nachdem, welche Printserver-Box bzw. welcher Netzwerkdrucker welches Protokoll unterstützt, verwendet der Print-Server das LPD Protokoll oder die direkt Datenübertragung via TCP-Socket, um die zu druckenden Daten an die Printserver-Box bzw. den Netzwerkdrucker zu schicken.

Fall 2: Einige wenige Workstations, kein Print-Server und ein oder mehrere Printserver-Boxen bzw. Netzwerkdrucker:

Workstations mit LPRng-Drucksystem oder CUPS-Server-Drucksystem:

- Für jeden an einer Printserver-Box angeschlossenen Drucker bzw. für jeden Netzwerkdrucker gibt es auf jeder Workstation eine eigene Warteschlange. Da auf jeder Workstation alle Warteschlangen eingerichtet werden müssen, ist das nur bei wenigen Workstations sinnvoll.
- Je nachdem, welche Printserver-Box bzw. welcher Netzwerkdrucker welches Protokoll unterstützt, verwendet die Workstation das LPD Protokoll oder die direkt Datenübertragung via TCP-Socket, um die zu druckenden Daten an die Printserver-Box bzw. den Netzwerkdrucker zu schicken.
- Wenn mehrere Workstations gleichzeitig Daten an dieselbe Printserver-Box bzw. an denselben Netzwerkdrucker schicken, kann es zu Datenverlust und allen möglichen anderen Problemen kommen — insbesondere wenn zur Datenübertragung das LPD-Protokoll verwendet wird, denn die Implementierung des LPD-Empfängers in der Printserver-Box bzw. im Netzwerkdrucker ist oft mangelhaft, weil es dort zumeist nicht genug Speicherplatz gibt, um mehrere Druckaufträge annehmen und zwischenspeichern zu können. Erfolgt dagegen die Datenübertragung ausschliesslich via TCP-Socket, so kann das je nach Printserver-Box bzw. Netzwerkdrucker auch sehr zuverlässig funktionieren.

5.9.4 Filterung beim Drucken im Netzwerk

Im vorigen Abschnitt wurde beschrieben, wie Druckaufträge bzw. wie rohe Daten von der Workstation zum Drucker übertragen werden. Etwas ganz anderes ist es, wie die Filterung (also das Umwandeln der ursprünglichen Daten in druckerspezifische Daten) beim Drucken im Netzwerk erfolgen

kann. Die Umwandlung in druckerspezifische Daten beim Drucken im Netzwerk erfolgt ganz genau so, wie bei einem lokal an einem Einzelplatzsystem angeschlossenen Drucker. In Hinblick auf den Druckerfilter gibt es überhaupt keinen Unterschied zwischen Netzwerkdruck und Einzelplatzsystem. Lediglich der Datenstrom von der Workstation zum Drucker ist komplizierter und läuft über mehrere Stationen; z. B. folgendermaßen:

```
Workstation ->  
Print-Server ->  
Printserver-Box ->  
Drucker
```

An genau einer Stelle dieser Kette muss die Ausgangsdatei in das Format umgewandelt werden, das der Drucker letztlich drucken kann (PostScript, PCL, ESC/P).

Die Umwandlung wird vom Druckerfilter erledigt und dieser kann nur auf einem Rechner mit genügend Rechenleistung und Speicherkapazität laufen, also entweder auf der Workstation, oder auf einem Print-Server, aber weder in einer Printserver-Box noch in einem Netzwerkdrucker. Printserver-Boxen und Netzwerkdrucker haben normalerweise keinen Druckerfilter eingebaut, sie können nur druckerspezifische Daten annehmen und an den Drucker bzw. an das eigentliche Druckwerk weiterleiten.

Eine Warteschlange kann mit Filterung oder ohne angelegt werden. Da in der YaST2-Druckerkonfiguration zuerst ausgewählt wird, über welche Art Hardware der Druck erfolgen soll (z. B. via CUPS-Netzwerk-Server, via LPD-Netzwerk-Server oder Druck direkt auf einem Netzwerkdrucker bzw. einer Printserver-Box), ist die Voreinstellung, ob Filterung erfolgen soll oder nicht, so, dass es normalerweise funktionieren sollte — bei Bedarf ist die Voreinstellung in der YaST2-Druckerkonfiguration passend zu ändern.

Die Voreinstellungen sind:

Druck via CUPS-Netzwerk-Server:

keine Filterung (da diese normalerweise auf dem CUPS-Netzwerk-Server erfolgt)

Druck via LPD-Netzwerk-Server: keine Filterung (da diese normalerweise auf dem LPD-Netzwerk-Server erfolgt)

Druck direkt auf einem Netzwerkdrucker bzw. einer Printserver-Box:
Filterung

Wird die Warteschlange mit Filterung angelegt, dann werden in der Warteschlange die ursprünglichen Daten zwischengespeichert. Erst wenn die Daten an den Empfänger gesendet werden, durchlaufen sie dabei auf dem Rechner, auf dem die Warteschlange liegt, den Filter. Dem eigentlichen Sender der Daten wird also der Filter vorgeschaltet, so dass beim Empfänger die umgewandelten Daten ankommen (Abbildung 5.2).

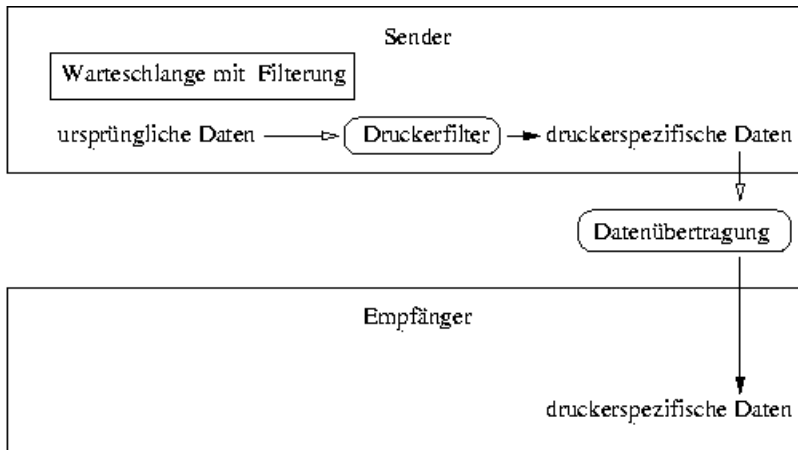


Abbildung 5.2: Überblick über den Ablauf beim Filtern

Im folgenden werden die Möglichkeiten der Filterung bei den obigen Beispielen aufgezeigt.

Fall B1: Mehrere Workstations, ein Print-Server und ein oder mehrere Printserver-Boxen bzw. Netzwerkdrucker: Am einfachsten und sinnvollsten ist die folgende Konfiguration in Abbildung 5.3 auf der nächsten Seite.

Fall B1b Man kann für jede Warteschlange mit Filterung auf dem Print-Server eine entsprechend konfigurierte Warteschlange ohne Filterung auf jeder Workstation anlegen, damit bei einem vorübergehenden Ausfall oder bei Überlast des Print-Servers die Druckaufträge auf den Workstations zwischengespeichert werden können. Dadurch können auf den Workstations jederzeit Ausdrücke erzeugt werden ohne warten zu müssen, bis der Print-Server wieder verfügbar ist. Der Nachteil hierbei ist, dass nun alle Warteschlangen auch auf jeder Workstation

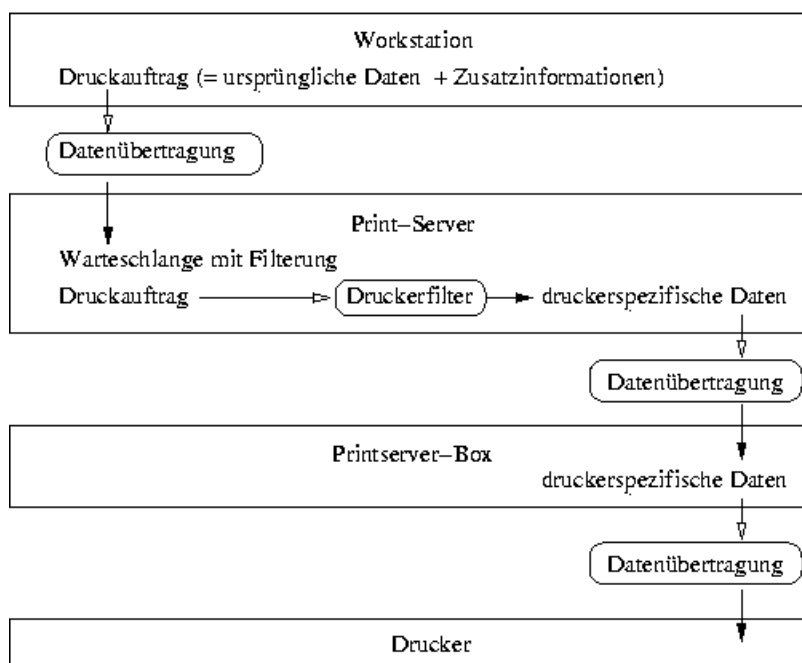


Abbildung 5.3: Konfiguration 1

zu konfigurieren sind (allerdings ohne die Filterung) und dass bei gewissen Änderungen der Warteschlangen auf dem Print-Server (z.B. Namensänderung oder neu hinzugekommene bzw. gelöschte Warteschlangen, aber nicht bei Änderungen der Filterung) die Konfigurationen auf allen Workstations anzupassen sind. Diese hochentwickelte Konfiguration sieht dann wie in Abbildung 5.4 auf der nächsten Seite aus.

Fall B1c Theoretisch könnte die Filterung auf jeder Workstation erfolgen und der Print-Server leitet die druckerspezifischen Daten nur noch an die Printserver-Boxen bzw. an die Netzwerkdrucker weiter. Aber damit würde der Print-Server zu einer Art grosse Printserver-Box kastriert, was in der Praxis normalerweise keinen Sinn macht, es sei denn, der Print-Server ist so leistungsschwach, dass die Filterung dort zur Überlastung führt. Der Nachteil hierbei ist, dass nun alle Warteschlangen auch auf jeder Workstation zu konfigurieren sind

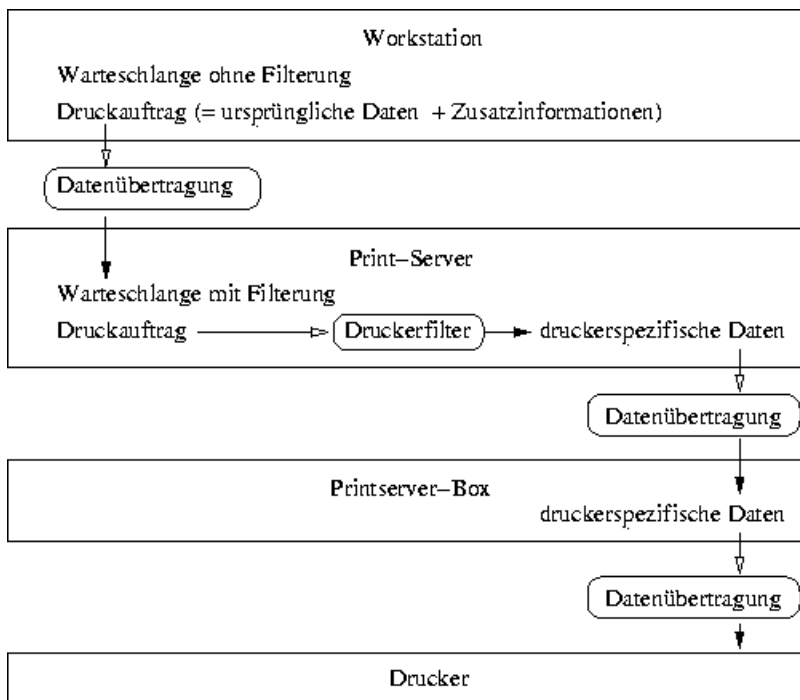


Abbildung 5.4: Konfiguration 2

(jeweils mit Filterung) und dass bei jeder Änderung die Konfigurationen auf allen Workstations anzupassen sind.

Diese Konfiguration sieht dann wie in Abbildung 5.5 auf der nächsten Seite aus.

Fall B2 Einige wenige Workstations, kein Print-Server und ein oder mehrere Printserver-Boxen bzw. Netzwerkdrucker: Die einzig mögliche Konfiguration ist auf jeder Workstation für jeden Drucker eine Warteschlange mit Filterung zu haben. Der Nachteil hierbei ist, dass nun alle Warteschlangen auf jeder Workstation zu konfigurieren sind (jeweils mit Filterung) und dass bei jeder Änderung die Konfigurationen auf allen Workstations anzupassen sind. Die Konfiguration sieht dann wie in Abbildung 5.6 auf Seite 160 aus.

Fall B3 Der vorige Fall sieht schon fast genauso aus, wie die Konfiguri-

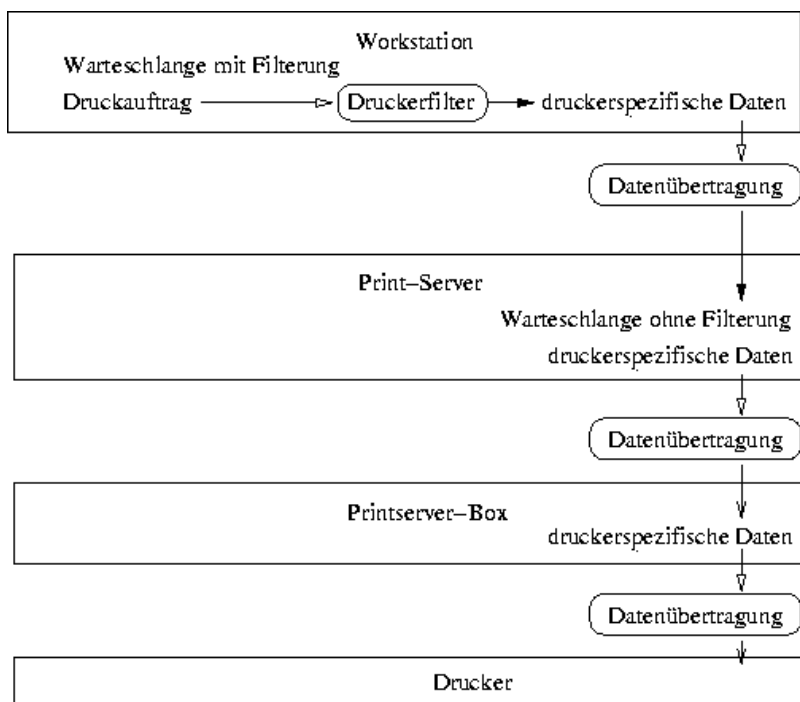


Abbildung 5.5: Konfiguration 3

on auf einem Einzelplatzsystem mit lokal angeschlossenem Drucker. Zum Vergleich die Konfiguration in Abbildung 5.7 auf Seite 161 für ein Einzelplatzsystem:

Wenn man die obigen Fälle von hier aus nacheinander rückwärts betrachtet, sieht man die Fortentwicklung von der Konfiguration auf einem Einzelplatzsystem mit lokal angeschlossenem Drucker zur hochentwickeltesten bzw. sinnvollsten Konfiguration für mehrere Workstations mit einem Print-Server für mehrere Printserver-Boxen bzw. Netzwerkdrucker.

5.9.5 Problemlösungen

TCP/IP-Netzwerk überprüfen Das TCP/IP-Netzwerk inklusive Namensauflösung muss ordnungsgemäß funktionieren.

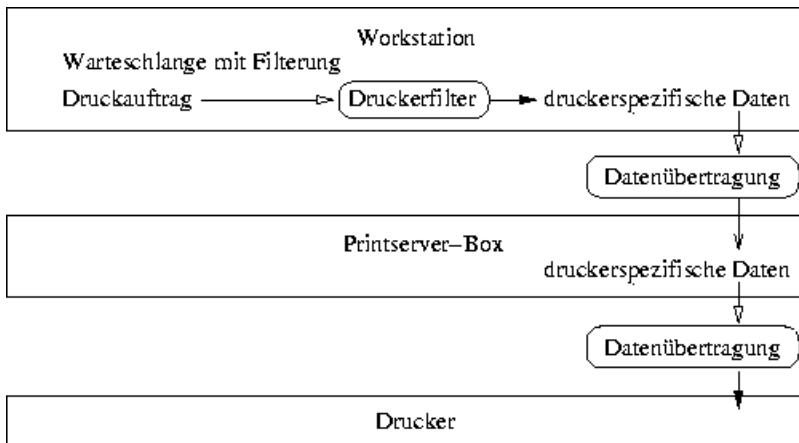


Abbildung 5.6: Konfiguration 4

Konfiguration des Filters prüfen Schließen Sie den Drucker direkt an der ersten parallelen Schnittstelle am Rechner an. Konfigurieren Sie den Drucker nur zum Test als lokalen Drucker, um etwaige Netzprobleme auszuschließen. Wenn der Drucker lokal funktioniert, haben Sie den passenden Ghostscript-Treiber und die anderen Parameter für die Konfiguration des Filters erhalten.

Einen entfernten lpd prüfen Mit den folgenden Kommando kann man testen, ob überhaupt eine TCP-Verbindung zum lpd (Port 515) auf dem Rechner *<host>* möglich ist:

```
netcat -z <host> 515 && echo ok || echo failed
```

Wenn keine Verbindung zum lpd möglich ist, dann läuft entweder der lpd nicht, oder grundlegende Netzwerkprobleme sind die Ursache.

Als Benutzer *root* kann man mit folgendem Kommando einen (ggf. sehr langen) Statusbericht für die Warteschlange *<queue>* auf dem (entfernten) Rechner *<host>* abfragen, sofern der dortige lpd läuft und Anfragen dorthin geschickt werden können:

```
echo -e "\004<queue>" \  
| netcat -w 2 -p 722 <host> 515
```

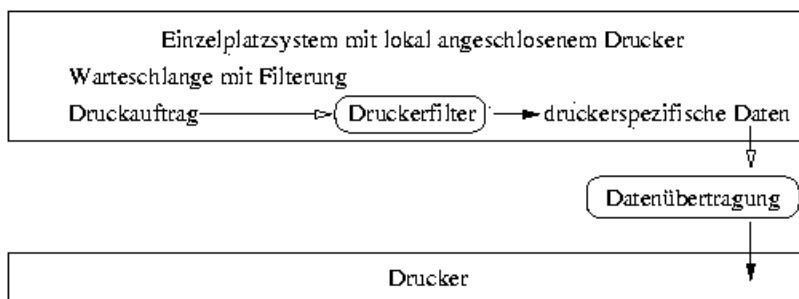


Abbildung 5.7: Konfiguration 5

Wenn keine Antwort vom lpd kommt, dann dann läuft entweder der lpd nicht, oder grundlegende Netzwerkprobleme sind die Ursache. Wenn eine Antwort vom lpd kommt, sollte diese klären, warum auf der Warteschlange queue auf dem Rechner host nicht gedruckt werden kann – Beispiele:

Beispiel 5.1: Fehlermeldung von lpd

```

lpd: your host does not have line printer access
lpd: queue does not exist
printer: spooling disabled
printer: printing disabled

```

Wenn eine derartige Antwort vom lpd kommt, liegt das Problem beim entfernten lpd.

Einen entfernten cupsd prüfen Mit folgendem Kommando kann man testen, ob es im Netzwerk einen CUPS-Netzwerk-Server gibt, denn dieser sollte über den UDP Port 631 seine Warteschlange standardmäßig alle 30 Sekunden broadcasten:

```
netcat -u -l -p 631 & PID=$! ; sleep 40 ; kill $PID
```

Nach 40 Sekunden Wartezeit sollte es eine Ausgabe in der folgenden Art geben, wenn ein CUPS-Netzwerk-Server broadcastet:

Beispiel 5.2: Broadcast vom CUPS-Netzwerk-Server

```
...
ipp://<host>.<domain>:631/printers/<queue>
```

Mit folgendem Kommando testet man, ob überhaupt eine TCP-Verbindung zum cupsd (Port 631) auf dem Rechner *<host>* möglich ist:

```
netcat -z <host> 631 && echo ok || echo failed
```

Wenn keine Verbindung zum cupsd möglich ist, dann läuft entweder der cupsd nicht, oder grundlegende Netzwerkprobleme sind die Ursache.

```
lpstat -h <host> -l -t
```

Damit erhält man einen (ggf. sehr langen) Statusbericht für alle Warteschlangen auf dem Rechner *<host>*, sofern der dortige cupsd läuft und Anfragen dorthin geschickt werden können.

```
echo -en "\r" \
| lp -d <queue> -h <host>
```

Damit kann man testen, ob die Warteschlange *<queue>* auf dem Rechner *<host>* einen Druckauftrag annimmt, wobei der Druckauftrag hier aus einem einzelnen Carriage-Return-Zeichen besteht — d. h. hierbei wird nur getestet, aber normalerweise sollte nichts gedruckt werden — und wenn, dann nur ein leeres Blatt.

Einen entfernten SMB-Server prüfen

Die grundlegende Funktion kann mit folgendem Befehl getestet werden:

```
echo -en "\r" \
| smbclient '/<HOST>/<SHARE>' '<<PASSWORD>' \
-c 'print -' -N -U '<USER>' \
&& echo ok || echo failed
```

Für *<HOST>* ist der Rechnernamen des Samba-Servers, für *<SHARE>* der Namen der entfernten Warteschlange (d. h. der Namen des Samba-Shares), für *<PASSWORD>* das Passwort und für *<USER>* den Benutzernamen einzusetzen. Hierbei wird nur getestet, aber normalerweise sollte nichts gedruckt werden — und wenn, dann nur ein leeres Blatt.

Mit folgendem Befehl können die frei verfügbaren Shares auf dem Rechner *<host>* angezeigt werden — siehe die Manualpage von `smbclient`:

```
smbclient -N -L <host>
```

Netzwerkdrucker oder Printserver-Box arbeitet nicht zuverlässig

Es gibt mitunter Probleme mit dem Druckerspooler, der in einer Printserver-Box läuft, sobald ein höheres Druckaufkommen vorliegt. Da es am Druckerspooler in der Printserver-Box liegt, kann man das nicht ändern. Man kann aber den Druckerspooler in der Printserver-Box umgehen, indem man den an der Printserver-Box angeschlossenen Drucker direkt via TCP-Socket anspricht.

Dadurch arbeitet die Printserver-Box nur noch als Umwandler zwischen den verschiedenen Formen der Datenübertragung (TCP/IP-Netzwerk und lokaler Druckeranschluss). Somit verhält sich der an der Printserver-Box angeschlossene Drucker wie ein lokal angeschlossener Drucker. So bekommt man auch direktere Kontrolle über den Drucker, als wenn der Spooler auf der Printserver-Box zwischengeschaltet wäre. Dazu muss der entsprechende TCP-Port auf der Printserver-Box bekannt sein. Bei angeschlossenem und eingeschaltetem Drucker an der Printserver-Box kann dieser TCP-Port normalerweise einige Zeit nach dem Einschalten der Printserver-Box mit dem Programm `nmap` aus dem Paket `nmap` ermittelt werden.

So liefert `nmap <IP-address>` bei einer Printserver-Box beispielsweise:

Port	State	Service
23/tcp	open	telnet
80/tcp	open	http
515/tcp	open	printer
631/tcp	open	cups
9100/tcp	open	jetdirect

Diese Ausgabe bedeutet:

- Man kann sich via `telnet` auf der Printserver-Box anmelden. So können dort grundlegende Informationen abfragt und grundlegende Konfigurationen vorgenommen werden.
- Via HTTP kann ein in der Printserver-Box laufender Web-Server angesprochen werden. Dieser liefert normalerweise detaillierte Informationen und ermöglicht detaillierte Konfigurationen.

- Über den Port 515 ist der in der Printserver-Box laufende Druckerspooler via LPD-Protokoll ansprechbar.
- Über den Port 631 ist der in der Printserver-Box laufende Druckerspooler via IPP-Protokoll ansprechbar.
- Über den Port 9100 ist der an der Printserver-Box angeschlossene Drucker via TCP-Socket ansprechbar.

Standardmässig prüft nmap nur eine gewisse Liste von allgemein bekannten Ports, die in `/usr/share/nmap/nmap-services` verzeichnet sind. Um alle möglichen Ports zu überprüfen, verwenden Sie `nmap -p <from_port>-<to_port> <IP-address>` (das kann dann sehr lange dauern) — vergleichen Sie dazu die Manual-Page `man nmap`.

Mit Befehlen der Art

```
echo -en "\rHello\r\n" | netcat -w 1 <IP-address> <port>
cat <file> | netcat -w 1 <IP-address> <port>
```

können Zeichenfolgen oder Dateien direkt an den betreffenden Port geschickt werden, um zu testen, ob der Drucker über diesen Port ansprechbar ist.

5.9.6 LPD-und-IPP Print-Server

LPD, IPP und CUPS

Standardmäßig unterstützt ein CUPS-Server nur das IPP-Protokoll. Aber das Programm `/usr/lib/cups/daemon/cups-lpd` aus dem Paket `cups` ermöglicht es, dass ein CUPS-Server auch Druckaufträge annehmen kann, die ihm via LPD-Protokoll an den Port 515 geschickt werden. Dazu ist der entsprechende Dienst für den `xinetd` zu aktivieren — normalerweise mit YaST2 oder manuell, indem die entsprechende Zeile in der Datei `/etc/xinetd.d/cups-lpd` aktiviert wird.

LPRng/lpfilter und CUPS

Es mag der Wunsch bestehen, beide Drucksysteme LPRng/lpfilter und CUPS auf demselben Rechner laufen zu haben, etwa weil ein bestehender LPD-Print-Server durch CUPS erweitert werden soll, oder weil für gewisse Spezialfälle das LPRng/lpfilter Drucksystem benötigt wird.

Grundsätzlich gibt es Schwierigkeiten, wenn beide Drucksysteme auf demselben Rechner laufen sollen. Hier werden die Problemstellen und die damit verbundenen Einschränkungen kurz angesprochen. Das Thema ist aber zu komplex, als dass hier eine Lösung beschrieben werden könnte.

- Die Druckerkonfiguration sollte nicht mit YaST2 erfolgen, denn die YaST2-Druckerkonfiguration ist für diesen Fall nicht ausgerichtet.
- Die Pakete `lprng` und `cups-client` stehen in Konflikt miteinander, denn sie enthalten Dateien, die denselben Namen haben z. B. `/usr/bin/lpr` und `/usr/bin/lp`. Das Paket `cups-client` darf daher nicht installiert sein. Die Folge ist, dass keine CUPS-Kommandozeilentools zur Verfügung stehen, sondern nur die für den LPRng. Dennoch kann unter der graphischen Oberfläche mit `xpp` oder `kprinter` auf CUPS-Warteschlangen gedruckt werden und auch von allen Anwendungsprogrammen, die CUPS direkt unterstützen.
- Standardmäßig legt der `cupsd` beim Starten die Datei `/etc/printcap` neu an, die nur die Namen aller CUPS-Warteschlangen enthält. Dies geschieht aus Kompatibilitätsgründen, denn viele Anwendungsprogramme lesen die Warteschlangennamen aus `/etc/printcap`, um diese im Drucken-Menü anbieten zu können. Das muss für den `cupsd` abgeschaltet werden, so dass `/etc/printcap` zur alleinigen Verwendung für das LPRng/lpfilter Drucksystem dient. Die Folge ist, dass Anwendungsprogramme, die nur die Warteschlangennamen aus `/etc/printcap` verwenden, auch nur diese lokalen Warteschlangen anzeigen, aber nicht die netzwerkweit verfügbaren CUPS-Warteschlangen.

Weiterführende Hinweise zum Druckerbetrieb

In diesem Kapitel wird weiterführendes Hintergrundwissen zum Druckerbetrieb geliefert. Das Nachvollziehen der Beispiele ermöglicht eine Einsicht in die Zusammenhänge beim Druckerbetrieb. Es dient dazu, Lösungen für spezielle Anwendungsfälle zu finden.

6.1	Manuelle Konfiguration lokaler Druckerschnittstellen	168
6.2	Manuelle Konfiguration von LPRng/lpdfilter	173
6.3	Der Druckerspooler LPRng	173
6.4	Kommandozeilentools für den LPRng	175
6.5	Der Druckerfilter des LPRng/lpdfilter Drucksystems	180
6.6	Etwas über Ghostscript	190
6.7	Etwas über a2ps	194
6.8	PostScript-Umformatierung mit den psutils	195
6.9	Zur Kodierung von ASCII-Text	199

6.1 Manuelle Konfiguration lokaler Druckerschnittstellen

6.1.1 Parallele Schnittstellen

Der Anschluss eines Druckers an ein Linux-System erfolgt in der Regel über eine parallele Schnittstelle. Ein Drucker an einer parallelen Schnittstelle wird über das `parport`-Subsystem des Kernels angesprochen. Die grundlegende Konfiguration einer parallelen Schnittstelle mit YaST ist im Abschnitt 5.3.4 auf Seite 112 erläutert, daher sollen hier nur einige weitergehende Informationen vorgestellt werden.

Dem `parport`-Subsystem sind die parallelen Schnittstellen durch Laden architekturenspezifischer Kernelmodule bekannt zu machen. So können mehrere, in Kette geschaltete Geräte (zum Beispiel ein Parallelport-ZIP-Laufwerk und ein Drucker) über *eine* parallele Schnittstelle *gleichzeitig* bedient werden. Die Zählung der Gerätedateien für Parallelport-Drucker beginnt bei `/dev/lp0`. Um über die erste parallele Schnittstelle drucken zu können, müssen beim SUSE Standardkernel die Module `parport`, `parport_pc` und `lp` geladen werden. Dies erledigt der `kmod` *Kernel Module Loader* in der Regel automatisch, sobald auf die Gerätedatei (zum Beispiel `/dev/lp0`) zum ersten Mal zugegriffen wird.

Wenn das Kernelmodul `parport_pc` ohne spezielle Parameter geladen wird, versucht es, die parallelen Schnittstellen automatisch zu erkennen und zu konfigurieren. In seltenen Fällen funktioniert das nicht und es kann zum plötzlichen Systemstillstand kommen. Dann müssen die korrekten Parameter für das `parport_pc` Modul explizit manuell konfiguriert werden. Deswegen kann, wie im Abschnitt 5.3 auf Seite 109 beschrieben, die automatische Druckererkennung bei YaST verhindert werden.

Manuelle Konfiguration der parallelen Schnittstelle

Die parallele Schnittstelle `/dev/lp0` wird durch einen Eintrag in `/etc/modules.conf` konfiguriert (Datei 6.1).

Beispiel 6.1: /etc/modules.conf: Erste parallele Schnittstelle

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378 irq=None
```

Bei `io` steht die IO-Adresse der parallelen Schnittstelle. Bei `irq` steht `none` als Voreinstellung für den Polling-Betrieb oder der Interrupt der parallelen Schnittstelle. Der Polling-Betrieb ist unproblematischer als der Interrupt-Betrieb, da Interrupt-Konflikte vermieden werden. Allerdings gibt es Motherboards und/oder Drucker, die nur im Interrupt-Betrieb korrekt funktionieren. Außerdem ermöglicht der Interrupt-Betrieb, dass der Drucker auch bei hoher Systemlast noch hinreichend Daten erhält.

Damit diese Einstellungen funktionieren, müssen im BIOS oder über die Firmware des Rechners folgende Werte (sofern vorhanden) für die parallele Schnittstelle eingestellt sein:

- IO-Adresse 378 (hexadezimal)
- Interrupt 7 (im Polling-Betrieb nicht relevant)
- Modus `Normal`, `SPP` oder `Output-Only` (andere Modi funktionieren nicht immer)
- DMA ist abgeschaltet (sollte im Modus `Normal` abgeschaltet sein)

Wenn der Interrupt 7 noch frei ist, dann kann mit dem Eintrag in der Datei 6.2 der Interrupt-Betrieb aktiviert werden.

***Beispiel 6.2:** `/etc/modules.conf`: Interrupt-Betrieb für die erste parallele Schnittstelle*

```
alias parport_lowlevel parport_pc
options parport_pc io=0x378 irq=7
```

Bevor der Interrupt-Betrieb aktiviert wird, ist der Datei `/proc/interrupts` zu entnehmen, welche Interrupts bereits verwendet werden, wobei hier nur die Interrupts angezeigt werden, die momentan in Gebrauch sind. Dies kann sich je nach aktiv benutzter Hardware ändern. Der Interrupt für die parallele Schnittstelle darf nicht anderweitig in Gebrauch sein. Im Zweifel ist der Polling-Betrieb zu nehmen.

Aktivierung und Test einer parallelen Schnittstelle

Nach einem Reboot steht die parallele Schnittstelle zur Verfügung. Statt eines Reboots genügt es, als Benutzer `root` die Liste der Abhängigkeiten der Kernelmodule zu aktualisieren, die Kernelmodule, die die parallele Schnittstelle betreffen, zu entladen. . .

```
depmod -a 2>/dev/null
rmmod lp
rmmod parport_pc
rmmod parport
```

...und wieder neu zu laden:

```
modprobe parport
modprobe parport_pc
modprobe lp
```

Ist der Drucker in der Lage, ASCII-Text zu drucken, sollte man als Benutzer `root` mit folgendem Befehl eine Seite mit dem Wort `Hello` ausdrucken können:

```
echo -en "\rHello\r\f" >/dev/lp0
```

Hierbei ist das Wort `Hello` passend für einen Ausdruck umgeben von dem ASCII-Zeichen `\r` für Wagenrücklauf und gefolgt von dem ASCII-Zeichen `\f` das einen Seitenvorschub auslöst.

6.1.2 USB-Anschluss

Im BIOS des Rechners muss ein Interrupt für USB aktiviert sein. Bei einem Award-BIOS ist dazu im Menü 'PNP AND PCI SETUP' der Eintrag 'USB IRQ' auf `Enabled` zu setzen. Je nach BIOS-Version werden auch andere Bezeichnungen verwendet.

Testen Sie, ob der USB-Drucker ansprechbar ist, indem Sie als Benutzer `root` eingeben:

```
echo -en "\rHello\r\f" >/dev/usb/lp0
```

Vorausgesetzt, es ist nur ein einziger USB-Drucker angeschlossen und dieser Drucker kann ASCII-Text drucken, sollte eine Seite mit dem Wort `Hello` ausgegeben werden.

Manche USB-Drucker brauchen eine spezielle Steuersequenz, bevor Daten über USB angenommen werden. Informationen hierzu finden sich auch in der Support-Datenbank <http://sdb.suse.de/de/sdb/html> unter dem Stichwort `Epson` und `usb`.

Normalerweise sollte Hersteller und Produktbezeichnung des Druckers in der Ausgabe des folgenden Kommandos erscheinen:

```
cat /proc/bus/usb/devices
```

Wenn hier weder Hersteller noch Produkt angezeigt werden, hat das normalerweise folgende Ursachen:

- Das USB-System hat das Gerät (noch) nicht erkannt – evtl. weil der USB-Drucker ausgeschaltet ist. Der USB-Drucker kann dann nicht angesprochen werden.
- Das USB-System hat zwar das Gerät erkannt, aber es kennt weder Hersteller- noch Produktbezeichnung des Druckers und zeigt daher nichts an. Der USB-Drucker kann dann aber angesprochen werden.

Manchmal kommt es vor, dass der USB-Drucker nicht mehr angesprochen werden kann, zum Beispiel wenn man während eines Ausdrucks den USB-Stecker abzieht. Zumeist sollte es genügen, diese Befehle zu verwenden, um das USB-System neu zu starten:

```
rhotplug stop  
rhotplug start
```

Wenn das nicht hilft, müssen alle Prozesse, die auf `/dev/usb/lp0` zugreifen, beendet und die Kernelmodule, die den USB-Drucker betreffen, entladen und wieder neu geladen werden. Prüfen Sie vorher mit `lsmod`, welche USB-Module geladen sind (ob `usb-uhci` oder `usb-ohci` oder `uhci`) und ob noch weitere Modul-Abhängigkeiten bestehen; folgende Anzeige besagt, dass das Modul `usbcore` noch von den Modulen `printer` und `usb-uhci` benötigt wird:

```
usbcore ... [printer usb-uhci]
```

Daher müssen in diesem Fall die Module `printer` und `usb-uhci` vor dem Modul `usbcore` entladen werden. Geben Sie als Benutzer `root` folgende Befehle ein (anstelle von `usb-uhci` je nach System auch `uhci` oder `usb-ohci`):

```
fuser -k /dev/usb/lp0  
rhotplug stop  
rmmod printer  
rmmod usb-uhci  
umount usbdevfs  
rmmod usbcore
```

```
modprobe usbcore
mount usbdevfs
modprobe usb-uhci
modprobe printer
rchtotplug start
```

Sind mehrere USB-Drucker angeschlossen, ist Folgendes zu beachten: Das USB-Subsystem erkennt angeschlossene USB-Drucker automatisch. Der erste USB-Drucker, der erkannt wird, ist über das Device `/dev/usb/lp0` ansprechbar. Der zweite USB-Drucker, der erkannt wird, ist über das Device `/dev/usb/lp1` ansprechbar. Je nach Druckermodell werden ausgeschaltete Drucker trotzdem noch automatisch erkannt oder nicht. Das liegt daran, dass manche Drucker auch im ausgeschalteten Zustand noch über den USB-Anschluss abgefragt werden können. Um ein Durcheinander der USB-Devices zu vermeiden, sollten vor dem Booten von Linux immer alle USB-Drucker eingeschaltet sein und während des Betriebs möglichst eingeschaltet bleiben.

6.1.3 IrDA-Druckerschnittstelle

Es wird eine parallele Schnittstelle über die Infrarotverbindung emuliert. Der Treiber im Linuxkernel stellt eine simulierte parallele Schnittstelle unter dem Device `/dev/irlpt0` zur Verfügung. Ein Drucker über die Infrarotschnittstelle wird also genauso angesprochen wie ein Drucker am Parallelport, nur dass `/dev/irlpt0` statt `/dev/lp0` verwendet wird.

Testen Sie, ob der IrDA-Drucker ansprechbar ist, indem Sie als Benutzer `root` eingeben:

```
echo -en "\rHello\r\f" >/dev/irlpt0
```

Vorausgesetzt, der Drucker kann ASCII-Text drucken, sollte eine Seite mit dem Wort `Hello` ausgegeben werden.

In jedem Fall sollte der Drucker in der Ausgabe des folgenden Kommandos `irdadump` erscheinen. Gibt es den `irdadump`-Befehl nicht, dann ist das `irda` zu installieren. Wenn bei `irdadump` der Drucker nicht angezeigt wird, kann er nicht angesprochen werden. Wird hier überhaupt nichts angezeigt, dann ist wahrscheinlich der IrDA-Systemdienst nicht gestartet, denn dieser wird nicht automatisch beim Booten gestartet. Der IrDA-Systemdienst kann mit folgenden Kommandos gestartet und gestoppt werden:

```
rcirda start
rcirda stop
```


6.1.4 Serielle Schnittstellen

Wie ein Drucker an der seriellen Schnittstelle betrieben werden kann, ist für den LPRng-Spooler im *LPRng-Howto* unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html` und dort insbesondere unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#SECSERIAL` und in der Manualpage von `printcap` beschrieben. Informationen finden sich auch in der Support-Datenbank unter dem Stichwort `seriell`.

6.2 Manuelle Konfiguration von LPRng/lpfilter

Normalerweise wird das Drucksystem mit YaST konfiguriert, wie es im Abschnitt 5.3 auf Seite 109 beschrieben ist. Zusätzlich gibt es für das LPRng/lpfilter Drucksystem das Programm `lprsetup`, das rein kommandozeilenorientiert verwendet wird.

Wenn ein Drucker mit YaST konfiguriert wird, sammelt YaST die nötigen Informationen und ruft dann zur Konfiguration des LPRng/lpfilter Drucksystems `lprsetup` mit den nötigen Optionen auf, welches die Konfiguration dann tatsächlich anlegt.

Das Programm `lprsetup` ist als Experten-Tool gedacht. Im Gegensatz zu YaST hilft `lprsetup` dem Anwender nicht dabei, die richtigen Werte für die einzelnen Optionen zu finden. Mit `lprsetup -help` werden die möglichen Optionen kurz erläutert und die Manualpage von `lprsetup` bzw. die Manualpage von `lpfilter` liefern weitere Informationen.

Zu Informationen bzgl. Ghostscript-Treiber und treiberspezifischer Parameter siehe die Abschnitte 5.2.2 auf Seite 106 und 6.6 auf Seite 190.

6.3 Der Druckerspooler LPRng

Als Druckerspooler des LPRng/lpfilter Drucksystems wird der LPRng (`lprng`) verwendet.

Der Druckerspooler `lpd` *Line Printer Daemon* wird normalerweise beim Systemstart automatisch aktiviert, indem das Skript `/etc/init.d/lpd` aufgerufen wird. Manuell kann der Druckerspooler, der als Daemon im Hintergrund läuft, so gestartet und gestoppt werden:

```
rclpd start  
rclpd stop
```

Die Konfigurationsdateien für den LPRng sind:

/etc/printcap Konfiguration der einzelnen Warteschlangen

/etc/lpd.conf globale Konfiguration des Spoolers

/etc/lpd.perms Konfiguration der Zugriffsrechte

Bei `rclpd start` wird gemäß `/etc/init.d/lpd` auch `checkpc -f` aufgerufen, was anhand der Einträge in `/etc/printcap` ggf. die Spoolverzeichnisse `/var/spool/lpd/*` anlegt und die Zugriffsrechte passend setzt.

Der Druckerspooler stellt beim Start anhand der Einträge in `/etc/printcap` fest, welche Druckwarteschlangen definiert sind. Seine Aufgabe ist, die Ausführung der gespoolten Aufträge (*Jobs*) zu organisieren:

- Er verwaltet die lokalen Warteschlangen und schickt die Datendatei eines Jobs ggf. durch den Druckerfilter und dann entweder direkt zum Drucker oder weiter an eine andere Warteschlange.
- Er berücksichtigt die Reihenfolge der Jobs in den Druckwarteschlangen.
- Er überwacht den Status der Warteschlangen und Drucker und gibt auf Verlangen Auskunft darüber.
- Er lauscht am Port 515, um Druckaufträge von entfernten Rechnern für lokale Warteschlangen anzunehmen (bzw. er weist sie ggf. ab).
- Er leitet Druckaufträge an Warteschlangen auf entfernten Rechnern an den dortigen Druckerspooler (also den dortigen Port 515) weiter.

Die Details sind für den LPRng-Spooler im *LPRng-Howto* unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html` und in der Manualpage von `printcap` und der Manualpage `lpd` beschrieben.

6.3.1 Drucken aus Anwendungsprogrammen

Anwendungsprogramme verwenden hier den `lpr`-Befehl zum Drucken. Wählen Sie dazu im Anwendungsprogramm den Namen einer bestehenden Warteschlange (zum Beispiel `color`) oder geben Sie in der Druckmaske des Anwendungsprogramms das passende Druck-Kommando (zum Beispiel `lpr -Pcolor`) ein.

Auf der Kommandozeile druckt man mit dem Befehl `lpr -Pcolor <Dateiname>`, wobei `<Dateiname>` durch den Namen der zu druckenden Datei zu ersetzen ist. Durch die Option `-P` kann die Warteschlange explizit bestimmt werden. Mit `lpr -Pcolor Dateiname` wird beispielsweise die Warteschlange `color` verwendet.

6.4 Kommandozeilentools für den LPRng

Die Kommandozeilentools sind detailliert im *LPRng-Howto* unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPRNGCLIENTS` erläutert, daher hier nur eine kurze Zusammenfassung:

6.4.1 Für lokale Warteschlangen

Druckaufträge erzeugen

Der `lpr`-Befehl ist im *LPRng-Howto* unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPR` erläutert, hier nur grundlegende Informationen:

Normalerweise druckt man mit `lpr -P <warteschlange> <datei>`. Wenn Sie die Option `-P<warteschlange>` weglassen, ist die Voreinstellung der Inhalt der Umgebungsvariablen `PRINTER`. Dies gilt ebenso für die Befehle `lpq` und `lprm` — siehe die Manualpage von `lpr`, die Manualpage von `lpq` und die Manualpage von `lprm`. Die Umgebungsvariable `PRINTER` wird beim Anmelden automatisch gesetzt, kann mit dem Befehl `echo $PRINTER` angezeigt werden und mit `export PRINTER=<warteschlange>` auf eine (andere) Warteschlange gesetzt werden.

Status anzeigen

`lpq -P<warteschlange>` zeigt die Druckaufträge in der angegebenen Warteschlange an. Wird beim LPRng-Spooler als Warteschlange `all` eingegeben, werden alle Druckaufträge in allen Warteschlangen angezeigt.

Mit `lpq -s -P<warteschlange>` wird minimale Information angezeigt und `lpq -l -P<warteschlange>` liefert mehr Information.

Mit `lpq -L -P<warteschlange>` wird ein detaillierter Statusbericht ausgegeben, der zur Fehlerdiagnose dient.

Für weitere Informationen siehe unten den Abschnitt *Status für entfernte Warteschlangen anzeigen* und die Manualpage von `lpq` und `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPQ` im *LPRng-Howto*.

Druckaufträge löschen

`lprm -P<warteschlange> <jobnummer>` löscht den Druckauftrag mit der angegebenen Jobnummer aus der angegebenen Warteschlange, sofern der Druckauftrag dem Benutzer gehört, der den `lprm`-Befehl aufgerufen hat. Ein Druckauftrag gehört dem Benutzer auf dem Rechner, der den Druckauftrag gestartet hat. Dieser Benutzer wird durch den `lpq`-Befehl angezeigt. Auch die Jobnummer wird durch den `lpq`-Befehl angezeigt.

Mit dem Befehl `lprm -Pa11 all` werden alle Druckaufträge aus allen Warteschlangen gelöscht, die der Benutzer, der den `lprm`-Befehl gegeben hat, löschen darf. Der Benutzer `root` darf jegliche Druckaufträge (auch in allen Warteschlangen) löschen.

Weitere Informationen in der Manualpage von `lprm` und unter `file:/usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPRM` im *LPRng-Howto*.

Steuerung der Warteschlangen

Der Befehl `lpc option <warteschlange>` zeigt den Status der angegebenen Warteschlangen an und ermöglicht es, diesen zu verändern. Die wichtigsten Optionen sind:

help liefert eine kurze Übersicht der Optionen.

status <warteschlange> gibt einen Statusbericht.

disable <warteschlange> stoppt die Aufnahme neuer Jobs in die Warteschlange.

enable *(warteschlange)* gibt die Warteschlange für die Aufnahme neuer Jobs frei.

stop *(warteschlange)* stoppt das Ausdrucken von Jobs aus der Warteschlange; der gerade im Druck befindliche Job wird noch beendet.

start *(warteschlange)* nimmt das Ausdrucken von Jobs aus der Warteschlange wieder auf.

down *(warteschlange)* wirkt wie `disable` plus `stop`.

up *(warteschlange)* hat dieselbe Wirkung wie `enable` plus `start`.

abort *(warteschlange)* ist identisch zu `down`, nur dass ein gerade im Druck befindlicher Job sofort abgebrochen wird. Die Jobs bleiben erhalten und können nach einem Restart der Warteschlange (`up`) weiter bearbeitet werden.

Für Veränderungen an den Druckwarteschlangen brauchen Sie `root`-Rechte. Sie können diese Kommandos gleich in der Kommandozeile mitgeben (zum Beispiel `lpc status all`). Oder Sie rufen `lpc` ohne Parameter auf: Dann wird ein Dialogmodus mit der Eingabeaufforderung *Prompt* `lpc>` gestartet, der die Eingabe obiger Optionen erwartet. Mit `quit` oder `exit` beenden Sie den Dialog.

Liefert `lpc status all` beispielsweise

Printer	Printing	Spooling	Jobs	Server	Subserver
lp@earth	enabled	enabled	2	123	456
color@earth	disabled	disabled	0	none	none
laser@earth	disabled	enabled	8	none	none

so ist die Warteschlangen `lp` komplett eingeschaltet und enthält zwei Druckaufträge wovon einer gerade gedruckt wird. Die Warteschlange `color` ist komplett abgeschaltet. Bei der Warteschlange `laser` ist zum Beispiel wegen vorübergehender Reparaturarbeiten am Drucker nur das Ausdrucken abgeschaltet, aber es können dennoch weiterhin Druckaufträge erzeugt werden, die sich in der Warteschlange sammeln (hier acht Stück).

Weitere Informationen in der Manualpage von `lpc` und unter `file: /usr/share/doc/packages/lprng/LPRng-HOWTO.html#LPC` im *LPRng-Howto*.

6.4.2 Für entfernte Warteschlangen

Hier ist *<print-server>* durch den Namen oder die IP-Adresse des Print-Servers zu ersetzen und *<warteschlange>* muss eine Warteschlange auf dem Print-Server sein.

Druckaufträge erzeugen

Beim LPRng-Spooler können direkt mit dem `lpr`-Befehl auch entfernte Warteschlangen wie folgt angesprochen werden: `lpr -P<warteschlange>@<print-server> <datei>`. Voraussetzung ist, dass der Print-Server so konfiguriert wurde, dass man auf dessen Warteschlangen auch drucken darf, was beim LPRng standardmäßig möglich ist.

Status anzeigen

Mit folgenden Befehlen kann die entfernte Warteschlange abgefragt werden:

```
lpq -P<warteschlange>@<print-server>
lpq -s -P<warteschlange>@<print-server>
lpq -l -P<warteschlange>@<print-server>
lpq -L -P<warteschlange>@<print-server>
```

und

```
lpc status <warteschlange>@<print-server>
lpc status all@<print-server>
```

Insbesondere mit `lpq -s -Pall@<print-server>` oder `lpc status all@<print-server>` können die Namen aller Warteschlangen auf dem Print-Server ermittelt werden, wenn auch auf diesem LPRng verwendet wird.

Ist kein Ausdruck über die entfernte Warteschlange möglich, dann sollte die Statusabfrage helfen. Mit `lpq -L -P<warteschlange>@<print-server>` kann der detaillierte Statusbericht zur Ferndiagnose angezeigt werden, sofern auch auf dem Print-Server LPRng verwendet wird.

Druckaufträge löschen

Mit folgenden Befehlen können alle die Druckaufträge in entfernten Warteschlangen gelöscht werden, die man selbst erzeugt hat:

```
lprm -P<warteschlange>@<print-server> <jobnummer>
lprm -P<warteschlange>@<print-server> all
lprm -Pall@<print-server> all
```

Insbesondere hat `root` keine Sonderrechte bei entfernten Warteschlangen. Die Angabe `all` funktioniert nur, wenn auch auf dem Print-Server LPRng verwendet wird.

6.4.3 Störungsbehebung mit obigen Befehlen beim LPRng

Druckaufträge bleiben in den Warteschlangen erhalten, wenn Sie während eines Druckvorgangs den Rechner herunterfahren und dann Linux neu starten – einen eventuell fehlerhaften Druckauftrag müssen Sie mit den oben vorgestellten Befehlen aus der Warteschlange entfernen.

Kommt es zum Beispiel zu einer Störung in der Kommunikation zwischen Rechner und Drucker so kann der Drucker mit den gesendeten Daten nichts sinnvolles anfangen und es kommt zu dem Problem, dass Unmengen Papier mit sinnlosen Zeichen vollgedruckt werden.

1. Entnehmen Sie zuerst alles Papier bei Tintenstrahldruckern bzw. öffnen Sie die Papierschächte bei Laserdruckern, damit das Drucken abgebrochen wird.
2. Da der Druckauftrag erst dann aus der Warteschlange entfernt wird, nachdem er komplett an den Drucker geschickt wurde, wird er meist noch in der Warteschlange stehen. Prüfen Sie mit `lpq` oder `lpc status`, aus welcher Warteschlange gerade gedruckt wird, und löschen Sie mit `lprm` den Druckauftrag.
3. Evtl. werden noch einige Daten an den Drucker übertragen, obwohl der Druckauftrag aus der Warteschlange gelöscht ist. Mit dem Befehl `fuser -k /dev/lp0` für einen Drucker am Parallelport bzw. `fuser -k /dev/usb/lp0` für einen USB-Drucker können ggf. alle Prozesse beendet werden, die noch auf den Drucker zugreifen.
4. Setzen Sie den Drucker komplett zurück, indem Sie ihn einige Zeit vom Stromnetz trennen. Danach legen Sie das Papier wieder ein und schalten den Drucker an.

6.5 Der Druckerfilter des LPRng/lpfilter Drucksystems

Als Druckerfilter wird der `lpfilter` (`lpfilter`) verwendet. Es folgt eine detaillierte Beschreibung des Ablauf eines Druckauftrages. Für eine exakte Analyse des Druckerfilters sind die Skripte des Druckerfilters (insbesondere `/usr/lib/lpfilter/bin/if`) durchzusehen, ggf. ist gemäß dem Abschnitt 6.5.3 auf Seite 189 vorzugehen.

1. Der Druckerfilter (`/usr/lib/lpfilter/bin/if`) bestimmt die an ihn direkt vom Druckerspooler übergebenen Optionen bzw. liest er sie aus dem control file des Druckjobs und passend zur verwendeten Warteschlange aus den Dateien `/etc/printcap` und `/etc/lpfilter/<warteschlange>/conf` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen).
2. Wenn es eine `ascii`-Warteschlange ist, wird der Druckerfilter gezwungen, die zu druckenden Daten wie ASCII-Text zu behandeln. Wenn es keine `ascii`-Warteschlange ist, versucht der Druckerfilter den Typ der zu druckenden Daten automatisch zu bestimmen. Der Typ der zu druckenden Daten wird durch das Skript `/usr/lib/lpfilter/bin/guess` bestimmt, das das Kommando `file` auf die zu druckenden Daten angewendet und mit dessen Ausgabe wird gemäß der Angaben in der Datei `/etc/lpfilter/types` der Typ der zu druckenden Daten festgesetzt.
3. Je nach Typ der zu druckenden Daten und nach Art der Warteschlange erfolgt die weitere Umwandlung in druckerspezifische Daten:
 - Wenn es eine `raw`-Warteschlange ist, werden die zu druckenden Daten normalerweise direkt an den Drucker (oder an eine andere Warteschlange) weitergeleitet, es kann aber auch gemäß der Einstellungen in `/etc/lpfilter/<warteschlange>/conf` eine einfache Umkodierung mit `recode` erfolgen. Für eine absolute `raw`-Warteschlange – also ganz ohne den `lpfilter` – ist die Zeile `:if=/usr/lib/lpfilter/bin/if:\ in /etc/printcap` bei der entsprechenden Warteschlange zu entfernen.
 - Wenn es keine `raw`-Warteschlange ist:
 - (a) Wenn die Daten nicht PostScript sind, werden sie zuerst durch einen Aufruf von `/usr/lib/lpfilter/filter/typ2ps` nach PostScript umgewandelt (hier ist `typ` durch

den tatsächlich bestimmten Typ der zu druckenden Daten zu ersetzen). Insbesondere ASCII-Text wird gemäß `/usr/lib/lpddfilter/filter/ascii2ps` mit dem Programm `a2ps` gemäß der für die Warteschlange konfigurierten länderspezifischen Kodierung passend in PostScript umgewandelt, so dass länderspezifische Sonderzeichen auch in einfachem Text korrekt druckbar sind; siehe dazu die Manupage von `a2ps`.

- (b) Die PostScript-Daten können ggf. nochmals umformatiert werden, sofern ein passendes Skript unter `/etc/lpddfilter/<warteschlange>/pre` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen) existiert.
- (c) Die PostScript-Daten werden ggf. in eine andere Druckersprache umgewandelt.
 - ▷ Wenn ein PostScript-Drucker angeschlossen ist, werden die PostScript-Daten direkt an den Drucker (oder an eine andere Warteschlange) geschickt. Ggf. werden aber zusätzlich die Bash-Funktionen `duplex` und `tray`, die in `/usr/lib/lpddfilter/global/functions` definiert sind, aufgerufen, um Duplexdruck oder Papierschachtauswahl über PostScript-Kommandos zu ermöglichen – vorausgesetzt der PostScript-Drucker kann diese Kommandos entsprechend verarbeiten.
 - ▷ Wenn kein PostScript-Drucker angeschlossen ist, wird Ghostscript mit einem zur Druckersprache des jeweiligen Druckermodells passenden Ghostscript-Treiber verwendet, um die druckerspezifischen Daten zu erzeugen, die dann an den Drucker (oder an eine andere Warteschlange) geschickt werden.
 Die Parameter für den Ghostscript-Aufruf sind entweder in der `/etc/printcap` direkt in der `cm`-Zeile oder in der Datei `/etc/lpddfilter/<warteschlange>/upp` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen) gespeichert.
 Die Ausgabe von Ghostscript kann ggf. nochmals umformatiert werden, sofern ein passendes Skript unter `/etc/lpddfilter/<warteschlange>/post` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen) existiert.
- (d) Die druckerspezifischen Daten werden an den Drucker

(oder an eine andere Warteschlange) geschickt. Dabei können vor und nach den druckerspezifischen Daten noch druckerspezifische Steuersequenzen geschickt werden, wenn diese in `/etc/lpfilter/<warteschlange>/conf` eingetragen wurden.

6.5.1 Konfiguration des lpdfilter

Normalerweise wird das Drucksystem mit YaST konfiguriert, wie es im Abschnitt 5.3 auf Seite 109 beschrieben ist, insbesondere wird dabei auch der lpdfilter konfiguriert.

Wird der lpdfilter mit YaST konfiguriert, so läuft die Filterung beim LPRng/lpfilter-Drucksystem wie folgt ablaufen:

```
zu druckende Daten
|
v
lpdfilter: nur noch Umwandlung nach PostScript
|
| |---- PPD-Datei passend zum Druckermodell
| |    (/etc/lpfilter/warteschlange/ppd)
v   v
foomatic-rip: Umwandlung in die Druckersprache mit Ghostscript
|
v
Drucker
```

Für spezielle Einstellungen sind die Konfigurationsdateien des Druckerfilters manuell anzupassen. Jede Warteschlange hat ihre eigene separate Konfigurationsdatei `/etc/lpfilter/<warteschlange>/conf` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen), die auch Informationen zu jeder Option enthält.

6.5.2 Eigene Ergänzungen für den lpdfilter

1. Wenn die zu druckenden Daten nicht PostScript sind, werden sie standardmäßig durch einen Aufruf von `/usr/lib/lpfilter/filter/typ2ps` nach PostScript umgewandelt (hier ist `typ` durch den Typ der zu druckenden Daten zu ersetzen).

Wenn unter `/etc/lpfilter/<warteschlange>/typ2ps` ein passendes Skript abgelegt wird, wird dieses verwendet, um die Daten nach

PostScript umzuwandeln. Dieses Skript bekommt die zu druckenden Daten über `stdin` und hat sie über `stdout` PostScript auszugeben.

2. Die PostScript-Daten können ggf. nochmals umformatiert werden, sofern ein passendes Skript unter `/etc/lpfilter/<warteschlange>/pre` existiert. Auch eigene sog. PostScript-Preloads können hier mit einem passenden Skript hinzugeladen werden. Dieses Skript bekommt PostScript-Daten über `stdin` und hat sie über `stdout` PostScript auszugeben. Programme, um PostScript-Daten umzuformatieren, finden sich im `psutils`. Insbesondere das Programm `pstops` ermöglicht weitreichende Umformatierungen; siehe dazu die Manualpage von `pstops`.
3. Spezielle Ghostscript Parameter: Bei der Konfiguration mit YaST2 werden die Parameter für den Ghostscript-Aufruf in der Datei `/etc/lpfilter/<warteschlange>/upp` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen) gespeichert und in dieser Datei können spezielle Ghostscript Parameter manuell eingetragen werden. Zu Ghostscript Parametern siehe den Abschnitt 6.6 auf Seite 190.
4. Auch die Ausgabe von Ghostscript kann ggf. nochmals umformatiert werden, sofern ein passendes Skript unter `/etc/lpfilter/<warteschlange>/post` (hier ist `<warteschlange>` durch den tatsächlichen Namen der Warteschlange zu ersetzen) existiert. Dieses Skript bekommt die Ausgabe von Ghostscript über `stdin` und es hat druckerspezifische Daten über `stdout` auszugeben.

Ein hardwareunabhängiges Beispiel

Angenommen es gibt eine Warteschlange `test` bei der ASCII-Text mit vorangestellten Zeilennummern gedruckt werden soll und bei jeglicher Druckausgabe sollen immer zwei Seiten verkleinert auf einem Blatt gedruckt werden, dann könnten die folgenden Skripten `/etc/lpfilter/test/ascii2ps` und `/etc/lpfilter/test/pre` erstellt werden:

Beispiel 6.3: `/etc/lpfilter/test/ascii2ps`: ASCII nach PostScript Umwandlung

```
#!/bin/bash
cat -n - | a2ps -l --stdin=' ' -o -
```

Beispiel 6.4: /etc/lpdfilter/test/pre: PostScript Umformatierung

```
#!/bin/bash
pstops -q '2:0L@0.6(20cm,2cm)+1L@0.6(20cm,15cm)'
```

Diese Skripte müssen für jeden Benutzer ausführbar sein, was mit `chmod` erreicht werden kann:

```
chmod -v a+rx /etc/lpdfilter/test/ascii2ps
chmod -v a+rx /etc/lpdfilter/test/pre
```

Der `pstops`-Aufruf funktioniert nur für PostScript-Dateien, die so erstellt wurden, dass eine Umformatierung möglich ist (was normalerweise der Fall sein sollte).

Selbsterstellte PostScript-Preloads verwenden

PostScript-Preloads sind kleine Dateien, die spezielle PostScript-Befehle enthalten und vor die eigentlichen Druckdaten vorgeschaltet werden, um einen PostScript-Drucker oder auch Ghostscript mit diesen speziellen Befehlen passend zu initialisieren. Üblicherweise werden Preloads verwendet, um bei einem PostScript-Drucker Duplex-Druck oder spezielle Papierschächte zu aktivieren oder um Randeinstellungen und Gammakorrektur passend zu setzen.

Voraussetzung ist, dass der PostScript-Drucker bzw. Ghostscript die unten angegebenen speziellen Befehle auch entsprechend verarbeiten kann (Ghostscript reagiert nicht auf Befehle für Duplex-Druck oder Papierschächte).

Angenommen, die betreffende Warteschlange heißt `test`.

Duplex-Druck Um Duplex-Druck ein- und auszuschalten, können Sie folgende Dateien `/etc/lpdfilter/test/duplexon.ps` und `/etc/lpdfilter/test/duplexoff.ps` erstellen:

Beispiel 6.5: /etc/lpdfilter/test/duplexon.ps: Duplex-Druck einschalten

```
%!PS
statusdict /setduplexmode known
{statusdict begin true setduplexmode end} if {} pop
```

Beispiel 6.6: */etc/lpfilter/test/duplexoff.ps: Duplex-Druck ausschalten*

```
%!PS
statusdict /setduplexmode known
{statusdict begin false setduplexmode end} if {} pop
```

Um die Rückseite bei Duplex-Druck um 180 Grad zu drehen, kann der folgedes PostScript-Code verwendet werden:

```
%!PS
statusdict /setduplexmode known
{statusdict begin true setduplexmode end} if {} pop
statusdict /setumble known
{statusdict begin true settumble end} if {} pop
```

Papierschachtwahl Um den Standardpapierschacht mit der Nummer 0 oder den Papierschacht zum Beispiel mit der Nummer 2 zu aktivieren, erstellen Sie die Dateien `/etc/lpfilter/test/tray0.ps` und `/etc/lpfilter/test/tray2.ps`:

Beispiel 6.7: */etc/lpfilter/test/tray0.ps: Papierschacht 0 aktivieren*

```
%!PS
statusdict /setpapertray known
{statusdict begin 0 setpapertray end} if {} pop
```

Beispiel 6.8: */etc/lpfilter/test/tray2.ps: Papierschacht 2 aktivieren*

```
%!PS
statusdict /setpapertray known
{statusdict begin 2 setpapertray end} if {} pop
```

Randeinstellungen Um Randeinstellungen zu verändern, erstellen Sie folgende Datei `/etc/lpfilter/test/margin.ps`:

Beispiel 6.9: */etc/lpfilter/test/margin.ps: Randeinstellungen*

```
%!PS
<<
/.HWMargins [left bottom right top]
/PageSize [width height]
/Margins [left-offset top-offset]
>>
setpagedevice
```

Die Randeinstellungen `left`, `bottom`, `right` und `top` und die Papiergröße `width` und `height` sind in sog. Punkten anzugeben wobei ein Punkt die Größe 1/72 Zoll (also ca. 0.35 mm) hat. Die Rand-Offsets `left-offset` und `top-offset` dagegen sind in Rasterpunkten anzugeben und somit von der jeweiligen Auflösung abhängig.

Soll nur die Position des Ausdrucks auf dem Papier verschoben werden, genügt folgende Datei `/etc/lpfilter/test/offset.ps`

Beispiel 6.10: */etc/lpfilter/test/offset.ps: Position des Ausdrucks*

```
%!PS
<< /Margins [left-offset top-offset] >> setpagedevice
```

Gammakorrektur Um die Helligkeitsverteilung der Farben zu verändern, erstellen Sie die Dateien `/etc/lpfilter/test/cmyk.ps` und `/etc/lpfilter/test/rgb.ps`:

Beispiel 6.11: */etc/lpfilter/test/cmyk.ps: CMYK Gammakorrektur*

```
%!PS
{cyan exp} {magenta exp} {yellow exp} {black exp}
setcolortransfer
```

Beispiel 6.12: */etc/lpfilter/test/rgb.ps: RGB Gammakorrektur*

```
%!PS
{red exp} {green exp} {blue exp} currenttransfer
setcolortransfer
```

Das Farbmodell (CMYK oder RGB) muss zu Ihrem Drucker passen. Die Werte, die für `cyan`, `magenta`, `yellow`, `black`, `red`, `green` und `blue` einzusetzen sind, müssen Sie durch Tests ermitteln. Normalerweise sind Werte zwischen 0.001 und 9.999 sinnvoll.

Die Wirkung der obigen Dateien testen Sie unter der grafischen Oberfläche am Bildschirm mit folgenden Befehlen: Ohne Gammakorrektur:

```
gs -r60 \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
```

Mit Gammakorrektur eines dieser Beispiele:

```
gs -r60 /etc/lpdfilter/test/cmyk.ps \
    /usr/share/doc/packages/ghostscript/examples/colorcir.ps
gs -r60 /etc/lpdfilter/test/rgb.ps \
    /usr/share/doc/packages/ghostscript/examples/colorcir.ps
```

Mit (Strg) + (C) wieder beenden.

Reset des Druckers Um den Drucker in den Grundzustand zurückzusetzen, erstellen Sie die Datei `/etc/lpdfilter/test/reset.ps`:

Beispiel 6.13: */etc/lpdfilter/test/reset.ps: Reset des Druckers*

```
%!PS
serverdict begin 0 exitserver
```

Zur Aktivierung einer PostScript-Preload-Datei kann folgendes Skript `/etc/lpdfilter/test/pre` erstellt werden:

Beispiel 6.14: */etc/lpdfilter/test/pre: PostScript-Preload laden*

```
#!/bin/bash
cat /etc/lpdfilter/test/preload.ps -
```

Dabei ist für `preload.ps` der passende Preload-Dateiname einzusetzen und außerdem muss das Skript für jeden Benutzer ausführbar und die Preload-Datei für jeden Benutzer lesbar sein, was mit `chmod` zu erreichen ist:

```
chmod -v a+rx /etc/lpdfilter/test/pre
chmod -v a+r /etc/lpdfilter/test/preload.ps
```

Derselbe Mechanismus kann auch verwendet werden, um eine PostScript-Datei nicht nur vor, sondern auch nach den eigentlichen PostScript-Druckdaten an den Drucker zu schicken. Beispielsweise um den Drucker am Ende eines Druckjobs wieder in den Grundzustand zurückzusetzen, kann das Skript `/etc/lpdfilter/test/pre` folgendermaßen ergänzt werden:

Beispiel 6.15: */etc/lpdfilter/test/pre: PostScript-Preload und PostScript-Reset*

```
#!/bin/bash
cat /etc/lpdfilter/test/preload.ps - /etc/lpdfilter/test/reset.ps
```

Beispiel zur Konfiguration eines GDI-Druckers

Es soll eine Warteschlange `gdi` für einen GDI-Drucker eingerichtet werden. Derartige Drucker können zumeist nicht unter Linux verwendet werden; siehe oben den Abschnitt 5.2.3 auf Seite 107. Allerdings gibt es für manche GDI-Drucker spezielle Treiberprogramme, die normalerweise als Zusatz nach Ghostscript verwendet werden, indem das Treiberprogramm spezielle Ausgaben von Ghostscript in das druckerspezifische Format konvertiert. Solche Treiberprogramme ermöglichen aber oft nur eingeschränkten Ausdruck – zum Beispiel nur Schwarzweißdruck. Ghostscript und Treiberprogramm arbeiten dann wie folgt zusammen (vgl. unten den Abschnitt 6.6 auf Seite 190.)

1. Die PostScript-Daten werden von Ghostscript in ein Raster einzelner Bildpunkte aufgelöst und die Rasterdaten durch einen zum nachgeschalteten Treiberprogramm passenden Ghostscript-Treiber in geeignetem Format und in geeigneter Auflösung ausgegeben.
2. Die Rasterdaten werden durch das Treiberprogramm in das druckerspezifische Format konvertiert.

Es wird im Folgenden vorausgesetzt, dass ein zur vorliegenden Version von SUSE LINUX passendes Treiberprogramm für den Drucker vorhanden ist bzw. aus dem Internet heruntergeladen werden kann, dass dieses Treiberprogramm in obiger Weise arbeitet und dass Sie ggf. im Umgang mit Unix-Quellen (zum Beispiel mit `.zip`- oder `.tar.gz`-Archiven oder `.rpm`-Paketen) vertraut sind.

Nach dem Entpacken eines solchen Archivs gibt es normalerweise aktuelle Installationshinweise in Dateien namens `README` oder `INSTALL` oder in einem Unterverzeichnis namens `doc`. Bei `.tar.gz`-Archiven ist das eigentliche Treiberprogramm in der Regel zu compilieren und zu installieren. Im Folgenden wird beispielsweise angenommen:

- Das Treiberprogramm ist `/usr/local/bin/printerdriver`.
- Als Ghostscript-Treiber wird `pbmraw` mit einer Auflösung von 600 dpi benötigt.
- Der Drucker ist an der ersten parallelen Schnittstelle `/dev/lp0` angeschlossen.

Welcher Ghostscript-Treiber und welche Auflösung tatsächlich zu nehmen ist, muss in der Dokumentation zum Treiberprogramm angegeben sein. Zuerst wird die `gdi`-Warteschlange mit `lpsetup` (als Benutzer `root`) angelegt:


```
lprsetup -add gdi -lprng -device /dev/lp0 \  
-driver pbmraw -dpi 600 -size a4dj -auto -sf
```

Dann ist folgendes Skript `/etc/lpfilter/gdi/post` zu erstellen:

```
#!/bin/bash  
/usr/local/bin/printerdriver <treiberspezifische_parameter>
```

Gegebenenfalls sind `<treiberspezifische_parameter>` passend einzutragen. Welche treiberspezifischen Parameter tatsächlich zu nehmen sind, muss in der Dokumentation zum Treiberprogramm angegeben sein. Das Skript muss für jeden Benutzer ausführbar gemacht und dann der Druckerspooler neu gestartet werden:

```
chmod -v a+rx /etc/lpfilter/gdi/post  
rclpd stop  
rclpd start
```

Nun kann jeder Benutzer so drucken:

```
lpr -Pgdi <datei>
```

6.5.3 Fehlersuche beim lpfilter

Der passende Debug-Level wird aktiviert, indem das Kommentarzeichen `#` vor der entsprechenden Zeile im Haupt-Skript `/usr/lib/lpfilter/bin/if` des Druckerfilters entfernt wird.

Beispiel 6.16: `/usr/lib/lpfilter/bin/if`: Debug-Level

```
# DEBUG="off"  
# DEBUG="low"  
# DEBUG="medium"  
# DEBUG="high"
```

Bei `DEBUG="low"` werden nur die `stderr`-Ausgaben von `/usr/lib/lpfilter/bin/if` in einer Datei `/tmp/lpfilter.if-$$.XXXXXX` (hierbei wird `$$` durch die Prozessnummer und `XXXXXX` durch eine zufällige aber eindeutige Zeichenkombination ersetzt) gespeichert.

Bei `DEBUG="medium"` werden zusätzlich die `stderr`-Ausgaben der Skripte unter `/usr/lib/lpddfilter/filter/`, die von `/usr/lib/lpddfilter/bin/if` aufgerufen werden, in Dateien der Form `/tmp/lpddfilter.name-$$.XXXXXX` (hierbei wird `name` durch den Namen des aufgerufenen Skripts und `$$.XXXXXX` analog zu oben ersetzt) gespeichert.

Bei `DEBUG="high"` wird zusätzlich die Ausgabe nicht an den Drucker geschickt, sondern in einer Datei der Form `/tmp/lpddfilter.out-$$.XXXXXX` (hierbei wird `$$.XXXXXX` analog zu oben ersetzt) gespeichert.

Um nicht die Übersicht zu verlieren, sollten diese Dateien vor jedem neuen Test mit `rm -v /tmp/lpddfilter*` gelöscht werden.

6.6 Etwas über Ghostscript

Ghostscript akzeptiert PostScript- und PDF-Daten als Eingabe und beinhaltet zur Konvertierung in andere Formate eine Vielzahl von Ghostscript-Treibern, die bei Ghostscript Devices heißen.

Ghostscript arbeitet bei der Konvertierung in zwei Schritten:

1. Die PostScript-Daten werden gerastert, das heisst die in der PostScript-Sprache beschriebene Grafik wird in ein feines Raster einzelner Bildpunkte zerlegt. Dieser Schritt ist unabhängig vom jeweiligen Ghostscript-Treiber. Je feiner das Raster (also je höher die Auflösung), desto höher ist einerseits die Ausgabequalität, aber bei doppelter Auflösung horizontal und vertikal vervierfacht sich die Anzahl der Rasterpunkte und damit vervierfacht sich der Rechenaufwand und Speicherverbrauch.
2. Die in Rasterpunkte aufgelöste Grafik wird nun durch den jeweiligen Ghostscript-Treiber in das letztlich gewünschte Format (zum Beispiel in die gewünschte Druckersprache) umgewandelt.

Ghostscript stellt nicht nur Druckertreiber zur Verfügung. Ghostscript kann auch PostScript-Dateien zur Bildschirmausgabe verarbeiten oder nach PDF umwandeln. Zur komfortableren Bildschirmausgabe von PostScript-Dateien sollte das Programm `gv` (`gv`) verwendet werden, weil es eine grafische Anwenderschnittstelle zu Ghostscript bietet.

Ghostscript ist ein sehr umfangreiches Programm mit zahlreichen Kommandozeilenoptionen. Die wichtigste Dokumentation ist neben der Manu-
alpage von `gs` und der Liste der Ghostscript-Treiber zu finden unter:

```
file:/usr/share/doc/packages/ghostscript/catalog.
devices
```

sowie in den Dateien:

```
file:/usr/share/doc/packages/ghostscript/doc/index.
html file:/usr/share/doc/packages/ghostscript/doc/
Use.htm file:/usr/share/doc/packages/ghostscript/doc/
Devices.htm file:/usr/share/doc/packages/ghostscript/
doc/hpdj/gs-hpdj.txt file:/usr/share/doc/packages/
ghostscript/doc/hpijs/hpijs_readme.html file:/usr/share/
doc/packages/ghostscript/doc/stp/README
```

Ein Direktaufruf von Ghostscript startet nach Abarbeitung der Kommandozeile einen Dialog mit eigener Eingabeaufforderung `GS>`, der mit dem Befehl `quit` beendet wird.

Der Hilfe-Befehl `gs -h` listet die nötigsten Optionen auf und gibt die aktuelle Liste der unterstützten Devices aus. Dabei erscheint nur die allgemeine Treiberbezeichnung wie `uniprint` oder `stp`, wenn ein einziger Treiber eine Vielzahl von Modellen unterstützt. Die Parameterdateien für `uniprint` und die Modelle von `stp` sind in `file:/usr/share/doc/packages/ghostscript/catalog.devices` explizit aufgezählt.

6.6.1 Beispiele für die Arbeit mit Ghostscript

In `file:/usr/share/doc/packages/ghostscript/examples` finden Sie PostScript-Beispieldateien. Die Farbellipse `file:/usr/share/doc/packages/ghostscript/examples/colorcir.ps` eignet sich gut für einen Druckertest.

X11-Ausgabe

Unter X, der graphischen Oberfläche, können Sie eine PostScript-Datei mit dem Befehl `gs` am Bildschirm anzeigen lassen:

```
gs -r60 \
/usr/share/doc/packages/ghostscript/examples/colorcir.ps
```

Mit der Option `-r` wird die Auflösung angegeben, die aber zum jeweiligen Ausgabegerät (Drucker oder Bildschirm) passen muss (probieren Sie zum Beispiel `-r30`). Zum Beenden drücken Sie in dem Terminalfenster, von dem aus Sie `gs` aufgerufen haben, **(Strg) + (C)**.

Umwandlung in PCL5e

Die Umwandlung einer PostScript-Datei in das druckerspezifische Format für einen PCL5e- oder PCL6-Drucker geschieht zum Beispiel mit dem Befehl

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.prn \  
-sDEVICE=ljet4 -r300x300 \  
/usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
quit.ps
```

wobei der Befehl in einer einzigen Zeile einzugeben und der Rückstrich (\) zu unterdrücken ist. Weiterhin wird vorausgesetzt, dass die Datei /tmp/out.prn noch nicht existiert.

Umwandlung in PCL3

Die Umwandlung einer PostScript-Datei in das druckerspezifische Format für einen PCL3-Drucker geschieht zum Beispiel mit:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.prn \  
-sDEVICE=deskjet -r300x300 \  
/usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
quit.ps
```

Je nach Modell können Sie anstelle von *deskjet* als Device auch *cdjmomo*, *cdj500* oder *cdj550* verwenden oder auf den alternativen Treiber *hpdj* ausweichen:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.prn \  
-sDEVICE=hpdj -r300x300 \  
-sModel=500 -sColorMode=mono -dCompressionMethod=0 \  
/usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
quit.ps
```

Jeder Befehl kann auch ohne \, jedoch dann in *einer einzigen Zeile* eingegeben werden.

Umwandlung in ESC/P, ESC/P2 oder ESC/P-Raster

Die Umwandlung einer PostScript-Datei in das druckerspezifische Format für einen ESC/P2- oder ESC/P- oder ESC/P-Raster-Drucker geschieht zum Beispiel mit einem der folgenden Befehle:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.prn \  
  @stcany.upp \  
  /usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
  quit.ps
```

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.prn \  
  -sDEVICE=stcolor -r360x360 \  
  -dBitsPerPixel=1 -sDithering=gsmono -dnoWeave \  
  -sOutputCode=plain \  
  /usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
  quit.ps
```

Hier sieht man den Unterschied im Aufruf bei Verwendung einer Parameterdatei `stcany.upp` für den `uniprint`-Treiber und bei einem der anderen Ghostscript-Treiber. Da alle treiberspezifischen Parameter in der `uniprint`-Parameterdatei stehen, sind keine weiteren treiberspezifischen Parameter anzugeben, ganz im Gegensatz zu den anderen Ghostscript-Treibern.

Direkte Druckerausgabe

Nach jedem der obigen Befehle stehen die druckerspezifischen Daten in `/tmp/out.prn`, die nun mit folgendem Befehl von `root` direkt an den Drucker (also ohne Druckerspooler oder Druckerfilter) geschickt werden, sofern der Drucker an der ersten parallelen Schnittstelle `/dev/lp0` angeschlossen ist: `cat /tmp/out.prn >/dev/lp0`

PostScript- und PDF-Bearbeitung

Ghostscript kann PostScript- und PDF-Dateien erzeugen, beide Formate ineinander umwandeln und PostScript- und PDF-Dateien auch in gemischter Reihenfolge aneinanderhängen.

Umwandlung von PostScript nach PDF:

```
gs -q -dNOPAUSE -dSAFER \  
  -sOutputFile=/tmp/colorcir.pdf -sDEVICE=pdfwrite \  
  /usr/share/doc/packages/ghostscript/examples/colorcir.ps \  
  quit.ps
```

Umwandlung der eben erzeugten PDF-Datei `/tmp/colorcir.pdf` nach PostScript:

```
gs -q -dNOPAUSE -dSAFER \
-sOutputFile=/tmp/colorcir.ps -sDEVICE=pswrite \
/tmp/colorcir.pdf quit.ps
```

Nach der Rückumwandlung von PDF nach PostScript stimmt die Datei /tmp/colorcir.ps nicht mit dem Original /usr/share/doc/packages/ghostscript/examples/colorcir.ps überein, aber im Ausdruck sollte kein Unterschied erkennbar sein.

Aneinanderhängen von PostScript- und PDF-Dateien zu einer PostScript-Datei:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.ps \
-sDEVICE=pswrite \
/usr/share/doc/packages/ghostscript/examples/escher.ps \
/tmp/colorcir.pdf quit.ps
```

Aneinanderhängen von PostScript- und PDF-Dateien zu einer PDF-Datei:

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.pdf \
-sDEVICE=pdfwrite /tmp/out.ps \
/usr/share/doc/packages/ghostscript/examples/golfer.ps \
/tmp/colorcir.pdf quit.ps
```

Das Aneinanderhängen von PostScript- und PDF-Dateien funktioniert abhängig von der verwendeten Dateien leider nicht in jedem Fall.

6.7 Etwas über a2ps

Soll eine ASCII-Textdatei mit Ghostscript gedruckt werden, muss diese zuerst nach PostScript umgewandelt werden, da Ghostscript als Eingabe PostScript erwartet. Dazu wird das Programm a2ps (a2ps) verwendet. Da das a2ps nicht standardmäßig installiert wird, muss es normalerweise nachinstalliert werden. a2ps ist ein mächtiges Werkzeug, um aus einfachem Text eine qualitativ hochwertige PostScript-Ausgabe zu erzeugen. a2ps ist ein sehr umfangreiches Programm mit zahlreichen Kommandozeilenoptionen. Die wichtigste Dokumentation ist in der Manualpage a2ps zu finden – die vollständige Dokumentation findet sich in der Info-Page von a2ps.

6.7.1 Direkte Druckerausgabe einer Textdatei mit a2ps

Um eine Textdatei mit a2ps nach PostScript umzuwandeln, so dass zwei Seiten verkleinert auf einem Blatt dargestellt werden, kann folgender Befehl verwendet werden:

```
a2ps -2 --medium=A4dj --output=/tmp/out.ps textdatei
```

Die Ausgabe von a2ps kann dann zum Beispiel mit

```
gs -r60 /tmp/out.ps
```

unter der graphischen Oberfläche zur Kontrolle angezeigt werden, wobei ggf. in dem Terminalfenster, von dem aus Sie gs aufgerufen haben, die Eingabetaste zu drücken ist, um das jeweils nächste Blatt angezeigt zu bekommen ((Strg) + (C) zum Beenden).

Die Ausgabe von a2ps kann mit

```
gs -q -dNOPAUSE -dSAFER -sOutputFile=/tmp/out.prn \  
<driver-parameter> /tmp/out.ps quit.ps
```

in das druckerspezifische Format umgewandelt werden, wobei *<driver-parameter>* passend zum Drucker gemäß dem vorigen Abschnitt einzugeben ist.

Die Ausgabe von Ghostscript kann dann als root mit

```
cat /tmp/out.prn >/dev/lp0
```

direkt (also ohne Druckerspooler oder Druckerfilter) an den Drucker geschickt wird, sofern der Drucker an der ersten parallelen Schnittstelle /dev/lp0 angeschlossen ist.

6.8 PostScript-Umformatierung mit den psutils

Zur Umformatierung ist aus dem Anwendungsprogramm zunächst in eine Datei /tmp/in.ps zu drucken. Gegebenenfalls kann mit file

`/tmp/in.ps` überprüft werden, dass auch wirklich eine PostScript-Datei erzeugt wurde.

Programme, zur Umformatierung von PostScript-Daten, befinden sich im `psutils`. Insbesondere das Programm `pstops` ermöglicht weitreichende Umformatierungen. Vergleichen Sie dazu die Manual-Page von `pstops`. Da das `psutils` nicht standardmäßig installiert wird, muss es normalerweise nachinstalliert werden.

Die folgenden Aufrufe funktionieren nur für PostScript-Dateien, die so gutartig erstellt wurden, dass eine Umformatierung möglich ist. Das ist normalerweise der Fall, kann aber auch je nach Anwendungsprogramm, was die PostScript-Datei erstellt hat, unmöglich sein.

6.8.1 `psnup`

Mit `psnup -2 /tmp/in.ps /tmp/out.ps` wird `/tmp/in.ps` nach `/tmp/out.ps` umgewandelt, wobei je zwei Seiten verkleinert nebeneinander auf einem Blatt dargestellt werden. Da sich die Komplexität des Ausdrucks pro Blatt erhöht, wenn mehreren Seiten verkleinert auf ein Blatt gedruckt werden, können dadurch manche PostScript-Drucker, die nur über geringe Speicherkapazität verfügen, scheitern, wenn die zu komplex gewordenen Seiten zu Papier gebracht werden sollen.

6.8.2 `pstops`

Eine individuelle Größe und Positionierung ist mit `pstops` wie folgt möglich:

```
pstops '1:0@0.8(2cm,3cm)' /tmp/in.ps /tmp/out.ps
```

Hier wird mit dem Faktor 0.8 skaliert, was eine A4-Seite von ca. 21x30 cm auf ca. 17x24 cm verkleinert. Dadurch entstehen rechts ca. 4 cm und oben ca. 6 cm zusätzlicher freier Rand. Dann wird noch alles um 2 cm nach rechts und 3 cm nach oben verschoben, um die freien Ränder überall etwa gleich groß zu bekommen.

Dieser `pstops`-Aufruf verkleinert recht stark und verwendet großzügige Ränder, so dass es auch für Anwendungsprogramme funktioniert, die recht optimistische Vorstellungen haben, was alles auf eine Seite passen soll - d.h. wo die Druckausgabe des Anwendungsprogramms in `/tmp/in.ps` eigentlich zu groß war.

Ein weiteres Beispiel:


```
pstops '1:0@0.8(2cm,3cm)' /tmp/in.ps /tmp/out1.ps
psnup -2 /tmp/out1.ps /tmp/out.ps
```

Damit bekommt man je zwei Seiten stark verkleinert nebeneinander auf einem Blatt dargestellt – allerdings mit viel Raum zwischen den beiden verkleinerten Seiten. Besser wird es mit der individuellen Positionierung jeder einzelnen Seite:

```
pstops '2:0L@0.6(20cm,2cm)+1L@0.6(20cm,15cm)' \
/tmp/in.ps /tmp/out.ps
```

Der Befehl ist ohne \ in einer einzigen Zeile einzugeben.

Zur Wirkungsweise von pstops

```
'2:0L@0.6(20cm,2cm)+1L@0.6(20cm,15cm)':
```

2:0 ... +1 bedeutet, dass je 2 Seiten übereinandergelegt werden wobei die Seiten modulo 2 also abwechselnd als Seite 0 (modulo 2) und Seite 1 (modulo 2) gezählt werden.

0L@0.6(20cm,2cm) bedeutet, dass die jeweilige Seite 0 (modulo 2) nach links um 90 Grad gedreht wird, mit dem Faktor 0.6 skaliert wird und dann um 20cm nach rechts und 2cm nach oben verschoben wird.

1L@0.6(20cm,15cm) Analog wird hiermit die jeweilige Seite 1 (modulo 2) nach links um 90 Grad gedreht, mit dem Faktor 0.6 skaliert und dann um 20cm nach rechts und 15cm nach oben verschoben.

Bei PostScript ist der Nullpunkt des Koordinatensystems die linke untere Ecke auf dem Blatt Papier in normaler Lage, der hier mit + gekennzeichnet ist (s. Abb. 6.1 auf der nächsten Seite):

1. Eine Seite 0 (modulo 2) mit drei Zeilen Text.
2. Nach der Linksdrehung um 90 Grad.
3. Nach der Skalierung mit dem Faktor 0.6.
4. Nach der Verschiebung um 20cm nach rechts und 2cm nach oben.
5. Darüber gelegt eine Seite 1 (modulo 2) mit zwei Zeilen Text.
6. Nach der Linksdrehung von Seite 1 (modulo 2) um 90 Grad.
7. Nach der Skalierung von Seite 1 (modulo 2) mit dem Faktor 0.6.
8. Nach der Verschiebung von Seite 1 (modulo 2) um 20 cm nach rechts und 15 cm nach oben.

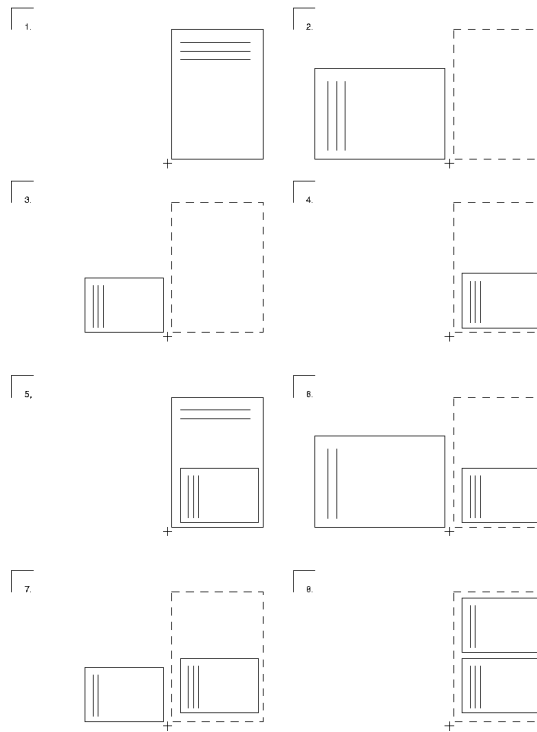


Abbildung 6.1: Veranschaulichung der Schritte mit pstops

6.8.3 psselect

Mit `psselect` können einzelne Seiten selektiert werden. Der Befehl `psselect -p2-5 /tmp/in.ps /tmp/out.ps` selektiert aus `/tmp/in.ps` die Seiten 2,3,4 und 5 und gibt sie nach `/tmp/out.ps` aus. Mit `psselect -p-3 /tmp/in.ps /tmp/out.ps` werden alle Seiten bis zur Seite 3 selektiert. Der Befehl `psselect -r -p4- /tmp/in.ps /tmp/out.ps` selektiert von Seite 4 bis zur letzten Seite und gibt sie dann in umgekehrter Reihenfolge ausgegeben.

6.8.4 Kontrolle am Bildschirm mit Ghostscript

Die PostScript-Datei `/tmp/out.ps` kann unter der grafischen Oberfläche von Ghostscript mit `gs -r60 /tmp/out.ps` Seite für Seite ange-

zeigt werden. Durch Drücken der Eingabetaste im Terminalfenster, in dem Ghostscript aufgerufen wurde, wird die PostScript-Datei Seite für Seite angezeigt und zum Beenden drücken Sie die Tasten (Strg) + (C).

Ein grafisches Bedienfrontend zu Ghostscript ist das Programm gv aus dem gv. Es wird unter der grafischen Oberfläche mit `gv /tmp/out.ps` aufgerufen und ermöglicht eine passende Darstellung bei Querformat, Vergrößerung oder Verkleinerung der Darstellung (aber nicht in der eigentlichen PostScript-Datei) und Selektion einzelner Seiten, insbesondere auch zum Druck direkt aus gv.

6.9 Zur Kodierung von ASCII-Text

Bei einfachem Text ist jedes Zeichen als Zahl kodiert abgespeichert. Welche Zeichendarstellung einem Zeichencode entspricht ist in Code-Tabellen festgelegt. Je nachdem, welche Code-Tabelle ein Anwendungsprogramm bzw. der Druckerfilter verwendet, kann die Darstellung desselben Codes auf dem Bildschirm und auf dem Drucker verschieden sein.

Bei Standardzeichensätzen sind nur Codes von 0 bis 255 möglich. Die Zeichen mit den Codes 0 bis 127 sind die ASCII-Zeichen (insbesondere die normalen Buchstaben, Ziffern und Sonderzeichen, aber keine länderspezifische Sonderzeichen), die immer gleich festgelegt sind.

Die Codes 128 bis 255 werden für länderspezifische Sonderzeichen (zum Beispiel Umlaute) benutzt. Da es aber deutlich mehr als 128 verschiedene länderspezifische Zeichen gibt, sind die Codes 128 bis 255 nicht mehr überall gleich belegt, sondern je nach geographischer Lage wird derselbe Code für verschiedene länderspezifische Zeichen verwendet.

ISO-8859-1 (bzw. Latin 1) ist die Kodierung für Westeuropäische Sprachen und ISO-8859-2 (bzw. Latin 2) ist die Kodierung für Zentral- und Osteuropäische Sprachen. So bedeutet zum Beispiel der Code 241 (octal) gemäß ISO-8859-1 ein umgedrehtes Ausrufungszeichen, aber gemäß ISO-8859-2 ein großes A mit Ogonek. ISO-8859-15 entspricht im wesentlichen ISO-8859-1, aber insbesondere hat ISO-8859-15 unter dem Code 244 (octal) das Eurozeichen. Da die deutschen Umlaute in ISO-8859-1 und ISO-8859-2 vorhanden und gleich codiert sind, kann bei deutschen Texten ohne Eurozeichen sowohl ISO-8859-1 als auch ISO-8859-2 verwendet werden.

6.9.1 Veranschaulichung

Alle Befehle sind in einer einzigen Zeile einzugeben ohne den Rückstrich (\) am *Zeilenende*.

ASCII-Text Beispieldatei erzeugen mit:

```
echo -en "\rCode 241(octal): \
\r\nCode 244(octal): \244\r\f" >example
```

Anzeige am Bildschirm

Öffnen Sie unter der graphischen Oberfläche drei Terminalfenster mit den folgenden drei Befehlen:

```
xterm -fn -*-***-14-***-iso8859-1 -title iso8859-1 &
xterm -fn -*-***-14-***-iso8859-15 -title iso8859-15 &
xterm -fn -*-***-14-***-iso8859-2 -title iso8859-2 &
```

In jedem der Terminalfenster lassen Sie die Beispieldatei anzeigen mit dem Befehl `cat example`.

In iso8859-1 wird angezeigt: Code 241 als umgedrehtes Ausrufungszeichen (Spanisch) Code 244 als Kreis mit Häkchen (allgemeines Währungssymbol)

In iso8859-15 wird angezeigt: Code 241 als umgedrehtes Ausrufungszeichen (Spanisch) Code 244 als Eurosymbol

In iso8859-2 wird angezeigt: Code 241 als großes A mit Krummhaken (A mit Ogonek) Code 244 als Kreis mit Häkchen (allgemeines Währungssymbol)

Wegen der festgelegten Kodierungen können nicht beliebige länderspezifische Sonderzeichen gleichzeitig verwendet werden. So kann zum Beispiel das Eurosymbol nicht zusammen mit einem A mit Ogonek in demselben Text dargestellt werden.

Weitere Informationen in der jeweils korrekten Darstellung: Zu iso8859-1: `man iso_8859-1`. Zu iso8859-2: `man iso_8859-2`. Zu iso8859-15: `man iso_8859-15`.

Ausdruck

Je nach Kodierung der jeweiligen Druckerwarteschlange erfolgt der Ausdruck von ASCII-Text (zum Beispiel der Ausdruck der Datei `example`) analog zu diesen Fällen. Der Ausdruck von Dokumenten, die mit Textverarbeitungssystemen erstellt wurden, ist davon normalerweise unbeeinflusst, denn Textverarbeitungssysteme liefern zur Druckausgabe PostScript und nicht ASCII-Text.

Wird die Datei `example` gedruckt, dann erhält man den Ausdruck in der Kodierung, die im Drucksystem für ASCII-Text verwendet wird. Mit `a2ps` kann man die Datei `example` nach PostScript umwandeln und dabei die Kodierung individuell festlegen:

```
a2ps -l -X ISO-8859-1 -o example-ISO-8859-1.ps example
a2ps -l -X ISO-8859-15 -o example-ISO-8859-15.ps example
a2ps -l -X ISO-8859-2 -o example-ISO-8859-2.ps example
```

Werden die PostScript-Dateien `example-ISO-8859-1.ps`, `example-ISO-8859-15.ps` und `example-ISO-8859-2.ps` gedruckt, dann erhält man den Ausdruck in der Kodierung, die jeweils mit `a2ps` festgelegt wurde.

Booten und Bootmanager

Im Folgenden werden verschiedene Methoden zum Booten des fertig installierten System vorgestellt. Um das Verständnis der einzelnen Methoden zu erleichtern, werden zunächst einige technische Details des Bootprozesses erläutert. Im Anschluss daran folgen Erläuterungen zum aktuell verwendeten Bootmanager GRUB.

7.1	Der Bootvorgang auf dem PC	204
7.2	Bootkonzepte	205
7.3	Map Files, GRUB und LILO	206
7.4	Booten mit GRUB	207
7.5	Linux-Bootloader entfernen	218
7.6	Für alle Fälle: Boot-CD erstellen	219

7.1 Der Bootvorgang auf dem PC

Nach dem Einschalten des Rechners werden vom BIOS (engl. *Basic Input Output System*) Bildschirm und Tastatur initialisiert sowie der Hauptspeicher getestet. Bis zu diesem Zeitpunkt verfügt der Rechner über keine Massenspeichermedien.

Anschließend werden Informationen über aktuelles Datum, Zeit und die wichtigsten Peripheriegeräte aus den CMOS-Werten (*CMOS Setup*) ausgelesen. Da nun die erste Festplatte einschließlich ihrer Geometrie bekannt sein sollte, kann das Laden des Betriebssystems von dort beginnen.

Dazu wird von der ersten Festplatte der physikalisch erste Datensektor von 512 Byte Größe in den Speicher geladen und die Kontrolle geht auf das Programm zu Beginn dieses Sektors über. Die Abfolge der auf diese Weise ausgeführten Anweisungen bestimmt den weiteren Ablauf des Bootvorgangs. Die ersten 512 Byte auf der ersten Festplatte werden deshalb auch als *Master Boot Record* bezeichnet.

Bis zu diesem Zeitpunkt (Laden des MBR) läuft der Bootvorgang völlig unabhängig vom installierten System auf jedem PC immer gleich ab und der Computer hat bis dahin für den Zugriff auf die Peripherie lediglich die im BIOS gespeicherten Routinen (Treiber) zur Verfügung.

7.1.1 Master Boot Record

Die Struktur des MBR ist durch eine betriebssystemübergreifende Konvention festgelegt. Die ersten 446 Byte sind für Programmcode reserviert. Die nächsten 64 Byte bieten Platz für eine Partitionstabelle mit bis zu vier Einträgen; siehe Abschnitt 1.7 auf Seite 25. Ohne die Partitionstabelle gibt es keine Dateisysteme, d.h. die Festplatte ist praktisch nicht zu verwenden. Die letzten zwei Byte müssen eine feste „magische Zahl“ (AA55) enthalten: ein MBR, der dort etwas anderes stehen hat, wird vom BIOS und von allen PC-Betriebssystemen als ungültig angesehen.

7.1.2 Bootsektoren

Bootsektoren sind die jeweils ersten Sektoren der Festplatten-Partitionen, außer bei der erweiterten Partition, die nur ein „Behälter“ für andere Partitionen ist. Diese Bootsektoren bieten 512 Byte Platz und sind dazu gedacht, Code aufzunehmen, der ein auf dieser Partition befindliches Betriebssystem starten kann. Dies gilt für Bootsektoren formatierter DOS-,

Windows- oder OS/2-Partitionen, die zusätzlich noch wichtige Grunddaten des Dateisystems enthalten. Im Gegensatz dazu sind Bootsektoren von Linux-Partitionen auch nach der Anlage eines Dateisystems erst einmal leer. Eine Linux-Partition ist daher *nicht von selbst startbar*, auch wenn sie einen Kernel und ein gültiges Root-Dateisystem enthält.

Ein Bootsektor mit gültigem Code für den Systemstart trägt in den letzten 2 Bytes dieselbe „magische“ Kennung wie der MBR (AA55).

7.1.3 Booten von DOS oder Windows

Im DOS-MBR der ersten Festplatte ist ein Partitionseintrag als *aktiv* (engl. *bootable*) gekennzeichnet, was bedeutet, dass dort nach dem zu ladenden System gesucht werden soll. Deshalb muss DOS zwingend auf der ersten Festplatte installiert sein. Der DOS-Programmcode im MBR ist die erste Stufe des Bootloaders (engl. *first stage bootloader*) und überprüft, ob auf der angegebenen Partition ein gültiger Bootsektor vorhanden ist.

Falls dies der Fall ist, kann der Code in diesem Bootsektor als „zweite Stufe“ des Bootloaders (engl. *secondary stage loader*) nachgestartet werden. Dieser lädt nun die Systemprogramme und schließlich erscheint der gewohnte DOS-Prompt bzw. es startet die Benutzeroberfläche von Windows.

Unter DOS lässt sich nur eine einzige primäre Partition als aktiv markieren. Folglich kann das DOS-System nicht auf logischen Laufwerken in einer erweiterten Partition untergebracht werden.

7.2 Bootkonzepte

Das einfachste „Bootkonzept“ betrifft einen Rechner mit einem einzigen Betriebssystem. Die Abläufe in der Startphase in diesem Fall haben wir soeben geschildert. Ein solcher Bootvorgang ist auch für einen Nur-Linux-Rechner denkbar. Dann kann theoretisch auf die Installation eines Bootloaders verzichtet werden, allerdings wäre es so nicht möglich, dem Kernel während des Startens eine Kommandozeile (mit Spezialwünschen zum Startvorgang, zusätzlichen Hardware-Informationen usw.) mitzugeben. Sobald mehr als ein Betriebssystem auf einem Rechner installiert ist, bieten sich verschiedene Bootkonzepte an:

Zusätzliche Systeme von Diskette booten

Ein Betriebssystem wird von Platte geladen, mit Hilfe von Boot-Disketten können alternativ weitere Betriebssysteme vom Disketten-Laufwerk aus gestartet werden.

- *Bedingung:* Ein bootfähiges Diskettenlaufwerk ist vorhanden.
- *Beispiel:* Sie installieren Linux zusätzlich zu Ihrem Windows-System und starten Linux stets von Bootdiskette.
- *Vorteil:* Sie ersparen sich die Bootloader-Installation.
- *Nachteile:* Sie müssen *sehr* darauf bedacht sein, einen Sicherheitsvorrat funktionierender Bootdisketten zu haben und der Start dauert länger.
- Dass Ihr Linux ohne Bootdiskette nicht starten kann, mag je nach beabsichtigtem Einsatz Ihres Rechners ein Vor- oder Nachteil sein.

Zusätzliche Systeme von einem USB-Speichermedium booten

Ebenso wie von einer Diskette können die zum Booten notwendigen Informationen von einem USB-Speichermedium eingelesen werden.

Installation eines Bootmanagers Ein Bootmanager erlaubt, mehrere Systeme gleichzeitig auf einem Rechner zu halten und sie abwechselnd zu nutzen. Der Benutzer wählt das zu ladende System bereits während des Bootvorgangs aus; ein Wechsel erfordert den Neustart des Rechners. Bedingung ist dabei, dass der gewählte Bootmanager mit allen Betriebssystemen „harmoniert“. Die Bootmanager von SUSE LINUX (LILO und sein Nachfolger GRUB) können alle gängigen Betriebssysteme starten. SUSE LINUX installiert daher den gewünschten Bootmanager standardmäßig in den MBR, so Sie diese Einstellung nicht während des Installationsdialogs ändern.

7.3 Map Files, GRUB und LILO

Das größte Problem beim Booten eines Betriebssystems besteht darin, dass der Kernel eine Datei auf einem Dateisystem auf einer Partition auf einer Festplatte ist. Für das BIOS allerdings sind Dateisysteme und Partitionen völlig unbekannte Konzepte.

Um dieses Problem zu umgehen, wurden so genannte „Maps“ und „Map Files“ eingeführt. In den Maps werden die physikalischen Blöcke auf der Festplatte notiert, die von den logischen Dateien belegt sind. Wenn so eine Map verarbeitet wird, lädt das BIOS die physikalischen Blöcke in der Reihenfolge, wie sie in der Map-Datei angegeben ist, und baut so die logische Datei im Speicher auf.

Im Gegensatz zu LILO, der sich vollständig auf Maps verlässt, versucht GRUB, sich sobald als möglich von den festen Maps zu lösen. Dies erreicht GRUB durch *File System Code*, der es ermöglicht, auf Dateien durch die Pfadangabe zuzugreifen und nicht mehr durch die Blocknummern.

Dieser Unterschied hat historische Gründe. In den frühen Tagen von Linux kämpften viele verschiedenen Dateisysteme um die Vorherrschaft. Werner Almesberger entwickelte einen Bootloader (LILO), der nicht wissen musste, auf welchem Filesystem der zu bootende Kernel angelegt war. Die Idee hinter GRUB geht sogar noch weiter zurück in die Tage des traditionellen Unix und BSD. Diese hatten sich gewöhnlich auf ein Dateisystem festgelegt und am Anfang desselben einen bestimmten Platz für den Bootloader reserviert. Dieser Bootloader kannte die Struktur des Dateisystems, in das er eingebunden war, und fand dort die Kernel mit ihren Namen im root-Verzeichnis.

Hinweis

Wann wird welcher Bootloader installiert?

Wenn Sie ein Update von einer früheren SUSE LINUX Version durchführen, die LILO benutzte, wird auch wieder LILO eingerichtet. Bei einer Neuinstallation wird dagegen GRUB verwendet, außer die Root-Partition wird auf folgenden Raid-Systemen installiert:

- CPU-abhängige Raid-Controller (wie z.B. viele Promise- oder Highpoint Controller)
- Software-Raid
- LVM

Informationen zur Installation und Konfiguration von LILO erhalten Sie unter dem Stichwort „LILO“ in der Support-Datenbank (<http://portal.suse.de/sdb/de/index.html>).

Hinweis

7.4 Booten mit GRUB

GRUB (*Grand Unified Bootloader*) besteht wie LILO aus zwei Stufen. Die erste Stufe (stage1) besteht aus 512 Byte und wird in den MBR oder den Bootsektor einer Plattenpartition oder Diskette geschrieben. Die zweite, größere Stufe (stage 2) wird im Anschluss daran geladen und enthält

den eigentlichen Programmcode. Einzige Aufgabe der ersten Stufe ist bei GRUB, die zweite Stufe des Bootloaders zu laden.

Ab diesem Punkt unterscheidet sich GRUB von LILO. `stage2` kann auf Dateisysteme zugreifen. Derzeit werden Ext2, Ext3, ReiserFS, JFS, XFS, Minix und das von Windows verwendete DOS FAT FS unterstützt. GRUB kann noch vor dem Booten auf Dateisysteme unterstützter BIOS-Disk-Devices (Diskette oder vom BIOS erkannte Festplatten) zugreifen, weshalb Änderungen an der GRUB-Konfigurationsdatei keine Neuinstallation des Bootmanagers mehr bedeuten. Beim Booten liest GRUB die Menüdatei samt der aktuellen Pfade und Partitionsangaben zu Kernel oder initialer Ramdisk (`initrd`) neu ein und findet diese Dateien selbständig.

GRUB hat den großen Vorteil, dass alle Bootparameter *vor* dem Booten bequem geändert werden können. Wurde zum Beispiel beim Editieren der Menüdatei ein Fehler gemacht, kann er auf diese Weise „ausgewetzt“ werden. Darüber hinaus können Boot-Kommandos interaktiv über eine Art von Eingabeaufforderung eingegeben werden. GRUB bietet die Möglichkeit, noch vor dem Booten die Lage von Kernel und `initrd` festzustellen. So booten Sie Betriebssysteme, die noch keinen Eintrag im Bootmenü haben.

7.4.1 Das GRUB-Bootmenü

Hinter dem grafischen Splash-Screen mit dem Bootmenü steht die GRUB-Konfigurationsdatei `/boot/grub/menu.lst`, die alle Informationen zu allen Partitionen oder Betriebssystemen enthält, die mit Hilfe des Menüs gebootet werden können.

GRUB liest bei jedem Systemstart die Menüdatei vom Dateisystem neu ein. Es besteht also kein Bedarf, GRUB nach jeder erfolgten Änderung an der Datei zu aktualisieren — verwenden Sie einfach YaST oder Ihren favorisierten Editor.

Die Menüdatei enthält Befehle. Die Syntax ist sehr einfach. Jede Zeile enthält einen Befehl, gefolgt von optionalen Parametern, die wie bei der Shell durch Leerzeichen getrennt werden. Einige Befehle erlauben aus historischen Gründen ein Gleichheitszeichen vor dem ersten Parameter. Kommentare werden durch einen Hash (`#`) eingeleitet.

Zur Erkennung der Menüeinträge in der Menü-Übersicht, müssen Sie für jeden Eintrag einen Namen oder einen `title` vergeben. Der nach dem Schlüsselwort `title` stehende Text wird inklusive Leerzeichen im Menü als selektierbare Option angezeigt. Alle Befehle bis zum nächsten `title` werden nach Auswahl dieses Menüeintrages ausgeführt.

Einfachster Fall ist das Verzweigen zu Bootloadern anderer Betriebssysteme. Der Befehl lautet `chainloader` und das Argument ist normalerweise der Boot-Block einer anderen Partition in GRUBs Block-Notation, zum Beispiel:

```
chainloader (hd0,3)+1
```

Die Devicenamen unter GRUB werden in Abschnitt 7.4.1 auf der nächsten Seite erklärt. Obiges Beispiel spezifiziert den ersten Block der vierten Partition auf der ersten Festplatte.

Mit dem Kommando `kernel` wird ein Kernel-Image spezifiziert. Das erste Argument ist der Pfad zum Kernel-Image auf einer Partition. Die restlichen Argumente werden dem Kernel auf der Kommandozeile übergeben.

Wenn der Kernel nicht die erforderlichen Treiber für den Zugriff auf die root-Partition einkompiliert hat, dann muss `initrd` angegeben werden. Hierbei handelt es sich um einen separaten GRUB-Befehl, der den Pfad zur `initrd`-Datei als einziges Argument hat. Da die Ladeadresse der `initrd` in das bereits geladene Kernel-Image geschrieben wird, muss der Befehl `initrd` auf den `kernel`-Befehl folgen.

Der Befehl `root` vereinfacht die Spezifikation der Kernel- und `initrd`-Dateien. `root` hat als einziges Argument entweder ein GRUB-Device oder eine Partition auf einem solchen. Allen Kernel-, `initrd`- oder anderen Dateipfaden, bei denen nicht explizit ein Device angegeben ist, wird bis zum nächsten `root`-Befehl das Device vorangestellt. In einer `menu.lst`-Datei, die während der Installation generiert wurde, kommt dieser Befehl nicht vor. Er dient der Vereinfachung beim händischen Editieren.

Am Ende jeden Menü-Eintrags steht implizit das `boot`-Kommando, so dass dieses nicht in die Menüdatei geschrieben werden muss. Sollten Sie jedoch in die Situation kommen, GRUB interaktiv zum Booten zu benutzen, müssen Sie am Ende das `boot`-Kommando eingeben. `boot` hat keine Argumente, es führt lediglich das geladene Kernel-Image oder den angegebenen Chain Loader aus.

Wenn Sie alle Menü-Einträge geschrieben haben, müssen Sie einen Eintrag als `default` festlegen. Andernfalls wird der erste (Eintrag 0) verwendet. Sie haben auch die Möglichkeit, einen Timeout in Sekunden anzugeben, nach dem dies geschehen soll. `timeout` und `default` werden üblicherweise vor die Menüeinträge geschrieben. Eine Beispieldatei samt Erläuterung finden Sie im Abschnitt 7.4.1 auf Seite 211.

Namenskonventionen für Festplatten und Partitionen

GRUB verwendet für die Bezeichnung von Festplatten und Partitionen andere Konventionen, als Sie es von den normalen Linux-Devices (z.B. `/dev/hda1`) her gewohnt sind. Die erste Festplatte wird immer `hd0` genannt, das Diskettenlaufwerk `fd0`.

Hinweis

Partitionenzählung in GRUB

Die Zählung der Partitionen in GRUB beginnt bei Null. (`hd0, 0`) entspricht der ersten Partition auf der ersten Festplatte; in einem gewöhnlichen Desktop-Rechner mit einer Platte als Primary Master angeschlossen lautet der Device-Name `/dev/hda1`.

Hinweis

Die vier möglichen primären Partitionen belegen die Partitionsnummern 0 bis 3. Ab 4 werden die logischen Partitionen hochgezählt:

(hd0,0)	erste primäre Partition auf der ersten Festplatte
(hd0,1)	zweite primäre Partition
(hd0,2)	dritte primäre Partition
(hd0,3)	vierte primäre (und meist die erweiterte) Partition
(hd0,4)	erste logische Partition
(hd0,5)	zweite logische Partition
...	

Hinweis

IDE, SCSI oder RAID

GRUB unterscheidet nicht zwischen IDE-, SCSI- oder RAID-Devices. Alle Festplatten, die vom BIOS oder weiteren Controllern erkannt werden, werden der im BIOS voreingestellten Bootreihenfolge entsprechend durchgezählt.

Hinweis

Das Problem, dass Linux-Device-Namen nicht eindeutig zu BIOS-Device-Namen zugeordnet werden können, besteht für sowohl LILO als auch GRUB. Beide benutzen vergleichbare Algorithmen, um diese Zuordnung zu generieren. Jedoch speichert GRUB diese Zuordnung in einer Datei (`device.map`) ab, die bearbeitet werden kann. Mehr Informationen zur Datei `device.map` finden Sie im Abschnitt 7.4.2 auf Seite 214.

Ein kompletter GRUB-Pfad besteht aus einem Device-Namen, der in Klammern geschrieben wird sowie dem Pfad der Datei in dem Dateisystem auf

der angegebenen Partition. Der Pfad wird durch einen Slash eingeleitet. Als Beispiel, auf einem System mit einer einzelnen IDE-Festplatte und Linux auf der ersten Partition, könnte der bootbare Kernel wie folgt aussehen:

```
(hd0,0)/boot/vmlinuz
```

Beispiel einer Menü-Datei

Zum besseren Verständnis des Aufbaus einer GRUB-Menüdatei stellen wir ein kurzes Beispiel vor. Diese Beispiel-Installation beinhaltet eine Linux-Bootpartition unter `/dev/hda5`, eine Root-Partition unter `/dev/hda7` und eine Windows-Installation unter `/dev/hda1`.

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8

title linux
    kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
    initrd (hd0,4)/initrd
title windows
    chainloader(hd0,0)+1
title floppy
    chainloader(fd0)+1
title failsafe
    kernel (hd0,4)/vmlinuz.shipped root=/dev/hda7 ide=nodma \
    apm=off acpi=off vga=normal nosmp maxcpus=0 3
    initrd (hd0,4)/initrd.shipped
```

Der erste Block behandelt die Konfiguration des Splash-Screens:

gfxmenu (hd0,4)/message Das Hintergrundbild liegt auf `/dev/hda5` und trägt den Namen `message`

color white/blue black/light-gray Das Farbschema: weiss (Vordergrund), blau (Hintergrund), schwarz (Auswahl) und hellgrau (Hintergrund der Auswahl). Das Farbschema wirkt sich nicht auf den Splashscreen aus, sondern erst, wenn Sie diesen mit `(Esc)` verlassen.

default 0 Der erste Menüeintrag mit `title linux` soll standardmäßig gebootet werden.

timeout 8 Nach acht Sekunden ohne Benutzerfeedback bootet GRUB automatisch durch.

Der zweite und größte Block listet die verschiedenen bootbaren Betriebssysteme auf.

- Der erste Eintrag (`title linux`) ist für das Booten von SUSE LINUX zuständig. Der Kernel (`vmlinuz`) liegt auf der ersten Festplatte in den ersten logischen Partition (hier der Bootpartition). Kernelparameter wie zum Beispiel die Angabe der Rootpartition, des VGA-Modus etc. werden hier angehängt. Die Angabe der Rootpartition erfolgt nach dem Linux-Schema (`/dev/hda7/`) da diese Information für den Kernel bestimmt ist und nichts mit GRUB zu tun hat. Die `initrd` liegt ebenfalls in der ersten logischen Partition der ersten Festplatte.
- Der zweite Eintrag ist für das Laden von Windows zuständig. Windows wird von der ersten Partition der ersten Festplatte aus gestartet (`hd0, 0`). Mittels `chainloader +1` wird das Auslesen und Ausführen des ersten Sektors der angegebenen Partition gesteuert.
- Der nächste Abschnitt dient dazu, das Booten von Diskette zu ermöglichen, ohne dass dazu das BIOS umgestellt werden müsste.
- Die Bootoption `failsafe` dient dazu, Linux mit einer bestimmten Auswahl an Kernelparametern zu starten, die selbst auf problematischen Systemen ein Hochfahren von Linux ermöglichen.

Die Menüdatei kann jederzeit geändert werden und wird von GRUB automatisch beim nächsten Booten übernommen. Sie können diese Datei mit dem Editor Ihrer Wahl oder mit YaST permanent editieren. Alternativ können Sie temporäre Änderungen interaktiv über die Edit-Funktion von GRUB vornehmen.

Hinweis

Festplattenreihenfolge tauschen

Manche Betriebssysteme (z.B. Windows) können nur von der ersten Festplatte starten. Wenn Sie ein solches Betriebssystem auf einer anderen als der ersten Festplatte installiert haben, können Sie beim entsprechenden Menüeintrag einen logischen Tausch veranlassen. Dies funktioniert allerdings nur, wenn das Betriebssystem beim Start über das BIOS auf die Festplatten zugreift.

```
...
title windows
    map (hd0) (hd1)
    map (hd1) (hd0)
    chainloader(hd1,0)+1
...
```

Hier soll Windows von der zweiten Festplatte gestartet werden. Dazu wird die logische Reihenfolge der Festplatten mit `map` getauscht. Beachten Sie dabei jedoch, dass sich durch den Tausch die Logik innerhalb der GRUB-Menüdatei *nicht* ändert. Nach wie vor müssen Sie bei `chainloader` die zweite Festplatte angeben.

Hinweis

Ändern von Menü-Einträgen während des Bootvorgangs

Aus dem grafischen Bootmenü von GRUB können Sie mittels der Cursorstasten auswählen, welches der verfügbaren Betriebssysteme gestartet werden soll. Wählen Sie ein Linux-System, können Sie am Bootprompt – wie auch von LILO gewohnt – eigene Bootparameter einfügen. GRUB geht noch über dieses Konzept hinaus. Drücken Sie (`Esc`) und verlassen Sie den Splash-Screen, können Sie nach der Eingabe von (`e`) (edit) einzelne Menü-Einträge gezielt direkt editieren. Änderungen, die Sie auf diese Weise vornehmen, gelten nur für diesen einen Bootvorgang und werden nicht dauerhaft übernommen.

Hinweis

Tastaturbelegung während des Bootens

Bitte beachten Sie, dass beim Booten nur die amerikanische Tastaturbelegung verfügbar ist. Achten Sie auf die vertauschten Sonderzeichen.

Hinweis

Nach Aktivieren des Editiermodus wählen Sie mittels der Cursortasten den Menü-Eintrag, dessen Konfiguration Sie verändern wollen. Um die Konfiguration editierbar zu machen, geben Sie ein weiteres Mal **(e)** ein. So korrigieren Sie falsche Partitions- oder Pfadangaben, bevor diese sich negativ auf den Bootprozess auswirken. Mit **(Enter)** verlassen Sie den Editiermodus, kehren ins Menü zurück und booten diesen Eintrag mit **(b)**. Im Hilfetext am unteren Rand werden weitere Handlungsmöglichkeiten angezeigt.

Möchten Sie geänderte Bootoptionen dauerhaft eintragen, öffnen Sie als Benutzer `root` die Datei `menu.lst` und hängen die zusätzlichen Kernelparameter durch ein Leerzeichen getrennt an die bestehende Zeile an:

```
title linux
kernel (hd0,0)/vmlinuz root=/dev/hda3 <zusätzlicher parameter>
initrd (hd0,0)/initrd
```

GRUB übernimmt den neuen Parameter beim nächsten Booten automatisch. Alternativ können Sie für diese Änderung auf das YaST-Bootloadermodul aufrufen. Auch hier wird der neue Parameter lediglich durch ein Leerzeichen getrennt an die bestehende Zeile angehängt.

7.4.2 Die Datei `device.map`

Die schon erwähnte Datei `device.map` enthält die Zuordnungen von GRUB-Devicenamen und Linux-Devicenamen. Sollten Sie ein Mischsystem aus IDE- und SCSI-Festplatten vorliegen haben, muss GRUB anhand eines bestimmten Verfahrens versuchen, die Bootreihenfolge zu ermitteln. Die BIOS-Informationen zur Bootreihenfolge sind GRUB nicht zugänglich. Das Ergebnis dieser Überprüfung speichert GRUB unter `/boot/grub/device.map` ab. Eine Beispieldatei `device.map` für ein Beispielsystem – angenommen wird eine im BIOS eingestellte Bootreihenfolge von IDE vor SCSI – sieht so aus:

```
(fd0)  /dev/fd0
(hd0)  /dev/hda
(hd1)  /dev/hdb
(hd2)  /dev/sda
(hd3)  /dev/sdb
```

Da die Reihenfolge von IDE, SCSI und anderen Festplatten abhängig von verschiedenen Faktoren ist und Linux die Zuordnung nicht erkennen kann, besteht die Möglichkeit, die Reihenfolge in der `device.map` manuell festzulegen. Sollten Sie Probleme beim Booten haben, kontrollieren Sie, ob die

Reihenfolge in dieser Datei der BIOS-Reihenfolge entspricht und ändern Sie sie notfalls mithilfe der GRUB-Shell (siehe Abschnitt 7.4.3) ab. Ist das Linux-System erst gebootet, können Sie die `device.map` mithilfe des YaST Bootloader-Moduls oder eines Editors Ihrer Wahl dauerhaft abändern.

Nach manuellen Änderungen der `device.map` Datei, rufen Sie den folgenden Befehl auf, um GRUB neu zu installieren:

```
grub --batch --device-map=/boot/grub/device.map < /etc/grub.conf
```

7.4.3 Die Datei `/etc/grub.conf`

Die dritte wichtige Konfigurationsdatei von GRUB neben `menu.lst` und `device.map` ist `/etc/grub.conf`. Hier werden die Parameter und Optionen aufgeführt, die der Befehl `grub` benötigt, um den Bootloader korrekt zu installieren:

```
root (hd0,4)
install /grub/stage1 d (hd0) /grub/stage2 0x8000 (hd0,4)/grub/menu.lst
quit
```

Die Bedeutung der einzelnen Einträge im Detail:

root (hd0,4) Mit diesem Befehl wird GRUB angewiesen, sich bei den folgenden Befehlen auf die erste logische Partition der ersten Festplatte zu beziehen, auf denen er seine Bootdateien findet.

install parameter Der Befehl `grub` soll mit dem `install-Parameter` gestartet werden. `stage1` als erste Stufe des Bootloaders soll in den MBR der ersten Festplatte installiert werden (`/grub/stage1 d (hd0)`). `stage2` soll in die Speicheradresse `0x8000` geladen werden (`/grub/stage2 0x8000`). Der letzte Eintrag `(hd0,4)/grub/menu.lst` weist `grub` an, wo die Menüdatei zu finden ist.

Die GRUB-Shell

GRUB existiert in zwei Versionen. Einmal als Bootloader und einmal als normales Linux-Programm unter `/usr/sbin/grub`. Dieses Programm wird als *GRUB-Shell* bezeichnet. Die Funktionalität, GRUB als Bootloader auf eine Festplatte oder Diskette zu installieren, ist direkt in GRUB integriert in Form der Kommandos `install` oder `setup`. Damit ist sie in der

GRUB-Shell verfügbar, wenn Linux geladen ist. Diese Befehle sind aber auch schon *während* des Bootvorgangs verfügbar, ohne dass Linux dazu laufen müsste. Dadurch vereinfacht sich die Rettung eines defekten Systems.

Nur wenn die GRUB-Shell als Linux-Programm läuft (aufzurufen mit `grub` wie beispielsweise unter Abschnitt 7.4.2 auf Seite 214 beschrieben), kommt der Zuordnungsalgorithmus von GRUB-Device und Linux-Device-Namen ins Spiel. Das Programm liest hierzu die Datei `device.map`. Mehr dazu im Abschnitt 7.4.2 auf Seite 214.

7.4.4 Bootpasswort setzen

GRUB unterstützt schon zum Bootzeitpunkt den Zugriff auf Dateisysteme, das heißt es können auch solche Dateien Ihres Linux-Systems eingesehen werden, zu denen Benutzer ohne Root-Rechte im einmal gestarteten System keinen Zugriff hätten. Durch Vergabe eines Passworts verhindern Sie solche Zugriffe. Einerseits können Sie lediglich den Dateisystemzugriff zur Bootzeit für Unbefugte sperren oder auch das Ausführen bestimmter Betriebssysteme für die Benutzer sperren.

Zur Vergabe eines Boot-Passworts gehen Sie als Benutzer `root` folgendermaßen vor:

- Geben Sie am Rootprompt `grub` ein.
- Verschlüsseln Sie in der GRUB-Shell das Passwort:

```
grub> md5crypt
Password: ****
Encrypted: $1$lS2dv/$JOYcdxIn7CJk9xShzzJVw/
```

- Fügen Sie den verschlüsselten Wert in den globalen Abschnitt der Datei `menu.lst` ein:

```
gfxmenu (hd0,4)/message
color white/blue black/light-gray
default 0
timeout 8
password --md5 $1$lS2dv/$JOYcdxIn7CJk9xShzzJVw/
```

Nun ist das Ausführen von GRUB-Befehlen am Bootprompt geschützt. Erst nach Eingabe von (P) und des Passworts wird diese Möglichkeit wieder freigegeben. Das Starten eines Betriebssystems aus dem Bootmenü heraus ist weiterhin für alle Benutzer möglich.

- Um zusätzlich das Starten einer oder mehrerer Betriebssysteme aus dem Bootmenü zu verhindern, ergänzen Sie in der Datei `menu.lst` den Eintrag `lock` für jeden Abschnitt, der nicht ohne Passwortheingabe starten soll. Im Beispiel sähe dies so aus:

```
title linux
kernel (hd0,4)/vmlinuz root=/dev/hda7 vga=791
initrd (hd0,4)/initrd
lock
```

Nach einem Reboot des Systems und der Auswahl des Linux-Eintrags im Bootmenü erscheint zunächst folgende Fehlermeldung:

```
Error 32: Must be authenticated
```

Drücken Sie **(Enter)**, um ins Menü zu gelangen und anschließend **(p)**, um einen Prompt für das Passwort zu erhalten. Nach Eingabe des Passworts und **(Enter)** bootet das gewünschte Betriebssystem (in diesem Fall Linux).

Hinweis

Bootpasswort und Splashscreen

Verwenden Sie ein Bootpasswort für GRUB, steht Ihnen der gewohnte Splashscreen nicht zur Verfügung.

Hinweis

7.4.5 Mögliche Probleme und weiterführende Informationen

Hinweis

Bootprobleme mit GRUB

GRUB überprüft die Geometrie der angeschlossenen Festplatten erst beim Booten. In seltenen Fällen macht das BIOS hier inkonsistente Angaben, so dass GRUB einen GRUB Geom Error meldet (siehe hierzu http://portal.suse.com/sdb/de/2003/02/fhassel_geom-error.html). In solchen Fällen verwenden Sie LILO oder aktualisieren ggf. das BIOS. Detaillierte Informationen zur Installation, Konfiguration und Wartung von LILO finden Sie in der Support-Datenbank http://portal.suse.de/sdb/de/2004/01/lilo_overview.html.

Hinweis

Auf der Webseite <http://www.gnu.org/software/grub/> finden Sie ausführliche Informationen zu GRUB in den Sprachen Deutsch und Englisch. Das Online-Handbuch existiert allerdings nur in Englisch.

Wenn `texinfo` auf dem Rechner installiert ist, können Sie sich in der Shell mit `info grub` die Info-Seiten zu GRUB anzeigen lassen. Suchen Sie auch in der Support-Datenbank <http://portal.suse.de/sdb/de/index.html> nach dem Stichwort GRUB, um Informationen zu speziellen Themen zu erhalten.

7.5 Linux-Bootloader entfernen

Es stehen zwei Methoden zur Auswahl, um den Linux-Bootloader zu deinstallieren:

- Spielen Sie das Backup des ursprünglichen MBRs mittels des YaST-Bootloader-Moduls wieder ein. YaST erstellt dieses Backup automatisch. Eine Beschreibung des YaST Bootloader-Moduls finden Sie im Installationsteil des *Benutzerhandbuchs*.
- Installieren Sie einen anderen Bootloader oder stellen Sie den DOS/Windows-MBR wieder her.

Achtung

Ein Backup des Bootsektors wird ungültig, wenn die betreffende Partition ein neues Dateisystem erhalten hat. Die Partitionstabelle in einem MBR-Backup wird ungültig, wenn die betreffende Festplatte zwischenzeitlich anders partitioniert worden ist. Solche Backups sind „Zeitbomben“: am Besten sofort in `/boot/backup.mbr` löschen!

Achtung

7.5.1 MBR wiederherstellen (DOS/Win9x/ME)

Einen DOS- oder Windows-MBR stellt man mit dem MS-DOS-Befehl `fdisk /MBR` (verfügbar ab DOS Version 5.0) wieder her. Oder unter OS/2 mit dem Befehl: `fdisk /newmbr`.

Diese Befehle schreiben nur die 446 ersten Bytes (den Boot-Code) in den MBR zurück und lassen die gegenwärtige Partitionstabelle unangetastet.

Außer, wenn der MBR (siehe 7.1.1 auf Seite 204) wegen einer falschen „magischen Zahl“ als im ganzen ungültig behandelt wird: dann wird die Partitionstabelle genullt. *Nicht vergessen:* Mit `fdisk` die von jetzt an gewünschte Startpartition wieder als *aktiv bootable* kennzeichnen; die MBR-Routinen von DOS, Windows, OS/2 brauchen das.

7.5.2 MBR wiederherstellen (Windows XP)

Booten Sie von der Windows XP CD, drücken Sie im Setup die Taste (R), um die Wiederherstellungskonsole zu starten. Wählen Sie aus der Liste Ihre Windows XP Installation aus und geben Sie das Administratorpasswort ein. Geben Sie in die Eingabeaufforderung den Befehl `FIXMBR` ein und bestätigen Sie die Sicherheitsabfrage mit `j`. Mit `exit` können Sie den Computer anschließend neu starten.

7.5.3 MBR wiederherstellen (Windows 2000)

Booten Sie von der Windows 2000 CD, drücken Sie im Setup die Taste (R), sowie im darauf folgenden Menü die Taste (K), um die Wiederherstellungskonsole zu starten. Wählen Sie aus der Liste Ihre Windows 2000 Installation aus und geben Sie das Administratorpasswort ein. Geben Sie in die Eingabeaufforderung den Befehl `FIXMBR` ein und bestätigen Sie die Sicherheitsabfrage mit `j`. Mit `exit` können Sie den Computer anschließend neu starten.

7.6 Für alle Fälle: Boot-CD erstellen

Falls Sie Probleme haben, Ihr installiertes System über einen Bootmanager zu booten oder Lilo oder Grub nicht in den MBR Ihrer Festplatte, noch auf eine Diskette installieren wollen oder können, ist es auch möglich, eine bootfähige CD zu erstellen, auf die Sie die Linux Startdateien brennen. Voraussetzung hierfür ist natürlich, dass ein Brenner in Ihrem System vorhanden und eingerichtet ist.

7.6.1 Boot-CD mit ISOLINUX

Um eine bootfähige CD zu erstellen, ist es am einfachsten, den Bootmanager Isolinux zu verwenden. Auch die SuSE Installations-CDs werden übrigens per Verwendung von Isolinux bootfähig gemacht.

- Booten Sie Ihr installiertes System zunächst auf folgendem Umweg: Legen Sie die Installations-CD oder -DVD in Ihr Laufwerk und booten Sie von dieser wie bei der Installation. Wählen Sie dann im Bootmenü die Option 'Installation' aus und im nächsten Menu den Punkt 'Installiertes System booten'. Dabei wird die root-Partition automatisch erkannt, sodass von dieser das System gebootet werden kann.
- Installieren Sie mit Hilfe von YaST das `syslinux`.
- Öffnen Sie eine Root-Shell. Mit Hilfe der folgenden Aufrufe wird für die Boot-CD ein temporäres Verzeichnis erstellt und die zum Booten des Linux Systems notwendigen Dateien (der Bootloader Isolinux sowie der Kernel und die Initrd) hineinkopiert.

```
mkdir /tmp/CDroot
cp /usr/share/syslinux/isolinux.bin /tmp/CDroot/
cp /boot/vmlinuz /tmp/CDroot/linux
cp /boot/initrd /tmp/CDroot
```

- Erstellen Sie mit einem Editor die Bootloader-Konfigurationsdatei `/tmp/CDroot/isolinux.cfg`. Tragen Sie folgenden Inhalt ein:

```
DEFAULT linux
LABEL linux
    KERNEL linux
    APPEND initrd=initrd root=/dev/hdXY [bootparameter]
```

Für den Parameter `root=/dev/hdXY` tragen Sie bitte Ihre root-Partition ein. Wenn Sie sich nicht sicher sind, welche Partitionsbezeichnung diese hat, schauen Sie einfach in der Datei `/etc/fstab` nach. Für den Wert `[bootparameter]` können Sie zusätzliche Optionen eingeben, die beim Booten verwendet werden sollen. Die Konfigurationsdatei könnte zum Beispiel folgendermaßen aussehen:

```
DEFAULT linux
LABEL linux
    KERNEL linux
    APPEND initrd=initrd root=/dev/hda7 hdd=ide-scsi
```

- Anschließend wird mit folgendem Aufruf aus den Dateien ein ISO9660-Dateisystem für die CD erstellt (schreiben Sie das folgende Kommando in eine Zeile):


```
mkisofs -o /tmp/bootcd.iso -b isolinux.bin -c boot.cat  
-no-emul-boot -boot-load-size 4  
-boot-info-table /tmp/CDroot
```

- Brennen Sie die Datei `/tmp/bootcd.iso` auf CD, entweder mit graphischen Brennprogrammen wie K3b oder einfach von der Kommandozeile: `cdrecord -v -eject speed=2 dev=0,0,0 /tmp/bootcd.iso`.

Evtl. muss der Parameter `dev=0,0,0` an die SCSI-ID des Brenners angepasst werden (diese erfahren Sie durch Eingabe des Aufrufs `cdrecord -scanbus`, vergleiche auch die Manualpage mit `man cdrecord`).

- Testen Sie die Boot-CD! Rebooten Sie dazu den Computer und überprüfen Sie ob Ihr Linux-System korrekt von der CD gestartet wird.

Mobiles Arbeiten unter Linux

Dieses Kapitel befasst sich mit den Besonderheiten des mobilen Arbeitens — schwerpunktmäßig auf Notebooks — unter Linux. Die Konfiguration von PC-Karten (PCMCIA) wird ebenso behandelt wie die Verwaltung verschiedener Systemprofile mittels SCPM und die drahtlose Kommunikation mittels IrDA und Bluetooth.

8.1	PCMCIA	224
8.2	SCPM – System Configuration Profile Management	235
8.3	IrDA – Infrared Data Association	244
8.4	Bluetooth – Geräte drahtlos verbinden	247

8.1 PCMCIA

PCMCIA steht für *Personal Computer Memory Card International Association* und wird als Sammelbegriff für sämtliche damit zusammenhängende Hard- und Software verwendet.

8.1.1 Die Hardware

Die wesentliche Komponente ist die PCMCIA-Karte; hierbei unterscheidet man zwei Typen:

PC-Karten Das sind die derzeit noch am häufigsten vorkommenden Karten. Sie verwenden einen 16 Bit breiten Bus zur Datenübertragung, sind meist relativ günstig und werden in der Regel problemlos und stabil unterstützt.

CardBus-Karten Dies ist ein neuerer Standard. Sie verwenden einen 32 Bit breiten Bus, sind dadurch schneller, aber auch teurer. Da die Datenübertragungsrate aber häufig an anderer Stelle eingeschränkt wird, lohnt sich der Aufwand oftmals nicht. Es gibt mittlerweile auch für diese Karten etliche Treiber, wobei manche immer noch instabil sind – abhängig auch vom vorhandenen PCMCIA-Controller.

Welche Karte eingeschoben ist, sagt bei aktivem PCMCIA-Dienst das Kommando `cardctl ident`. Eine Liste von unterstützten Karten findet man in `SUPPORTED.CARDS` in `/usr/share/doc/packages/pcmcia/`. Dort gibt es auch die jeweils aktuelle Version des PCMCIA-HOWTO.

Die zweite notwendige Komponente ist der PCMCIA-Controller, oder auch die PC-Card/CardBus-Bridge. Diese stellt die Verbindung zwischen der Karte und dem PCI-Bus her, in älteren Geräten auch die Verbindung zum ISA-Bus. Diese Controller sind fast immer zu dem Intel-Chip i82365 kompatibel; es werden alle gängigen Modelle unterstützt. Der Typ des Controllers lässt sich mit dem Kommando `pcic_probe` ermitteln. Falls es ein PCI-Gerät ist, gibt das Kommando `lspci -vt` weitere Auskünfte.

8.1.2 Die Software

Unterschiede zwischen beiden PCMCIA-Systemen

Gegenwärtig gibt es zwei PCMCIA-Systeme: externes PCMCIA und Kernel-PCMCIA. Das externe PCMCIA-System von David Hinds ist das

ältere, somit auch besser erprobte und wird immer noch weiterentwickelt. Die Quellen der verwendeten Module sind nicht in die Kernelquellen integriert, deshalb „externes“ System. Seit Kernel 2.4 gibt es alternative Module in den Kernelquellen. Diese bilden das PCMCIA-System des Kernels („Kernel-PCMCIA“). Die Basismodule wurden von Linus Torvalds geschrieben und unterstützen vor allem neuere CardBus-Bridges besser.

Leider sind diese beiden Systeme zueinander inkompatibel. Außerdem gibt es in beiden Systemen unterschiedliche Sätze von Kartentreibern. Deswegen kommt je nach verwendeter Hardware nur ein System in Frage. Die Voreinstellung unter SUSE LINUX ist das neuere Kernel-PCMCIA. Es ist jedoch möglich, das System zu wechseln. Dazu muss in der Datei `/etc/sysconfig/pcmcia` der Variablen `PCMCIA_SYSTEM` entweder der Wert `external` oder `kernel` zugewiesen und PCMCIA mit `rcpcmcia restart` neu gestartet werden. Für vorübergehende Wechsel können Sie auch `rcpcmcia [re]start external,kernel` verwenden. Detailinformationen dazu befindet sich in `/usr/share/doc/packages/pcmcia/LIESMICH.SuSE`.

Die Basismodule

Die Kernelmodule für beide Systeme befinden sich in den Kernelpaketen. Zusätzlich werden noch die Pakete `pcmcia` und `hotplug` benötigt.

Beim Start von PCMCIA werden die Module `pcmcia_core`, `i82365` (externes PCMCIA) oder `yenta_socket` (Kernel-PCMCIA) und `ds` geladen. In sehr seltenen Fällen wird alternativ zu `i82365` bzw. `yenta_socket` das Modul `tcic` benötigt. Sie initialisieren die vorhandenen PCMCIA-Controller und stellen Basisfunktionen zur Verfügung.

Der Cardmanager

Da PCMCIA-Karten zur Laufzeit gewechselt werden können, muss es einen Daemonen geben, der die Aktivitäten in den Steckplätzen überwacht. Diese Aufgabe erledigen je nach gewähltem PCMCIA-System und verwendeter Hardware der Cardmanager oder das Hotplug-System des Kernels. Bei externem PCMCIA kommt nur der Cardmanager zum Einsatz. Bei Kernel-PCMCIA handhabt der Cardmanager nur die PC-Card-Karten, wohingegen CardBus-Karten von Hotplug behandelt werden. Der Cardmanager wird vom PCMCIA-Startskript nach dem Laden der Basismodule gestartet. Da Hotplug neben PCMCIA auch noch andere Subsysteme bedient, gibt es hierfür ein eigenes Startskript.

Ist eine Karte eingeschoben, ermittelt der Cardmanager bzw. Hotplug Typ und Funktion und lädt die passenden Module. Wurden diese erfolgreich

geladen, startet der Cardmanager bzw. Hotplug je nach Funktion der Karte bestimmte Initialisierungsskripten, die ihrerseits die Netzwerkverbindung aufbauen, Partitionen von externen SCSI-Platten einhängen (mounten) oder andere hardwarespezifische Aktionen ausführen. Die Skripte des Cardmanagers befinden sich in `/etc/pcmcia/`. Die Skripte für Hotplug sind in `/etc/hotplug/` zu finden. Wenn die Karte wieder entfernt wird, beendet der Cardmanager bzw. Hotplug mit den selben Skripten sämtliche Kartenaktivitäten. Anschließend werden die nicht mehr benötigten Module wieder entladen.

Sowohl der Startvorgang von PCMCIA als auch die Kartenereignisse werden in der Systemprotokolldatei (`/var/log/messages`) protokolliert. Dort wird festgehalten, welches PCMCIA System gerade verwendet wird und welcher Daemon welche Skripte zur Einrichtung verwendet hat. Theoretisch kann eine PCMCIA-Karte einfach entnommen werden. Dies funktioniert auch hervorragend für Netzwerk-, Modem- oder ISDN-Karten, solange keine aktiven Netzwerkverbindungen mehr bestehen. Es funktioniert nicht im Zusammenhang mit eingehängten Partitionen einer externen Platte oder mit NFS-Verzeichnissen. Hier müssen Sie dafür sorgen, dass die Einheiten synchronisiert und sauber ausgehängt werden (unmounten). Das ist natürlich nicht mehr möglich, wenn die Karte bereits herausgenommen wurde. Im Zweifelsfall hilft ein `cardctl eject`.

Dieser Befehl deaktiviert alle Karten, die sich noch im Notebook befinden. Um nur eine der Karten zu deaktivieren, können Sie zusätzlich die Slotnummer angeben, zum Beispiel `cardctl eject 0`.

8.1.3 Die Konfiguration

Ob PCMCIA bzw. Hotplug beim Booten gestartet wird, lässt sich mit dem Runleveleditor von YaST oder auf der Kommandozeile mittels `chkconfig` einstellen.

In `/etc/sysconfig/pcmcia` befinden sich vier Variablen:

PCMCIA_SYSTEM bestimmt, welches PCMCIA-System verwendet wird.

PCMCIA_PCIC enthält den Namen des Moduls, das den PCMCIA-Controller ansteuert. Im Normalfall ermittelt das Startskript diesen Namen selbstständig. Nur wenn dies fehlschlägt, kann das Modul hier eingetragen werden. Ansonsten sollte diese Variable leer bleiben.

PCMCIA_CORE_OPTS ist für Parameter für das Modul `pcmcia_core` gedacht; sie werden aber nur selten benötigt. Diese Optionen sind in der Manualpage von `pcmcia_core` beschrieben.

PCMCIA_PCIC_OPTS nimmt Parameter für das Modul `i82365` auf; vgl. die Manualpage von `i82365`. Falls `yenta_socket` verwendet wird, werden diese Optionen ignoriert, da `yenta_socket` keine Optionen kennt.

Die Zuordnung von Treibern zu PCMCIA-Karten für den Cardmanager befindet sich in den Dateien `/etc/pcmcia/config` und `/etc/pcmcia/*.conf`. Zuerst wird `config` gelesen und dann die `/*.conf` in alphabetischer Reihenfolge. Der zuletzt gefundene Eintrag für eine Karte ist ausschlaggebend. Details über die Syntax dieser Dateien befinden sich in der Manualpage von `pcmcia`.

Netzwerkkarten (Ethernet, Wireless LAN und TokenRing)

Diese lassen sich wie gewöhnliche Netzwerkkarten mit YaST einrichten. Dort muss lediglich PCMCIA als Kartentyp ausgewählt werden. Alle weiteren Details zur Netzwerkeinrichtung befinden sich im Kapitel 14.4 auf Seite 370. Beachten Sie dort die Hinweise zu hotplugfähigen Karten.

ISDN

Auch bei ISDN-PC-Karten erfolgt die Konfiguration größtenteils wie bei sonstigen ISDN-Karten mit YaST. Es spielt keine Rolle welche der dort angebotenen PCMCIA ISDN-Karten ausgewählt wird; wichtig ist nur, dass es eine PCMCIA-Karte ist. Bei der Einrichtung der Hardware und der Provider ist darauf zu achten, dass der Betriebsmodus immer auf `hotplug`, nicht auf `onboot` steht.

So genannte ISDN-Modems gibt es auch bei PCMCIA-Karten. Dies sind Modem- oder Multifunktionskarten mit einem zusätzlichen ISDN-Connection-Kit und werden wie ein Modem behandelt.

Modem

Bei Modem-PC-Karten gibt es im Normalfall keine PCMCIA-spezifischen Einstellungen. Sobald ein Modem eingeschoben wird, steht dieses unter `/dev/modem` zur Verfügung.

Es gibt auch bei PCMCIA-Karten so genannte Softmodems. Diese werden in der Regel nicht unterstützt. Falls es Treiber gibt, müssen diese individuell ins System eingebunden werden.

SCSI und IDE

Das passende Treibermodul wird vom Cardmanager oder Hotplug geladen. Sobald also eine SCSI- oder IDE-Karte eingeschoben wird, stehen die daran angeschlossenen Geräte zur Verfügung. Die Gerätenamen werden dynamisch ermittelt. Informationen über vorhandene SCSI- bzw. IDE-Geräte sind unter `/proc/scsi/` bzw. unter `/proc/ide/` zu finden.

Externe Festplatten, CD-ROM-Laufwerke und ähnliche Geräte müssen eingeschaltet sein, bevor die PCMCIA-Karte in den Steckplatz eingeschoben wird. SCSI-Geräte müssen aktiv terminiert werden.

Achtung

Bevor eine SCSI- oder IDE-Karte entnommen wird, müssen sämtliche Partitionen der daran angeschlossenen Geräte ausgehängt werden.

Wurde dies vergessen, kann erst nach einem Reboot des Systems erneut auf diese Geräte zugegriffen werden, auch wenn der Rest des Systems durchaus stabil weiterläuft.

Achtung

Sie können Linux auch vollständig auf solchen externen Platten installieren. Allerdings gestaltet sich dann der Bootvorgang etwas komplizierter. Es wird auf alle Fälle eine Bootdisk benötigt, die den Kernel und eine Initial-Ramdisk (`initrd`) enthält; mehr Informationen dazu finden Sie in Abschnitt 12.3 auf Seite 305.

Die `initrd` enthält ein virtuelles Dateisystem, das alle benötigten PCMCIA-Module und -Programme enthält. Die Bootdisk bzw. die Bootdisk-Images sind ebenso aufgebaut. Damit könnten Sie Ihre externe Installation immer booten. Es ist jedoch umständlich, jedes Mal die PCMCIA-Unterstützung von Hand zu laden. Fortgeschrittene Anwender können sich eine auf das jeweilige System zugeschnittene Bootdiskette selbst erstellen. Hinweise finden Sie dazu in dem englischsprachigem PCMCIA-HOWTO in Abschnitt 5.3 *Booting from a PCMCIA device*.

8.1.4 Wenn's trotzdem nicht geht

Bisweilen kommt es bei der Verwendung von PCMCIA auf manchen Notebooks oder mit manchen Karten zu Problemen. Die meisten Schwierigkeiten lassen sich mit wenig Aufwand bewältigen, solange man die Sache systematisch angeht.

Achtung

Da es in SUSE LINUX sowohl externes als auch Kernel-PCMCIA nebeneinander gibt, muss beim manuellen Laden von Modulen eine Besonderheit beachtet werden. Die beiden PCMCIA-Systeme verwenden Module gleichen Namens und sind in unterschiedlichen Unterverzeichnissen unter `/lib/modules/<kernelversion>/` untergebracht. Diese Unterverzeichnisse heißen `pcmcia/` für Kernel-PCMCIA und `pcmcia-external/` für externes PCMCIA. Deshalb muss beim manuellen Laden von Modulen dieses Unterverzeichnis angegeben werden:

```
modprobe -t <verzeichnis> <modulname>
```

Achtung

Zuerst ist herauszufinden, ob das Problem mit einer Karte zusammenhängt, oder ob ein Problem des PCMCIA-Basissystems vorliegt. Deshalb sollten Sie in jedem Fall den Computer zunächst ohne eingeschobene Karten starten. Erst wenn das Basissystem einwandfrei zu funktionieren scheint, wird die Karte eingeschoben. Alle aufschlussreichen Meldungen werden in `/var/log/messages/` protokolliert. Deshalb sollte die Datei mit `tail -f /var/log/messages` während der notwendigen Tests beobachtet werden. So lässt sich der Fehler auf einen der beiden folgenden Fälle einschränken.

Das PCMCIA-Basissystem funktioniert nicht

Wenn das System beim Booten bereits bei der Meldung "PCMCIA: Starting services" stehen bleibt oder andere merkwürdige Dinge geschehen, kann das Starten von PCMCIA beim nächsten Booten durch die Eingabe von `NOPCMCIA=yes` am Bootprompt verhindert werden. Um den Fehler weiter einzugrenzen, werden nun die drei Basismodule des verwendeten PCMCIA Systems von Hand nacheinander geladen.

Dazu dienen die Kommandos (als Benutzer `root` aufzurufen):

```
modprobe -t <dir> pcmcia_core
modprobe -t pcmcia-external i82365
```

bei externem PCMCIA bzw.

```
modprobe -t pcmcia yenta_socket
```

bei Kernel-PCMCIA

bzw. – in sehr seltenen Fällen – `modprobe -t <dir> tcic` und

`modprobe -t <dir> ds`

Die kritischen Module sind die beiden ersten.

Tritt der Fehler beim Laden von `pcmcia_core` auf, hilft die Manpage zu `pcmcia_core` weiter. Die darin beschriebenen Optionen können zunächst zusammen mit dem Kommando `modprobe` getestet werden. Als Beispiel können wir die APM-Unterstützung der PCMCIA-Module abschalten; in wenigen Fällen kann es damit Probleme geben. Dafür gibt es die Option `doapm`; mit `do_apm=0` wird das Powermanagement deaktiviert:

`modprobe -t <dir> pcmciacore do_apm=0`

Führt die gewählte Option zum Erfolg, wird sie in der Datei `/etc/sysconfig/pcmcia` in die Variable `PCMCIA_CORE_OPTS` geschrieben:

`PCMCIA_CORE_OPTS="do_apm=0"`

Vereinzelt kann das Prüfen freier IO-Bereiche Ärger machen, wenn sich dadurch andere Hardwarekomponenten gestört fühlen. Das umgeht man dann mit `probe_io=0`. Sollen mehrere Optionen verwendet werden, müssen sie durch Leerzeichen getrennt werden:

`PCMCIA_CORE_OPTS=do_apm=0 probe_io=0`

Wenn es beim Laden des Moduls `i82365` zu Fehlern kommt, hilft die Manpage von `i82365`.

Ein Problem in diesem Zusammenhang ist ein Ressourcenkonflikt, ein Interrupt, IO-Port oder Speicherbereich wird doppelt belegt. Das Modul `i82365` prüft zwar diese Ressourcen, bevor sie für eine Karte zur Verfügung gestellt werden, jedoch führt manchmal genau diese Prüfung zum Problem. So führt das Prüfen des Interrupt 12 (PS/2-Geräte) bei manchen Computern zum Blockieren von Maus und/oder Tastatur. In diesem Fall hilft der Parameter `irq_list=<ListevonIRQs>`. Die Liste soll alle IRQs enthalten, die verwendet werden dürfen. Also `modprobe i82365 irq_list=5,7,9,10` oder dauerhaft in `/etc/sysconfig/pcmcia`:

```
PCMCIA_PCIC_OPTS="irq_list=5,7,9,10"
```

Weiterhin gibt es `/etc/pcmcia/config` und `/etc/pcmcia/config.opts`. Diese Dateien werden vom Cardmanager ausgewertet. Die darin gemachten Einstellungen sind erst für das Laden der Treiber-Module für die PCMCIA-Karten relevant.

In `/etc/pcmcia/config.opts` können auch IRQs, IO-Ports und Speicherbereiche ein- oder ausgeschlossen werden. Der Unterschied zur Option `irqlist` ist, dass die in der Datei `config.opts` ausgeschlossenen Ressourcen zwar nicht für eine PCMCIA-Karte verwendet, aber dennoch vom Basis-Modul `i82365` geprüft werden.

Die PCMCIA-Karte funktioniert nicht (richtig)

Hier gibt es im Wesentlichen drei Varianten: Die Karte wird nicht erkannt, der Treiber kann nicht geladen werden oder das Interface, das vom Treiber bereitgestellt wird, wurde falsch eingerichtet.

Man sollte beachten, ob die Karte vom Cardmanager oder von Hotplug behandelt wird. Nochmal zur Erinnerung: Bei externem PCMCIA regiert immer der Cardmanager, bei Kernel-PCMCIA behandelt der Cardmanager PC-Card-Karten und Hotplug behandelt CardBUS-Karten. Hier wird nur der Cardmanager besprochen.

Die Karte wird nicht erkannt Wenn die Karte nicht erkannt wird, erscheint in `/var/log/messages` die Meldung "unsupported Card in Slot x". Diese Meldung besagt lediglich, dass der Cardmanager der Karte keinen Treiber zuordnen kann. Zu dieser Zuordnung wird `/etc/pcmcia/config` bzw. `/etc/pcmcia/*.conf` benötigt. Diese Dateien sind sozusagen die Treiberdatenbank. Diese Treiberdatenbank lässt sich am leichtesten erweitern, wenn man vorhandene Einträge als Vorlage nimmt. Sie können mit dem Kommando `cardctl ident` herausfinden, wie die Karte sich identifiziert. Weitere Informationen dazu befinden sich im PCMCIA-HOWTO (Abschnitt 6) und in der Manualpage von `pcmcia`. Nach der Änderung von `/etc/pcmcia/config` bzw. `/etc/pcmcia/*.conf` muss die Treiberzuordnung neu geladen werden; dazu genügt ein `rcpcmcia reload`.

Treiber wird nicht geladen Eine Ursache hierfür besteht darin, dass in der Treiberdatenbank eine falsche Zuordnung gespeichert ist. Das kann zum Beispiel daher kommen, dass ein Hersteller in ein äußerlich unverändertes Kartenmodell einen anderen Chip einbaut. Manchmal

gibt es auch alternative Treiber, die bei bestimmten Modellen besser (oder überhaupt erst) funktionieren als der voreingestellte Treiber. In diesen Fällen werden genaue Informationen über die Karte benötigt. Hier hilft auch, eine Mailingliste oder den Advanced Support Service zu fragen.

Eine weitere Ursache ist ein Ressourcenkonflikt. Bei den meisten PCMCIA-Karten ist es nicht relevant, mit welchem IRQ, IO-Port oder Speicherbereich sie betrieben werden, aber es gibt auch Ausnahmen. Dann sollte man zuerst immer nur eine Karte testen und evtl. auch andere Systemkomponenten wie zum Beispiel Soundkarte, IrDA, Modem oder Drucker vorübergehend abschalten. Die Ressourcenverteilung des Systems kann man mit `lsdev` einsehen (Es ist durchaus normal, dass mehrere PCI Geräte denselben IRQ verwenden).

Eine Lösungsmöglichkeit wäre, eine geeignete Option für das Modul `i82365` zu verwenden (siehe oben `PCMCIA_PCIC_OPTS`). Es gibt jedoch auch für manche Kartentreibermodule Optionen. Diese lassen sich mit `modinfo /lib/modules/<richtigesPCMCIA-Verzeichnis>/<treiber>.o` herausfinden (der vollständige Pfad ist hier wieder nötig, um den Treiber vom richtigen PCMCIA-System anzusprechen). Für die meisten Module gibt es auch eine Manualpage.

`rpm -ql pcmcia | grep man` listet alle im `pcmcia` enthaltene Manualpages auf. Zum Testen der Optionen können die Kartentreiber auch von Hand entladen werden. Hierbei ist wieder zu beachten, dass das Modul des gerade verwendeten PCMCIA-Systems zu verwenden. Siehe die Warnung weiter oben.

Wenn eine Lösung gefunden wurde, kann in `/etc/pcmcia/config.opts` die Verwendung einer bestimmten Ressource allgemein erlaubt bzw. verboten werden. Auch die Optionen für Kartentreiber finden hier Platz. Soll zum Beispiel das Modul `pcnet_cs` ausschließlich mit dem IRQ 5 betrieben werden, wird folgender Eintrag benötigt:

```
module pcnet_cs opts irq_list=5
```

Mit 10/100-MBit-Netzwerkkarten tritt manchmal das Problem auf, dass die Übertragungsart nicht automatisch richtig erkannt wird. Hier hilft das Kommando `ifport` oder `mii_tool`. Damit lässt sich die eingestellte Übertragungsart anzeigen und verändern. Um diese Kommandos automatisch ausführen zu lassen, muss das Skript `/etc/pcmcia/network` individuell angepasst werden.

Interface wird falsch konfiguriert In diesem Fall ist es empfehlenswert, die Konfiguration des Interfaces nochmal genau zu überprüfen, um seltene Konfigurationsfehler auszuschließen. Bei Netzwerkkarten kann außerdem die Dialograte der Netzwerkskripten erhöht werden, in dem man in `/etc/sysconfig/network/config` der Variable `DEBUG` den Wert `yes` zuweist. Bei anderen Karten, oder wenn das noch nicht hilft, gibt es noch die Möglichkeit, in das vom Cardmanager aufgerufene Skript (siehe `/var/log/messages`) eine Zeile `set -x` einzubauen. Dadurch wird jedes einzelne Kommando des Skripts im Systemlog protokolliert. Hat man die kritische Stelle in einem Skript gefunden, können die entsprechenden Kommandos auch in einem Terminal eingegeben und getestet werden.

8.1.5 Installation via PCMCIA

In manchen Fällen wird PCMCIA bereits zum Installieren benötigt, wenn man über Netzwerk installieren möchte oder das CD-ROM via PCMCIA betrieben wird. Dazu muss man mit einer Bootdiskette starten. Des Weiteren wird eine der Moduldisketten benötigt.

Nach dem Booten von Diskette (oder auch nach der Auswahl 'manuelle Installation' beim Booten von CD) wird das Programm `linuxrc` gestartet. Dort muss unter dem Menüpunkt 'Kernel-Module (Hardware-Treiber)' der Punkt 'Lade PCMCIA Module' ausgewählt werden. Zuerst erscheinen zwei Eingabefelder, in denen man Optionen für die Module `pcmcia_core` und `i82365` eingeben kann. Im Normalfall bleiben diese Felder jedoch leer. Die Manualpages für `pcmcia_core` und `i82365` befinden sich als Textdateien auf der ersten CD im Verzeichnis `docu/`.

Bei SUSE LINUX wird mit dem externen PCMCIA-System installiert. Während der Installation werden Systemmeldungen auf verschiedenen virtuellen Konsolen ausgegeben, auf die man mit `(Alt) + (Funktionstaste)` umschalten kann.

Später, wenn bereits eine grafische Oberfläche aktiv ist, muss man `(Strg) + (Alt) + (Funktionstaste)` verwenden.

Es gibt auch schon während der Installation Terminals, auf denen Kommandos ausgeführt werden können. Solange `linuxrc` läuft, ist das die Konsole 9 (eine sehr spartanisch ausgestattete Shell); sobald das Installationssystem geladen ist (YaST wurde gestartet) gibt es auf Konsole 2 eine `bash` und viele gängige Systemtools.

Wenn während der Installation ein falsches Treibermodul für eine PCMCIA Karte geladen wird, muss die Bootdiskette von Hand angepasst werden. Dazu benötigt man jedoch fortgeschrittene Linuxkenntnisse. Wenn

der erste Teil der Installation abgeschlossen ist, wird das System teilweise oder ganz neu gestartet. Dabei kann in seltenen Fällen beim Starten von PCMCIA das System stehen bleiben. Zu diesem Zeitpunkt ist die Installation aber schon weit genug fortgeschritten, sodass mit der Boot-Option `NOPCMCIA=yes` Linux ohne PCMCIA gestartet werden kann, zumindest im Textmodus. Hier hilft der Abschnitt 8.1.4 auf Seite 228 weiter. Evtl. kann man schon vor Abschluss des ersten Teils der Installation auf Konsole 2 einige Einstellungen am System verändern, so dass der Neustart erfolgreich verläuft.

8.1.6 Weitere Hilfsprogramme

Das Programm `cardctl` wurde schon mehrfach erwähnt. Diese Applikation ist das wesentliche Werkzeug, um Informationen von PCMCIA zu erhalten bzw. bestimmte Aktionen auszuführen. In der Datei `cardctl` finden Sie Details; oder Sie geben `cardctl` ein und erhalten eine Liste der gültigen Kommandos.

Zu diesem Programm gibt es auch ein grafisches Frontend `cardinfo`, mit dem die wichtigsten Dinge kontrollierbar sind. Dazu muss jedoch das Paket `pcmcia-cardinfo` installiert sein.

Weitere Helfer aus dem `pcmcia` Paket sind `ifport`, `ifuser`, `probe` und `rcpcmcia`. Diese werden aber nicht immer benötigt. Um genau zu erfahren, was alles im Paket `pcmcia` steckt, verwendet man den Befehl `rpm -ql pcmcia`.

8.1.7 Kernel oder PCMCIA Paket aktualisieren

Wenn Sie den Kernel aktualisieren möchten, sollten Sie die von SUSE bereitgestellten Kernelpakete verwenden. Ist es notwendig, einen eigenen Kernel zu kompilieren, dann müssen auch die PCMCIA-Module neu kompiliert werden. Wichtig ist, dass während der Neuübersetzung bereits der richtige Kernel läuft, da aus diesem einige Informationen extrahiert werden. Das `pcmcia` Paket sollte bereits installiert, aber nicht gestartet sein; im Zweifelsfall sollte Sie noch `rcpcmcia stop` ausführen. Dann installiert man das PCMCIA-Quellpaket mit und gibt anschließend ein:

```
rpm -ba /usr/src/packages/SPECS/pcmcia.spec
```

Danach liegen unter `/usr/src/packages/RPMS/` neue Pakete. Das Paket `pcmcia-modules` enthält die PCMCIA-Module für externes PCMCIA. Dieses Paket muss mit `rpm --force` installiert werden, da die Moduldateien offiziell zum Kernel-Paket gehören.

8.1.8 Weiterführende Informationen

Wer an Erfahrungen mit bestimmten Notebooks interessiert ist, sollte auf alle Fälle die Linux Laptop Homepage unter <http://linux-laptop.net> besuchen. Eine weitere gute Informationsquelle ist die TuxMobil-Hompage unter <http://tuxmobil.org/>. Dort findet man neben viele interessanten Informationen auch ein Laptop-Howto und ein IrDA-Howto. Außerdem gibt es in der Supportdatenbank mehrere Artikel zum mobilen Arbeiten unter SUSE LINUX. Suchen Sie unter <http://portal.suse.de/sdb/de/index.html> unter dem Stichwort *Laptop*.

8.2 SCPM – System Configuration Profile Management

Es gibt Situationen, in denen eine veränderte Konfiguration des Computersystems benötigt wird. Am häufigsten trifft dies auf mobile Computer zu, die an verschiedenen Standorten betrieben werden. Es kann aber auch sein, dass man auf einem Desktopsystem zeitweilig andere Hardwarekomponenten verwendet. In jedem Fall sollte eine Rückkehr zum ursprünglichen System einfach sein. Noch besser ist es, wenn diese Umkonfiguration auch noch einfach reproduzierbar ist.

Eine Lösung für dieses Problem gab es bisher nur für PCMCIA-Hardware. Dort konnte man verschiedene Konfigurationen in bestimmten Schemata ablegen. Ausgehend davon haben wir SCPM (engl. *System Configuration Profile Management*) entwickelt, das die Beschränkung auf PCMCIA aufhebt. Mit SCPM lässt sich ein frei wählbarer Teil der Systemkonfiguration festlegen, von dem verschiedene Zustände in eigenen Konfigurationsprofilen festgehalten werden können. Etwas freier ausgedrückt ist das, als ob man Schnappschüsse von seiner Systemkonfiguration macht, die man jederzeit wiederherstellen kann.

Das Hauptanwendungsgebiet wird vermutlich bei der Netzwerkkonfiguration von Laptops liegen. Aber unterschiedliche Netzwerkeinstellungen be-

einflussen meist auch noch andere Elemente, zum Beispiel die Einstellungen für e-Mail oder Proxies. Hierzu kommen unterschiedliche Drucker zu Hause und in der Firma oder die gesonderte XFree-Konfiguration für den Beamer bei Vorträgen, besonders sparsame Stromverbrauchseinstellungen für unterwegs oder eine andere Zeitzone in der Auslandsniederlassung.

Mit vermehrten Einsatz dieses Werkzeugs bilden sich immer wieder neue Anforderungen heraus. Wenn Sie selbst Anregungen und Kritik zu SCPM haben, dann nehmen Sie mit uns Kontakt auf. Wir sind sehr an Rückmeldungen interessiert. Wir haben SCPM auf ein flexibles Grundgerüst gestellt, sodass zum Beispiel auch ein serverbasiertes Profil Management möglich ist. Bitte teilen Sie uns Ihre Wünsche, Anregungen und Fehlerbeschreibungen über unser Webfrontend <http://www.suse.de/feedback> mit.

8.2.1 Grundbegriffe und Grundlagen

Vorab sollen einige Grundbegriffe festgelegt werden, die auch in der restlichen Dokumentation zu SCPM und im YaST-Modul so verwendet werden.

- Unter *Systemkonfiguration* verstehen wir die gesamte Konfiguration des Computers. Alle grundlegenden Einstellungen, wie die zum Beispiel Verwendung von Festplattenpartitionen oder Netzwerkeinstellungen, Zeitzonenauswahl oder Tastatureinstellungen.
- Ein *Profil* oder auch *Konfigurationsprofil* ist ein Zustand der Systemkonfiguration, der festgehalten wurde und der bei Bedarf einfach wiederhergestellt werden kann.
- Als *aktives Profil* wird immer das Profil bezeichnet, in das zuletzt geschaltet wurde. Das heißt nicht, dass die aktuelle Systemkonfiguration exakt diesem Profil entspricht, denn die Konfiguration kann jederzeit individuell verändert werden.
- *Resource* im Sinne von SCPM sind alle Elemente, die zur Systemkonfiguration beitragen. Das kann eine Datei oder ein Softlink einschließlich ihrer Metadaten, wie Benutzer, Rechte oder Zugriffszeit sein. Das kann aber auch ein Systemdienst sein, der einmal läuft und in einem anderen Profil ausgeschaltet ist.
- Die Ressourcen sind in sogenannten *Resource Groups* organisiert. Diese Gruppen enthalten jeweils Ressourcen, die logisch zusammenpassen. Für die meisten Gruppen bedeutet das, dass sie einen Dienst und

die dazugehörigen Konfigurationsdateien enthalten. Dieser Mechanismus erlaubt das einfache Zusammenstellen der Ressourcen, die von SCPM behandelt werden, ohne wissen zu müssen, welche Konfigurationsdateien für welche Dienste notwendig wären. SCPM beinhaltet bereits eine Vorauswahl an aktivierten Resource Groups, die für die meisten Benutzer ausreichend sein sollte.

8.2.2 Der YaST Profil-Manager und weiterführende Dokumentation

Als grafisches Frontend zu SCPM (Paket `scpm`) gibt es ein YaST-Modul (Paket `yast2-profile-manager`) als Alternative zu dem Kommandozeilen-Frontend. Da die Funktionalität beider Frontends im Wesentlichen dieselbe ist und die Kenntnis des Kommandozeilen-Frontends für viele Zwecke interessant ist, wird hier nur das letztere beschrieben. Die Bedienung des SCPM YaST-Moduls ist danach zusammen mit den dort angebotenen Hilfetexten sehr leicht. Die wenigen Besonderheiten des YaST-Moduls werden an passender Stelle erwähnt.

Die aktuellste Dokumentation ist in den Infoseiten zu SCPM zu finden. Diese kann mit Werkzeugen wie Konqueror oder Emacs eingesehen werden (`konqueror info:scpm`). In der Konsole verwendet man `info` oder `pinfo`. Technische Dokumentation für Benutzer, die selbst Hand an SCPM legen möchten, gibt es unter `/usr/share/doc/packages/scpm/`. Der Aufruf von `scpm` ohne weitere Argumente gibt eine Kommandoübersicht aus.

8.2.3 SCPM einrichten

Bevor mit SCPM gearbeitet werden kann, muss es erst einmal eingeschaltet werden. Standardmäßig behandelt SCPM Netzwerk- und Druckereinstellungen, sowie die XFree86 Konfiguration und einige Netzwerkdienste. Falls Sie darüber hinaus Dienste oder Konfigurationsdateien verwalten möchten, sollten Sie noch die entsprechenden Resource Groups aktivieren. Die bereits definierten Resource Groups können Sie mit dem Befehl `scpm list_groups` anzeigen lassen, wenn Sie nur die bereits aktiven Gruppen sehen möchten, geben Sie `scpm list_groups -a` ein. Die Kommandozeilenbefehle müssen als Benutzer `root` ausgeführt werden. Aktivieren bzw. Deaktivieren können Sie die Gruppen mit `scpm activate_group NAME` bzw. `scpm deactivate_group NAME`, wobei `NAME` durch

den entsprechenden Gruppennamen zu ersetzen ist. Sie können die Resource Groups auch bequem mit Hilfe des YaST Profil-Managers konfigurieren.

Mit dem Aufruf von `scpm enable` wird SCPM eingeschaltet. Beim ersten Einschalten wird SCPM initialisiert, was einige Sekunden in Anspruch nimmt. SCPM kann mit `scpm disable` jederzeit ausgeschaltet werden, um unbeabsichtigte Profilumschaltungen zu vermeiden. Beim anschließenden Wiedereinschalten wird der Betrieb einfach fortgesetzt.

8.2.4 Profile anlegen und verwalten

Nachdem SCPM eingeschaltet wurde, gibt es bereits ein Profil namens `default`. Eine Liste aller verfügbaren Profile gibt das Kommando `scpm list` aus. Dieses bisher einzige Profil ist zwangsläufig auch das aktive Profil. Das erfährt man mit `scpm active`. Das Profil `default` ist als Grundkonfiguration gedacht, von der die anderen Profile abgeleitet werden. Deshalb sollten zuerst alle Einstellungen, die in allen Profilen einheitlich sein sollen, vorgenommen werden. Mit `scpm reload` werden diese Änderungen dann im aktiven Profil gespeichert. Das Profil `default` kann dennoch beliebig verwendet, umbenannt oder gelöscht werden.

Es gibt zwei Möglichkeiten, ein neues Profil hinzuzufügen. Wenn das neue Profil (hier mit Namen `work`) zum Beispiel auf dem Profil `default` basieren soll, geschieht dies mit `scpm copy default work`. Danach kann man mit `scpm switch work` in das neue Profil umschalten und es dann konfigurieren. Manchmal hat man aber auch die Systemkonfiguration schon für bestimmte Zwecke verändert und möchte diese danach in einem neuen Profil festhalten. Das erledigt der Aufruf von `scpm add work`. Jetzt ist die aktuelle Systemkonfiguration im Profil `work` gesichert und das neue Profil als aktiv markiert; das heisst ein `scpm reload` sichert Änderungen jetzt im Profil `work`.

Selbstverständlich können Profile auch umbenannt oder gelöscht werden. Dafür gibt es die Kommandos `scpm rename x y` und `scpm delete x`. Um zum Beispiel `work` nach `arbeit` umzubenennen und es hinterher zu löschen, gibt man `scpm rename work arbeit` und dann `scpm delete arbeit` ein. Nur das aktive Profil kann nicht gelöscht werden.

Nochmal die einzelnen Kommandos:

`scpm list` gibt alle verfügbaren Profile aus

`scpm active` gibt das aktive Profil aus

scpm add <name> speichert die gegenwärtige Systemkonfiguration in einem neuen Profil und macht dieses zum aktiven

scpm copy <quellname> <zielname>

kopiert ein Profil

scpm rename <quellname> <zielname>

benennt ein Profil um

scpm delete <name> löscht ein Profil

Hinweis zum YaST-Modul: Hier gibt es nur den Knopf 'Hinzufügen'. Es erscheint dann aber die Frage, ob man ein existierendes Profil kopieren oder die gegenwärtige Systemkonfiguration sichern möchte. Zum Umbenennen verwendet man dort den Knopf 'Ändern'.

8.2.5 Zwischen Konfigurationsprofilen umschalten

Das Umschalten zu einem anderen Profil (hier `work`) wird mit dem Kommando `scpm switch work` ausgelöst. Es ist zulässig, zum gerade aktiven Profil umzuschalten um geänderte Einstellungen an der Systemkonfiguration zu sichern. Alternativ kann dafür aber auch das Kommando `scpm reload` verwendet werden.

Um den Umschaltvorgang und die dabei eventuell auftretenden Fragen besser zu verstehen, soll dieser hier näher erläutert werden. Zuerst prüft SCPM, welche Ressourcen des aktiven Profils seit dem letzten Umschalten verändert wurden. Aus der Liste der veränderten Ressourcen wird die Liste der geänderten Resource Groups erzeugt. Für jede dieser Gruppen wird anschließend nachgefragt, ob die Änderungen in das noch aktive Profil übernommen werden sollen. Falls Sie – wie es bei früheren Versionen von SCPM der Fall war – lieber die einzelnen Ressourcen angezeigt bekommen möchten, dann rufen Sie den Switch-Befehl mit dem Parameter `-r` auf, etwa so: `scpm switch -r work`.

Danach vergleicht SCPM die aktuelle Systemkonfiguration mit dem neuen Profil, in das umgeschaltet werden soll. Dabei wird ermittelt, welche Systemdienste aufgrund von Konfigurationsänderungen oder wegen gegenseitiger Abhängigkeiten angehalten bzw. (wieder) gestartet werden müssen. Das kann man sich wie einen teilweisen Systemreboot vorstellen, nur dass eben nur ein kleiner Teil des Systems betroffen ist und der Rest unverändert weiterarbeitet.

Erst jetzt laufen folgende Aktionen ab:

1. Die Systemdienste werden angehalten.
2. Alle veränderten Ressourcen (zum Beispiel Konfigurationsdateien) werden geschrieben.
3. Die Systemdienste werden (wieder) gestartet.

8.2.6 Erweiterte Profileinstellungen

Sie können für jedes Profil eine Beschreibung eingeben, die dann bei `scpm list` mit ausgegeben wird. Eingeben kann man diese Beschreibung für das gerade aktive Profil mit dem Kommando `scpm set description "text"`. Für nicht aktive Profile muss noch das Profil angegeben werden, also `scpm set description "text" work`

Manchmal kommt es vor, dass beim Umschalten in ein anderes Profil zusätzliche Aktionen ausgeführt werden sollen, die in SCPM (noch) nicht vorgesehen sind. Dafür können für jedes Profil vier ausführbare Programme oder Skripte eingehängt werden, die zu verschiedenen Zeitpunkten während das Umschaltens ausgeführt werden. Diese Zeitpunkte sind:

prestop vor dem Anhalten von Diensten beim Verlassen des Profils

poststop nach dem Anhalten von Diensten beim Verlassen des Profils

prestart vor dem Starten von Diensten beim Aktivieren des Profils

poststart nach dem Starten von Diensten beim Aktivieren des Profils

Das Umschalten von Profil `work` zu Profil `home` läuft dann folgendermaßen ab:

1. Prestop-Aktion des Profils `work` wird ausgeführt.
2. Anhalten von Diensten
3. Poststop-Aktion des Profils `work` wird ausgeführt.
4. Verändern der Systemkonfiguration
5. Prestart-Aktion des Profils `home` wird ausgeführt.
6. Starten von Diensten
7. Poststart-Aktion des Profils `home` wird ausgeführt.

Diese Aktionen werden auch mit dem `set` Kommando eingehängt, nämlich mit `scpm set prestop <dateiname>`, `scpm set poststop <dateiname>`, `scpm set prestart <dateiname>` oder `scpm set poststart <dateiname>`. Es muss sich dabei um ein ausführbares Programm handeln, das heisst Skripte müssen den richtigen Interpreter beinhalten und zumindest für den Superuser (`root`) ausführbar sein.

Achtung

Da diese Skripte oder Programme mit den Rechten des Superusers ausgeführt werden sollten sie nicht von beliebigen Anwendern änderbar sein. Da in Skripten durchaus vertrauliche Informationen enthalten sein können, ist es sogar angeraten, dass diese nur vom Superuser lesbar sind. Am besten versieht man diese Programme mit den Rechten `-rwx---- root root`.

```
chmod 700 <dateiname>
chown root.root <dateiname>
```

Achtung

Alle Zusatzeinstellungen, die mit `set` eingegeben wurden, lassen sich mit `get` abfragen. Zum Beispiel liefert `scpm get poststart` den Namen des Poststartprogramms oder einfach keine Information, wenn nichts eingehängt wurde. Gelöscht werden solche Einstellungen durch Überschreiben mit `" "`; das heisst der Aufruf von `scpm set prestop " "` hängt das Poststop-Programm wieder aus.

Genau wie beim Anlegen der Beschreibung können alle `set` und `get` Kommandos für ein beliebiges Profil angewandt werden. Dazu wird zuletzt noch der Name des Profils angegeben. Zum Beispiel `scpm get prestop <dateiname> work` oder `scpm get prestop work`.

8.2.7 Profilauswahl beim Booten

Es ist möglich, schon vor dem Booten ein Profil auszuwählen. Dazu muss lediglich der Bootparameter `PROFILE=<Name des Profils>` am Bootprompt eingegeben werden.

In der Bootloaderkonfiguration (`/boot/grub/menu.lst`) wird für die Option `title` ebenfalls der Name des Profils verwendet. Standardmäßig wird GRUB als Bootloader verwendet. Eine ausführlichere Beschreibung finden Sie Abschnitt 7.4 auf Seite 207; alternativ geben Sie `info grub` ein. Die Konfiguration von GRUB sieht dann zum Beispiel wie folgt aus:

Beispiel 8.1: Die Datei /boot/grub/menu.lst

```
gfxmenu (hd0,5)/boot/message
color white/green black/light-gray
default 0
timeout 8

title work
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 PROFILE=work
    initrd (hd0,5)/boot/initrd

title home
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 PROFILE=home
    initrd (hd0,5)/boot/initrd

title road
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6 PROFILE=road
    initrd (hd0,5)/boot/initrd
```

Für Systeme, die noch den Bootloader LILO verwenden, kann das Beispiel 8.2 benutzt werden.

Beispiel 8.2: Die Datei /etc/lilo.conf

```
boot      = /dev/hda
change-rules
reset
read-only
menu-scheme = Wg:kw:Wg:Wg
prompt
timeout = 80
message = /boot/message

    image = /boot/vmlinuz
    label = home
    root = /dev/hda6
    initrd = /boot/initrd
    append = "vga=0x317 hde=ide-scsi PROFILE=home"

    image = /boot/vmlinuz
    label = work
    root = /dev/hda6
    initrd = /boot/initrd
```

```

append = "vga=0x317 hde=ide-scsi PROFILE=work"

image  = /boot/vmlinuz
label  = road
root   = /dev/hda6
initrd = /boot/initrd
append = "vga=0x317 hde=ide-scsi PROFILE=road"

```

Jetzt kann beim Booten sehr einfach das gewünschte Profil ausgewählt werden.

8.2.8 Probleme und deren Lösung

Normalerweise sollte der Betrieb von SCPM reibungslos funktionieren. Es gibt aber einige Fallstricke, die hier beschrieben werden.

SCPM ist zum jetzigen Zeitpunkt noch nicht in der Lage, ein Systemupdate zu verwalten. Die Schwierigkeit liegt darin, dass bei einem Systemupdate die Daten, die in den Profilen gespeichert sind, von den verschiedenen Updatemechanismen nicht aktualisiert werden können. SCPM erkennt daher, falls ein Systemupdate gemacht wurde, und verweigert seinen Dienst. Sie sollten in dieser Situation eine Fehlermeldung von SCPM erhalten, die "Ihre Betriebssysteminstallation hat sich geändert oder ist unbekannt" enthält. In diesem Fall reinitialisieren Sie SCPM einfach mit `scpm -f enbale neu`. Die Profile sind dann allerdings verloren, d.h. Sie müssen sie neu einrichten.

Unter Umständen kann es auch vorkommen, dass SCPM während eines Switch-Vorgangs unvermittelt abbricht. Das kann entweder aufgrund äußerer Einwirkung eintreten – zum Beispiel Abbruch durch den Benutzer, Leerlaufen des Notebookakkus o.ä. – oder es kann ein Fehler in SCPM selbst sein. Dann werden Sie beim nächsten SCPM Aufruf die Meldung erhalten, dass SCPM gesperrt ist. Dies dient zum Schutz Ihres Systems, da die Daten, die SCPM in seiner Datenbank gespeichert hat, eventuell nicht zu dem Zustand Ihres System passen. Löschen Sie in diesem Fall die Lock-Datei mit `rm /var/lib/scpm/#LOCK` und stellen Sie mit `scpm -s reload` wieder einen konsistenten Status her. Anschließend können Sie wie gewohnt weiterarbeiten.

Noch ein Hinweis: Wenn Sie bei bereits initialisiertem SCPM die Resource Group Konfiguration ändern möchten, ist das prinzipiell kein Problem. Sie müssen nur darauf achten, daß Sie, nachdem Sie mit dem Hinzufügen

oder Entfernen von Gruppen fertig sind, `scpm rebuild` aufrufen. Dies fügt neue Ressourcen zu allen Profilen hinzu und löscht die entfernten. Letztere sind dann allerdings endgültig gelöscht, wenn Sie diese in den verschiedenen Profilen verschieden konfiguriert haben, verlieren Sie diese Konfigurationsdaten – bis auf die aktuelle Version in Ihrem System natürlich, diese wird von SCPM nicht angefasst. Falls Sie die Konfiguration mit YaST verändern, ist kein Rebuild-Aufruf nötig, dies erledigt dann YaST für Sie.

8.3 IrDA – Infrared Data Association

IrDA (engl. *Infrared Data Association*) ist ein Industriestandard für drahtlose Kommunikation über Infrarotlicht. Viele heute ausgelieferte Laptops sind mit einem IrDA-kompatiblen Sender/Empfänger ausgestattet, der die Kommunikation mit anderen Geräten, wie Druckern, Modems, LAN oder anderen Laptops ermöglicht. Die Übertragungsrate reicht von 2400 bps bis hin zu 4 Mbps.

Es gibt zwei Betriebsmodi für IrDA. Im Standardmodus SIR wird der Infrarotport über eine serielle Schnittstelle angesprochen. Dieser Modus funktioniert auf fast allen Geräten und genügt für viele Anforderungen. Der schnellere Modus FIR benötigt einen speziellen Treiber für den IrDA-Chip. Es gibt aber nicht für alle Chips solche Treiber. Außerdem muss der gewünschte Modus im BIOS-Setup des Computers eingestellt werden. Dort erfährt man meist auch, welche serielle Schnittstelle für den SIR-Modus verwendet wird.

Informationen zu IrDA finden Sie im IrDA-Howto von Werner Heuser unter <http://tuxmobil.org/Infrared-HOWTO/Infrared-HOWTO.html> und auf der Homepage des Linux IrDA Projekts: <http://irda.sourceforge.net/>.

8.3.1 Software

Die notwendigen Kernelmodule sind im Kernelpaket enthalten. Das Paket `irda` stellt die nötigen Hilfsprogramme zur Unterstützung der Infrarotschnittstelle bereit. Nach der Installation des Paketes findet man die Dokumentation unter `/usr/share/doc/packages/irda/README`.

8.3.2 Konfiguration

Der IrDA Systemdienst wird nicht automatisch beim Booten gestartet. Verwenden Sie das YaST Runlevel-Modul um die Einstellungen zu den Systemdiensten zu verändern. Alternativ kann auch das Programm `chkconfig` verwendet werden. Leider benötigt IrDA merklich mehr (Batterie-)Strom, da alle paar Sekunden ein Discovery-Paket verschickt wird, um andere Peripheriegeräte automatisch zu erkennen. Deshalb sollte man, wenn man auf Batteriestrom angewiesen ist, IrDA am besten nur bei Bedarf starten. Mit dem Kommando `rcirda start` können Sie die Schnittstelle jederzeit manuell aktivieren bzw. deaktivieren (mit dem Parameter `stop`). Beim Aktivieren der Schnittstelle werden die notwendigen Kernel-Module automatisch geladen.

In der Datei `/etc/sysconfig/irda` gibt es nur eine Variable `IRDA_PORT`. Dort können Sie einstellen, welche Schnittstelle im SIR-Modus verwendet wird; dies wird über das Skript `/etc/irda/drivers` beim Start der Infrarotunterstützung eingestellt.

8.3.3 Verwendung

Will man nun über Infrarot drucken, kann man dazu über die Gerätedatei `/dev/irlpt0` die Daten schicken. Die Gerätedatei `/dev/irlpt0` verhält sich wie die normale drahtgebundene Schnittstelle `/dev/lp0`, nur dass die Druckdaten drahtlos über infrarotes Licht verschickt werden.

Einen Drucker, der über die Infrarotschnittstelle betrieben wird, können Sie wie einen Drucker am Parallelport oder an der seriellen Schnittstelle einrichten. Beachten Sie bitte beim Drucken, dass sich der Drucker in Sichtweite der Infrarotschnittstelle des Computers befindet und dass die Infrarotunterstützung gestartet wird.

Will man über die Infrarotschnittstelle mit anderen Rechnern, mit Handys oder ähnlichen Geräten kommunizieren, so kann man dies über die Gerätedatei `/dev/ircomm0` erledigen. Mit dem Siemens S25 Handy beispielsweise kann man sich über das Programm `wvdial` mittels Infrarot drahtlos ins Internet einwählen. Auch ein Datenabgleich mit dem Palm Pilot ist so möglich, dazu muss im entsprechenden Programm als Gerät einfach `/dev/ircomm0` eingegeben werden.

Beachten Sie bitte auch, dass Sie ohne weiteres nur Geräte ansprechen können, die die Protokolle Printer oder IrCOMM unterstützen. Mit speziellen Programmen (`irobexpalm3`, `irobexreceive`, bitte beachten Sie hierzu die Beschreibung im IR-HOWTO) können Sie auch Geräte ansprechen, die das

IROBEX-Protokoll verwenden (3Com Palm Pilot). Die vom Gerät unterstützten Protokolle werden bei der Ausgabe von `irdadump` nach dem Gerätenamen in eckigen Klammern angegeben. Die Unterstützung des IrLAN-Protokolls ist „work in progress“ – es ist leider zur Zeit noch nicht stabil, wird aber sicher in naher Zukunft auch unter Linux zur Verfügung stehen.

8.3.4 Troubleshooting

Falls Geräte am Infrarotport nicht reagieren, können Sie als Benutzer `root` mit dem Kommando `irdadump` überprüfen, ob das andere Gerät vom Computer erkannt wird.

Bei einem Canon BJC-80 Drucker in Sichtweite des Computers erscheint dann eine Ausgabe ähnlich der folgenden in regelmäßiger Wiederholung (vgl. Ausgabe 8.3).

Beispiel 8.3: Ausgabe von irdadump

```
21:41:38.435239 xid:cmd 5b62bed5 > ffffffff S=6 s=0 (14)
21:41:38.525167 xid:cmd 5b62bed5 > ffffffff S=6 s=1 (14)
21:41:38.615159 xid:cmd 5b62bed5 > ffffffff S=6 s=2 (14)
21:41:38.705178 xid:cmd 5b62bed5 > ffffffff S=6 s=3 (14)
21:41:38.795198 xid:cmd 5b62bed5 > ffffffff S=6 s=4 (14)
21:41:38.885163 xid:cmd 5b62bed5 > ffffffff S=6 s=5 (14)
21:41:38.965133 xid:rsp 5b62bed5 < 6cac38dc S=6 s=5 BJC-80
                    hint=8804 [Printer IrCOMM ] (23)
21:41:38.975176 xid:cmd 5b62bed5 > ffffffff S=6 s=* erde
                    hint=0500 [ PnP Computer ] (21)
```

Sollte überhaupt keine Ausgabe erfolgen oder das andere Gerät sich nicht zurückmelden, so überprüfen Sie bitte die Konfiguration der Schnittstelle. Verwenden Sie überhaupt die richtige Schnittstelle? Manchmal ist die Infrarotschnittstelle auch unter `/dev/ttyS2` oder `/dev/ttyS3` zu finden oder ein anderer Interrupt als Interrupt 3 wird verwendet. Diese Einstellungen können Sie aber bei fast jedem Laptop im BIOS-Setup konfigurieren.

Mit einer einfachen Video-Kamera können Sie auch überprüfen, ob die Infrarot-LED überhaupt aufleuchtet – im Gegensatz zum menschlichen Auge können die meisten Videokameras Infrarotlicht sehen.

8.4 Bluetooth – Geräte drahtlos verbinden

Bei Bluetooth handelt es sich um eine Funktechnologie, die verschiedene Geräte miteinander verbindet. Bluetooth unterscheidet sich dabei in einigen wesentlichen Punkten von IrDA: Zum einen müssen die einzelnen Geräte sich nicht direkt „sehen“, zum anderen können mehrere Geräte zusammen ganze Netzwerke aufbauen. Allerdings sind nur Datenraten bis maximal 720 Kbps erreichbar (in der aktuellen Version 1.1). Theoretisch kann man mittels Bluetooth auch durch Wände hindurch „funken“. In der Praxis hängt dies aber sehr stark von den Wänden und der Geräteklasse ab. Letztere bestimmt die maximale Sendereichweite, die in drei Klassen von 10 bis 100 Metern reicht.

8.4.1 Profile

Dienste werden bei Bluetooth mittels sogenannter Profile definiert. Im Bluetooth-Standard sind z.B. Profile für den Dateitransfer („File Transfer“-Profile), Drucken („Basic Printing“-Profil) und Netzwerkverbindungen („Personal Area Network“-Profil) festgelegt.

Damit ein Gerät den Dienst eines anderen benutzen kann, müssen beide das gleiche Profil verstehen — eine Information, die manchmal leider weder der Verpackung noch dem Handbuch des Gerätes entnehmbar ist. Erschwerend kommt hinzu, dass sich zwar manche Hersteller streng an die Definitionen der einzelnen Profile halten, andere dagegen weniger. In der Regel klappt die Verständigung zwischen den Geräten aber.

8.4.2 Software

Um Bluetooth verwenden zu können braucht man einen Bluetooth-Adapter (entweder eingebaut im Gerät oder als externes Dongle), Treiber und einen sogenannten „Bluetooth Protocol Stack“.

Im Linuxkernel befindet sich bereits die Grundausstattung an Treibern für den Gebrauch von Bluetooth. Als „Protocol Stack“ kommt das Bluez-System zur Anwendung. Zusätzlich sollten noch alle mit Bluetooth in Verbindung stehenden Pakete (`bluez-libs`, `bluez-bluefw`, `bluez-pan`, `bluez-sdp` und `bluez-utils`) installiert werden, da diese einige benötigte Dienste und Dienstprogramme bereitstellen. Auf einige davon wird später noch eingegangen.

8.4.3 Konfiguration

Die nachstehend beschriebenen Konfigurationsdateien können nur als User `root` verändert werden. Eine grafische Benutzeroberfläche um die entsprechenden Parameter einzustellen, gibt es im Moment leider nicht. Die Dateien müssen also mit einem Textverarbeitungsprogramm verändert werden.

Einen ersten Schutz vor ungewollten Verbindungen bietet die Absicherung durch eine PIN-Nummer. Mobiltelefone fragen den PIN normalerweise beim ersten Kontakt (bzw. dem Einrichten eines Gerätekontaktes auf dem Telefon) ab. Damit sich zwei Geräte miteinander unterhalten können, müssen beide sich mit dem selben PIN identifizieren. Dieser befindet sich auf dem Rechner in der Datei `/etc/bluetooth/pin`. Momentan gibt es unter Linux nur einen PIN, unabhängig von der Anzahl der installierten Bluetoothgeräte. Das Ansprechen von mehreren Geräte mit unterschiedlichen PINs wird zur Zeit leider nicht unterstützt, hier müssen entweder alle Geräte auf die gleiche PIN-Nummer gesetzt werden, oder die PIN-Authentifizierung ganz deaktiviert werden.

Hinweis

Sicherheit von Bluetooth-Verbindungen

Trotz des PINs sollte man davon ausgehen, dass eine Übertragung zwischen zwei Geräten nicht abhörsicher ist!

Hinweis

Das Aktivieren geschieht in der Konfigurationsdatei `/etc/bluetooth/hcid.conf`. Hier kann man verschiedene Einstellungen wie Gerätenamen und Sicherheitsmodus ändern. Im Wesentlichen sollten die Einstellungen ausreichend sein, lediglich zwei sollen hier kurz erwähnt werden. Die Datei enthält Kommentare, die die Optionen bei den verschiedenen Einstellungen beschreiben.

Eine der wichtigsten Einstellungen ist `security auto`. Mit dieser wird die beschriebene Notwendigkeit eines PINs zur Identifikation aktiviert, wobei durch das `auto` im Problemfall auf keine PIN verwendet geschaltet wird. Ob Sie diese Einstellung auf `none` setzen, so dass nie eine PIN-Nummer verwendet wird, oder auf `user` (immer verwenden) setzen, bleibt Ihnen und Ihrem Bedarf an Sicherheit überlassen.

Interessant ist der Abschnitt, der mit `device {` eingeleitet wird. Hier kann man festlegen, unter welchem Namen der Rechner bei den Gegenstellen angezeigt wird. Die Geräteklasse (`Laptop`, `Server`, etc.) wird hier ebenso definiert wie Authentifizierung und Verschlüsselung.

8.4.4 Systemkomponenten und nützliche Hilfsmittel

Erst durch das Zusammenspiel verschiedener Dienste wird Bluetooth überhaupt benutzbar. Zwei im Hintergrund laufende Daemonen werden minimal benötigt: Zum einen der *hcid* (*Host Controller Interface*). Dieser dient als Schnittstelle zum Bluetoothgerät und steuert dieses. Zum anderen braucht man den *sdpd* (*Service Discovery Protocol*). Über den *sdpd* kann ein Gerät herausbekommen, welche Dienste der Rechner zur Verfügung stellt. Sowohl *hcid* als auch *sdpd* können — falls nicht bereits automatisch beim Systemstart geschehen — mit dem Kommando `rcbluetooth start` in Betrieb genommen werden. Es ist dazu allerdings nötig `root` zu sein.

Im Folgenden wird kurz auf die wichtigsten Werkzeuge eingegangen, die für das Arbeiten mit Bluetooth nötig sind. Leider sind im Moment nur Kommandozeilenprogramme verfügbar. Ob eine Erweiterung des Konqueror- (KDE-Desktop) bzw. des Nautilus- (GNOME-Desktop) Browsers bis zur Distributionserstellung fertig wird stand zum Redaktionsschluss noch nicht fest. Falls ja, so sollte Ihnen die URL `sdp://lokale` (d.h. physikalisch mit dem Rechner verbundene) und entfernte (d.h. nur mittels Funk erreichbaren) Bluetooth-Geräte anzeigen.

Hinweis

Alle erwähnten Programme verfügen über weitere Funktionalitäten, die sich mittels `man <programmname>` in Erfahrung bringen lassen.

Hinweis

Einige Kommandos lassen sich leider nur als `root` ausführen. Hierzu gehört z.B. `l2ping <Geräteadresse>`, mit dem die Verbindung zu einem entfernten Gerät getestet werden kann.

hcitool

Mittels des `hcitool` kann einfach festgestellt werden, ob lokale und/oder entfernte Geräte gefunden wurden. Der Kommandoaufruf `hcitool dev` sollte das eigene Gerät anzeigen. Die Ausgabe erzeugt für jedes gefundene lokale Gerät eine Zeile in der Form `<interfacename> <Geräteadresse>`.

Mit `hcitool name <Geräteadresse>` kann der Gerätenamen eines entfernten Gerätes ermittelt werden. Handelt es sich dabei z.B. um einen weiteren Rechner, so würde die ausgegebene Klasse und der Gerätenamen der Information aus dessen `/etc/bluetooth/hcid.conf` Datei entsprechen. Lokale Geräteadressen erzeugen eine Fehlerausgabe.

hciconfig

Weitere Informationen über das lokale Gerät erhält man mittels `/sbin/hciconfig`. Entfernte Geräte (die nicht physikalisch mit dem Rechner verbunden sind) werden mit `hcitool inq` gesucht. Hier werden drei Werte pro gefundenem Gerät ausgegeben: Die Geräteadresse, eine Uhrendifferenz und die Geräteklasse. Wichtig ist die Geräteadresse. Diese wird bei anderen Kommandos benutzt um das Zielgerät zu identifizieren. Die Uhrendifferenz ist im Prinzip nur aus technischer Sicht interessant. In der Klasse wird sowohl Gerätetyp als auch Servicetyp als Hexadezimalwert kodiert.

sdptool

Die Information, welcher Dienst von einem bestimmten Gerät zur Verfügung gestellt wird, erhält man durch das Programm `sdptool`. `sdptool browse <Geräteadresse>` liefert alle Dienste eines Gerätes, während man mit `sdptool search <Dienstekürzel>` nach einem bestimmten Dienst suchen kann. Dieser Aufruf befragt alle erreichbaren Geräte nach dem gewünschten Dienst. Wird er von einem der Geräte angeboten, so gibt das Programm den vom Gerät gelieferten (vollen) Dienstnamen und eine kurze Beschreibung dazu aus. Eine Liste aller möglichen Dienstkürzel erhält man durch Aufruf von `sdptool` ohne irgendwelche Parameter.

8.4.5 Beispiele

Um zu zeigen, was mit Bluetooth alles möglich ist, sind nachstehend zwei Beispiele aufgeführt.

Netzwerkverbindung zwischen zwei Rechnern R1 und R2

Im ersten Beispiel soll eine Netzwerkverbindung zwischen zwei Rechnern aufgebaut werden. Dies geschieht mit Hilfe des `pand` (*Personal Area Networking*). Die nachstehenden Kommandos müssen vom Benutzer `root` durchgeführt werden. Auf eine genauere Erläuterung der Netzwerkkommandos (`ip`) wird verzichtet und nur auf die Bluetooth bedingten Aktionen eingegangen:

Auf einem der beiden Rechner (im folgenden als *R1* bezeichnet) wird der `pand` mit dem Kommando `pand -s` gestartet. Auf dem zweiten Rechner *R2* wird mittels `hcitool inq` dessen Geräteadresse ermittelt. Mit `pand -c <Geräteadresse>` kann dann eine Verbindung aufgebaut werden. Ruft man jetzt eine Liste der zur Verfügung stehenden Netzwerkschnittstellen mit `ip link show` auf, so sollte ein Eintrag in der Form:

```
bnep0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000
    link/ether 00:12:34:56:89:90 brd ff:ff:ff:ff:ff:ff
```

zu finden sein (an Stelle von 00:12:34:56:89:90 sollte die lokale Geräteadresse stehen). Dieser Schnittstelle muss jetzt eine IP-Adresse zugewiesen werden, und sie in den aktiven Zustand gebracht werden.

Dies geschieht z.B. durch die beiden Kommandos (auf *R1*)

```
ip addr add 192.168.1.3/24 dev bnep0
ip link set bnep0 up
```

bzw. analog auf *R2*

```
ip addr add 192.168.1.4/24 dev bnep0
ip link set bnep0 up
```

Bitte beachten Sie, die 3 ist jetzt eine 4. *R1* ist damit von *R2* unter der IP 192.168.1.3 erreichbar. Mit `ssh 192.168.1.4` können Sie sich jetzt von *R1* aus einloggen (sofern *R2* einen `sshd`, wie er standardmässig unter SUSE LINUX läuft, im Betrieb hat). Der Aufruf `ssh 192.168.1.4` funktioniert im übrigen jetzt auch als „normaler“ Benutzer.

Datentransfer vom Mobiltelefon auf den Rechner

Im zweiten Beispiel soll ein mit einem Fotomobiltelefon erzeugtes Bild (ohne zusätzliche Kosten z.B. durch den Versand einer Multimediamail zu erzeugen) auf einen Rechner transportiert werden. Bitte beachten Sie, dass jedes Mobiltelefon eine andere Menüstruktur besitzt, aber die Vorgehensweise meist ähnlich ist. Konsultieren Sie nötigenfalls die Anleitung für Ihr Telefon. Nachstehend wird der Transfer eines Bildes von einem Sony Ericsson auf einen Laptop beschrieben. Dazu muss einerseits auf dem Rechner der Dienst Obex-Push vorhanden sein, andererseits der Rechner auch dem Mobiltelefon den Zugriff erlauben. Im ersten Schritt wird der Dienst auf dem Laptop zur Verfügung gestellt. Dies geschieht mit dem Daemon `opd`, der aus dem Paket `bluez-utils` kommt. Starten Sie diesen mit:

```
opd --mode OBEX --channel 10 --daemonize --path /tmp --sdp
```

Wichtig sind dabei zwei Parameter. `--sdp` meldet den Dienst beim `sdpd` an. Der Parameter `--path /tmp` teilt dem Programm mit, wohin es empfangene Daten speichern soll — in diesem Fall nach `/tmp/`. Genauso können Sie auch andere Pfade angeben. Sie brauchen nur Schreibberechtigung im angegebenen Verzeichnis.

Jetzt muss das Mobiltelefon den Rechner „kennenlernen“. Suchen Sie dazu das Menü ‘Verbindungen’ auf dem Telefon auf, und wählen Sie dort ‘Bluetooth’ an. Gehen Sie gegebenenfalls auf ‘Einschalten’, bevor Sie den Punkt ‘Eigene Geräte’ auswählen. Wählen Sie ‘Neues Gerät’ aus und lassen Sie Ihr Telefon nach dem Laptop suchen. Wenn ein Gerät gefunden wird, so erscheint es mit seinem Namen im Display. Wählen Sie das zum Laptop gehörende Gerät aus. Jetzt sollte eine PIN-Abfrage kommen, bei der Sie bitte den PIN aus `/etc/bluetooth/pin` eingeben. Damit kennt das Telefon jetzt den Laptop, und kann mit diesem auch Daten austauschen. Verlassen Sie jetzt das Menü und suchen Sie das Bildermenü auf. Wählen Sie ein Bild aus, dass Sie transferieren möchten und drücken Sie dann den ‘Mehr’-Button. Im jetzt erscheinenden Menü kommen Sie über ‘Senden’ zu einer Auswahl wie Sie es verschicken möchten. Wählen Sie ‘Über Bluetooth’ aus. Jetzt sollte der Laptop als Zielgerät selektierbar sein. Nach der Auswahl des Rechners erfolgt die Übertragung, und das Bild wird in das beim Aufruf des `opd` angegebene Verzeichnis gelegt. Genauso könnte Sie natürlich ein Musikstück auf den Laptop übertragen.

8.4.6 Troubleshooting

Bei Verbindungsproblemen empfiehlt es sich, folgende Liste abzuarbeiten:

- Überprüfen Sie die Ausgabe von `hcitool dev`. Wird das lokale Gerät angezeigt? Wenn nicht, ist entweder der `hcid` nicht gestartet, oder das Gerät wird nicht als Bluetooth-Gerät erkannt (entweder weil der Treiber dies nicht kann oder weil das Gerät kaputt ist). Starten Sie den Daemon mit `rcbluetooth restart` neu und werfen Sie einen Blick in `/var/log/messages`, ob irgendwelche Fehler aufgetreten sind.
- „Sieht“ der Rechner andere Geräte wenn Sie `hcitool inq` aufrufen? Probieren Sie das ruhig zweimal, evtl. war die Verbindung nicht ganz in Ordnung. Das Frequenzband für Bluetooth wird auch von anderen Geräten benutzt.
- Überprüfen Sie, ob die PIN in `/etc/bluetooth/pin` und die PIN des anderen Gerätes übereinstimmen.
- Versuchen Sie, die Verbindung vom anderen Gerät aus zu initiieren. Überprüfen Sie, ob dieses Gerät den Rechner sieht.
- Das erste Beispiel (Netzwerkverbindung) klappt nicht. Hier gibt es verschiedene Problemmöglichkeiten: Zum einen kann es sein, dass einer der beiden Rechner das `ssh`-Protokoll nicht versteht. Probieren

Sie, ob `ping 192.168.1.3` bzw. `ping 192.168.1.4` klappt. Wenn ja überprüfen Sie, ob der `sshd` läuft. Ein anderes Problem kann sein, dass Sie bereits andere Adressen haben, die mit den im Beispiel genannten `192.168.1.x` Konflikte erzeugen. Versuchen Sie einfach andere Adressen, z.B. `10.123.1.2` und `10.123.1.3`.

- Im zweiten Beispiel erscheint der Laptop nicht als Zielgerät: Erkennt das Mobilgerät den Dienst Obex-Push auf dem Laptop? Gehen Sie dazu im 'Eigene Geräte'-Menü zum betreffenden Gerät, und lassen Sie sich die 'Dienstliste' anzeigen. Steht hier (auch nach dem Aktualisieren der Liste) kein Obex-Push, so liegt das Problem am `opd` auf dem Laptop. Ist der `opd` gestartet? Haben Sie Schreibberechtigung auf das angegebene Verzeichnis?
- Geht das zweite Beispiel auch umgekehrt? Ja, wenn Sie `obexftp` installiert haben, geht dies mit `obexftp -b <Geräteadresse> -B 10 -p <bild>` auch bei einigen Geräten (Siemens und Sony Ericsson sind getestet, andere können, müssen aber nicht).

8.4.7 Weitere Informationen

Eine gute Übersicht über verschiedene Anleitungen zum Umgang und zur Konfiguration von Bluetooth findet sich unter: <http://www.holtmann.org/linux/bluetooth/>

Gute Informationen und Anleitungen:

- GPRS über Bluetooth (deutschsprachige Seite): http://www.van-schelve.de/edv-wissen/linux/bluetooth_1.htm
- Verbindung mit PalmOS PDA (englischsprachige Seite): <http://www.cs.ucl.ac.uk/staff/s.zachariadis/btpalmlinux.html>

Offizielles Howto für den im Kernel integrierten *Bluetooth Protocol Stack*: <http://bluez.sourceforge.net/howto/index.html>

Powermanagement

Dieses Kapitel bietet einen Überblick über die verschiedenen Powermanagement-Techniken unter Linux. Die Konfiguration aller einsetzbaren Techniken von APM (engl. *Advanced Power Management*) über ACPI (engl. *Advanced Configuration and Power Interface*) bis hin zum CPU Frequency Scaling werden hier detailliert beschrieben.

9.1	Stromsparfunktionen	256
9.2	APM	258
9.3	ACPI	261
9.4	Pause für die Festplatte	267
9.5	Das powersave Paket	268
9.6	Das YaST Powermanagement-Modul	275

Vom reinen Powermanagement auf Laptops mit APM ist die Entwicklung weitergegangen in Richtung ACPI, das ein auf allen modernen Rechnern (Laptops, Desktops und Servern) verfügbares Hardwareinformations- und -konfigurationswerkzeug darstellt. Auf vielen modernen Hardwaretypen kann ausserdem die CPU-Frequenz der Situation entsprechend angepasst werden, was gerade bei mobilen Geräten kostbare Akkulaufzeit einsparen hilft (*CPU Frequency Scaling*).

Alle Powermanagement-Techniken setzen eine dafür ausgelegte Hardware und passende BIOS-Routinen voraus. Die meisten Notebooks und viele moderne Desktops und Server bringen diese Voraussetzungen mit. Auf älterer Hardware wurde oft APM verwendet (engl. *Advanced Power Management*). Da APM im Wesentlichen aus einem im BIOS implementierten Satz von Funktionen besteht, ist die APM-Unterstützung auf unterschiedlicher Hardware unter Umständen unterschiedlich gut. ACPI ist wesentlich komplexer und variiert in der Unterstützung durch die Hardware noch stärker als APM. Aus diesem Grund macht es keinen Sinn, die Verwendung des einen oder anderen Systems zu propagieren. Testen Sie die unterschiedlichen Verfahren auf Ihrer Hardware und nutzen Sie die Technologie, die am besten unterstützt wird.

Hinweis

Powermanagement auf AMD64-Prozessoren

Die AMD64-Prozessoren unterstützen mit einem 64-bit Kernel ausschließlich ACPI.

Hinweis

9.1 Stromsparfunktionen

Viele dieser Funktionen sind von allgemeinem Interesse, aber so richtig wichtig sind diese vor allem im mobilen Einsatz. Im Folgenden werden diese Funktionen beschrieben und erklärt von welchem System diese ausgeführt werden können.

Stand-by In dieser Betriebsart wird nur das Display ausgeschaltet und bei manchen Geräten die Prozessorleistung gedrosselt. Nicht jede APM-Implementierung stellt diese Funktion zur Verfügung. Bei ACPI entspricht das dem Zustand *S1*.

Suspend (to memory) Hier wird der gesamte Systemzustand in den Arbeitsspeicher geschrieben und außer diesem das gesamte System schlafen gelegt. In diesem Zustand braucht der Computer nur sehr wenig Strom, sodass man damit je nach Gerät von zwölf Stunden bis mehrere Tage mit Batterie überbrücken kann. Der Vorteil dieses Zustands ist, dass man innerhalb weniger Sekunden wieder an derselben Stelle weiterarbeiten kann, ohne erst booten und benötigte Programme neu laden zu müssen. Bei den meisten modernen Geräten genügt es, den Deckel zu schließen, um zu suspendieren, und ihn zum Weiterarbeiten einfach wieder zu öffnen. Bei ACPI entspricht das dem Zustand S3. Die Unterstützung dieses Zustands ist stark hardwareabhängig.

Hibernation (Suspend to disk) In dieser Betriebsart hält es der Computer länger als einen Winter aus (Hibernation bedeutet „Winterschlaf“), denn der Systemzustand wird vollständig auf der Festplatte gespeichert und das System danach ausgeschaltet. Die Rückkehr aus dem „Winterschlaf“ dauert zwischen 30 und 90 Sekunden und auch hier wird der Zustand vor dem Suspend genau wiederhergestellt. Einige Hersteller bieten in ihrem APM sinnvolle Mischformen davon an (zum Beispiel RediSafe bei IBM Thinkpads). Hibernation entspricht bei ACPI dem Zustand S4.

Kontrolle des Akkuzustands Neben der reinen Information über den Ladezustand, ist es auch wichtig, dass etwas unternommen wird, wenn die Energiereserven knapp werden. Diese Kontrollfunktion übernehmen ACPI oder APM.

Automatisches Ausschalten Nach einem Shutdown wird der Computer vollständig ausgeschaltet. Das ist vor allem von Bedeutung, wenn ein automatischer Shutdown ausgeführt wird, kurz bevor der Akku leer ist.

Abschalten von Systemkomponenten

Die wesentliche Komponente, um Energie zu sparen, ist die Festplatte. Je nach Zuverlässigkeit des gesamten Systems kann diese mehr oder weniger lang schlafen gelegt werden. Allerdings steigt das Risiko eines Datenverlusts mit der Länge der Ruhepausen der Platte. Andere Komponenten können via ACPI (zumindest theoretisch) oder dauerhaft im BIOS-Setup deaktiviert werden.

Kontrolle der Prozessorleistung AMDs PowerNow! und Intels SpeedStep sind zwei Konzepte, die darauf ausgelegt sind, den Stromverbrauch des Gesamtsystems zu senken. Hierzu wird der Stromverbrauch der

stromhungrigsten Komponente, des Prozessors, gesenkt. Ein angenehmer Nebeneffekt der reduzierten Prozessorleistung ist eine geringere Wärmeentwicklung, sodass auch regelbare Lüfter leiser arbeiten können. Die *CPU Frequency Scaling*-Funktionen des Linux-Kernels steuern dies. Hierbei werden im Wesentlichen drei verschiedene Level der Prozessorleistung unterschieden:

performance höchstmögliche Leistung des Prozessors — sinnvoll, wenn das System am Stromnetz betrieben wird.

powersave geringstmögliche Prozessorleistung für den Akkubetrieb

dynamic dynamische Anpassung der Prozessorleistung an die aktuelle Prozessorauslastung — bevorzugte Einstellung sowohl im Akkubetrieb als auch am Netz, um die Batterie zu schonen, Lärm zu vermeiden und bestmögliche Performance zu erreichen. Die Umschaltung zwischen den Frequenzen/Zuständen erfolgt in der Regel so nahtlos, dass der Benutzer im normalen Betrieb nichts davon mitbekommt.

Mehr Informationen zur Kontrolle der Prozessorleistung in Abschnitt 9.5 auf Seite 268.

9.2 APM

Einige der Stromsparfunktionen führt das APM-BIOS selbstständig aus. Stand-by und Suspend kann man auf vielen Notebooks mit Tastenkombinationen oder mit Schließen des Deckels aktivieren. Dazu ist erstmal keinerlei Funktion seitens des Betriebssystems nötig. Wer diese Betriebsarten jedoch per Kommando einleiten möchte, ist darauf angewiesen, dass vor dem Suspend noch bestimmte Aktionen ausgeführt werden. Wer ausserdem einfach nur den Ladezustand der Batterie angezeigt bekommen möchte, muss entsprechende Pakete und einen geeigneten Kernel installiert haben.

Bei den fertigen Kernels von SUSE LINUX ist der APM-Support fest eingebaut, wird aber nur aktiviert, falls kein ACPI im BIOS implementiert ist und ein APM-BIOS gefunden wird. Um den APM-Support einzuschalten, muss ACPI am Bootprompt mit `acpi=off` ausgeschaltet werden. Ob APM aktiviert wurde, lässt sich leicht mit dem Kommando `cat /proc/apm` nachprüfen. Wenn hier eine Zeile mit diversen Zahlen erscheint, ist alles in Ordnung. Jetzt sollte ein `shutdown -h` zum Ausschalten des Computers führen.

Da manche BIOS-Implementierungen sich nicht exakt an Standards halten, kommt es manchmal zu merkwürdigem Verhalten. Manche Probleme kann man mit speziellen Bootparametern umgehen (früher waren dies Kernel-konfigurationsoptionen). Alle Parameter werden am Bootprompt in der Form `apm=<parameter>` eingegeben:

on/off APM Support ein- oder ausschalten

(no-)allow-ints Während des Ausführens von BIOS-Funktionen Interrupts zulassen.

(no-)broken-psr BIOS hat eine nicht ordnungsgemäß funktionierende „GetPowerStatus“-Funktion.

(no-)realmode-power-off Den Prozessor vor dem Shutdown in den Real Mode zurückschalten.

(no-)debug APM Ereignisse im Syslog protokollieren.

(no-)power-off Nach dem Shutdown das System ausschalten.

bounce-interval=<n> Zeit in 1/100 Sekunden, in der nach einem Suspendereignis weitere Suspendereignisse ignoriert werden.

idle-threshold=<n> Prozentsatz der Systeminaktivität, ab der die BIOS-Funktion `idle` aufgerufen wird (0=immer, 100=nie).

idle-period=<n> Zeitraum in 1/100 Sekunden, über dem die System(in)aktivität ermittelt wird.

9.2.1 Der APM-Daemon (apmd)

Der Daemon `apmd` dient zur Überwachung der Batterie und kann bestimmte Aktionen auslösen, wenn ein Stand-by- oder Suspend-Ereignis eintritt. Er befindet sich im Paket `apmd`. Er ist nicht unbedingt zum Betrieb notwendig, kann jedoch bei manchen Problemen recht nützlich sein.

Der `apmd` wird nicht automatisch beim Booten gestartet. Ist dies jedoch erforderlich, kann man mit dem YaST Runlevel-Editor die Einstellungen zu den Systemdiensten verändern. Alternativ kann auch das Programm `chkconfig` verwendet werden. Manuell kann er mit dem Kommando `rcapmd start` gestartet werden.

Zur Konfiguration gibt es in `/etc/sysconfig/powermanagement` einige Variablen. Die Datei ist mit Kommentaren versehen, deshalb werden hier nur einige Hinweise gegeben.

APMD_ADJUST_DISK_PERF Damit wird veranlasst, dass das Verhalten der Festplatte an den Zustand der Stromversorgung angepasst wird. Dazu gibt es eine Reihe weiterer Variablen, die entweder mit **APMD_BATTERY** oder **APMD_AC** beginnen. Die ersteren enthalten die Einstellungen für den Batteriebetrieb, die letzteren die für den Betrieb mit externer Stromversorgung.

APMD_BATTERY/AC_DISK_TIMEOUT

Die Zeit von Platteninaktivität, nach der diese angehalten wird. Die Werte sind im Abschnitt 9.4 auf Seite 267 oder in der Manualpage zu **hdparm**, Option **-S**, beschrieben.

APMD_BATTERY/AC_KUPDATED_INTERVAL

Die Zeit zwischen zwei Läufen des Kernel Update Deamons.

APMD_BATTERY/AC_DATA_TIMEOUT

Das maximale Alter gepufferter Daten.

APMD_BATTERY/AC_FILL_LEVEL

Der maximale Füllstand des Festplattenpuffers.

APMD_PCMCIA_EJECT_ON_SUSPEND

Obwohl PCMCIA mit APM-Unterstützung übersetzt ist, gibt es hier manchmal Schwierigkeiten. Einige der Kartentreiber kehren von einem Suspend nicht ordentlich zurück **xirc2ps_cs**. Deshalb kann der **apmd** das PCMCIA-System vor dem Suspend deaktivieren und danach wieder aktivieren. Dazu wird die Variable **APMD_PCMCIA_EJECT_ON_SUSPEND** auf **yes** gesetzt.

APMD_INTERFACES_TO_STOP Hier können Netzwerk-Interfaces eingetragen werden, die vor dem Suspendieren angehalten und danach wieder gestartet werden sollen.

APMD_INTERFACES_TO_UNLOAD

Wenn außerdem noch die Treibermodule dieser Interfaces entladen werden müssen, ist diese Variable zu verwenden.

APMD_TURN_OFF_IDEDMA_BEFORE_SUSPEND

Manchmal kommt es auch vor, dass das Wiederaufwachen nach einem Suspend nicht funktioniert, wenn ein IDE-Gerät (Festplatte) noch im DMA-Modus ist.

Es gibt noch weitere Möglichkeiten, wie zum Beispiel Tastaturwiederholrate oder die Uhrzeit nach einem Suspend zu korrigieren oder den Laptop automatisch herunterzufahren, wenn das APM-BIOS ein „Batterie

kritisch“-Ereignis sendet. Wer noch speziellere Aktionen ausführen möchte, der kann das Skript `/usr/sbin/apmd_proxy`, das die oben aufgeführten Jobs ausführt, an seine Bedürfnisse anpassen.

9.2.2 Weitere Befehle

Im `apmd` sind noch einige nützliche Programme enthalten. Mit `apm` kann die aktuelle Batteriekapazität abgefragt werden und das System in Stand-by (`apm -s`) oder Suspend (`apm -s`) geschickt werden; vgl. die Manualpage von `apm`. Das Kommando `apmsleep` suspendiert das System für eine vorgegebene Zeit. Wer eine Logdatei beobachten möchte, ohne die Festplatte ständig am Laufen zu halten, der kann `tailf` als Ersatz für `tail -f` verwenden.

Natürlich gibt es auch hier Tools für das X Window System. Ebenfalls im `apmd` findet man `xqpm`, was den Ladezustand der Batterie grafisch anzeigt. Wer den KDE-Desktop verwendet – oder zumindest `kpanel` –, kann sich auch von `kbatmon` den Ladestand des Akkus anzeigen lassen und das System suspendieren. Als Alternative ist auch `xosview` interessant.

9.3 ACPI

ACPI steht für *Advanced Configuration and Power Interface* und soll dem Betriebssystem ermöglichen, die einzelnen Hardwarekomponenten individuell einzurichten und zu steuern. Damit ersetzt ACPI sowohl „Plug and Play“, als auch APM. Weiterhin stellt ACPI noch diverse Informationen über Batterie, Netzteil, Temperatur und Lüfter zur Verfügung und unterrichtet über Systemereignisse, wie zum Beispiel „Deckel schließen“ oder „Batterieladung niedrig“.

Das BIOS stellt Tabellen zur Verfügung, in denen Informationen über die Einzelkomponenten und Methoden für den Zugriff auf die Hardware enthalten sind. Diese Informationen werden vom Betriebssystem verwendet, um zum Beispiel Interrupts zuzuweisen oder Komponenten bedarfsweise an- und abzuschalten. Da das Betriebssystem allerdings Anweisungen ausführt, die im BIOS abgelegt sind, ist man auch hier wieder von der Implementierung des BIOS abhängig. In `/var/log/boot.msg` findet man die Bootmeldungen. Dort meldet ACPI, welche Tabellen es gefunden hat und erfolgreich auslesen konnte. Mehr Information zum Troubleshooting bei ACPI-Problemen lesen Sie unter Abschnitt 9.3.1 auf Seite 265.

9.3.1 Praxis

Wenn der Kernel beim Booten ein ACPI-BIOS erkennt, wird ACPI automatisch aktiviert (und APM deaktiviert). Der Bootparameter `acpi=on` kann höchstens bei älteren Maschinen notwendig sein. Natürlich muss der Computer ACPI 2.0 oder neuer unterstützen. Ob ACPI aktiviert wurde, kann in den Bootmeldungen des Kernels in `/var/log/boot.msg` nachgesehen werden. Es gibt dann auch ein Verzeichnis `/proc/acpi/`, welches im weiteren Verlauf beschrieben wird.

Danach müssen jedoch noch eine Reihe von Modulen geladen werden. Diese werden vom Startskript des ACPI-Daemons geladen. Wenn eines dieser Module Probleme bereitet, kann es in `/etc/sysconfig/powersave/common` vom Laden bzw. Entladen ausgeschlossen werden. Im Systemlog (`/var/log/messages`) findet man die Meldungen der Module und kann sehen, welche Komponenten erkannt wurden.

Jetzt findet man unter `/proc/acpi/` eine Reihe von Dateien, die über den Systemzustand informieren oder mit deren Hilfe man einige Zustände aktiv verändern kann. Allerdings funktioniert hier noch längst nicht alles, weil es sich noch in der Entwicklung befindet, und von der Implementierung des Herstellers abhängt.

Alle Dateien (außer `dsdt` und `fadt`) können mit `cat` gelesen werden. In einigen kann man Einstellungen verändern, indem man mit `echo X <datei>` geeignete Werte für X übergibt (alles unter `/proc` ist nicht wirklich eine Datei auf der Festplatte sondern, eine Schnittstelle zum Kernel). Im Folgenden werden die wichtigsten Dateien beschrieben:

`/proc/acpi/info` Allgemeine Information über ACPI

`/proc/acpi/alarm` Hier lässt sich einstellen, wann das System aus einem Schlafzustand zurückkehrt. Momentan ist dieses Feature noch nicht hinreichend unterstützt.

`/proc/acpi/sleep` Gibt Auskunft über die möglichen Schlafzustände.

`/proc/acpi/event` Hier werden alle Ereignisse gemeldet. Diese werden von einem Daemon wie `acpid` oder `powersaved` verarbeitet. Wenn kein Daemon darauf zugreift, kann man die Ereignisse mit `cat /proc/acpi/event` lesen (Mit `(Strg) + (C)` beenden). Ein kurzer Druck auf die Powertaste oder das Schließen des Deckels sind solche Ereignisse.

/proc/acpi/dsdt und /proc/acpi/fadt

Hierin stehen die ACPI-Tabellen DSDT (*Differentiated System Description Table*) und FADT (*Fixed ACPI Description Table*). Diese können mit `acpidmp`, `acpidisasm` und `dmdecode` ausgelesen werden. Diese Programme einschließlich Dokumentation finden Sie im Paket `pmtools`. Beispiel: `acpidmp DSDT | acpidisasm`.

/proc/acpi/ac_adapter/AC/state

Ist das Netzteil angeschlossen?

/proc/acpi/battery/BAT*/{alarm,info,state}

Ausführliche Information über den Zustand der Batterien. Um den Füllstand ablesen zu können, muss `last full capacity` aus `info` mit `remaining capacity` aus `state` verglichen werden. Komfortabler geht das mit speziellen Programmen, die nachher beschrieben werden. In `alarm` kann die Kapazität eingegeben werden, bei der ein Batterieereignis ausgelöst wird.

/proc/acpi/button In diesem Verzeichnis gibt es Informationen über diverse Schalter.

/proc/acpi/fan/FAN/state Dies zeigt an, ob der Lüfter gerade läuft. Er kann auch manuell ein- und ausgeschaltet werden, indem man 0 (=ein) bzw. 3 (=aus) in diese Datei schreibt. Es ist jedoch zu beachten, dass sowohl der ACPI-Code im Kernel als auch die Hardware (bzw. das BIOS) diese Einstellung überschreiben, wenn es zu warm wird.

/proc/acpi/processor/CPU*/info

Informationen über die Energiesparmöglichkeiten des Prozessors.

/proc/acpi/processor/CPU*/power

Information über den gegenwärtigen Prozessorzustand. Ein Sternchen bei 'C2' bedeutet Leerlauf; das ist der häufigste Zustand, wie an der Zahl `usage` zu Erkennen ist.

/proc/acpi/processor/CPU*/performance

Diese Schnittstelle wird nicht mehr verwendet.

/proc/acpi/processor/CPU*/throttling

Hier ist eine weitere lineare Drosselung des Prozessors möglich. Diese Schnittstelle ist veraltet. Ihre Funktion haben die Einstellungen unter `/etc/sysconfig/powersave/common` übernommen (siehe Abschnitt 9.5.2 auf Seite 271).

/proc/acpi/processor/CPU*/limit

Wenn Performance und Throttling von einem Daemon automatisch geregelt werden, lassen sich hier die Grenzen angeben, die nicht überschritten werden dürfen. Es gibt vom System festgelegte Limits und solche, die vom Benutzer einstellbar sind. Ihre Funktion haben die Einstellungen unter `/etc/sysconfig/powersave/common` übernommen (siehe Abschnitt 9.5.2 auf Seite 271).

/proc/acpi/thermal_zone/ Hier gibt es für jede Thermalzone ein Unterverzeichnis. Eine Thermalzone ist ein Bereich mit ähnlichen thermischen Eigenschaften, deren Anzahl und Namen vom Hardware-Hersteller gewählt werden. Viele der Möglichkeiten, die ACPI bietet, werden jedoch nur selten implementiert. Stattdessen, wird die Temperatursteuerung auf herkömmliche Weise direkt vom BIOS übernommen, ohne dem Betriebssystem ein wesentliches Mitspracherecht einzuräumen, denn es geht um nicht weniger als die Lebensdauer der Hardware. Die folgenden Beschreibungen sind also teilweise theoretischer Natur.

/proc/acpi/thermal_zone/*/temperature

Die aktuelle Temperatur der Thermalzone.

/proc/acpi/thermal_zone/*/state

Der Status sagt aus, ob alles „ok“ ist oder ob ACPI „aktiv“ oder „passiv“ kühlt. Bei ACPI-unabhängiger Lüftersteuerung ist hier immer alles „ok“.

/proc/acpi/thermal_zone/*/cooling_mode

Hier kann man unter voller ACPI-Kontrolle die bevorzugte Kühlmethode wählen. Entweder passiv (weniger Leistung, aber sparsam) oder aktiv (immer volle Leistung und voller Lüfterlärm).

/proc/acpi/thermal_zone/*/trip_points

Hier kann eingestellt werden, ab welcher Temperatur etwas unternommen werden soll. Das reicht von passiver oder aktiver Kühlung über Suspendierung („hot“) bis zum Abschalten des Computers („critical“).

/proc/acpi/thermal_zone/*/polling_frequency

Wenn der Wert in `temperature` nicht automatisch aktualisiert wird, sobald sich die Temperatur ändert, kann hier auf den „Polling Modus“ umgeschaltet werden. Der Befehl `echo X > /proc/acpi/thermal_zone/*/polling_frequency` bewirkt, dass die Temperatur alle X Sekunden abgefragt wird. Mit `X=0` wird das „Polling“ wieder ausgeschaltet.

Der ACPI-Daemon (acpid)

Ähnlich wie der APM Daemon verarbeitet der ACPI Daemon bestimmte ACPI Ereignisse. Diese sind zur Zeit lediglich die Bestätigung bestimmter Schalter wie der Ein/Aus-Schalter oder der Deckelkontakt. Alle Ereignisse werden im Systemlog protokolliert. In `/etc/sysconfig/powermanagement` kann in den Variablen `ACPI_BUTTON_POWER` und `ACPI_BUTTON_LID` festgelegt werden, was bei diesen Ereignissen geschehen soll. Wem das nicht genügt, der kann das Skript `/usr/sbin/acpid_proxy` anpassen oder die Konfiguration des `acpid` unter `/etc/acpi/` verändern.

Im Gegensatz zum `apmd` ist hier nicht sehr viel vorkonfiguriert, da sich ACPI unter Linux noch stark entwickelt. Bei Bedarf muss man sich den `acpid` selbst zurecht konfigurieren. Für Vorschläge zu vorbereiteten Aktionen sind wir jederzeit unter <http://www.suse.de/feedback> zu erreichen.

Weitere Tools

Es gibt eine Reihe von mehr oder weniger umfangreichen ACPI-Werkzeugen. Darunter reine Informationstools, die Batteriezustand, Temperatur usw. anzeigen (`acpi`, `klaptopdaemon`, `wmacpimon` etc.). Andere vereinfachen den Zugriff auf die Strukturen unter `/proc/acpi` oder helfen Veränderungen zu beobachten (`akpi`, `acpiw`, `gtkacpiw`). Des weiteren gibt es noch Werkzeuge zum Bearbeiten der ACPI Tabellen im BIOS (Paket `pmttools`).

Mögliche Probleme und Lösungen

Es gibt zwei unterschiedliche Gruppen von Problemen. Einerseits können natürlich Fehler im ACPI-Code des Kernels enthalten sein, die nicht rechtzeitig bemerkt wurden. Dann wird es jedoch eine Lösung zum Download geben. Unangenehmer und leider auch häufiger sind Probleme im BIOS eines Computers. Es kommt leider sogar vor, dass Abweichungen von der ACPI-Spezifikation im BIOS eingebaut werden, um Fehler der ACPI-Implementierung in anderen sehr verbreiteten Betriebssystemen zu umgehen. Es gibt auch Hardware, bei der gravierende Fehler in der ACPI-Implementierung bekannt sind und die deshalb in einer Blacklist vermerkt sind, damit der Linuxkernel ACPI dort nicht verwendet.

Falls ein Problem auftritt, sollte man also zunächst das BIOS aktualisieren. Viele Probleme lösen sich dabei einfach in Luft auf. Falls der Rechner überhaupt nicht vernünftig bootet, hilft eventuell einer der folgenden Bootparameter:

pci=noacpi Kein ACPI zur Konfiguration der PCI-Geräte verwenden.

acpi=oldboot Nur einfache Ressourcenkonfiguration durchführen, sonst ACPI nicht verwenden.

acpi=off Kein ACPI verwenden.

Achtung

Probleme beim Booten ohne ACPI

Manche Rechner der neueren Generation, insbesondere SMP-Systeme und AMD64-Systeme benötigen ACPI für eine korrekte Hardwarekonfiguration. Ein Abschalten von ACPI kann zu Problemen führen.

Achtung

Wichtig ist dann jedenfalls, sich die Bootmeldungen genauer anzusehen. Am besten verwendet man dafür nach dem Booten das Kommando `dmesg | grep -2i acpi` (oder auch alle Meldungen, denn das Problem muss ja nicht an ACPI hängen). Wenn ein Fehler beim Parsen einer ACPI Tabelle auftritt, gibt es zumindest für die wichtigste Tabelle, die DSDT, die Möglichkeit, dem System eine verbesserte Version unterzuschieben. Dann wird die fehlerhafte DSDT des BIOS ignoriert. Das Vorgehen wird unter Abschnitt 9.5.4 auf Seite 272 näher beschrieben.

Es gibt bei der Kernelkonfiguration auch einen Schalter, um Debug-Meldungen von ACPI zu aktivieren. Wenn man sich einen Kernel mit ACPI Debugging kompiliert und installiert hat, kann man Experten, die einen Fehler suchen, mit detaillierter Information unterstützen.

Auf alle Fälle ist es bei BIOS- oder Hardwareproblemen immer eine gute Idee, sich an den Hersteller des Gerätes zu wenden. Auch wenn diese einem bei Linux nicht immer weiterhelfen können, ist es jedoch von Bedeutung, dass diese den Begriff Linux so häufig als möglich hören. Erst wenn die Hersteller merken, dass genug ihrer Kunden Linux verwenden, werden sie es ernst nehmen. Es schadet auch nichts, wenn Sie ohne Probleme dem Hersteller Ihrer Hardware erzählen, dass Sie Linux darauf verwenden.

Weitere Dokumentation und Hilfe finden Sie unter:

- c't 2002, Heft 25: Schöne neue Welt (Dominik Brodowski, Oliver Dierich)
- <http://www.cpqlinux.com/acpi-howto.html> (etwas genaueres ACPI HowTo, enthält Patches der DSDT)

- <http://www.intel.com/technology/iapc/acpi/faq.htm> (ACPI FAQ @Intel)
- <http://acpi.sourceforge.net/> (Das ACPI4Linux-Projekt bei Sourceforge)
- <http://www.poupinou.org/acpi/> (DSDT Patches von Bruno Ducrot)

9.4 Pause für die Festplatte

Man kann unter Linux die Festplatte abschalten, wenn sie nicht benötigt wird. Dazu dient das Programm `hdparm`, mit dem man diverse Einstellungen an den Festplatten vornehmen kann. Mit der Option `-y` wird die Platte sofort in den Stand-by-Modus geschickt, mit `-Y` (Vorsicht!) wird sie vollständig abgeschaltet. Mit `hdparm -S <x>` wird erreicht, dass die Platte nach einer bestimmten Zeit Inaktivität abgeschaltet wird. Der Platzhalter `<x>` hat folgende Bedeutung: 0 schaltet diesen Mechanismus aus, die Platte läuft immer. Werte von 1 bis 240 werden mit 5 Sekunden multipliziert. 241 bis 251 entsprechen 1 bis 11 mal 30 Minuten.

Häufig ist es aber nicht ganz so einfach, die Festplatte in den Ruhezustand zu versetzen. Unter Linux gibt es eine Vielzahl von Prozessen, die durch Schreibvorgänge die Platte immer wieder aufwecken. Deshalb ist es an dieser Stelle wichtig zu verstehen, wie Linux mit Daten umgeht, die auf die Platte geschrieben werden sollen. Alle Daten werden zuerst in einen Puffer im Arbeitsspeicher zwischengespeichert. Dieser Puffer wird vom „Kernel Update Daemon“ (`kupdated`) überwacht. Immer wenn Daten ein bestimmtes Alter erreichen oder wenn der Puffer bis zu einem gewissen Grad gefüllt ist, wird der Puffer geleert und die Daten der Festplatte übergeben. Die Größe des Puffers ist übrigens dynamisch und hängt von der Speichergröße und der Systemauslastung ab. Da das vorrangige Ziel Datensicherheit ist, wird der `kupdated` standardmäßig auf kleine Zeitintervalle eingestellt. Er prüft den Puffer alle 5 Sekunden und benachrichtigt den `bdflush`-Daemon, wenn Daten älter als 30 Sekunden sind oder der Puffer zu 30% gefüllt ist. Der `bdflush`-Daemon schreibt dann die Daten auf die Platte. Er schreibt auch unabhängig vom `kupdated` wenn zum Beispiel der Puffer voll ist. Wer ein stabiles System hat, kann diese Einstellungen nun verändern. Man muss sich jedoch immer darüber im Klaren sein, dass dies auf Kosten der Datensicherheit geht.

Achtung

Beeinträchtigung der Datensicherheit

Änderungen an den Einstellungen des Kernel Update Daemon beeinflussen die Datensicherheit. Wer sich unsicher ist, lässt lieber die Finger davon.

Achtung

Die Einstellungen für den Festplattentimeout, den `kupdated`-Intervall, den Füllgrad des Puffers und das Alter der Daten können in `/etc/sysconfig/powermanagement` zweifach abgelegt werden: einmal für den Batteriebetrieb und einmal für den Betrieb mit externer Stromversorgung. Die Variablen sind im Abschnitt über den `qpm` 9.2.1 auf Seite 259 und in der Datei selbst beschrieben. Weiterhin finden Sie einige Information zum Thema unter `/usr/share/doc/packages/powersave`.

Neben all diesen Vorgängen schreiben so genannte „Journaling Dateisysteme“ wie zum Beispiel ReiserFS oder Ext3 unabhängig von `bdflush` ihre Metadaten auf die Festplatte, was natürlich auch ein Einschlafen der Platte verhindert. Um das zu vermeiden, gibt es jetzt eine Erweiterung im Kernel, die speziell für mobile Geräte entwickelt wurde. Die genaue Beschreibung dazu findet man in `/usr/src/linux/Documentation/laptop-mode.txt`.

Weiterhin ist natürlich zu beachten, wie sich die Programme verhalten, die man gerade verwendet. Zum Beispiel schreiben gute Texteditoren regelmäßig versteckte Sicherungen der gerade geänderten Datei auf die Platte. Das weckt dann die Platte immer wieder auf. Solche Eigenschaften von Programmen können auch abgeschaltet werden, aber auch hier wieder auf Kosten der Datensicherheit.

In diesem Zusammenhang gibt es für den Maildaemon `postfix` eine Variable `POSTFIX_LAPTOP`. Wenn diese auf `yes` gesetzt wird, greift `postfix` wesentlich seltener auf die Festplatte zu. Das ist jedoch nicht von Bedeutung, wenn das Intervall für den `kupdated` verlängert wurde.

9.5 Das powersave Paket

Das `powersave` Paket ist hauptsächlich für die Anwendung in Laptops ausgelegt, wo es für die Stromsparfunktion beim Batteriebetrieb zuständig ist. Manche seiner Features sind aber auch für normale Arbeitsplatzrechner

und Server interessant (z.B. Suspend/Standby, ACPI-Button-Funktionalität und Abstellen von IDE-Festplatten).

In diesem Paket sind alle Powermanagementfunktionen Ihres Rechners zusammengefasst. Es unterstützt Hardware, die ACPI, APM, IDE-Platten und PowerNow!- bzw. SpeedStep-Technologien nutzt. Die Funktionalitäten aus den Paketen `apmd`, `acpid`, `ospm` und `cpufreqd` (mittlerweile `cpuspeed`) wird im Paket `powersave` zusammengefasst. Aus diesem Grund sollten Daemons aus diesen Paketen nicht parallel zum `powersave`-Daemon betrieben werden.

Selbst wenn Ihr System nicht alle der oben genannten Hardwareelemente enthält (APM und ACPI schließen sich gegenseitig aus), sollten Sie den `powersave` Daemon zur Regelung der Stromsparfunktion nutzen. Eventuelle Änderungen der Hardwarekonfiguration erkennt der Daemon automatisch.

Hinweis

Informationen zu powersave

Neben diesem Kapitel sind aktuelle Informationen zum `powersave` Paket auch unter `/usr/share/doc/packages/powersave/README_POWERSAVE` verfügbar.

Hinweis

9.5.1 Konfiguration des powersave Pakets

Generell ist die Konfiguration von `powersave` über mehrere Dateien verteilt:

/etc/powersave.conf Diese Datei wird vom `powersave`-Daemon benötigt, um die Bearbeitung auftretender Systemereignisse (*Events*) an den `powersave_proxy` zu delegieren. Darüber hinaus werden hier benutzerdefinierte Einstellungen zum genauen Verhalten des Daemons vorgenommen.

/etc/sysconfig/powersave/common

Diese Datei dient der allgemeinen Konfiguration des Startupskripts (`rcpowersave`) und des Proxies. Die Voreinstellungen können meistens unverändert übernommen werden.

/etc/sysconfig/powersave/scheme_*

Dies sind die verschiedenen Schemes oder auch Profile, die die Anpassung des Stromverbrauchs an bestimmte Einsatzszenarien

regeln. Einige sind vorkonfiguriert und ohne weitere Änderungen einsatzbereit. Sie können allerdings auch eigene Profile hier ablegen.

9.5.2 Konfiguration von APM und ACPI

Suspend und Standby

In der Datei `/etc/sysconfig/powersave/common` legen Sie fest, welche kritischen Module und Dienste vor einem Suspend- oder Standby-Ereignis entladen bzw. gestoppt werden sollen. Wird das System später wieder hochgefahren, werden diese wieder geladen bzw. gestartet. Die Voreinstellungen betreffen in der Hauptsache USB- und PCMCIA-Module.

POWERSAVE_SUSPEND_RESTART_SERVICES=""

Listen Sie hier die nach einem Suspend neu zu startenden Dienste auf.

POWERSAVE_STANDBY_RESTART_SERVICES=""

Listen Sie hier die nach einem Standby neu zu startenden Dienste auf.

POWERSAVE_UNLOAD_MODULES_BEFORE_SUSPEND=""

Listen Sie die vor einem Suspend zu entladenden Module auf.

POWERSAVE_UNLOAD_MODULES_BEFORE_STANDBY=""

Listen Sie die vor einem Standby zu entladenden Module auf.

Ausserdem stellen Sie sicher, dass die folgenden Standardoptionen zur korrekten Verarbeitung von Suspend/Standby, Occurrence/Resume gesetzt sind (dies sind normalerweise die Voreinstellungen nach der Installation von SUSE LINUX):

```
POWERSAVE_EVENT_GLOBAL_SUSPEND="prepare_suspend"
POWERSAVE_EVENT_GLOBAL_STANDBY="prepare_standby"
POWERSAVE_EVENT_GLOBAL_RESUME_SUSPEND="restore_after_suspend"
POWERSAVE_EVENT_GLOBAL_RESUME_STANDBY="restore_after_standby"
```

In der Konfigurationsdatei des powersave Daemons unter `/etc/powersave.conf` werden diese Ereignisse dem powersave_proxy-Skript zugeordnet, das ausgeführt wird, sobald diese Ereignisse eintreten (Voreinstellung nach der Installation):

```
global.suspend=/usr/sbin/powersave_proxy
global.standby=/usr/sbin/powersave_proxy
global.resume.suspend=/usr/sbin/powersave_proxy
global.resume.standby=/usr/sbin/powersave_proxy
```

Benutzerdefinierte Batteriezustände

Sie können in der Datei `/etc/powersave.conf` drei Ladezustände der Batterie (in Prozent) festlegen, bei deren Erreichen das System warnt bzw. bestimmte Aktionen ausführt.

```
POWERSAVED_BATTERY_WARNING=20
POWERSAVED_BATTERY_LOW=10
POWERSAVED_BATTERY_CRITICAL=5
```

Welche Aktionen/Skripte ausgeführt werden, sobald bestimmte Ladezustände unterschritten werden, ist in der Konfigurationsdatei des `powersave`-Daemons festgelegt (`/etc/powersave.conf`). Der Typ dieser Aktionen ist in `/etc/sysconfig/powersave/common` konfiguriert:

```
POWERSAVE_EVENT_BATTERY_NORMAL="ignore"
POWERSAVE_EVENT_BATTERY_WARNING="notify"
POWERSAVE_EVENT_BATTERY_LOW="notify"
POWERSAVE_EVENT_BATTERY_CRITICAL="suspend"
```

Weitere Optionen lesen Sie direkt in dieser Konfigurationsdatei nach.

Anpassungen des Stromverbrauchs an verschiedene Arbeitsbedingungen

Sie können das Verhalten des Systems von der Art seiner Stromversorgung abhängig machen. So sollte der Stromverbrauch des Systems vermindert werden, wenn das System vom Netz getrennt und per Batterie betrieben wird. Umgekehrt sollte die Performance des Systems automatisch wieder steigen, sobald es sich wieder am Netz befindet. Konkret beeinflussbar sind die CPU-Frequenz, die Stromsparfunktion von IDE-Platten und einige andere Parameter mehr.

In `/etc/powersave.conf` ist die Ausführung bestimmter Aktionen bei Trennung/Anbindung vom Stromnetz an den `powersave_proxy` delegiert. In `/etc/sysconfig/powersave/common` wählen Sie die zu verwendenden Szenarien (genannt „Schemes“ oder „Profile“) fest:

```
POWERSAVE_AC_SCHEME="performance"
POWERSAVE_BATTERY_SCHEME="powersave"
```

Die „Schemes“ sind in entsprechenden Dateien unter `/etc/sysconfig/powersave/` abgelegt. Ihr Name setzt sich zusammen aus:

scheme_<Name des Schemas>. Im Beispiel werden zwei Schemas referenziert: `scheme_performance` und `scheme_powersave`. Vorkonfiguriert werden `performance`, `powersave` und `acoustic` ausgeliefert. Sie können mittels des YaST Powermanagement-Moduls jederzeit existierende Schemata bearbeiten, neue anlegen, bestehende löschen oder deren Zuordnungen zum Stromversorgungszustand ändern.

9.5.3 Zusätzliche ACPI-Features

Sollten Sie ACPI verwenden, können Sie die Reaktion Ihres Systems auf die so genannten „ACPI-Buttons“ (`(Power)`, `(Sleep)` und „Deckel öffnen“, „Deckel geschlossen“) steuern. In `/etc/powersave.conf` ist die Ausführung der entsprechenden Aktionen an den `powersave_proxy` delegiert. Die eigentliche Aktion selbst legen Sie in der Datei `/etc/sysconfig/powersave/common` fest. Nähere Erläuterungen zu den einzelnen Optionen entnehmen Sie bitte dieser Konfigurationsdatei.

POWERSAVE_EVENT_BUTTON_POWER="wm_shutdown"

Wird der `(Power)`-Button gedrückt, reagiert das System mit dem Herunterfahren des jeweiligen Windowmanagers (KDE, GNOME, fvwm...).

POWERSAVE_EVENT_BUTTON_SLEEP="suspend"

Wird der `(Sleep)`-Button gedrückt, fällt das System in den Suspend-Modus.

POWERSAVE_EVENT_BUTTON_LID_OPEN="ignore"

Beim Öffnen des Deckels passiert nichts.

POWERSAVE_EVENT_BUTTON_LID_CLOSED="screen_saver"

Wird der Deckel geschlossen, aktiviert sich der Bildschirmschoner.

Wird der Prozessor für eine bestimmte Zeit nicht über ein festgelegtes Maß hinaus beansprucht, können Sie seine Leistung zusätzlich drosseln. Legen Sie mit `POWERSAVED_CPU_LOW_LIMIT` den Level fest, bei dessen dauerhafter Unterschreitung — die Zeitspanne legen Sie in `POWERSAVED_CPU_IDLE_TIMEOUT` fest — die CPU heruntergeregelt wird.

9.5.4 Troubleshooting

Die folgenden Fragen und Antworten decken die häufigsten Probleme mit `powersave` ab.

■ Es gibt ein Problem, ich kann es aber nicht lokalisieren...

Sehen Sie sich `/var/log/messages` an. Sämtliche Fehler- und Warnmeldungen werden hier protokolliert. Ergibt sich hier auf den ersten Blick kein Hinweis, weisen Sie `powersave` in der Datei `/etc/sysconfig/powersave/common` über die Variable `DEBUG` an, seine Meldungen etwas detaillierter und ausführlicher zu halten. Erhöhen Sie den Variablenwert hierzu auf 7 oder gar 15 und starten Sie den Daemon neu. Mithilfe der jetzt ausführlicheren Fehlermeldungen in `/var/log/messages` sollten Sie in der Lage sein, den Fehler einzugrenzen.

■ Ich habe ACPI aktiviert, aber die Batteriezustände und Buttons funktionieren nicht so wie konfiguriert...

Sollten Sie mit ACPI Probleme bekommen, durchsuchen Sie mit folgendem Befehl die Ausgabe von `dmesg` nach ACPI-spezifischen Meldungen: `dmesg | grep -i acpi`.

Um den Fehler zu beheben, kann ein BIOS-Update notwendig werden. Besuchen Sie daher die Homepage Ihres Laptopherstellers, suchen Sie nach einer aktuelleren BIOS-Version und spielen Sie diese ein. Geben Sie an den Hersteller Ihres Systems weiter, dass er sich an die aktuellste ACPI-Spezifikation halten soll.

Treten die Fehler nach dem BIOS-Update immer noch auf, suchen Sie auf den folgenden Webseiten nach einer aktuelleren DSDT für Ihr System, um die fehlerhafte DSDT-Tabelle in Ihrem BIOS zu ersetzen:

1. Laden Sie die für Ihr System passende DSDT von <http://acpi.sourceforge.net/dsdt/tables> herunter. Stellen Sie sicher, dass die Datei entzippt und kompiliert ist (zu erkennen an der Dateiendung `.aml` (ACPI Machine Language)). Ist dies der Fall, fahren Sie mit Punkt 3 fort.
2. Ist die Dateiendung der heruntergeladenen Tabelle `.asl` (ACPI Source Language), muss sie mit Hilfe von `iasl` aus dem Paket `pmttools` kompiliert werden. Rufen Sie hierzu `iasl -sa <Datei>.asl`. Die aktuellste Version von `iasl` (Intel ACPI Compiler) finden Sie ausserdem unter <http://developer.intel.com/technology/iapc/acpi/downloads.htm>.
3. Kopieren Sie die Datei `DSDT.aml` an eine beliebige Stelle (wir empfehlen `/etc/DSDT.aml`). Editieren Sie `/etc/sysconfig/kernel` und passen Sie den Pfad zur DSDT-Datei entsprechend an. Starten Sie `mkinitrd` (Paket `mkinitrd`). Wann immer Sie Ihren Kernel deinstallieren und `mkinitrd` verwenden, um eine `initrd` zu erstellen, wird die angepasste DSDT eingebunden und zur Bootzeit geladen.

■ CPU Frequency funktioniert nicht...

Überprüfen Sie anhand der Kernelquellen (`kernel-source`), ob Ihr Prozessor unterstützt wird und ob Sie eventuell ein bestimmtes Kernelmodul oder eine bestimmte Modulooption verwenden müssen, um CPU-Frequency zu aktivieren. Diese Informationen finden Sie unter `/usr/src/linux/Documentation/cpu-freq/*`. Wenn ein bestimmtes Modul oder eine bestimmte Option nötig sind, konfigurieren Sie dies in der Datei `/etc/sysconfig/powersave/common` über die Variablen `CPUFREQD_MODULE` und `CPUFREQD_MODULE_OPTS`.

■ Suspend/Standby funktioniert nicht...

Es sind mehrere, mit dem Kernel zusammenhängende Probleme bekannt, die auf ACPI Systemen Suspend/Standby verhindern:

- ▷ Systeme mit mehr als 1 GB RAM unterstützen im Moment (noch) kein Suspend
- ▷ Multiprozessorsysteme oder Systeme mit einem P4-Prozessor (mit Hyperthreading) unterstützen momentan kein Suspend.

Der Fehler kann auch in einer fehlerhaften Implementierung Ihrer DSDT (BIOS) liegen. In diesem Fall spielen Sie eine neue DSDT wie unter *Ich habe ACPI aktiviert, aber die Batteriezustände und Buttons funktionieren nicht so wie konfiguriert...* beschrieben ein.

Auf ACPI und APM Systemen:

Sobald Ihr System versucht, fehlerhafte Module zu entladen, hängt sich der Proxy auf und das Suspendereignis wird niemals getriggert. Der umgekehrte Weg ist auch möglich, wenn Sie Module/Dienste nicht entladen oder stoppen, die einen erfolgreichen Suspend verhindern. In beiden Fällen sollten Sie versuchen, durch Manipulation der folgenden Einstellungen unter `/etc/sysconfig/powersave/common` herauszufinden, welche Module das Problem hervorrufen:

```
POWERSAVE_UNLOAD_MODULES_BEFORE_SUSPEND=" "  
POWERSAVE_UNLOAD_MODULES_BEFORE_STANDBY=" "  
POWERSAVE_SUSPEND_RESTART_SERVICES=" "  
POWERSAVE_STANDBY_RESTART_SERVICES=" "
```

■ Bei der Verwendung von ACPI: Der Powersave-Daemon erkennt nicht, wenn ein bestimmtes Batterielimit erreicht wurde...

Unter ACPI kann das Betriebssystem vom BIOS eine Meldung über das Unterschreiten eines bestimmten Ladeniveaus der Batterie anfordern. Der Vorteil dieser Methode ist, dass nicht permanent der Batteriezustand ausgelesen werden muss, was sonst die Performance des Rechners schwächen würde. Trotzdem kann es vorkommen, dass diese Benachrichtigung laut BIOS zwar funktionieren sollte, tatsächlich aber nicht stattfindet, selbst bei Unterschreitung des Limits nicht.

Sollten Sie dies auf Ihrem System beobachten, setzen Sie in der Datei `/etc/powersave.conf` die Variable `POWERSAVED_FORCE_BATTERY_POLLING` auf `yes`, um das Auslesen des Batteriezustands zu erzwingen.

9.6 Das YaST Powermanagement-Modul

Mit Hilfe des YaST Powermanagement-Moduls können Sie alle Einstellungen zum Powermanagement vornehmen, die in den vorangegangenen Abschnitten erläutert wurden.



Abbildung 9.1: YaST-Powermanagement: Scheme selektieren

Nach dem Start des Moduls über das YaST-Controlcenter ('System' → 'Powermanagement') gelangen Sie in die erste Maske des Moduls (siehe Abbildung 9.1 auf der vorherigen Seite), in der Sie zur Auswahl der bei bestimmten Betriebszustände — Akkubetrieb oder Betrieb am Stromnetz — zu verwendenden Schemes aufgefordert werden.

Sie können sich an dieser Stelle per Drop-Down-Menü für jeweils eines der bereits existierenden Schemes entscheiden, oder aber über den Button 'Edit Schemes' in eine Übersicht der bereits vorhandenen Schemes gelangen (Abbildung 9.2).

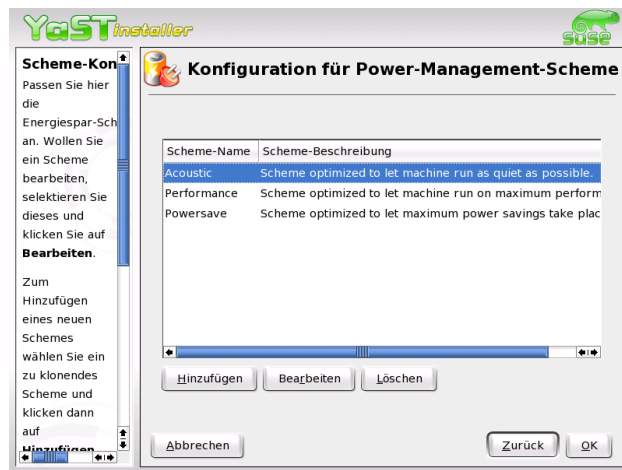


Abbildung 9.2: YaST-Powermanagement: Überblick der vorhandenen Schemes

In der Schemes-Übersicht selektieren Sie dasjenige Scheme, das Sie ändern möchten und klicken dann auf 'Edit', um in den Editierdialog zu gelangen (siehe Abbildung 9.3 auf der nächsten Seite). Alternativ können Sie ein neues Scheme erstellen, indem Sie den Button 'Add' drücken. In beiden Fällen ist der Folgedialog identisch.

Versehen Sie das neue oder zu ändernde Scheme zuerst mit einem (sprechenden) Namen und einer Beschreibung. Für die Festplatte legen Sie eine 'Standby Policy' fest, die entweder auf maximale Performance oder auf Energieersparnis ausgelegt ist. Die 'Acoustic Policy' regelt den Geräuschpegel der Festplatte. Klicken Sie auf 'Next', um in den Dialog zur Konfiguration der Optionen 'CPU' und 'Cooling Policy' zu gelangen. 'CPU' umfasst die Optionen 'CPU Frequency Scaling' und 'Throttling', mit deren

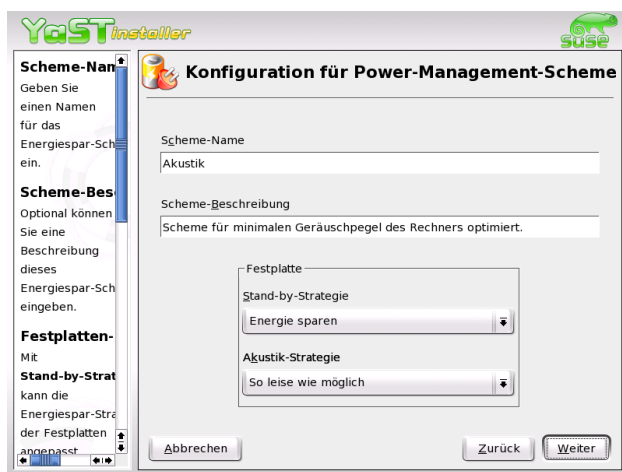


Abbildung 9.3: YaST-Powermanagement: Scheme erstellen

Hilfe Sie festlegen, ob und inwieweit die CPU-Frequenz bei Bedarf heruntergeregt werden darf. Die 'Cooling Policy' regelt, welche Art der Kühlung angewandt werden soll. Sobald Sie alle Einstellungen für das Scheme abgeschlossen haben, verlassen Sie diesen Dialog mit 'OK' und kehren in den Startdialog (Abbildung 9.1 auf Seite 275) zurück. Dort können Sie nun das selbsterstellte Scheme für einen der beiden Betriebszustände anwählen. Verlassen Sie diesen Dialog wiederum mit 'OK', werden Ihre Einstellungen aktiv.

Aus dem Startdialog heraus (siehe Abbildung 9.1 auf Seite 275), können Sie neben der Scheme-Auswahl für verschiedene Betriebszustände auch globale Einstellungen zum Powermanagement vornehmen. Klicken Sie hierzu auf 'Battery Warnings' oder 'ACPI Settings'. Um in den Dialog zum Ladezustand der Batterie zu gelangen, klicken Sie 'Battery Warnings' (9.4 auf der nächsten Seite).

Das BIOS Ihres Systems meldet dem Betriebssystem, sobald bestimmte, konfigurierbare Kapazitätsgrenzen unterschritten werden. Daraufhin können bestimmte Aktionen ausgelöst werden. In diesem Dialog legen Sie drei Grenzen fest, deren Unterschreitung bestimmte Aktionen auslösen soll. Dies sind 'Warning Capacity', 'Low Capacity' und 'Critical Capacity'. In den ersten beiden Fällen wird üblicherweise nur eine Warnmeldung an den Benutzer weitergereicht, während Unterschreitung des letzten kritischen

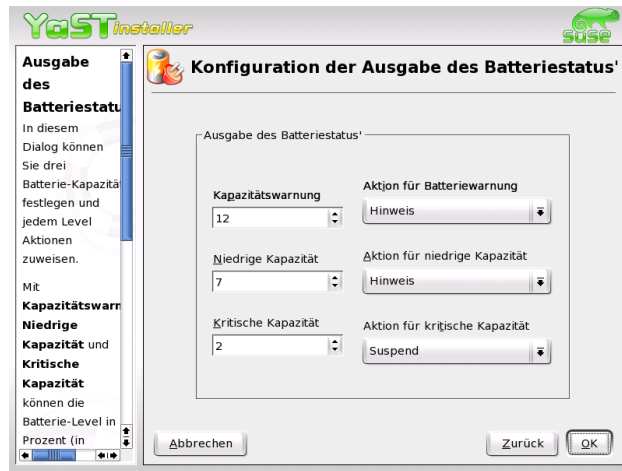


Abbildung 9.4: YaST-Powermanagement: Ladezustand der Batterie

Levels ein Suspend des Rechners auslöst, da die verbliebene Energie kaum noch für längere Zeit einen sinnvollen Betrieb des Systems erlaubt. Wählen Sie die für Ihre Zwecke passenden Ladezustände und entsprechenden Aktionen aus und verlassen Sie diesen Dialog mit 'OK', um zurück in den Startdialog zu gelangen. Von dort aus gelangen Sie über 'ACPI Settings' in den Dialog zur Konfiguration der ACPI-Buttons (siehe Abbildung 9.5 auf der nächsten Seite).

Mit den Einstellungen zu den ACPI-Buttons legen Sie fest, wie das System auf die Betätigung bestimmter Schalter reagieren soll. Diese Schalter/Ereignisse kennt ACPI als „Buttons“. Konfigurieren Sie die Antwort des Systems auf Drücken der (Power)-Taste, einer (Sleep)-Taste und auf Schließen des Laptopdeckels. Mit 'OK' schließen Sie die Konfiguration ab und gelangen zurück in den Startdialog (Abbildung 9.1 auf Seite 275). Verlassen Sie das gesamte Modul durch ein erneutes Drücken von 'OK', um alle Ihre Einstellungen zum Powermanagement zu übernehmen.

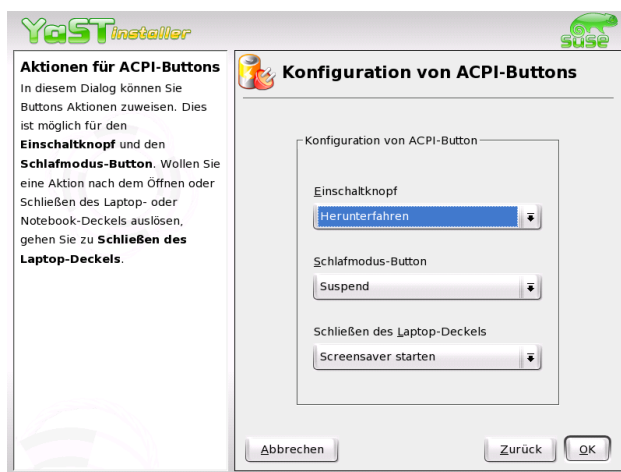


Abbildung 9.5: YaST-Powermanagement: ACPI-Einstellungen

Teil III

System

SUSE LINUX auf AMD64 Systemen

AMD hat im September 2003 den AMD Athlon64-Prozessor der Öffentlichkeit vorgestellt. Dieser neue Prozessor ist ein 64-bit Prozessor und kann somit neue 64-bit AMD64-Programme ausführen. Desweiteren ermöglicht er die Ausführung bestehender 32-bit x86-Programme mit gleicher Performance.

10.1	64-bit SUSE LINUX für AMD64	284
10.2	Weitere Informationen	285

64-bit Programme erlauben einen größeren Adressraum und können aufgrund von Extra-Registern, die nur im 64-bit Modus unterstützt werden, sowie von weiteren Änderungen, wie den moderneren Aufrufkonventionen „Calling conventions“ für Funktionen, eine bessere Performance bieten.

SUSE LINUX unterstützt den neuen Prozessor mit diesem Produkt in doppelter Hinsicht:

- Die 32-bit SUSE LINUX für x86 unterstützt diesen Prozessor als 32-bit Prozessor, so wie es auch den AMD Athlon und die Intel Pentium Prozessoren unterstützt.
- Die neue 64-bit SUSE LINUX für AMD64 unterstützt den Prozessor im 64-bit Modus. Desweiteren wird sowohl Ausführung als auch Entwicklung von 32-bit x86 Programmen unterstützt.

Hinweis

Aus historischen Gründen ist die Ausgabe von `uname -m` **x86_64**, da dies der Name der ersten Spezifikation von AMD war.

Hinweis

10.1 64-bit SUSE LINUX für AMD64

10.1.1 Hardware

Auf der Hardware-Seite ist bei AMD64 aus Anwenderseite alles wie auf normalen AMD Athlon Systemen. Die gängigen Schnittstellen und Busse sind auf beiden Plattformen die gleichen und werden auch unterstützt.

Da die Hardware-Treiber für Linux auf AMD64 64-bit Treiber sein müssen, müssen diese teilweise erst angepaßt werden. Einige ältere Karten funktionieren zur Zeit nicht, aber die Hardwareunterstützung von aktueller Hardware sollte in 32-bit und 64-bit gleich sein.

10.1.2 Software

Auf der Software-Seite sind fast alle Pakete 64-bit. Desweiteren wird die Ausführung von 32-bit Programmen unterstützt. Hierzu wurden extra 32-bit Library-Pakete entwickelt, die in der Standardinstallation auch installiert werden. Um 32-bit und 64-bit Libraries mit gleichem Namen auf einem System zu installieren, werden 32-bit Libraries in Verzeichnisse `/lib/`

und 64-bit Libraries in `/lib64/` Verzeichnisse installiert. Die ermöglicht insbesondere, dass 32-bit RPMs ohne Änderungen installiert werden können.

Auf der Administrations- und Anwendungsseite ist eine Unterscheidung zwischen 32-bit und 64-bit nicht direkt erkennbar, alle Programme sehen gleich aus und haben gleiches Verhalten.

10.1.3 Verwendung von 32-bit Software

32-bit Software, welche `uname` aufruft um die Architektur festzustellen, muss eventuell angepasst werden um, damit sie auf einem AMD64 System läuft. Ändern Sie die Ausgabe von `uname -m` mit dem Programm `linux32`. Geben Sie dazu den Befehl `linux32` gefolgt von einem Leerzeichen und dann den Programmaufruf ein. Damit können Sie auch eine Shell starten, in der `uname` für alle Eingaben modifiziert ist. Aus dieser Shell können Sie problemlos eine beliebige Zahl von Programmen starten.

10.1.4 Softwareentwicklung unter 64-bit

Auf einem SUSE LINUX für AMD64 System lassen sich sowohl 32-bit als auch 64-bit Programme entwickeln. Die GNU Compiler erzeugen normalerweise 64-bit AMD64 Code. Der Schalter `-m32` sorgt für die Erzeugung von 32-bit x86 Code, der dann auch auf einem 32-bit AMD Athlon oder Intel Pentium System läuft.

Bei der Entwicklung von 64-bit Code müssen die 64-bit Libraries benutzt werden. Die Pfade `/lib64/` und `/usr/lib64/` werden immer durchsucht, aber für z.B. X11-Code, muß ein `-L/usr/X11R6/lib64` benutzt werden. Hier sind also teilweise Anpassungen an den Makefiles nötig.

Zum Debuggen von Code kann GDB benutzt werden, für 64-bit AMD64 Programm heißt das Programm `gdb`, aber für 32-bit x86 Programme `gdb32`. Das Tool `strace` kann sowohl 32-bit als auch 64-bit Programme untersuchen und bei dem Library Tracer `ltrace` gibt es auch ein extra 32-bit Programm `ltrace32`.

10.2 Weitere Informationen

Für weitere Informationen verweisen wir auf die Webseite von AMD (<http://www.amd.com>) und die Projektseite des Linuxports auf AMD64 (<http://www.x86-64.org>).

Der Linux Kernel

Der Kernel verwaltet die Hardware jedes Linux Systems und stellt diese den verschiedensten Prozessen zur Verfügung. Auf den folgenden Seiten wird man nicht lernen, wie man Kernel-„Hacker“ wird, aber man erfährt, wie man ein Kernel-Update durchführt, und wird in die Lage versetzt, sich einen selbstkonfigurierten Kernel zu kompilieren und zu installieren. Wenn Sie so vorgehen, wie in diesem Kapitel beschrieben, bleibt der bisherige Kernel funktionsfähig und kann jederzeit auf Wunsch gebootet werden.

11.1	Kernel-Update	288
11.2	Die Kernelquellen	289
11.3	Konfiguration des Kernels	289
11.4	Kernel-Module	291
11.5	Einstellungen bei der Kernelkonfiguration	294
11.6	Übersetzen des Kernels	294
11.7	Kernel installieren	295
11.8	Festplatte nach der Übersetzung aufräumen	296

Der Kernel, der bei der Installation im `/boot/`-Verzeichnis abgelegt wird, ist so konfiguriert, dass er ein möglichst breites Spektrum von Hardware unterstützt. Es ist meist *nicht erforderlich*, einen eigenen Kernel zu generieren, außer Sie wollen „experimentelle“ Features oder Treiber ausprobieren.

Zum Erzeugen eines neuen Kernels existieren bereits `Makefiles`, mit deren Hilfe der Ablauf fast völlig automatisiert ist. Lediglich die Auswahl der vom Kernel zu unterstützenden Hardware und Features muss interaktiv durchlaufen werden. Da Sie Ihr Computer-System ziemlich gut kennen müssen, um eine funktionierende Auswahl zu treffen, empfehlen wir – wenigstens für die ersten Versuche – eine bestehende und funktionierende Konfigurationsdatei abzuändern und damit die Gefahr falscher Einstellungen zu vermindern.

11.1 Kernel-Update

Um einen SUSE Update-Kernel zu installieren, laden Sie das Update-Paket vom SUSE FTP-Server oder einem Mirror wie zum Beispiel: `ftp://ftp.gwdg.de/pub/linux/suse/` herunter. Wenn Sie nicht wissen, welcher Kernel aktuell bei Ihnen läuft, so können Sie zum einen den version-String ansehen: `cat /proc/version`.

Sie können außerdem prüfen, zu welchem Paket der Kernel `/boot/vmlinuz` gehört: `rpm -qf /boot/vmlinuz`.

Vor der Installation sollten Sie den ursprünglichen Kernel und die dazugehörige `initrd` sichern. Geben Sie dazu als `root` die folgenden beiden Befehle ein:

```
cp /boot/vmlinuz /boot/vmlinuz.old
cp /boot/initrd /boot/initrd.old
```

Installieren Sie nun das neue Paket mit dem Befehl `rpm -Uvh <Paketname>`. Setzen Sie dabei die entsprechende Versionsnummer ein.

Seit SUSE LINUX 7.3 wird `reiserfs` als Standardfilesystem verwendet, was den Einsatz einer „initial ramdisk“ voraussetzt. Diese wird mit dem Befehl `mk_initrd` neu geschrieben. Bei aktuellen SUSE LINUX Versionen geschieht dies automatisch bei der Installation des Kernels.

Um gegebenenfalls den alten Kernel booten zu können, muss der Bootloader entsprechend konfiguriert werden. Genaue Informationen dazu finden Sie im Kapitel 7 auf Seite 203.

Wenn Sie den Original-Kernel von den SUSE LINUX CDs installieren möchten, gehen Sie ähnlich vor. Auf CD 1 oder der DVD finden Sie im Verzeichnis `boot/` den Standard Kernel als rpm-Paket. Installieren Sie diesen wie oben beschrieben. Falls Sie eine Fehlermeldung erhalten, dass bereits ein neueres Paket installiert ist, müssen Sie die Option `--force` beim rpm-Kommando zusätzlich angeben.

11.2 Die Kernelquellen

Um einen Kernel bauen zu können, müssen die Kernelquellen (Paket `kernel-source`) installiert werden. Andere erforderliche Pakete wie der C-Compiler (Paket `gcc`), die GNU Binutils (Paket `binutils`) und die Include-Dateien für den C-Compiler (Paket `glibc-devel`) werden dabei automatisch mit ausgewählt.

Die Kernelquellen befinden sich nach der Installation im Verzeichnis `/usr/src/linux-<kernel-version>/`. Sollten Sie vorhaben, mit dem Kernel zu experimentieren und verschiedene Versionen des Kernels gleichzeitig auf der Platte zu halten, so bietet es sich an, die einzelnen Versionen in verschiedene Verzeichnisse zu entpacken und die augenblicklich relevanten Quellen über einen Link anzusprechen, da es Software-Pakete gibt, die die Kernelquellen unter `/usr/src/linux` erwarten. Diese Form der Installation wird von YaST automatisch vorgenommen.

11.3 Konfiguration des Kernels

Die Konfiguration des aktuell laufenden Kernel ist in der Datei `/proc/config.gz` gespeichert. Um diese Konfiguration nach Ihren Wünschen anzupassen, wechseln Sie als Benutzer `root` in das Verzeichnis `/usr/src/linux/` und führen folgende Befehle aus:

```
zcat /proc/config.gz > .config
make oldconfig
```

Der Befehl `make oldconfig` verwendet die Datei `/usr/src/linux/.config` als Vorlage zur aktuellen Kernelkonfiguration. Wenn bei Ihren aktuellen Kernel-Sourcen neue Optionen hinzugekommen sind, so werden diese jetzt abgefragt.

Wenn die Datei `.config` fehlt, dann wird eine „default“ Konfiguration verwendet, die in den Kernel-Sourcen enthalten ist.

11.3.1 Kommandozeilenkonfiguration

Um den Kernel zu konfigurieren, wechseln Sie nach `/usr/src/linux` und geben den Befehl `make config` ein.

Sie werden nach einer Reihe von Systemfähigkeiten gefragt, die der Kernel unterstützen soll. Bei der Beantwortung der Fragen gibt es normalerweise zwei oder drei Möglichkeiten: Entweder einfaches **y** und **n**, oder eine der drei Möglichkeiten **y** (yes), **n** (no) und **m** (module). **m** bedeutet hierbei, dass der entsprechende Treiber nicht fest zum Kernel hinzugebunden wird, sondern vielmehr als Modul übersetzt wird, das zur Laufzeit zum Kernel hinzugeladen werden kann. Sämtliche Treiber, die zum Booten des Systems unbedingt benötigt werden, müssen fest zum Kernel hinzugebunden werden; in diesen Fällen also **y** wählen. Mit **Enter** bestätigen Sie die Vorauswahl, die aus der Datei `.config` eingelesen wird. Wenn Sie bei einer Frage eine andere Taste drücken, erhalten Sie einen kurzen Hilfetext zu der jeweiligen Option angezeigt.

11.3.2 Konfiguration im Textmodus

Angenehmer lässt sich die Konfiguration des Kernels mit „menuconfig“ durchführen; gegebenenfalls müssen Sie dazu das `ncurses-devel` mit `YaST` nachinstallieren. Starten Sie die Kernel-Konfiguration mit dem Befehl `make menuconfig`.

Bei einer geringfügigen Änderung der Konfiguration müssen Sie sich hier nicht durch alle Fragen „durchtasten“, sondern können über das Menü direkt bestimmte Bereiche wählen. Die Voreinstellungen werden der Datei `.config` entnommen. Um eine andere Konfiguration zu laden, wählen Sie den Menüpunkt ‘Load an Alternate Configuration File’ und geben den Dateinamen an.

11.3.3 Konfiguration unter dem X Window System

Haben Sie das X Window System (Paket `xf86`) sowie `Tcl/Tk` (Paket `tcl` und `tk`) installiert, können Sie die Konfiguration alternativ durch den Befehl `make xconfig` vornehmen.

Sie haben dann eine grafische Oberfläche, die das Konfigurieren komfortabler macht. Dazu müssen Sie das X Window System aber als Benutzer `root` gestartet haben oder in der Shell zuerst als normaler Benutzer `xhost +` eingeben, um `root` Zugriff auf das Display zu gewähren. Die Voreinstellungen werden aus der Datei `.config` ausgelesen. Beachten Sie,

daß die Konfiguration über `make xconfig` nicht so gut gepflegt ist, wie die anderen Konfigurationsmöglichkeiten. Sie sollten daher nach dieser Konfigurationmethode immer noch ein `make oldconfig` ausführen.

11.4 Kernel-Module

Es gibt eine große Vielfalt an PC-Hardware-Komponenten. Um diese Hardware richtig benutzen zu können, braucht man einen „Treiber“, über den das Betriebssystem (bei Linux der „Kernel“) die Hardware richtig ansprechen kann. Generell gibt es zwei Mechanismen, Treiber in den Kernel zu integrieren:

- Die Treiber können fest in den Kernel einkompiliert sein. Solche Kernel „aus einem Stück“ bezeichnen wir in diesem Buch auch als *monolithische* Kernel. Manche Treiber können nur in dieser Form verwendet werden.
- Die Treiber können erst bei Bedarf in den Kernel geladen werden, der in diesem Fall als *modularisierter* Kernel bezeichnet wird. Das hat den Vorteil, dass wirklich nur die benötigten Treiber geladen sind und dass der Kernel keinen unnötigen Ballast enthält.

Welche Treiber fest zum Kernel gebunden und welche als Module realisiert werden, wird bei der Konfiguration des Kernels festgelegt. Alle Kernel-Komponenten, die nicht zwingend während des Bootvorgangs benötigt werden, sollten als Module realisiert werden. So wird sichergestellt, dass der Kernel nicht zu groß wird und dass der Kernel ohne Schwierigkeiten vom BIOS und einem beliebigen Bootloader geladen werden kann. Der Festplatten-Treiber, Unterstützung für `ext2` und ähnliche Dinge sind also im Regelfall direkt in den Kernel hineinzukompilieren, Unterstützung für `isofs`, `msdos` oder `sound` sollten in jedem Fall als Module kompiliert werden.

Die Kernelmodule werden in dem Verzeichnis `/lib/modules/<Version>/` abgelegt, wobei `Version` der momentanen Version des Kernels entspricht.

11.4.1 Erkennung der aktuellen Hardware mit `hwinfo`

Unter SUSE LINUX steht Ihnen das Programm `hwinfo` zur Verfügung, mit der die aktuelle Hardware des Rechners erkannt werden kann, und die ver-

fügbaren Treiber zugeordnet werden. Eine kurze Hilfestellung zu diesem Programm bekommen Sie mit dem Befehl `hwinfo --help`.

Um zum Beispiel die Daten der eingebauten SCSI-Geräte zu bekommen geben Sie folgenden Befehl ein:

```
hwinfo --scsi
```

Die Ausgaben dieses Hilfsprogrammes stehen Ihnen auch in YaST im Modul Hardware-Information zur Verfügung.

11.4.2 Umgang mit Modulen

Folgende Befehle zum Umgang mit Modulen stehen zur Verfügung:

insmod Mit dem Befehl `insmod` wird das angegebene Modul geladen. Das Modul wird in einem Unterverzeichnis von `/lib/modules/<Version>` gesucht. Zugunsten von `modprobe` (s. u.) sollte `insmod` *nicht* mehr verwendet werden.

rmmod Entlädt das angegebene Modul. Dies ist natürlich nur dann möglich, wenn die entsprechende Funktionalität des Kernels nicht mehr verwendet wird. So ist es nicht möglich, das Modul `isofs` zu entladen, wenn noch eine CD gemountet ist.

depmod Dieser Befehl erzeugt eine Datei mit dem Namen `modules.dep` im Verzeichnis `/lib/modules/<Version>`, in der die Abhängigkeiten der einzelnen Module untereinander verzeichnet sind. Damit stellt man sicher, dass beim Laden eines Modules alle davon abhängigen Module ebenfalls automatisch geladen werden. Die Datei mit den Modul-Abhängigkeiten beim Start des Systems automatisch generiert, sofern sie noch nicht existiert.

modprobe Laden bzw. Entladen eines Modules mit Berücksichtigung der Abhängigkeiten von anderen Modulen. Dieser Befehl ist sehr mächtig und kann für eine Reihe weiterer Zwecke eingesetzt werden (etwa Durchprobieren aller Module eines bestimmten Typs, bis eines erfolgreich geladen werden kann). Im Gegensatz zum Laden mittels `insmod` wertet `modprobe` die Datei `/etc/modprobe.conf` aus und sollte daher generell zum Laden von Modulen verwendet werden. Für eine ausführliche Erklärung sämtlicher Möglichkeiten lesen Sie bitte die zugehörigen Manual-Pages.

lsmod Zeigt an, welche Module gegenwärtig geladen sind und von wie vielen anderen Modulen sie verwendet werden. Module, die vom Kernel-Daemon geladen wurden, sind durch ein nachfolgendes `autoclean` gekennzeichnet. Die Kennzeichnung mit `autoclean` weist darauf hin, dass diese Module automatisch wieder entfernt werden, wenn sie längere Zeit nicht benutzt wurden und man entsprechende Vorkehrungen getroffen hat; vgl. jedoch Abschnitt 11.4.4.

modinfo Zeigt Informationen zu einem Modul an.

11.4.3 /etc/modprobe.conf

Das Laden von Modulen wird über die Dateien `/etc/modprobe.conf` `/etc/modprobe.conf.local` und das Verzeichnis `/etc/modprobe.d` beeinflusst; vgl. die Manualpage `man modprobe.conf`. In dieser Datei können auch die Parameter für solche Module eingetragen werden, die direkt auf die Hardware zugreifen und daher auf das spezifische System eingestellt werden müssen (zum Beispiel CD-ROM-Treiber oder Netzwerk-treiber). Die hier eingetragenen Parameter werden in den Kernel Sources beschrieben. Installieren Sie dazu das Paket `kernel-source` und lesen Sie die Dokumentation im Verzeichnis `/usr/src/linux/Documentation/`.

11.4.4 Kmod – der Kernel Module Loader

Der eleganteste Weg bei der Verwendung von Kernel-Modulen ist der Einsatz des „Kernel Module Loader“. Kmod wacht im Hintergrund und sorgt dafür, dass benötigte Module durch `modprobe`-Aufrufe automatisch geladen werden, sobald auf die entsprechende Funktionalität des Kernels zugegriffen wird.

Um den Kmod verwenden zu können, müssen Sie bei der Kernel-Konfiguration die Option 'Kernel module loader' (`CONFIG_KMOD`) aktivieren.

Der Kmod ist nicht dafür ausgelegt, Module wieder automatisch zu entladen; bei der heutigen RAM-Ausstattung der Rechner wäre der Gewinn an Arbeitsspeicher nur marginal. Server-Rechner, die spezielle Aufgaben zu erfüllen haben und nur wenige Treiber benötigen, werden aus Performance-Gründen einen „monolithischen“ Kernel bevorzugen.

11.5 Einstellungen bei der Kernelkonfiguration

Die einzelnen Konfigurationsmöglichkeiten des Kernels können hier nicht im Detail dargestellt werden. Machen Sie bitte Gebrauch von den zahlreichen vorhandenen Hilfetexten zur Kernel-Konfiguration. Der neueste Stand der Dokumentation findet sich immer im Verzeichnis `/usr/src/linux/Documentation/` sofern das Paket `kernel-source` installiert ist.

11.6 Übersetzen des Kernels

Wir empfehlen, ein „bzImage“ zu generieren. So lässt es sich in der Regel umgehen, dass der Kernel „zu groß“ wird, wie dies leicht passieren kann, wenn man zu viele Features auswählt und ein „zImage“ herstellt (typisch sind dann die Meldungen "kernel too big" oder "System is too big").

Nachdem Sie den Kernel für Ihre Gegebenheiten konfiguriert haben, starten Sie die Kompilation (in `/usr/src/linux/`:

```
make clean
make bzImage
```

Diese beiden Befehle können Sie auch in einer Befehlszeile eingeben:

```
make clean bzImage
```

Nach der erfolgreichen Übersetzung finden Sie den komprimierten Kernel in `/usr/src/linux/arch/<arch>/boot/`. Das Kernel-Image – die Datei, die den Kernel enthält – heißt `bzImage`.

Finden Sie diese Datei nicht vor, ist aller Wahrscheinlichkeit nach ein Fehler während der Kernelübersetzung aufgetreten. Unter der Bash können Sie mit:

```
make bzImage 2> &1 | tee kernel.out
```

den Kompilationsvorgang erneut starten und in die Datei `kernel.out` „mitschreiben“ lassen.

Wenn Sie Teile des Kernels als ladbare Module konfiguriert haben, müssen Sie anschließend das Übersetzen dieser Module veranlassen. Dies erreichen Sie durch: `make modules`.

11.7 Kernel installieren

Nachdem Sie den Kernel übersetzt haben, müssen Sie dafür sorgen, dass dieser neue Kernel installiert wird, um ihn künftig booten zu können.

Wenn Sie LILO verwenden, so muss dies gleichfalls neu installiert werden. Im einfachsten Fall kopieren Sie dazu den neuen Kernel nach `/boot/vmlinuz` und rufen dann LILO auf; um sich vor unliebsamen Überraschungen zu schützen, empfiehlt es sich jedoch, den alten Kernel zunächst beizubehalten (als `/boot/vmlinuz.old`), um ihn notfalls booten zu können, wenn der neue Kernel nicht wie erwartet funktioniert:

```
cp /boot/vmlinuz /boot/vmlinuz.old
cp arch/i386/boot/bzImage /boot/vmlinuz
lilo
```

Das Makefile-Target `make bzlilo` erledigt diese drei Schritte übrigens in einem Rutsch.

Hinweis

Wenn Sie GRUB als Bootloader verwenden, muss dieser *nicht* neu installiert werden! Führen Sie also bitte nur die ersten beiden Schritte aus, um den Kernel an die richtige Stelle im System zu kopieren.

Hinweis

Die übersetzten Module müssen nun noch installiert werden; durch Eingabe von `make modules_install` können Sie diese in die korrekten Zielverzeichnisse unter `/lib/modules/<Version>/` kopieren lassen. Dabei werden die alten Module bei gleicher Kernelversion überschrieben; Sie können jedoch die ursprünglichen Module zusammen mit dem Kernel von den CDs wieder installieren.

Hinweis

Es ist darauf zu achten, dass Module, deren Funktionalität man jetzt eventuell direkt in den Kernel einkompiliert hat, unter `/lib/modules/<Version>/` entfernt werden. Sonst kann es zu unvorhersehbaren Effekten kommen! Dies ist ein Grund, weshalb dem Ungeübten vom Selbstkompilieren des Kernels *dringend* abgeraten wird.

Hinweis

Damit der alte Kernel (jetzt `/boot/vmlinuz.old`) von GRUB oder LILO gebootet werden kann, tragen Sie in der Datei `/boot/grub/menu.lst` bzw. in `/etc/lilo.conf` zusätzlich ein Label `linux.old` als Boot-Image ein. Dieses Vorgehen wird ausführlich im Kapitel 7 auf Seite 203 beschrieben. Falls Sie LILO als Bootloader verwenden, müssen Sie nach den Anpassungen in `/etc/lilo.conf` erneut `lilo` aufrufen. Bei GRUB ist keine Neuinstallation notwendig.

Weiterhin ist Folgendes zu beachten: Die Datei `/boot/System.map` enthält die Kernelsymbole, die die Kernelmodule benötigen, um Kernelfunktionen korrekt aufrufen zu können. Diese Datei ist abhängig vom aktuellen Kernel. Daher sollten Sie nach der Übersetzung und Installation des Kernels die aktuelle Datei `/usr/src/linux/System.map` in das Verzeichnis `/boot/` kopieren. Bei jeder Kernelübersetzung wird diese Datei neu erzeugt. Falls Sie Ihren Kernel mittels `make bzlilo` bzw. `make zlilo` erstellen, wird diese Aufgabe automatisch für Sie erledigt.

Sollten Sie beim Booten eine Fehlermeldung wie "System.map does not match actual kernel" erhalten, dann wurde wahrscheinlich nach der Kernelübersetzung die Datei `System.map` nicht nach `/boot/` kopiert.

11.8 Festplatte nach der Übersetzung aufräumen

Sie können die während der Kernel-Übersetzung erzeugten Objekt-Dateien löschen, falls Sie Probleme mit dem Plattenplatz haben:

```
cd /usr/src/linux
make clean
```

Falls Sie jedoch über ausreichend Plattenplatz verfügen und vorhaben, den Kernel des Öfteren neu zu konfigurieren, so überspringen Sie diesen letzten Schritt. Ein erneutes Übersetzen des Kernels ist dann erheblich schneller, da nur die Teile des Systems neu übersetzt werden, die von den entsprechenden Änderungen betroffen sind.

Systemmerkmale

In diesem Kapitel finden Sie Hinweise zu *Filesystem Hierarchy Standard* (FHS) und *Linux Standard Base* (LSB). Einzelne Softwarepakete und besondere Eigenheiten wie das Booten mit der `initrd`, das Programm `linuxrc` und das Rettungssystem werden detailliert beschrieben.

12.1	Linux-Standards	298
12.2	Hinweise zu speziellen Softwarepaketen	299
12.3	Booten mit der initial ramdisk	305
12.4	linuxrc	310
12.5	Das SUSE Rettungssystem	316
12.6	Virtuelle Konsolen	321
12.7	Tastaturbelegung	321
12.8	Lokale Anpassungen – I18N/L10N	322

12.1 Linux-Standards

12.1.1 Linux Standard Base (LSB)

SUSE unterstützt aktiv die Bemühungen des *Linux Standard Base*-Projekts; aktuelle Informationen dazu unter <http://www.linuxbase.org>.

Die LSB-Spezifikation liegt in der Version 1.3.x vor. Nunmehr ist der Filesystem Hierarchy Standard (FHS) Teil der Spezifikation und es sind u. a. das Paketformat und die Initialisierung des Systems festgelegt; vgl. Kapitel 13 auf Seite 325.

12.1.2 Filesystem Hierarchy Standard (FHS)

SUSE LINUX ist gemäß LSB-Spezifikation konform zum *Filesystem Hierarchy Standard* (FHS, fhs-Paket); vgl. <http://www.pathname.com/fhs/>. Aus diesem Grunde war es bisweilen erforderlich, Dateien oder Verzeichnisse an die richtigen Stelle zu verschieben, wie diese im FHS festgelegt ist.

Ein Ziel des FHS ist es beispielsweise, eine Struktur zu definieren, mit deren Hilfe es möglich ist, `/usr` read-only einzuhängen (mounten).

12.1.3 teTeX — TeX unter SuSE Linux

TeX ist ein komplexes Satzsystem, das auf zahlreichen Plattformen läuft. Es ist über Makro-Pakete wie LaTeX erweiterbar. Es besteht aus sehr vielen einzelnen Dateien, die gemäß der *TeX Directory Structure* (TDS) zusammenzustellen sind (vgl. <ftp://ftp.dante.de/tex-archive/tds/>; teTeX ist eine Zusammenstellung aktueller TeX-Software.

Unter SUSE LINUX kommt teTeX in einer Konfiguration zum Einsatz, die sowohl die Anforderungen des TDS als auch des FHS erfüllt.

12.1.4 Zu FTP

Um die Einrichtung eines FTP-Servers zu erleichtern, hält das Paket `ftplib` eine Beispielumgebung bereit. Diese Umgebung wird unter `/srv/ftp` installiert.

12.1.5 Zu HTTP

Apache ist der Standard-Webserver bei SUSE LINUX. Gleichzeitig mit der Installation des Apache werden Beispiel-Dokumente unter `/srv/www/` zur Verfügung gestellt. Wenn Sie einen eigenen Webserver aufbauen wollen, tragen Sie bitte eine eigene `DocumentRoot` in `/etc/httpd/httpd.conf` ein und legen Sie dort Ihre Dateien (Dokumente, Bilder etc.) ab.

12.2 Hinweise zu speziellen Softwarepaketen

12.2.1 Paket bash und `/etc/profile`

In dieser Reihenfolge wertet die `bash` die Initialisierungsdateien aus, wenn sie als Loginshell aufgerufen wird:

1. `/etc/profile`
2. `~/.profile`
3. `/etc/bash.bashrc`
4. `~/.bashrc`

Eigene Einträge können Benutzer in `~/.profile` bzw. `~/.bashrc` vornehmen. Um ordnungsgemäßes Abarbeiten dieser Dateien zu gewährleisten, ist es erforderlich, dass die aktuellen Grundeinstellungen von `/etc/skel/.profile` bzw. `/etc/skel/.bashrc` in das Benutzerverzeichnis übernommen werden. Nach einem Update empfiehlt sich deshalb, die Einstellungen aus `/etc/skel` zu übernehmen. Um keine eigenen Anpassungen zu verlieren, führen Sie bitte die folgenden Shellbefehle aus:

```
mv ~/.bashrc ~/.bashrc.old
cp /etc/skel/.bashrc ~/.bashrc
mv ~/.profile ~/.profile.old
cp /etc/skel/.profile ~/.profile
```

Danach sind die eigenen Anpassungen aus den Dateien `*.old` zurückzuschreiben.

12.2.2 Paket cron

Die cron-Tabellen liegen unter `/var/spool/cron/tabs`. Als systemweite Tabelle wird die Datei `/etc/crontab` eingerichtet. In der Datei `/etc/crontab` muss zusätzlich nach der Zeitangabe eingetragen werden, unter welchem Benutzer der jeweilige Auftrag ausgeführt werden soll (vgl. Datei 12.1, dort ist `root` angegeben); dem gleichen Format folgen paket-spezifische Tabellen, die in `/etc/cron.d/` liegen – vgl. die Manualpage `man cron`.

Beispiel 12.1: Beispiel eines Eintrags in /etc/crontab

```
1-59/5 * * * * root test -x /usr/sbin/atrun && /usr/sbin/atrun
```

`/etc/crontab` kann *nicht* mit `crontab -e` bearbeitet werden, sondern muss direkt in einen Editor geladen, bearbeitet und gespeichert werden.

Einige Pakete installieren in den Verzeichnissen `/etc/cron.hourly/`, `/etc/cron.daily/`, `/etc/cron.weekly/` und `/etc/cron.monthly/` Shellskripten, deren Abarbeitung von `/usr/lib/cron/run-crons` gesteuert wird. `/usr/lib/cron/run-crons` wird alle 15 Minuten von der Haupt-Tabelle (`/etc/crontab`) aufgerufen. So wird sichergestellt, dass eventuell versäumte Läufe rechtzeitig nachgeholt werden.

Die täglichen Wartungsarbeiten am System sind aus Gründen der Übersichtlichkeit auf mehrere Skripten verteilt worden (Paket `aaa_base`). In `/etc/cron.daily/` gibt es also neben `aaa_base` zum Beispiel die Komponenten `backup-rpmdb`, `clean-tmp` oder `clean-vi`.

12.2.3 Protokoll-Dateien – das Paket logrotate

Zahlreiche System-Dienste (*Daemons*) und auch der Kernel selbst protokollieren regelmäßig Systemzustände oder besondere Vorkommnisse in Protokoll-Dateien (*logfiles*). So kann der Administrator zuverlässig feststellen, in welchem Zustand sich das System zu einem bestimmten Zeitpunkt befand, Fehler oder Fehlfunktionen erkennen und gezielt beheben. Diese Protokoll-Dateien werden in der Regel gemäß FHS unter `/var/log` abgelegt und werden von Tag zu Tag größer. Mit Hilfe von `logrotate` ist es möglich, das Wachsen der Protokoll-Dateien zu steuern.

Umstellung auf logrotate (8.0)

Beim Update einer Version vor SUSE LINUX 8.0 werden alte Einstellungen übernommen:

- Einträge aus `/etc/logfile`, die keinem speziellen Paket zugeordnet sind, werden nach `/etc/logrotate.d/aaa_base` verschoben.
- Die ehemalige `rc.config`-Variable `MAX_DAYS_FOR_LOG_FILES` wird als `dateext` und `maxage` in der Konfigurationsdatei abgebildet; vgl. `man logrotate`.

Konfiguration

In der Konfigurationsdatei `/etc/logrotate.conf` wird das generelle Verhalten festgelegt. Mit der `include`-Angabe wird insbesondere konfiguriert, welche weiteren Dateien ausgewertet werden sollen. Bei SUSE LINUX ist vorgesehen, dass die einzelnen Pakete in `/etc/logrotate.d` Dateien installieren (beispielsweise `syslog` oder `yast`).

Beispiel 12.2: Beispiel für /etc/logrotate.conf

```
# see "man logrotate" for details
# rotate log files weekly weekly
# keep 4 weeks worth of backlogs rotate 4
# create new (empty) log files after rotating old ones create
# uncomment this if you want your log files compressed
#compress
# RPM packages drop log rotation information into this directory
include /etc/logrotate.d
# no packages own lastlog or wtmp - we'll rotate them here
#/var/log/wtmp {
#   monthly
#   create 0664 root utmp
#   rotate 1
#}
# system-specific logs may be also be configured here.
```

`logrotate` selbst wird über `cron` gesteuert und einmal täglich von `/etc/cron.daily/logrotate` angestoßen.

Hinweis

Die Option `create` liest etwaige Einstellungen des Administrator in den Dateien `/etc/permissions*` ein. Stellen Sie bitte sicher, dass es bei eigenen Anpassungen zu keinen Konflikten kommt.

Hinweis

12.2.4 Manual-Pages

Für einige GNU-Programme (zum Beispiel `tar`) werden die Manual-Pages nicht mehr weiter gepflegt. An ihre Stelle treten als Schnellübersicht die `--help`-Ausgabe sowie als ausführliche Handbücher die Info-Dateien. `info` ist GNUs Hypertext-System. Mit `info info` erhält man erste Hilfe zur Benutzung; `info` kann entweder über Emacs `emacs -f info` aufgerufen werden, oder direkt mit dem Befehl `info`. Angenehm zu bedienen sind `tkinfo`, `xinfo` oder der Zugriff über das Hilfesystem.

12.2.5 Der Befehl `ulimit`

Mit dem Befehl `ulimit user limits` ist es möglich, Limits für die Nutzung von Systemressourcen zu setzen, bzw. sich diese anzeigen zu lassen. Insbesondere ist `ulimit` dazu geeignet, den zur Verfügung stehenden Speicher für Anwendungen zu begrenzen. Dadurch wird verhindert, dass eine Anwendung übermäßig viel (allen) Speicherplatz für sich beschlagnahmt und das System zum Stillstand kommt.

Der Aufruf von `ulimit` kann mit verschiedenen Optionen erfolgen. Um den Speicherverbrauch zu begrenzen, sind zum Beispiel die Optionen in Tabelle 12.1 tauglich.

Tabelle 12.1: *ulimit: Ressourcen für den Anwender einstellen*

<code>-m</code>	max. Größe des physikalischen Speichers
<code>-v</code>	max. Größe des virtuellen Speichers
<code>-s</code>	max. Größe des Stacks
<code>-c</code>	max. Größe der Core-Dateien
<code>-a</code>	Anzeige der gesetzten Limits

Systemweit können die Einstellungen in `/etc/profile` vorgenommen werden. Dort muss beispielsweise das Erzeugen von Core-Dateien freigeschaltet werden, die Programmierer zum „Debuggen“ benötigen. Als Anwender kann man die vom Systemadministrator in `/etc/profile` vorgegebenen Werte nicht erhöhen, aber man kann spezielle Einstellung in die eigene `~/ .bashrc` eintragen.

Beispiel 12.3: *ulimit-Einstellungen in `./bashrc`*

```
# Begrenzung des realen Speichers
ulimit -m 98304

# Begrenzung des virtuellen Speichers
ulimit -v 98304
```

Die Speicherangaben müssen in KB gemacht werden. Für detailliertere Informationen werfen Sie bitte einen Blick in die Manualpage `man bash`.

Hinweis

Nicht alle Shells unterstützen `ulimit`-Angaben. Wenn Sie auf übergreifende Einstellungen für derartige Beschränkungen angewiesen sind, dann bietet PAM (zum Beispiel `pam_limits`) weitgehende Einstellungsmöglichkeiten.

Hinweis

12.2.6 Der Befehl `free`

Der Befehl `free` ist etwas irreführend, wenn es darum geht herauszufinden, wie der Arbeitsspeicher gerade verwendet wird ... Informationen findet man in `/proc/meminfo`. Heutzutage sollte sich eigentlich kein Anwender darum Gedanken machen, dem ein modernes Betriebssystem wie Linux zur Verfügung steht. Das Konzept vom „freien Arbeitsspeicher“ datiert von der Zeit her, als es noch keine vereinheitlichte Speicherverwaltung *unified memory management* gab – unter Linux gilt das Motto: „freier Speicher ist schlechter Speicher“ *free memory is bad memory*. Infolgedessen ist Linux immer bestrebt, verschiedene Caches auszubalancieren, nie aber wirklich freien (= ungenutzten) Speicher zuzulassen.

Der Kernel weiß im Grunde nichts direkt von Programmen oder Benutzerdaten. Er verwaltet Programme und Benutzerdaten im so genannten „Page Cache“. Wenn der Speicher knapp wird, werden Teile davon entweder in den Swapbereich oder in die Dateien geschrieben, aus denen sie ursprünglich mit Hilfe des Systemaufrufs `mmap` gelesen wurden; vgl. die Manualpage von `mmap`.

Des Weiteren hält der Kernel auch noch andere Zwischenspeicher, wie den „slab cache“, der zum Beispiel die für den Netzwerkzugriff benutzten Puffer enthält. Dadurch werden eventuelle Differenzen zwischen den Zählern in `/proc/meminfo` erklärt. Die meisten, aber nicht alle, sind über `/proc/slabinfo` abfragbar.

12.2.7 Die Datei `/etc/resolv.conf`

Die Namensauflösung wird über die Datei `/etc/resolv.conf` geregelt; vgl. Abschnitt 14.6 auf Seite 375. Diese Datei wird stets nur von dem Skript `/sbin/modify_resolvconf` aktualisiert. Es ist keinem Programm erlaubt, `/etc/resolv.conf` direkt zu manipulieren. Nur wenn diese Regel beachtet wird, kann sichergestellt werden, dass die Netzwerkkonfiguration und die zugehörigen Daten konsistent gehalten werden.

12.2.8 Einstellungen für GNU Emacs

GNU Emacs ist eine komplexe Arbeitsumgebung. Weiterführende Informationen finden Sie unter <http://www.gnu.org/software/emacs/>.

In den folgenden Absätzen werden die Konfigurationsdateien genannt, die GNU Emacs beim Start abarbeitet. Beim Start liest Emacs mehrere Dateien ein, um gemäß den Vorgaben des Benutzers, des Systemadministrators und oder des Distributors für die jeweilige Bedürfnisse angepasst oder vorkonfiguriert zu werden.

Für jeden Benutzer wird im Home-Verzeichnis die Initialisierungsdatei `~/.emacs` von `/etc/skel/` installiert; `.emacs` wiederum liest die Datei `/etc/skel/.gnu-emacs` ein. Wenn ein Benutzer eigene Anpassungen vornehmen möchte, empfiehlt es sich, diese Datei `.gnu-emacs` in das eigene Home-Verzeichnis zu kopieren und dort die gewünschten Einstellungen vorzunehmen:

```
cp /etc/skel/.gnu-emacs ~/.gnu-emacs
```

In `.gnu-emacs` wird die Datei `~/.gnu-emacs-custom` als `custom-file` festgelegt; wenn der Benutzer mit den `customize-`Möglichkeiten eigene Einstellungen vornimmt, werden diese in `~/.gnu-emacs-custom` gespeichert.

Mit dem Paket `emacs` wird bei SUSE LINUX die Datei `site-start.el` im Verzeichnis `/usr/share/emacs/site-lisp` installiert. Die Datei `site-start.el` wird vor der Initialisierungsdatei `~/.emacs` geladen. `site-start.el` sorgt beispielsweise dafür, dass besondere Konfigurationsdateien automatisch geladen werden, die mit Emacs-Zusatzpaketen der Distribution installiert werden (zum Beispiel Paket `psgml`). Derartige Konfigurationsdateien befinden sich gleichfalls in `/usr/share/emacs/site-lisp` und beginnen stets mit `suse-start-`.

Der lokale Systemadministrator kann in `default.el` systemweite Einstellungen vornehmen. Mehr Informationen zu diesen Dateien finden Sie in der Info-Datei zu Emacs, im Knoten `Init File: info:/emacs/InitFile`. Dort ist auch beschrieben, wie man — falls notwendig — das Laden dieser Dateien verhindern kann.

Die Bestandteile des Emacs' sind auf mehrere Pakete verteilt:

- Basispaket `emacs`
- Dazu ist in der Regel das Paket `emacs-x11` zu installieren, in dem das Programm *mit* X11-Unterstützung enthalten ist.
- Im Paket `emacs-nox` ist das Programm *ohne* X11-Unterstützung enthalten.
- Das Paket `emacs-info` stellt Online-Dokumentation im Info-Format bereit.
- Das Paket `emacs-el` enthält die nicht kompilierten Bibliotheksdateien in Emacs Lisp — zur Laufzeit nicht erforderlich!
- Zahlreiche Zusatzpakete, die nach Bedarf installiert werden können: Paket `emacs-auctex` (für LaTeX); `psgml` (für SGML/XML); `gnuserv` (für Client-/Serverbetrieb) usw.

12.3 Booten mit der initial ramdisk

12.3.1 Problemstellung

Sobald der Linux-Kernel geladen und das Root-Dateisystem (/) gemountet ist, können Programme ausgeführt und weitere Kernel-Module eingebunden werden, um zusätzliche Funktionalitäten bereitzustellen. Um aber das Root-Dateisystem überhaupt mounten zu können, müssen verschiedene Bedingungen erfüllt sein: Der Kernel benötigt die entsprechenden Treiber, um das Gerät ansprechen zu können, auf dem das Root-Dateisystem liegt (insbesondere SCSI-Treiber). Weiter muss der Kernel den Code enthalten, der benötigt wird, um das Dateisystem lesen zu können (`ext2`, `reiserfs`, `romfs` usw.). Weiterhin ist es denkbar, dass bereits das Root-Dateisystem verschlüsselt ist. Zum Mounten ist in diesem Fall die Eingabe des Schlüssels/Passworts erforderlich.

Betrachtet man nur einmal das Problem der SCSI-Treiber, so sind verschiedene Lösungsansätze denkbar: Der Kernel kann alle denkbaren Treiber

enthalten. Dies ist problematisch, da sich die verschiedenen Treiber beißen können. Außerdem wird der Kernel dadurch sehr groß. Eine andere Möglichkeit besteht darin, verschiedene Kernel zur Verfügung zu stellen, die jeweils nur einen oder sehr wenige SCSI-Treiber enthalten. Auch dieser Weg ist problematisch, da er eine sehr große Zahl unterschiedlicher Kernel notwendig macht. Ein Problem, das durch verschieden optimierte Kernel (Athlon-Optimierung, SMP) noch weiter verschärft wird.

Der Ansatz, den SCSI-Treiber als Modul zu laden, führt zur generellen Problematik, der durch das Konzept der *initial ramdisk* begegnet wird: Das Bereitstellen einer Möglichkeit, Userspace-Programme bereits vor dem Mounten des Root-Dateisystems ausführen zu können.

12.3.2 Konzept der initial ramdisk

Die *initial ramdisk* (auch *initdisk* oder *initrd* genannt) löst genau diese oben beschriebenen Probleme. Der Linux-Kernel bietet die Möglichkeit, ein (kleines) Dateisystem in eine Ramdisk zu laden, und darin Programme ausführen zu lassen, bevor das eigentliche Root-Dateisystem gemountet wird. Das Laden der *initrd* wird dabei vom Bootloader (GRUB, LILO usw.) übernommen; all diese Bootloader benötigen lediglich BIOS-Routinen, um Daten vom Bootmedium zu laden. Wenn der Bootloader den Kernel laden kann, kann er auch die *initial ramdisk* laden. Spezielle Treiber sind somit nicht erforderlich.

12.3.3 Ablauf des Bootvorgangs mit initrd

Der Bootloader lädt den Kernel und die *initrd* in den Speicher und startet den Kernel, wobei der Bootloader dem Kernel mitteilt, dass eine *initrd* vorhanden ist und wo im Speicher diese liegt. Ist die *initrd* komprimiert (was typischerweise der Fall ist), so dekomprimiert der Kernel die *initrd* und mountet sie als temporäres Root-Dateisystem. Hierauf wird in der *initrd* ein Programm mit dem Namen *linuxrc* gestartet. Dieses Programm kann nun all die Sachen tun, die erforderlich sind, um das richtige Root-Dateisystem mounten zu können. Sobald *linuxrc* terminiert, wird die (temporäre) *initrd* wieder abgehängt *unmounted* und der Bootvorgang wie gewohnt mit dem Mounten des richtigen Root-Dateisystems fortgeführt. Das Mounten der *initrd* und das Ausführen von *linuxrc* kann somit als ein kurzes Intermezzo während eines normalen Bootvorgangs betrachtet werden. Der Kernel versucht nach dem Booten der tatsächlichen Root-Partition, die *initrd* auf das Verzeichnis */initrd*

umzumounten. Wenn das fehlschlägt, weil zum Beispiel der Mountpunkt `/initrd` nicht vorhanden ist, wird der Kernel versuchen, die `initrd` abzuhängen. Sollte auch dies fehlschlagen, ist das System zwar voll funktionsfähig, jedoch kann der durch die `initrd` belegte Speicher nie freigegeben werden; er steht somit nicht mehr zur Verfügung.

Das Programm `linuxrc`

Für das Programm `linuxrc` in der `initrd` gibt es lediglich die folgenden Anforderungen: Das Programm muss den speziellen Namen `linuxrc` tragen und im Root-Verzeichnis der `initrd` liegen. Abgesehen davon muss es lediglich vom Kernel ausgeführt werden können. Das bedeutet, dass `linuxrc` durchaus dynamisch gelinkt sein darf. In diesem Fall müssen natürlich die shared libraries wie gewohnt vollständig unter `/lib` in der `initrd` verfügbar sein. Weiter darf `linuxrc` auch ein Shellskript sein. In diesem Fall muss natürlich eine Shell in `/bin` existieren. Kurz gesagt, muss die `initrd` ein minimales Linux-System enthalten, das die Ausführung des Programmes `linuxrc` erlaubt. Bei der Installation von SUSE LINUX wird ein statisch gelinktes `linuxrc` verwendet, um die `initrd` so klein wie möglich halten zu können. `linuxrc` wird mit root-Rechten ausgeführt.

Das echte Root-Dateisystem

Sobald `linuxrc` terminiert, wird die `initrd` abgehängt und verworfen, der Bootvorgang geht normal weiter und der Kernel mountet das wirkliche Root-Dateisystem. Was als Root-Dateisystem gemountet werden soll, kann durch `linuxrc` beeinflusst werden. Dazu muss `linuxrc` lediglich das `/proc`-Dateisystem mounten und den Wert des echten Root-Dateisystems in numerischer Form nach `/proc/sys/kernel/real-root-dev` schreiben.

12.3.4 Bootloader

Die meisten Bootloader (vor allem GRUB, LILO und `syslinux`) können mit `initrd` umgehen. Die einzelnen Bootloader werden wie folgt angewiesen, eine `initrd` zu verwenden:

GRUB Eintrag der folgenden Zeile in `/boot/grub/menu.lst`:

```
initrd (hd0,0)/initrd
```

Da die Ladeadresse der `initrd` in das bereits geladene Kernel-Image geschrieben wird, muss der Befehl `initrd` auf den `kernel`-Befehl folgen.

LILO Eintrag der folgenden Zeile in `/etc/lilo.conf`:

```
initrd=/boot/initrd
```

Die Datei `/boot/initrd` ist die *initial ramdisk*. Sie kann komprimiert sein.

syslinux Eintrag der folgenden Zeile in `syslinux.cfg`:

```
append initrd=initrd
```

Weitere Parameter können in der Zeile folgen.

12.3.5 Anwendung von `initrd` bei SUSE

Installation des Systems

Die `initrd` wird bereits seit geraumer Zeit für die Installation verwendet: Bei manueller Installation kann der Anwender in `linuxrc` Kernel-Module laden und die für eine Installation notwendigen Eingaben vornehmen. `linuxrc` startet dann YaST, das die Installation durchführt. Hat YaST seine Arbeit getan, teilt es `linuxrc` mit, wo das Root-Dateisystem des frisch installierten Systems liegt. `linuxrc` schreibt diesen Wert nach `/proc` und führt dann einen Reboot durch. Danach startet YaST erneut und installiert die restlichen Pakete bereits in jenes System, das gerade installiert wird.

Booten des installierten Systems

In der Vergangenheit hat YaST mehr als 40 Kernel für die Installation im System angeboten, wobei sich die Kernel im Wesentlichen dadurch unterschieden, dass jeder Kernel einen bestimmten SCSI-Treiber enthielt. Dies war nötig, um nach dem Booten das Root-Dateisystem mounten zu können. Weitere Treiber konnten dann als Modul nachgeladen werden.

Da inzwischen aber auch optimierte Kernel zur Verfügung gestellt werden, ist dieses Konzept nicht mehr tragbar – es wären inzwischen weit über 100 Kernel-Images nötig.

Daher wird nun auch für das normale Starten des Systems eine `initrd` verwendet. Die Funktionsweise ist analog zu einer Installation. Das hier eingesetzte `linuxrc` ist jedoch einfach nur ein Shellskript, das lediglich die Aufgabe hat, einige vorgegebene Module zu laden. Typischerweise handelt es sich nur um ein einziges Modul, nämlich denjenigen SCSI-Treiber, der benötigt wird, um auf das Root-Dateisystem zugreifen zu können.

Erstellen einer `initrd`

Das Erstellen einer `initrd` erfolgt mittels des Skripts `mkinitrd` (früher `mk_initrd`). Die zu ladenden Module werden bei SUSE LINUX durch die Bezeichner `INITRD_MODULES` in `/etc/sysconfig/kernel` festgelegt. Nach einer Installation wird diese Variable automatisch durch die richtigen Werte vorbelegt (das Installations-`linuxrc` weiß ja, welche Module geladen wurden). Die Module werden in genau der Reihenfolge geladen, in der sie in `INITRD_MODULES` auftauchen. Das ist besonders wichtig, wenn mehrere SCSI-Treiber verwendet werden, da sich ansonsten die Benennung der Platten ändern würde. Streng genommen würde es reichen, nur denjenigen SCSI-Treiber laden zu lassen, der für den Zugriff auf das Root-Dateisystem benötigt wird. Da das automatische Nachladen zusätzlicher SCSI-Treiber jedoch problematisch ist, laden wir alle bei der Installation verwendeten SCSI-Treiber mittels der `initrd`.

Hinweis

Da das Laden der `initrd` durch den Bootloader genauso abläuft wie das Laden des Kernels selbst (LILO vermerkt in seiner `map`-Datei die Lage der Dateien), muss bei der Verwendung von LILO nach jeder Änderung der `initrd` der Bootloader neu installiert werden – bei der Verwendung von GRUB ist dies nicht notwendig!

Hinweis

12.3.6 Mögliche Schwierigkeit – Selbstkompilierte Kernel

Übersetzt man sich selbst einen Kernel, so kann es zu folgendem Problem kommen: Versehentlich wird der SCSI-Treiber fest in den Kernel gelinkt, die bestehende `initrd` bleibt aber unverändert. Beim Booten geschieht Folgendes: Der Kernel enthält bereits den SCSI-Treiber, die Hardware wird erkannt. Die `initrd` versucht nun jedoch, den Treiber nochmals als Modul zu laden. Dies führt bei einigen SCSI-Treibern (insbesondere beim `aic7xxx`) zum Stillstand des Systems. Streng genommen handelt es sich um einen Kernelfehler (ein bereits vorhandener Treiber darf nicht ein zweites Mal als Modul geladen werden können) – das Problem ist bereits im Zusammenhang mit seriellen Treibern bekannt.

Es gibt mehrere Lösungen: Entweder den Treiber als Modul konfigurieren (dann wird er korrekt in der `initrd` geladen) oder aber den Eintrag für die `initrd` aus `/etc/grub/menu.lst` bzw. `/etc/lilo.conf` entfernen. Äquivalent zur letzteren Lösung ist es, den Treiber aus `INITRD_`-

MODULES zu entfernen und `mkinitrd` aufzurufen, das dann feststellt, dass keine `initrd` benötigt wird.

12.3.7 Ausblick

Für die Zukunft ist denkbar, dass eine `initrd` für weitaus mehr (und anspruchsvollere) Dinge verwendet wird als nur für das Laden der Module, die für den Zugriff auf / benötigt werden.

- Root-Dateisystem auf Software RAID (`linuxrc` setzt die `md`-Devices auf)
- Root-Dateisystem auf LVM
- Root-Dateisystem ist verschlüsselt (`linuxrc` fragt nach Passwort)
- Root-Dateisystem auf einer SCSI-Platte am PCMCIA-Adapter

Weitere Informationen

- `/usr/src/linux/Documentation/initrd.txt`
(Nur verfügbar, wenn die Kernel-Quellen installiert wurden)
- Die man-page zu `initrd`.

12.4 linuxrc

`linuxrc` ist ein Programm, das in der Start-Phase des Kernels gestartet wird, bevor richtig gebootet wird. Diese angenehme Eigenschaft des Kernels erlaubt es, einen kleinen modularisierten Kernel zu booten und die wenigen Treiber, die man wirklich braucht, als Module nachzuladen. `linuxrc` hilft bei Bedarf die relevanten Treiber manuell zu laden. Im Regelfall kann jedoch auf die automatische Hardware-Erkennung vertraut werden, die vor dem Start von YaST durchgeführt wird. Sie können `linuxrc` nicht nur bei der Installation verwenden, sondern auch als Boot-Tool für ein installiertes System und sogar für ein autonomes (RAM-Disk basiertes) Rettungssystem. Näheres finden Sie in Abschnitt 12.5 auf Seite 316.

12.4.1 Hauptmenü

Nachdem Sprache und Tastatur eingestellt sind, gelangen Sie in das Hauptmenü von linuxrc (vgl. Abbildung 1.2 auf Seite 11). Normalerweise wird linuxrc benutzt, um Linux zu starten. Ziel ist also der Menüpunkt 'Installation / System starten'. Ob Sie direkt zu diesem Punkt gehen können, hängt von der Hardware des Rechners und dem Installationsvorhaben überhaupt ab. Informationen dazu finden Sie in Abschnitt 1.1 auf Seite 8.

12.4.2 Einstellungen

Einstellungen können bezüglich 'Sprache', 'Bildschirm' (Farbe oder monochrome Darstellung), 'Tastaturbelegung' und 'Debug (Experten)' vorgenommen werden.

12.4.3 System-Information

Unter 'System-Information' (Abbildung 12.1) können Sie neben den Meldungen des Kernels auch einige weitere Einzelheiten überprüfen, etwa die I/O-Adressen von PCI-Karten oder die Größe des Hauptspeichers, die von Linux erkannt wurde.

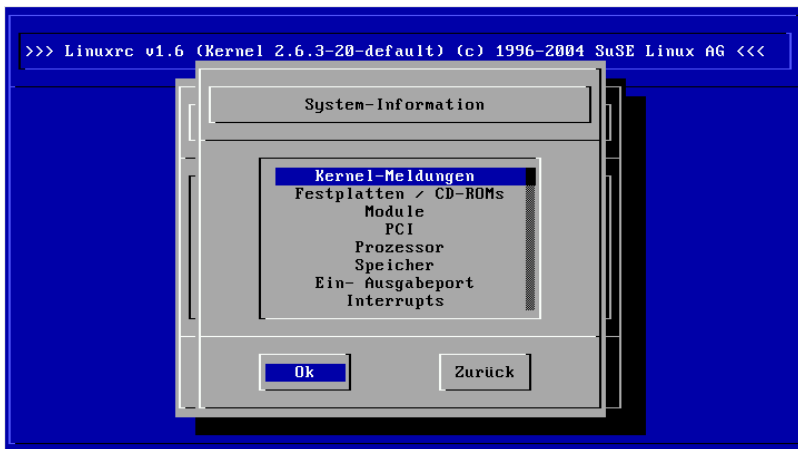


Abbildung 12.1: Systeminformationen

Die folgenden Zeilen zeigen, wie sich eine Festplatte und ein CD-ROM-Laufwerk an einem EIDE-Adapter melden. In diesem Fall müssen Sie keine Kernelmodule für eine Installation laden:

```
hda: ST32140A, 2015MB w/128kB Cache, LBA, CHS=1023/64/63
hdb: CD-ROM CDR-SlG, ATAPI CDROM drive
Partition check:
hda: hda1 hda2 hda3 < hda5 >
```

Haben Sie einen Kernel gestartet, der bereits einen SCSI-Treiber fest integriert hat, brauchen Sie natürlich ebenfalls kein SCSI-Modul mehr zu laden. Typische Meldungen bei Erkennung eines SCSI-Adapters und der daran angeschlossenen Geräte sind:

```
scsi : 1 host.
Started kswpd v 1.4.2.2
scsi0 : target 0 accepting period 100ns offset 8 10.00MHz FAST SCSI-II
scsi0 : setting target 0 to period 100ns offset 8 10.00MHz FAST SCSI-II
Vendor: QUANTUM Model: VP32210 Rev: 81H8
Type: Direct-Access ANSI SCSI revision: 02
Detected scsi disk sda at scsi0, channel 0, id 0, lun 0
scsi0 : target 2 accepting period 236ns offset 8 4.23MHz synchronous SCSI
scsi0 : setting target 2 to period 248ns offset 8 4.03MHz synchronous SCSI
Vendor: TOSHIBA Model: CD-ROM XM-3401TA Rev: 0283
Type: CD-ROM ANSI SCSI revision: 02
scsi : detected 1 SCSI disk total.
SCSI device sda: hdwr sector= 512 bytes. Sectors= 4308352 [2103 MB] [2.1 GB]
Partition check:
sda: sda1 sda2 sda3 sda4 < sda5 sda6 sda7 sda8 >
```

12.4.4 Laden von Modulen

Hier wählen Sie aus, welche Module (Treiber) Sie benötigen. `linuxrc` bietet Ihnen die verfügbaren Treiber in einer Liste an. Links sehen Sie den Namen des zuständigen Moduls, rechts eine Kurzbeschreibung der Hardware, für die der Treiber zuständig ist. Für einige Komponenten gibt es mitunter mehrere Treiber oder neuere Alpha-Treiber. Auch diese werden hier angeboten.

12.4.5 Parametereingabe

Haben Sie den Treiber gefunden, der für Ihre Hardware zuständig ist, drücken Sie `(Return)`. Es erscheint eine Maske, in der Sie etwaige Parameter für das zu ladende Modul eingeben können. Hier sei noch einmal darauf hingewiesen, dass im Gegensatz zur Parametereingabe am Kernel-Prompt mehrere Parameter für das gleiche Modul durch Leerzeichen voneinander getrennt werden müssen.

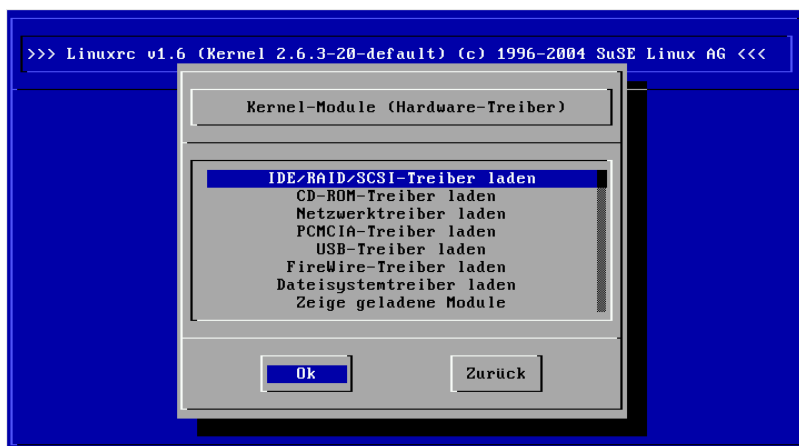


Abbildung 12.2: Module laden

In vielen Fällen ist die genaue Spezifizierung der Hardware gar nicht notwendig, denn die meisten Treiber finden Ihre Komponenten von alleine. Lediglich bei den Netzwerkkarten und bei älteren CD-ROM-Laufwerken mit eigener Controller-Karte ist die Angabe von Parametern mitunter erforderlich. Probieren Sie es jedenfalls erst einmal mit **(Return)**.

Bei einigen Modulen kann das Erkennen und Initialisieren der Hardware recht lange dauern. Durch Umschalten auf die virtuelle Konsole 4 (**(Alt) + (F4)**) können Sie die Meldungen des Kernels während des Ladens beobachten. Vor allem SCSI-Adapter lassen sich etwas Zeit beim Ladevorgang, da sie eine gewisse Zeit warten, bis sich alle angeschlossenen Geräte gemeldet haben.

Wurde das Modul erfolgreich geladen, werden die Meldungen des Kernels von linuxrc angezeigt, sodass Sie sich vergewissern können, dass alles wie vorgesehen gelaufen ist. Ansonsten weisen die Meldungen möglicherweise auf die Ursache des Scheiterns hin.

12.4.6 System / Installation starten

Haben Sie die komplette Kernel-Unterstützung für Ihre Hardware erreicht, können Sie zum Punkt 'System / Installation starten' weitergehen. Von hier aus lassen sich mehrere Vorgänge anstoßen: 'Installation/Update starten',

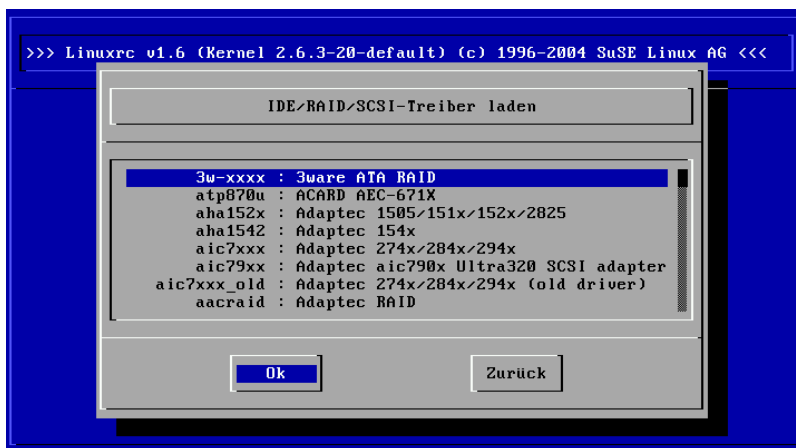


Abbildung 12.3: Auswahl der SCSI-Treiber

‘Installiertes System booten’ (die Rootpartition muss bekannt sein), ‘Rettungssystem starten’ (vgl. Abschnitt 12.5 auf Seite 316) und ‘CD auswerfen’.

Der Punkt ‘LiveEval-CD starten’ steht nur zur Verfügung, wenn Sie von einer „LiveEval-CD“ gebootet haben. ISO-Images können vom FTP-Server heruntergeladen werden (live-eval-`<VERSION>`): `ftp://ftp.suse.com/pub/suse/i386/`

Hinweis

Der Punkt ‘LiveEval-CD starten’ kann zum Beispiel immer dann nützliche Dienste leisten, wenn man *ohne* eigentliche Festplatten-Installation testen möchte, ob der fragliche Rechner oder das anzuschaffende Notebook kompatibel sind.

Hinweis

Für die Installation (Abbildung 12.5 auf Seite 316) und ähnlich auch für das Rettungssystem können Sie verschiedene Quellen wählen (Abbildung 12.6 auf Seite 317).

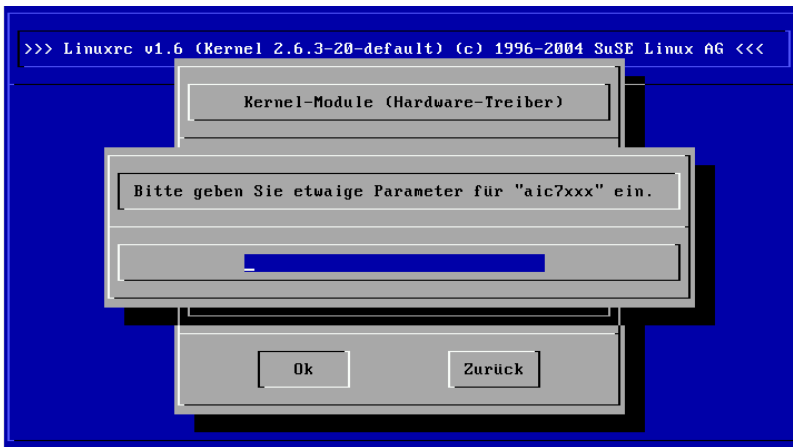


Abbildung 12.4: Eingabe der Parameter für das Laden eines Moduls

12.4.7 Parameter an linuxrc übergeben

Befindet sich linuxrc nicht im manuellen Modus, sucht es nach einer Info-Datei, entweder auf Diskette oder in der `initrd` unter `/info`. Erst danach liest linuxrc die Parameter am Kernel-Prompt ein. Die voreingestellten Werte können in der Datei `/linuxrc.config` verändert werden. Diese wird zuerst eingelesen. Allerdings empfiehlt es sich Änderungen vorzugsweise in der Info-Datei festzulegen.

Eine Info-Datei besteht aus Schlüsselwörtern und zugehörigen Werten der Form: `key: value`. Diese Schlüssel-Wert-Paare können in dieser Form auch am Kernel-Prompt übergeben werden. Eine Liste möglicher Schlüssel finden Sie in der Datei `/usr/share/doc/packages/linuxrc/linuxrc.html`. Einige der wichtigsten werden im Folgenden mit Beispielen aufgeführt:

- Install: URL (nfs, ftp, hd, ...)
- HostIP: 10.10.0.2
- Proxy: 10.10.0.1
- Netdevice: eth0
- Textmode: 0|1

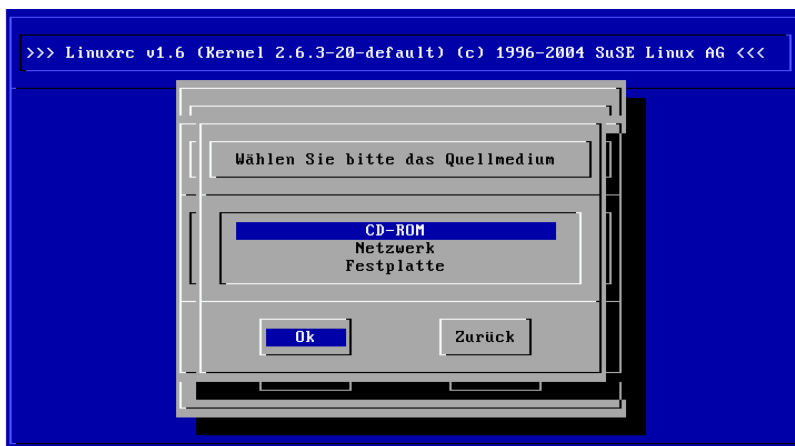


Abbildung 12.5: Auswahl des Quellmediums in linuxrc

- AutoYast: ftp://autoyastfile
- VNC: 0|1
- VNCPasswd: password
- UseSSH: 0|1
- SSHPasswd: password
- ForceInsmod: 0|1 (Benutzen Sie die Option -f, wenn insmod aufgerufen wird).
- Insmod: Modul-Parameter
- AddSwap: 0|3|dev/hda5
Bei 0 wird nie swap angefordert, bei einer positiven Zahl wird die Partition dieser Nummer aktiviert. Alternativ geben Sie den Namen der Partition an.

12.5 Das SUSE Rettungssystem

SUSE LINUX enthält ein Rettungssystem, mit dessen Hilfe Sie in Notfällen von außen auf Ihre Linux-Partitionen zugreifen können: Sie können

das *Rescue-System* von CD, Netzwerk oder vom SUSE-FTP-Server laden. Weiterhin gibt es eine bootbare SUSE LINUX-CD (die *LiveEval-CD*), die als Rettungssystem eingesetzt werden kann. Zum Rettungssystem gehören verschiedene Hilfsprogramme, mit denen Sie Probleme mit unzugänglich gewordenen Festplatten, fehlerhaften Konfigurationsdateien usw. beheben können. Teil des Rettungssystems ist auch Parted (*parted*) zum Verändern der Partitionsgrößen. Es kann bei Bedarf aus dem Rettungssystem heraus manuell aufgerufen werden, falls Sie nicht auf den in YaST integrierten Resizer zurückgreifen wollen. Informationen zu Parted finden Sie unter:

<http://www.gnu.org/software/parted/>

12.5.1 Das Rettungssystem starten

Das Rettungssystem wird von CD (oder DVD) gestartet. Voraussetzung ist, dass das CD-ROM/DVD-Laufwerk bootfähig ist. Gegebenenfalls müssen Sie im BIOS-Setup die Boot-Reihenfolge ändern.

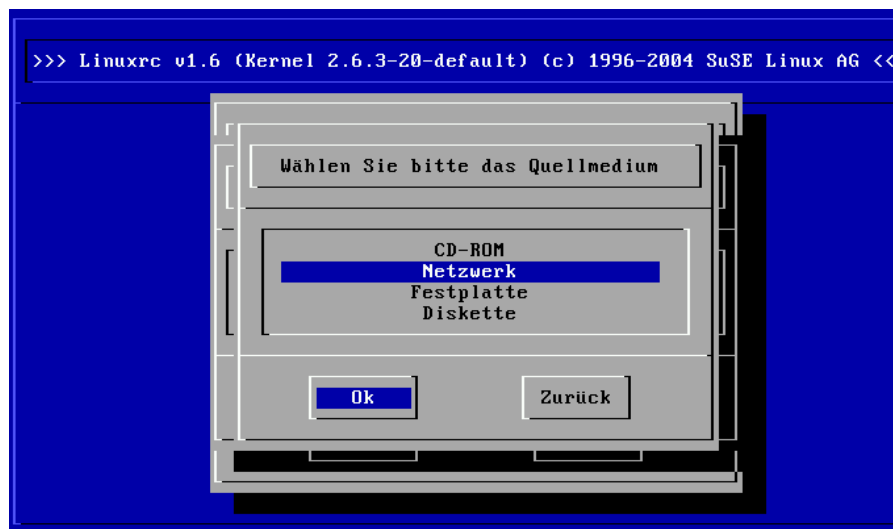


Abbildung 12.6: Quellmedium für das Rettungssystem

Nachfolgend die Schritte zum Starten des Rettungssystems:

1. Legen Sie die erste CD oder DVD von SUSE LINUX in das entsprechende Laufwerk ein und schalten Sie Ihr System ein.
2. Sie können entweder das System durchbooten lassen oder Sie wählen 'Manual Installation' aus, und können dann – falls notwendig – bei 'boot option' spezielle Boot-Parameter angeben.
3. Nehmen Sie im `linuxrc` die erforderlichen Einstellungen für die Sprache und die Tastatur vor.
4. Anschließend können die für ihr System benötigten Kernel-Module geladen werden. Laden Sie hier bitte *alle* Module, von denen Sie glauben, dass sie später im Rettungssystem gebraucht werden. Das Rettungssystem selbst enthält aus Platzgründen fast keine.
5. Wählen Sie im Hauptmenü den Punkt 'Installation/System starten'.
6. Wählen Sie im Menü 'Installation/System starten' den Punkt 'Rettungssystem starten' (s. Abb. 1.3 auf Seite 13) und geben Sie dann das gewünschte Quellmedium an (s. Abb. 12.6 auf der vorherigen Seite).

'CD-ROM' Das Rettungssystem auf der CD-ROM wird verwendet.

'Netzwerk' Das Rettungssystem wird über eine Netzverbindung gestartet. Hierfür muss vorher das richtige Kernel-Modul für die Netzwerkkarte geladen worden sein (vgl. die allgemeinen Hinweise in Abschnitt 1.3.2 auf Seite 17). In einem Untermenü stehen mehrere Protokolle zur Verfügung (s. Abb. 12.7 auf der nächsten Seite): NFS, FTP, SMB etc.

'Festplatte' Sollten Sie vorher schon ein Rettungssystem auf eine aktuell erreichbare Festplatte kopiert haben, können Sie hier angeben wo es liegt. Dieses Rettungssystem wird dann verwendet.

Welches Medium Sie auch gewählt haben, das Rettungssystem wird dekomprimiert, als neues Root-Dateisystem in eine RAM-Disk geladen, gemountet und gestartet. Es ist damit betriebsbereit.

12.5.2 Das Rettungssystem benutzen

Das Rettungssystem stellt Ihnen unter `(Alt) + (F1)` bis `(Alt) + (F3)` mindestens drei virtuelle Konsolen zur Verfügung, an denen Sie sich als Benutzer `root` ohne Passwort einloggen können. Mit `(Alt) + (F10)` kommen Sie zur Systemkonsole mit den Meldungen von Kernel und syslog.



Abbildung 12.7: Netzwerkprotokolle

In dem Verzeichnis `/bin` finden Sie die Shell und Utilities (zum Beispiel `mount`). Wichtige Datei- und Netz-Utilities, zum Beispiel zum Überprüfen und Reparieren von Dateisystemen (`e2fsck`), liegen im Verzeichnis `/sbin`. Des Weiteren finden Sie in diesem Verzeichnis auch die wichtigsten Binaries für die Systemverwaltung wie `fdisk`, `mkfs`, `mkswap`, `init`, `shutdown`, sowie für den Netzbetrieb `ifconfig`, `route` und `netstat`. Als Editor ist der `vi` unter `/usr/bin` verfügbar; hier sind auch weitere Tools (`grep`, `find`, `less` etc.) wie auch das Programm `telnet` zu finden.

Zugriff auf das normale System

Zum Mounten Ihres SUSE LINUX-Systems auf der Platte ist der Mount-point `/mnt` gedacht. Sie können für eigene Zwecke weitere Verzeichnisse erzeugen und als Mount-Punkte verwenden.

Nehmen wir als Beispiel einmal an, Ihr normales System setzt sich laut `/etc/fstab` wie in der Beispieldatei 12.4 beschrieben zusammen.

Beispiel 12.4: Beispiel `/etc/fstab`

<code>/dev/sdb5</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>/dev/sdb3</code>	<code>/</code>	<code>ext2</code>	<code>defaults</code>	<code>1</code>	<code>1</code>
<code>/dev/sdb6</code>	<code>/usr</code>	<code>ext2</code>	<code>defaults</code>	<code>1</code>	<code>2</code>

Achtung

Beachten Sie im folgendem Abschnitt die Reihenfolge, in welcher die einzelnen Geräte zu mounten sind.

Achtung

Um Zugriff auf Ihr gesamtes System zu haben, mounten Sie es Schritt für Schritt unter `/mnt` mit den folgenden Befehlen:

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

Nun haben Sie Zugriff auf Ihr ganzes System und können zum Beispiel Fehler in Konfigurationsdateien wie `/etc/fstab`, `/etc/passwd`, `/etc/inittab` beheben. Die Konfigurationsdateien befinden sich statt im Verzeichnis `/etc` jetzt im Verzeichnis `/mnt/etc`. Um selbst komplett verloren gegangene Partitionen mit dem Programm `fdisk` einfach wieder durch Neu-Anlegen zurückzugewinnen, sollten Sie sich *vorher* einen Ausdruck (Hardcopy) von dem Verzeichnis `/etc/fstab` und dem Output des Befehls `fdisk -l` machen.

Dateisysteme reparieren

Beschädigte Dateisysteme sind ein besonders ernster Anlass für den Griff zum Rettungssystem. Dateisysteme lassen sich grundsätzlich nicht im laufenden Betrieb reparieren. Bei schwereren Schäden lässt sich unter Umständen nicht einmal mehr das Root-Dateisystem mounten und der Systemstart endet in einer `kernel panic`. Dann bleibt nur noch der Weg, die Reparatur von außen unter einem Rettungssystem zu versuchen.

Im SUSE LINUX-Rettungssystem sind die Utilities `e2fsck` und `dumpe2fs` (zur Diagnose) enthalten. Damit beheben Sie die meisten Probleme. Und da auch im Notfall oft die Manual-Page von `e2fsck` nicht mehr zugänglich ist, ist sie im Anhang C auf Seite 603 ausgedruckt.

Beispiel: Wenn sich ein Dateisystem wegen eines *ungültigen Superblocks* nicht mehr mounten lässt, wird das Programm `e2fsck` vermutlich zunächst ebenfalls scheitern. Die Lösung ist, die im Dateisystem alle 8192 Blöcke (8193, 16385...) angelegt und gepflegten Superblock-Backups zu verwenden. Dies leistet zum Beispiel der Befehl:

```
e2fsck -f -b 8193 /dev/<Defekte_Partition>
```

Die Option `-f` erzwingt den Dateisystem-Check und kommt damit dem möglichen Irrtum von `e2fsck` zuvor, es sei – angesichts der intakten Superblock-Kopie – alles in Ordnung.

12.6 Virtuelle Konsolen

Linux ist multitasking- und multiuserfähig. Auch wenn nur Sie selbst an Ihrem Rechner arbeiten, werden Sie die Vorteile, die diese Fähigkeiten mitbringen, zu schätzen lernen.

Im Textmodus stehen sechs virtuelle Konsolen zur Verfügung, zwischen denen Sie über die Tastenkombinationen **(Alt) + (F1)** bis **(Alt) + (F6)** wechseln können. Die siebte Konsole ist für X11 reserviert, die achte für eine weitere X11-Sitzung. Durch Modifikation der Datei `/etc/inittab` können weitere oder weniger Konsolen zur Verfügung gestellt werden. Wenn Sie von X11 aus auf eine Textkonsole zurückschalten möchten, ohne X11 zu beenden, verwenden Sie **(Strg) + (Alt) + (F1)** bis **(Strg) + (Alt) + (F6)**. Mit **(Alt) + (F7)** kommen Sie zu X11 zurück.

12.7 Tastaturbelegung

Um die Tastaturbelegung von Programmen zu vereinheitlichen, wurden Änderungen an den folgenden Dateien vorgenommen:

```
/etc/inputrc
/usr/X11R6/lib/X11/Xmodmap
/etc/skel/.Xmodmap
/etc/skel/.exrc
/etc/skel/.less
/etc/skel/.lesskey
/etc/csh.cshrc
/etc/termcap
/usr/lib/terminfo/x/xterm
/usr/X11R6/lib/X11/app-defaults/XTerm
/usr/share/emacs/<VERSION>/site-lisp/term/*.el
```

Diese Änderungen wirken sich nur auf die Applikationen aus, die die `terminfo`-Einträge auslesen, bzw. deren Konfigurationsdateien direkt verändert wurden (`vi`, `less` etc.). Applikationen die nicht von SUSE stammen, sollten an diese Vorgaben angepasst werden.

Unter X ist die Compose-Taste (Multi_key) über die Tastenkombination **(Strg) + (Shift)** (rechts) zu erreichen. Beachten Sie dabei den entsprechenden Eintrag in `/usr/X11R6/lib/X11/Xmodmap`.

Zu Besonderheiten bei der Eingabe von Chinesisch, Japanisch oder Koreanisch (CJK) finden Sie detaillierte Informationen auf Mike Fabians Seite: <http://www.suse.de/~mfabian/suse-cjk/input.html>.

12.8 Lokale Anpassungen – I18N/L10N

SUSE LINUX ist internationalisiert und kann flexibel auf lokale Gegebenheiten abgestimmt werden. Die Internationalisierung (I18N) erlaubt spezielle Lokalisierungen (L10N). Die Abkürzungen I18N und L10N stehen für *internationalization* und *localization*: jeweils Anfangs- und Endbuchstabe und dazwischen die Anzahl der ausgelassenen Buchstaben.

Die Einstellungen werden über LC_-Variablen vorgenommen, die in der Datei `/etc/sysconfig/language` definiert sind. Dabei geht es nicht nur um die Einstellung der Sprache für die Programmoberfläche und -meldungen *native language support*, sondern im Einzelnen um die Kategorien für *Nachrichten* (Sprache), *Zeichenklassen*, *Sortierreihenfolge*, *Datum und Uhrzeit*, *Zahlen* und *Geld*. Jede dieser Kategorien kann entweder gezielt über eine eigene Variable oder indirekt über eine übergeordnete Variable in der Datei `language` festgelegt werden (vgl. die Manualpage `man locale`).

1. `RC_LC_MESSAGES`, `RC_LC_CTYPE`, `RC_LC_COLLATE`, `RC_LC_TIME`, `RC_LC_NUMERIC`, `RC_LC_MONETARY`: Diese Variablen werden ohne den `RC_`-Vorsatz an die Shell weitergereicht und bestimmen die oben genannten Kategorien; die betroffenen Dateien werden im Folgenden aufgezählt.

Die aktuelle Einstellung kann mit dem Befehl `locale` abgefragt werden.

2. `RC_LC_ALL`: Diese Variable überschreibt, falls gesetzt, die Werte der in Punkt 1 genannten Variablen.
3. `RC_LANG`: Wenn keine der o. g. Variablen gesetzt ist, ist diese der Fallback. SUSE LINUX setzt standardmäßig nur `RC_LANG`; dadurch kann der Anwender leichter eigene Werte eintragen.
4. `ROOT_USES_LANG`: Eine yes/no-Variable. Ist sie auf no gesetzt, dann arbeitet root immer in der POSIX-Umgebung.

Die Variablen sind über den `sysconfig`-Editor zu setzen. Der Wert einer solchen Variablen setzt sich aus Sprachangabe *language code*, Land oder Territorium *country code*, Zeichensatz *encoding* und Option *modifier* zusammen. Die einzelnen Angaben werden mit Spezialzeichen verbunden:

`LANG=⟨language⟩[_⟨COUNTRY⟩].⟨Encoding⟩[@⟨Modifier⟩]`

12.8.1 Einige Beispiele

Bitte setzen Sie die Sprach- und die Länderangabe immer zusammen. Die Angabe der Sprache folgt dem Standard ISO 639 (<http://www.evertype.com/standards/iso639/iso639-en.html> und <http://www.loc.gov/standards/iso639-2/>), die Ländercodes sind in ISO 3166 festgelegt (siehe http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/en_listp1.html). Sinnvollerweise dürfen aber nur die Werte gewählt werden, für die verwendbare Beschreibungsdateien unter `/usr/lib/locale` zu finden sind. Weitere Beschreibungsdateien lassen sich mit Hilfe von `localedef` aus den Dateien in `/usr/share/i18n` erzeugen.

LANG=de_DE.UTF-8 Dies ist die Standardeinstellung, wenn man in deutscher Sprache installiert; installiert man in einer anderen Sprache, wird auch UTF-8 als Zeichen-Kodierung gesetzt, aber die jeweils andere Sprache für das System eingestellt.

LANG=de_DE.ISO-8859-1 So stellt man die deutsche Sprache in Deutschland mit Zeichensatz ISO-8859-1 ein. Dieser Zeichensatz enthält nicht das Euro-Zeichen; man benötigt diesen Zeichensatz bisweilen noch, wenn ein Programm noch nicht an UTF-8 angepasst ist. Die Angabe des Zeichensatzes (hier ISO-8859-1) wertet zum Beispiel der Editor Emacs aus.

LANG=de_DE@euro Dies ist ein Beispiel für das Setzen einer Option (euro).

SUSEconfig liest die Variablen aus `/etc/sysconfig/language` aus und schreibt die Angaben nach `/etc/SUSEconfig/profile` und `/etc/SUSEconfig/csh.cshrc`. `/etc/SUSEconfig/profile` wird von `/etc/profile` eingelesen (gesourcet) und `/etc/SUSEconfig/csh.cshrc` von `/etc/csh.cshrc`. Somit stehen die Einstellungen systemweit zur Verfügung.

Die Benutzer können die Systemvorgaben in `~/ .bashrc` überschreiben. Wenn also die Systemvorgabe `de_DE` ist, kann der Benutzer, falls er mit deutschen Programmmeldungen nicht zufrieden ist, so auf englische Ausgaben umschalten: `LC_MESSAGES=en_US`.

12.8.2 Anpassung für Sprachunterstützung

Dateien der Kategorie *Nachrichten* werden in der Regel nur im Sprachverzeichnis (zum Beispiel `de`) abgelegt, um ein Fallback zu haben. Wenn

man also LANG auf de_AT setzt und die Message-Datei unter /usr/share/locale/de_AT/LC_MESSAGES nicht vorhanden ist, dann wird auf /usr/share/locale/de/LC_MESSAGES zurückgegriffen.

Auch kann man mit LANGUAGE eine Fallbackkaskade festlegen; zum Beispiel für bretonisch -> französisch oder für galizisch -> spanisch -> portugiesisch:

```
LANGUAGE="br_FR:fr_FR"
```

```
LANGUAGE="gl_ES:es_ES:pt_PT"
```

Oder um auf die norwegischen Varieten nynorsk bzw. bokmål auszuweichen (mit zusätzlichem Rückfall auf no):

```
LANG="nn_NO"
```

```
LANGUAGE="nn_NO:nb_NO:no"
```

oder

```
LANG="nb_NO"
```

```
LANGUAGE="nb_NO:nn_NO:no"
```

Bei Norwegisch ist auch zu beachten, dass LC_TIME unterschiedlich behandelt wird.

Mögliche Probleme

Der Tausenderpunkt wird nicht erkannt. Wahrscheinlich steht LANG beispielsweise auf de. Da die Beschreibung, auf die die glibc zurückgreift, in /usr/share/locale/de_DE/LC_NUMERIC zu finden ist, muss beispielsweise LC_NUMERIC auf de_DE gesetzt werden.

Weitere Informationen:

- *The GNU C Library Reference Manual*, Kapitel „Locales and Internationalization“; enthalten im glibc-info.
- Jochen Hein, unter dem Stichwort „NLS“.
- *German-Howto* von Winfried Trümper file:/usr/share/doc/howto/en/html/German-HOWTO.html.
- Markus Kuhn, *UTF-8 and Unicode FAQ for Unix/Linux*, aktuell unter <http://www.cl.cam.ac.uk/~mgk25/unicode.html>.
- *Unicode-Howto* von Bruno Haible file:/usr/share/doc/howto/en/html/Unicode-HOWTO.html.
- *CJK Support in SuSE Linux* auf Englisch von Mike Fabian <http://www.suse.de/~mfabian/suse-cjk/suse-cjk.html>.

Das Bootkonzept

Das Booten und die Initialisierung eines Unix-Systems sind selbst für einem erfahrenen System-Administrator keineswegs trivial. Dieses Kapitel gibt eine kurze Einführung in das Bootkonzept von SUSE LINUX. Die vorliegende Implementierung der Systeminitialisierung setzt den Standard LSB in Version 1.3.x um (vgl. Abschnitt 12.1.1 auf Seite 298).

13.1	Das init-Programm	326
13.2	Die Runlevels	327
13.3	Wechsel des Runlevels	328
13.4	Die Init-Skripten	330
13.5	Der YaST Runlevel-Editor	334
13.6	SuSEconfig und /etc/sysconfig	335
13.7	Der YaST Sysconfig-Editor	337

Mit den lapidaren Worten „Uncompressing Linux...“ übernimmt der Kernel die Kontrolle über die gesamte Hardware des Systems. Er prüft und setzt die Konsole — oder genauer: die BIOS-Register der Graphikkarte und das Ausgabeformat auf den Bildschirm —, um danach die Einstellungen im BIOS zu lesen und die elementaren Schnittstellen des Mainboards zu initialisieren. In den nächsten Schritten „proben“ die einzelnen Treiber — die ja Bestandteil des Kernels sind — die vorhandene Hardware, um sie gegebenenfalls zu initialisieren. Nach dem Überprüfen der Partitionen und dem Mounten des Root-Dateisystems startet der Kernel das Programm `init`. Durch `init` wird das eigentliche System „hochgefahren“ (Unix-Jargon) und die vielen Dienstprogramme und deren Konfiguration werden so gestartet. Der Kernel verwaltet dann das gesamte System; er überwacht Rechenzeit für die einzelnen Programme, er stellt Speicher zur Verfügung und steuert Hardware-Zugriffe.

13.1 Das `init`-Programm

Das Programm `init` ist der für die korrekte Initialisierung des Systems zuständige Prozess; alle Prozesse im System sind also „Kinder“ von `init`.

Unter allen Programmen nimmt `init` eine Sonderrolle ein: `init` wird direkt vom Kernel gestartet und ist immun gegen das Signal 9, mit dem normalerweise jeder Prozess „gekillt“ werden kann. Alle weiteren Prozesse werden entweder von `init` selbst oder von einem seiner „Kindprozesse“ gestartet.

Konfiguriert wird `init` zentral über die Datei `/etc/inittab`; hier werden die so genannten „Runlevels“ definiert (mehr dazu im Abschnitt 13.2 auf der nächsten Seite) und es wird festgelegt, welche Dienste und Daemonen in den einzelnen Levels zur Verfügung stehen sollen. Abhängig von den Einträgen in `/etc/inittab` ruft `init` verschiedene Skripten auf, die aus Gründen der Übersichtlichkeit im Verzeichnis `/etc/init.d/` zusammengefasst sind.

Der gesamte Hochlauf des Systems — und natürlich auch das Herunterfahren — wird somit einzig und allein vom `init`-Prozess gesteuert; insofern lässt sich der Kernel quasi als „Hintergrundprozess“ betrachten, dessen Aufgabe darin besteht, die gestarteten Prozesse zu verwalten, ihnen Rechenzeit zuzuteilen und den Zugriff auf die Hardware zu ermöglichen und zu kontrollieren.

13.2 Die Runlevels

Unter Linux existieren verschiedene *Runlevels*, die den jeweiligen Zustand des Systems definieren. Der Standard-Runlevel, in dem das System beim Booten hochfährt, ist in der Datei `/etc/inittab` durch den Eintrag `initdefault` festgelegt. Für gewöhnlich ist dies 3 oder 5 (siehe Überblick in Tabelle 13.1). Alternativ kann der gewünschte Runlevel beim Booten (zum Beispiel am Boot-Prompt) angegeben werden; der Kernel reicht die Parameter, die er nicht selbst auswertet, unverändert an den `init`-Prozess weiter.

Um zu einem späteren Zeitpunkt in einen anderen Runlevel zu wechseln, kann man `init` mit der Nummer des zugehörigen Runlevels aufrufen; das Wechseln des Runlevels kann nur vom Systemadministrator veranlasst werden. Beispielsweise gelangt man durch das Kommando `init 1` oder `shutdown now` in den Einzelbenutzerbetrieb (engl. *Single user mode*), der der Pflege und Administration des Systems dient. Nachdem der Systemadministrator seine Arbeit beendet hat, kann er durch `init 3` das System wieder in den normalen Runlevel hochfahren lassen, in dem alle für den Betrieb erforderlichen Programme laufen und sich die Benutzer beim System anmelden können. Mit `init 0` oder `shutdown -h now` kann das System angehalten, bzw. durch `init 6` oder `shutdown -r now` zu einem Neustart veranlasst werden.

Hinweis

Runlevel 2 bei NFS gemounteter `/usr/` Partition

Runlevel 2 sollte auf einem System, dessen `/usr/` Partition via NFS gemountet ist, nicht verwendet werden. Die `/usr/` Partition enthält wichtige Programme, die zur reibungslosen Bedienbarkeit des Systems notwendig sind. Da der NFS-Dienst im Runlevel 2 (Lokaler Multiuserbetrieb ohne entferntes Netzwerk) noch nicht zur Verfügung steht, würde Ihr System in seiner Funktion stark beeinträchtigt.

Hinweis

Tabelle 13.1: Liste der gültigen Runlevels unter Linux

Runlevel	Bedeutung
0	Systemhalt (engl. <i>System halt</i>)
S	Einzelbenutzerbetrieb (engl. <i>Single user mode</i>); vom Bootprompt aus mit US-Tastaturbelegung

1	Einzelbenutzerbetrieb (engl. <i>Single user mode</i>)
2	Lokaler Multiuserbetrieb ohne entferntes Netzwerk (engl. <i>Local multiuser without remote network</i>) (d.h. NFS)
3	Voller Multiuserbetrieb mit Netzwerk (engl. <i>Full multiuser with network</i>)
4	Frei (engl. <i>Not used</i>)
5	Voller Multiuserbetrieb mit Netzwerk und KDM (Standard), GDM oder XDM (engl. <i>Full multiuser with network and xdm</i>)
6	Systemneustart (engl. <i>System reboot</i>)

Bei einer Standardinstallation von SUSE LINUX wird normalerweise Runlevel 5 als Standard eingerichtet, so dass sich die Benutzer direkt an der grafischen Oberfläche beim System anmelden können.

Wenn Sie den Runlevel von 3 auf 5 setzen wollen, muss sichergestellt sein, dass das X Window System bereits korrekt konfiguriert ist; (siehe Kapitel 4 auf Seite 81). Ob das System so wie von Ihnen gewünscht funktioniert, testen Sie danach durch Eingabe von `init 5`. Ist dies der Fall, können Sie den Standard-Runlevel über YaST auf 5 ändern.

Achtung

Eigene Änderungen an `/etc/inittab`

Eine fehlerhafte `/etc/inittab` kann dazu führen, dass das System nicht korrekt hochgefahren wird. Gehen Sie bei Veränderungen dieser Datei mit äußerster Sorgfalt vor und behalten Sie immer eine Kopie einer intakten Datei. Zur Behebung des Schadens können Sie versuchen, am Bootprompt den Parameter `init=/bin/sh` zu übergeben, um direkt in eine Shell zu booten und von dort aus die Datei wiederherzustellen. Nach dem Booten spielen Sie mittels `cp` die Backupkopie wieder ein.

Achtung

13.3 Wechsel des Runlevels

Generell passieren bei einem Wechsel des Runlevels folgende Dinge: Die *Stopp-Skripten* des gegenwärtigen Runlevels werden ausgeführt — dabei

werden typischerweise verschiedene, in diesem Level laufende Programme beendet — und die *Start-Skripten* des neuen Runlevels werden ausgeführt. Während eines solchen Vorgangs werden in den meisten Fällen einige Programme gestartet.

Um dies zu verdeutlichen, betrachten wir an einem Beispiel den Wechsel von Runlevel 3 nach Runlevel 5:

- Der Administrator (`root`) teilt dem `init`-Prozess mit, dass der Runlevel gewechselt werden soll. In diesem Fall erreicht er dies durch Eingabe von `init 5`.
- `init` konsultiert die Konfigurationsdatei `/etc/inittab` und stellt fest, dass das Skript `/etc/init.d/rc` mit dem neuen Runlevel als Parameter aufgerufen werden muss.
- Nun ruft `rc` alle Stopp-Skripten des gegenwärtigen Runlevels auf, für die im neuen Runlevel kein Start-Skript existiert; in unserem Beispiel sind das alle Skripten, die sich im Verzeichnis `/etc/init.d/rc3.d/` befinden (der alte Runlevel war 3) und mit einem `K` beginnen. Die nach dem `K` folgende Zahl gewährleistet, dass hierbei eine bestimmte Reihenfolge eingehalten wird, da unter Umständen manche Programme von anderen abhängig sind.

Hinweis

Die Namen der Stopp-Skripten beginnen immer mit `K` (engl. *kill*), die der Start-Skripten mit `S` (engl. *start*).

Hinweis

- Als Letztes werden die Start-Skripten des neuen Runlevels aufgerufen; diese liegen in unserem Beispiel unter `/etc/init.d/rc5.d/` und beginnen mit einem `S`. Auch hierbei wird eine bestimmte Reihenfolge eingehalten, die durch die dem `S` folgende Zahl festgelegt ist.

Wenn Sie in denselben Runlevel wechseln, in dem Sie sich bereits befinden, liest `init` nur die `/etc/inittab` ein, prüft sie auf Veränderungen und nimmt bei Bedarf die entsprechenden Maßnahmen vor, etwa das Starten eines `getty` auf einer weiteren Schnittstelle.

13.4 Die Init-Skripten

Die Skripten unter `/etc/init.d/` unterteilen sich in zwei Kategorien:

- Skripte, die *direkt* von `init` aufgerufen werden: Dies ist nur beim Booten der Fall sowie bei einem sofortigen Herunterfahren des Systems (bei Stromausfall oder durch Drücken der Tastenkombination `(Strg) + (Alt) + (Entf)` durch den Anwender).
- Skripte, die *indirekt* von `init` aufgerufen werden: Das geschieht bei einem Wechsel des Runlevels; es wird generell das übergeordnete Skript `/etc/init.d/rc` ausgeführt, das dafür sorgt, dass die relevanten Skripten in der korrekten Reihenfolge aufgerufen werden.

Alle Skripten befinden sich unter `/etc/init.d/`. Die Skripten für das Wechseln des Runlevels befinden sich ebenfalls in diesem Verzeichnis, werden jedoch grundsätzlich als symbolischer Link aus einem der Unterverzeichnisse `/etc/init.d/rc0.d/` bis `/etc/init.d/rc6.d/` aufgerufen. Dies dient der Übersichtlichkeit und vermeidet, dass Skripten mehrfach vorhanden sein müssen, etwa weil sie in verschiedenen Runlevels verwendet werden sollen. Da jedes dieser Skripten sowohl als Start- wie auch als Stopp-Skript aufgerufen werden kann, müssen sie alle die beiden möglichen Parameter `start` und `stop` verstehen. Zusätzlich verstehen die Skripten die Optionen `restart`, `reload`, `force-reload` und `status`; die Bedeutung aller Optionen ist in Tabelle 13.2 aufgelistet.

Tabelle 13.2: Übersicht der Optionen der init-Skripten

Option	Bedeutung
<code>start</code>	Dienst starten
<code>stop</code>	Dienst stoppen
<code>restart</code>	Dienst stoppen und erneut starten, wenn der Dienst bereits lief; andernfalls den Dienst starten
<code>reload</code>	Konfiguration des Dienstes erneut einlesen, ohne den Dienst zu stoppen und neu zu starten
<code>force-reload</code>	Konfiguration des Dienstes erneut einlesen, wenn der Dienst dies unterstützt; andernfalls wie <code>restart</code>
<code>status</code>	aktuellen Status anzeigen

Die Links in den einzelnen Runlevel-spezifischen Unterverzeichnissen dienen somit also nur dazu, eine Zuordnung der einzelnen Skripten zu bestimmten Runlevels zu ermöglichen. Das Anlegen und Entfernen der notwendigen Links geschieht mit `insserv` (bzw. dem Link `/usr/lib/lsh/install_initd`) beim Installieren oder Deinstallieren des jeweiligen Paketes; vgl. die Manualpage von `insserv`.

Im Folgenden finden Sie eine kurze Beschreibung der ersten Boot- und der letzten Shutdown-Skripten sowie des Steuerskripts:

boot Wird beim Start des Systems ausgeführt und direkt von `init` gestartet. Es ist unabhängig vom gewünschten Default-Runlevel und wird nur genau ein einziges Mal ausgeführt: Im Wesentlichen werden `proc`- und `pts`-Dateisystem eingehängt („gemountet“), der `blogd` (engl. *Boot Logging Daemon*) wird aktiviert und — nach einer Erstinstallation oder einem Update des Systems — wird eine Grundkonfiguration angestoßen.

Der `blogd` Daemon ist ein Daemon, der vom `boot` und `rc` Skript vor allem anderen gestartet wird und nach getaner Arbeit (zum Beispiel dem Aufruf von Unterskripten) wieder beendet wird. Dieser Daemon schreibt in die Log-Datei `/var/log/boot.msg`, falls `/var/` les- und schreibbar gemountet ist bzw. puffert alle Bildschirmdaten bis das `/var/` les- und schreibbar gemountet wird. Weitere Informationen zu `blogd` finden Sie unter `man blogd`.

Diesem Skript ist des Weiteren das Verzeichnis `/etc/init.d/boot.d/` zugeordnet; alle in diesem Verzeichnis gefundenen Skripte, die mit `s` beginnen, werden automatisch beim Hochlauf des Systems ausgeführt. Es werden die Dateisysteme geprüft, etwaige überflüssige Dateien unter `/var/lock/` gelöscht und das Netzwerk für das Loopback-Device konfiguriert, sofern dies vorgesehen ist. Weiterhin wird die Systemzeit gesetzt.

Tritt beim Prüfen und automatischen Reparieren der Dateisysteme ein schwerer Fehler auf, hat der Systemadministrator nach Eingabe des Root-Passwortes die Möglichkeit, manuell eine Lösung des Problems herbeizuführen. Schließlich wird das Skript `boot.local` ausgeführt.

boot.local Hier können weitere Dinge eingetragen werden, die beim Start geschehen sollen, bevor das System in einen der Runlevels eintritt; es kann von seiner Funktion her mit der vielleicht von DOS her gewohnten `AUTOEXEC.BAT` verglichen werden.

- boot.setup** Grundlegende Einstellungen, die beim Übergang vom Einzelnutzerbetrieb in irgendeinen Runlevel vorgenommen werden müssen. Hier werden die Tastaturbelegung und die Konsolenkonfiguration geladen.
- halt** Dieses Skript wird nur beim Eintritt in den Runlevel 0 oder 6 ausgeführt. Dabei wird es entweder unter dem Namen `halt` oder dem Namen `reboot` aufgerufen. Abhängig davon, wie `halt` aufgerufen wurde, wird das System neu gestartet oder ganz heruntergefahren.
- rc** Das übergeordnete Skript, das bei jedem Wechsel des Runlevels aufgerufen wird. Es führt die Stopp-Skripten des gegenwärtigen Runlevels aus und danach die Start-Skripten des neuen.

13.4.1 Init-Skripten hinzufügen

Zusätzliche Init-Skripten lassen sich in das oben beschriebene Konzept leicht integrieren. Orientieren Sie sich bei Fragen zum Format, Namensgebung und Organisation der Init-Skripten an den Vorgaben des LSB und den Manualpages von `init`, `init.d/` und `insserv`. Hilfreich sind in diesem Zusammenhang weiterhin die Manualpages von `startproc` und `killproc`.

Achtung

Erstellung eigener Init-Skripten

Fehlerhafte Init-Skripten können das gesamte System „aufhängen“. Erstellen Sie eigene Skripte mit äußerster Sorgfalt und testen Sie sie — soweit möglich — vor dem Ernstfall in der Multiuserumgebung. Grundlageninformation zum Umgang mit Runleveln/Init-Skripten finden Sie im Abschnitt 13.2 auf Seite 327.

Achtung

- Wenn Sie für ein eigenes Programm oder eigenen Dienst (engl. *service*) ein Init-Skript erstellen, verwenden Sie die Datei `/etc/init.d/skeleton` als Vorlage. Speichern Sie diese Datei unter dem neuen Namen und editieren Sie die Nennung von Programm- oder Dateinamen und Pfaden und fügen Sie, wenn nötig, eigene Skriptbestandteile hinzu, die für ein korrektes Ausführen des Init-Aufrufes benötigt werden.
- Editieren Sie den obligatorischen `INIT INFO` Block am Anfang der Datei:

Beispiel 13.1: Eine minimale INIT INFO

```
### BEGIN INIT INFO
# Provides:          FOO
# Required-Start:    $syslog $remote_fs
# Required-Stop:     $syslog $remote_fs
# Default-Start:     3 5
# Default-Stop:      0 1 2 6
# Description:       Start FOO to allow XY and provide YZ
### END INIT INFO
```

In der ersten Zeile des INFO-Headers nennen Sie nach `Provides:` den Namen des Programms oder Dienstes, der mit diesem Init-Skript gesteuert werden soll. `Required-Start:` und `Required-Stop:` enthalten alle Dienste, die vor dem Start oder Stopp des betroffenen Dienstes oder Programms gestartet oder gestoppt werden müssen. Diese Information wird ausgewertet, um die Numerierung der resultierenden Start- und Stoppskripten in den Runlevel-Verzeichnissen zu generieren. Die Runlevel, in denen Ihre Anwendung automatisch gestartet bzw. gestoppt werden sollen, geben Sie bei `Default-Start:` und `Default-Stop:` an. Mit einer kurzen Beschreibung Ihrer Anwendung unter `Description:` schließen Sie Ihre Eingaben ab.

- Legen Sie mit dem Befehl `insserv <Name des neuen Skripts>` die Links von `/etc/init.d/` in die entsprechenden Runlevelverzeichnisse (`/etc/init.d/rc?.d/`) an. `insserv` wertet automatisch die im Header des Init-Skripts gemachten Angaben aus und legt die Links für Start- und Stoppskripte in den entsprechenden Runlevelverzeichnissen ab. Die korrekte Start- und Stoppreihenfolge innerhalb eines Runlevels wird ebenfalls über die Nummerierung der Skripte von `insserv` gewährleistet. Als grafisches Konfigurationswerkzeug zum Anlegen der Links steht der Runlevel-Editor von YaST zur Verfügung; vgl. Abschnitt 13.5 auf der nächsten Seite.

Soll lediglich ein in `/etc/init.d/` bereits vorliegendes Skript in das Runlevel-Konzept eingebunden werden, legen Sie mittels `insserv` oder dem YaST-Runlevel-Editor die Links in die entsprechenden Runlevelverzeichnisse an und aktivieren den Dienst. Beim nächsten Start des Systems werden Ihre Änderungen umgesetzt und der neue Dienst automatisch gestartet.

13.5 Der YaST Runlevel-Editor

Nach dem Start dieses Moduls gelangen Sie in eine Übersichtsmaske, die alle verfügbaren Dienste und deren Aktivierungszustand wiedergibt. Entscheiden Sie sich per Radiobutton für einen der beiden Modi 'Einfacher Modus' oder 'Experten-Modus'. Voreingestellt und für die meisten Anwendungssituationen ausreichend ist der 'Einfache Modus'. In einer tabellarischen Übersicht sind alphabetisch geordnet alle Dienste und Daemonen aufgelistet, die auf Ihrem System zur Verfügung stehen. In der linken Spalte stehen die Namen der Dienste, in der Mitte ihr Aktivierungszustand und in der rechten Spalte eine kurze Beschreibung. Unterhalb der Übersicht wird zum aktuell selektierten Dienst eine ausführlichere Beschreibung eingeblendet. Um einen Dienst zu aktivieren, selektieren Sie ihn in der Übersicht und klicken auf 'Aktivieren'. Entsprechend gehen Sie vor, um aktive Dienste zu deaktivieren.

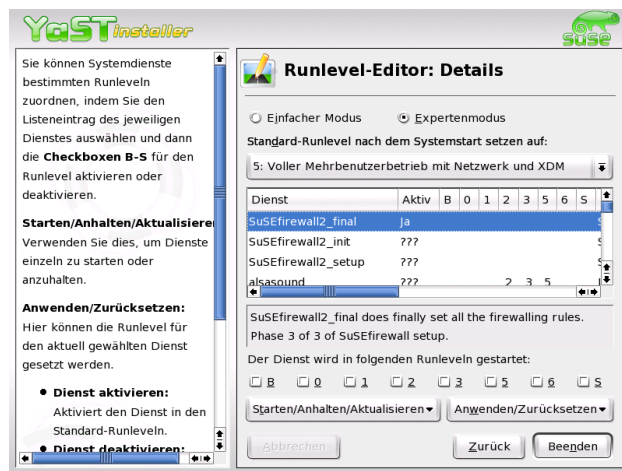


Abbildung 13.1: YaST Runlevel-Editor

Möchten Sie gezielt den Runlevel beeinflussen, in dem ein Dienst gestartet oder gestoppt werden soll oder den Standard-Runlevel verändern, wechseln Sie per Radiobutton in den 'Expertenmodus'. In dieser Maske wird zuerst der aktuelle Standard-Runlevel angezeigt. Dieser „Betriebsmodus“ wird nach dem Booten Ihres Systems hochgefahren. Bei SUSE LINUX ist dies üblicherweise Runlevel 5 (voller Multiuserbetrieb mit Netzwerk und

XDM). Geeignet wäre zum Beispiel auch Runlevel 3 (voller Multiuserbetrieb mit Netzwerk). An dieser Stelle lässt sich mit Hilfe von YaST ein anderer Default-Runlevel einstellen; vgl. Tabelle 13.1 auf Seite 327. Die De-/Aktivierung von Diensten und Daemonen geschieht über die tabellarische Übersicht. Sie erhalten dort Information darüber, welche Dienste und Daemonen vorhanden sind, ob diese in Ihrem System aktiv geschaltet sind und für welche Runlevels dies gilt. Wenn Sie eine Zeile per Mausklick markieren, haben Sie die Möglichkeit, die Checkboxen für die Runlevels 'B', '0', '1', '2', '3', '5', '6' und 'S' zu aktivieren und damit festzulegen, für welche Runlevels der entsprechende Dienst bzw. Daemon aktiv werden soll. Runlevel 4 ist nicht definiert — dieser ist stets frei für benutzereigene Einstellungen. Unmittelbar unterhalb der Übersicht wird eine kurze Beschreibung des jeweils selektierten Dienstes oder Daemons angezeigt.

Mit 'Starten/Anhalten/Aktualisieren' entscheiden Sie, ob ein Dienst eingesetzt werden soll. Mit 'Status aktualisieren' sind Sie in der Lage, den aktuellen Status zu prüfen, falls dies nicht automatisch geschieht. Über 'Anwenden/Zurücksetzen' selektieren Sie, ob der von Ihnen konfigurierte Zustand übernommen werden soll oder der Ausgangszustand vor Aufruf des Runlevel-Editors wiederhergestellt werden soll. Mit 'Beenden' speichern Sie die Systemkonfiguration.

Achtung

Editieren der Runlevel-Einstellungen

Fehlerhafte Einstellungen von Systemdiensten und Runleveln können Ihr System unbrauchbar machen. Informieren Sie sich vor einer Änderung dieser Einstellungen über die möglichen Folgen, um die Funktionsfähigkeit Ihres Systems zu gewährleisten.

Achtung

13.6 SuSEconfig und /etc/sysconfig

Die wesentliche Konfiguration von SUSE LINUX nehmen Sie über die Konfigurationsdateien unter `/etc/sysconfig/` vor. Frühere Versionen von SUSE LINUX verwendeten zur Systemkonfiguration die Datei `/etc/rc.config/`, die mittlerweile obsolet wurde. Bei einer Neuinstallation von SUSE LINUX wird diese Datei nicht mehr angelegt. Sämtliche Systemkonfiguration wird über die Dateien unter `/etc/sysconfig/` vorgenommen. Bei einem Update bleibt eine bestehende `/etc/rc.config/` jedoch erhalten.

Auf die Dateien in `/etc/sysconfig/` wird nur gezielt von einzelnen Skripten zugegriffen; dadurch wird gewährleistet, dass zum Beispiel die Netzwerkeinstellungen auch nur von dem Netzwerk-Skripten ausgewertet werden müssen. Darüber hinaus werden viele weitere Konfigurationsdateien des Systems in Abhängigkeit von den Dateien in `/etc/sysconfig/` generiert; diese Aufgabe erledigt `SuSEconfig`. So wird beispielsweise nach einer Änderung der Netzwerkkonfiguration die Datei `/etc/host.conf` neu erzeugt, da sie abhängig von der Art der Konfiguration ist.

Wenn Sie Änderungen an den genannten Dateien vornehmen, müssen Sie nachfolgend immer `SuSEconfig` aufrufen, so dass die neuen Einstellungen auch an allen relevanten Stellen wirksam werden. Verändern Sie die Konfiguration mit dem YaST Sysconfig-Editor, brauchen Sie sich darum nicht explizit zu kümmern; YaST startet automatisch `SuSEconfig`, wodurch die betroffenen Dateien auf den aktuellen Stand gebracht werden.

Dieses Konzept ermöglicht es, grundlegende Änderungen an der Konfiguration des Rechners vornehmen zu können, ohne die Maschine neu booten zu müssen. Da manche Einstellungen sehr tief greifend sind, müssen jedoch unter Umständen einige Programme neu gestartet werden, um die Änderungen wirksam werden zu lassen.

Wenn Sie zum Beispiel Änderungen an der Netzwerkkonfiguration vorgenommen haben, erreichen Sie durch manuelles Ausführen der Kommandos `rcnetwork stop` und `rcnetwork start`, dass die betroffenen Netzwerk-Programme neu gestartet werden.

Für das Konfigurieren des Systems ist folgender Weg zu empfehlen:

- Bringen Sie das System durch Eingabe von `init 1` in den *single user mode* (Runlevel 1).
- Nehmen Sie die gewünschten Änderungen an den Konfigurationsdateien vor. Dies entweder kann mit einem Texteditor geschehen oder besser mit dem Sysconfig-Editor von YaST; vgl. in Abschnitt 13.7 auf der nächsten Seite.

Achtung

Manuelles Editieren der Systemkonfiguration

Wenn Sie die Konfigurationsdateien in `/etc/sysconfig/` *nicht* mit YaST bearbeiten, achten Sie darauf, dass Sie einen leeren Parameter als zwei aufeinander folgende Anführungszeichen schreiben (zum Beispiel `KEYTABLE=" "`) und Parameter, die Leerzeichen enthalten, in Anführungsstriche einschließen. Bei Variablen, die nur aus einem Wort bestehen, sind die Anführungszeichen nicht notwendig.

Achtung

- Führen Sie `SuSEconfig` aus, um die Änderungen in die verschiedenen weiteren Konfigurationsdateien eintragen zu lassen. Dies geschieht automatisch, wenn Sie YaST verwendet haben, um den Runlevel zu setzen.
- Bringen Sie das System durch Eingabe von `init 3` in den vorherigen Runlevel zurück (hier im Beispiel 3).

Diese Vorgehensweise ist natürlich nur bei sehr weitreichenden Änderungen an den Einstellungen des Systems erforderlich (zum Beispiel Netzwerkkonfiguration); bei einfachen Aufgaben ist es nicht erforderlich, für die Administration in den „single user mode“ zu wechseln; jedoch stellen Sie damit sicher, dass auch wirklich alle von der Änderung betroffenen Programme neu gestartet werden.

Hinweis

Sie können die automatische Konfiguration via `SuSEconfig global` abschalten, indem Sie die Variable `ENABLE_SUSECONFIG` in `/etc/sysconfig/suseconfig` auf `no` setzen. Wollen Sie den Installationssupport in Anspruch nehmen, muss `ENABLE_SUSECONFIG` allerdings auf `yes` gesetzt sein. Einzelne Teile der Autokonfiguration können auch gezielt deaktiviert werden.

Hinweis

13.7 Der YaST Sysconfig-Editor

Im Verzeichnis `/etc/sysconfig/` sind die Dateien mit den wichtigsten Einstellungen für SUSE LINUX hinterlegt. Der YaST Sysconfig-Editor stellt

alle Einstellmöglichkeiten übersichtlich dar. Die Werte können geändert und anschließend in die einzelnen Konfigurationsdateien übernommen werden. Im Allgemeinen ist das manuelle Editieren allerdings nicht notwendig, da bei der Installation eines Paketes oder beim Einrichten eines Dienstes etc. die Dateien automatisch angepasst werden.

Achtung

Änderungen in den `/etc/sysconfig/*`-Dateien

Ihre Änderungen in `/etc/sysconfig/*` haben tief greifende Folgen für Ihr gesamtes System. Bitte informieren Sie sich vor jeder Änderung ausreichend über die möglichen Folgen. So stellen Sie sicher, dass Ihr System funktionsfähig bleibt. Sämtliche Sysconfig-Variablen in den `/etc/sysconfig/`-Dateien sind mit kurzen Kommentaren versehen, die die Funktion der jeweiligen Variablen dokumentieren.

Achtung

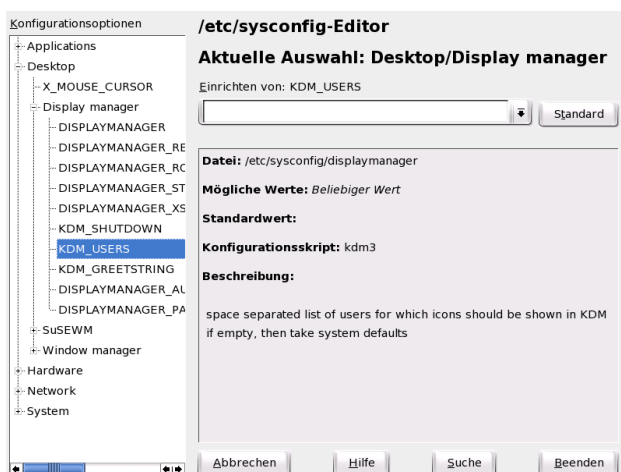


Abbildung 13.2: YaST: Systemkonfiguration mit dem Sysconfig-Editor

Der YaST Sysconfig-Editor startet mit einer in drei Teilbereiche gegliederten Maske. Im linken Teil der Maske können Sie in einer Baumansicht die zu konfigurierende Variable selektieren. Sobald Sie eine Variable selektieren, erscheint in der rechten Fensterhälfte die Bezeichnung der Selektion und die derzeit aktive Einstellung der Variablen. Unterhalb der Variablen wird

eine kurze Beschreibung, möglichen Werte, die Standardeinstellung, sowie die Datei, in der diese Variable gespeichert wird, angezeigt. Weiterhin wird in dieser Maske angezeigt, welches Konfigurationsskript bei Änderung dieser Variablen ausgeführt wird und welcher Dienst neu gestartet wird. YaST bittet Sie um eine Bestätigung der Änderungen und informiert Sie, welche Skripte im Anschluss an ein Verlassen des Moduls mit 'Beenden' ausgeführt werden sollen. Sie haben die Möglichkeit, das Starten bestimmter Dienste und Skripte zu überspringen, wenn Sie sie an dieser Stelle noch nicht starten wollen.

Teil IV

Netzwerk

Grundlagen der Vernetzung

Linux, ein wahres Kind des Internets, bietet Ihnen alle Voraussetzungen und notwendigen Netzwerktools zur Einbindung in diverse Netzwerkstrukturen. Im folgenden erhalten Sie eine Einführung in das normalerweise von Linux verwendete Protokoll TCP/IP, dessen Dienstleistungen und auch besonderen Eigenschaften. Anschließend zeigen wir Ihnen die Einrichtung eines Netzwerkzugangs mit einer Netzwerkkarte unter SUSE LINUX mit Hilfe von YaST. Es werden die zentralen Konfigurationsdateien besprochen und einige der wichtigsten Tools aufgeführt. Da die Konfiguration eines Netzwerks beliebig komplex sein kann, werden in diesem Kapitel nur die grundlegenden Mechanismen dargestellt.

Auch die Internet-Anbindung per PPP via Modem, ISDN oder DSL lässt sich mit YaST bequem konfigurieren und ist im *Benutzer-Handbuch* erklärt.

14.1	TCP/IP – Das von Linux verwendete Protokoll . . .	344
14.2	IPv6 – Internet der nächsten Generation	353
14.3	Manuelle Netzwerkkonfiguration	362
14.4	Die Einbindung ins Netzwerk	370
14.5	Routing unter SuSE Linux	374
14.6	DNS – Domain Name System	375
14.7	LDAP – Ein Verzeichnisdienst	387
14.8	NIS – Network Information Service	404
14.9	NFS – verteilte Dateisysteme	408
14.10	DHCP	413
14.11	Zeitsynchronisation mit xntp	419

14.1 TCP/IP – Das von Linux verwendete Protokoll

Linux und andere Unix-Betriebssysteme verwenden das TCP/IP-Protokoll. Genau genommen handelt es sich um eine Protokollfamilie, die ganz unterschiedliche Dienstleistungen bietet. TCP/IP wurde aus einer militärischen Anwendung heraus entwickelt und in der heute verwendeten Form ca. 1981 in einem so genannten RFC festgelegt. Bei RFC (engl. *Request for comments*) handelt es sich um Dokumente, die die verschiedenen Internetprotokolle und die Vorgehensweise bei der Implementierung des Betriebssystems und von Applikationen beschreiben. Auf diese RFC-Dokumente können Sie direkt über das Web zugreifen, die URL lautet <http://www.ietf.org/>. In der Zwischenzeit sind einige Verfeinerungen am TCP/IP Protokoll vorgenommen worden, am grundlegenden Protokoll hat sich seit 1981 aber nichts geändert.

Hinweis

Die RFC-Dokumente beschreiben den Aufbau der Internet Protokolle. Falls Sie Ihr Know-how über ein bestimmtes Protokoll vertiefen wollen, ist das passende RFC-Dokument die richtige Anlaufstelle: <http://www.ietf.org/rfc.html>

Hinweis

Die in Tabelle 14.1 genannten Dienste stehen zur Verfügung, um Daten zwischen zwei Linuxrechnern über TCP/IP auszutauschen:

Tabelle 14.1: Verschiedene Protokolle der TCP/IP Protokollfamilie

Protokoll	Beschreibung
TCP	(engl. <i>Transmission control protocol</i>) Ein verbindungsorientiertes, gesichertes Protokoll. Die zu übertragenden Daten werden aus der Sicht der Applikation als Datenstrom verschickt und vom Betriebssystem selbst in das passende Übertragungsformat gebracht. Die Daten kommen bei der Zielapplikation auf dem Zielrechner als exakt der Datenstrom an, als der sie abgeschickt wurden. TCP stellt sicher, dass unterwegs keine Daten verloren gehen und nichts durcheinander kommt. TCP wird dort verwendet, wo die Reihenfolge der Daten wichtig ist und der Begriff Verbindung Sinn macht.

UDP	(engl. <i>User Datagram protocol</i>) Ein verbindungsloses, ungesichertes Protokoll. Die zu übertragenden Daten werden paketorientiert verschickt, die Datenpakete werden dabei schon von der Applikation erzeugt. Die Reihenfolge der Daten beim Empfänger ist nicht garantiert, ebenso kann es passieren, dass einzelne Datenpakete verloren gehen. UDP eignet sich für datensatzorientierte Applikationen und bietet kleinere Latenzzeiten als TCP.
ICMP	(engl. <i>Internet Control Message Protocol</i>) Im Wesentlichen ist das kein für den Benutzer verwendbares Protokoll, sondern ein spezielles Steuerprotokoll, das Fehlerzustände übermittelt und das Verhalten der an der TCP/IP-Datenübertragung beteiligten Rechner steuern kann. Zusätzlich wird durch ICMP noch ein spezieller Echo-Modus bereitgestellt, den man mit dem Programm ping prüfen kann.
IGMP	(engl. <i>Internet group management protocol</i>) Dieses Protokoll steuert das Verhalten von Rechnern bei der Verwendung von IP-Multicast. Leider kann IP-Multicasting in diesem Rahmen nicht vorgestellt werden.

Fast alle Hardwareprotokolle arbeiten paketorientiert. Die zu übertragenden Daten müssen in kleine „Päckchen“ gepackt werden und können nicht „in einem Rutsch“ verschickt werden. Deshalb arbeitet auch TCP/IP mit kleinen Datenpaketen. Die Maximalgröße eines TCP/IP Paketes ist knapp 64 Kilobyte. In der Praxis sind die Pakete normalerweise viel kleiner, da die Netzwerkhardware der limitierende Faktor ist. So ist die zulässige Maximalgröße eines Datenpaketes auf dem Ethernet ca. 1500 Byte. Dementsprechend wird die Paketgröße des TCP/IP Pakets begrenzt, wenn die Daten über ein Ethernet geschickt werden. Will man mehr Daten übertragen, müssen vom Betriebssystem entsprechend mehr Datenpakete verschickt werden.

14.1.1 Schichtenmodell

Über IP (engl. *Internet protocol*) findet eine ungesicherte Datenübertragung statt. TCP (engl. *Transmission control protocol*) ist gewissermaßen nur ein Aufsatz auf das darunter liegende IP, um eine gesicherte Übertragung der Daten zu garantieren. IP selbst ist wiederum ein Aufsatz auf das

darunter liegende, hardwareabhängige Protokoll, zum Beispiel Ethernet. Kenner sprechen hier vom „Schichtenmodell“. Vergleichen Sie hierzu die Abbildung 14.1.

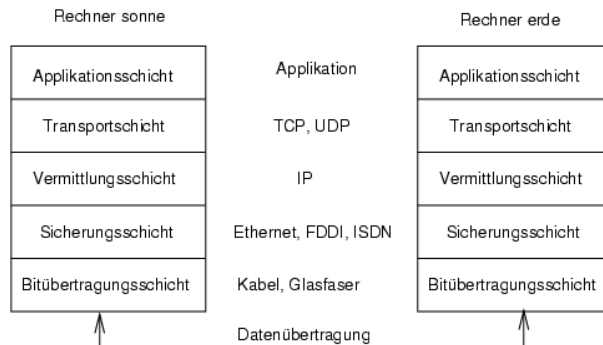


Abbildung 14.1: Vereinfachtes Schichtenmodell für TCP/IP

In der Abbildung sind jeweils ein oder zwei Beispiele für die jeweilige Schicht erwähnt. Wie Sie sehen, sind die Schichten nach „Abstraktions-ebenen“ geordnet, die unterste Schicht ist sehr nah an der Hardware. Die oberste Schicht hingegen abstrahiert die darunter liegende Hardware nahezu vollständig. Jede der Schichten hat eine ganz spezielle Funktion, die zum Großteil schon aus der Bezeichnung hervorgeht. So wird das verwendete Netzwerk (zum Beispiel Ethernet) durch die Bitübertragungsschicht und die Sicherungsschicht verkörpert.

- Während sich Schicht 1 mit solchen Dingen wie Kabeltypen, Signalformen, Signalkodierung und ähnlichem beschäftigt ist Schicht 2 für das Zugriffsverfahren (Welcher Rechner darf wann Daten schicken?) und eine Fehlerkorrektur (Datensicherung - deshalb *Sicherungsschicht*) zuständig. Die Schicht 1 nennt man die *Bitübertragungsschicht*.
- Schicht 3 wiederum, die *Vermittlungsschicht* ist für die Datenübertragung über weite Strecken verantwortlich. Die Vermittlungsschicht stellt sicher, dass die Daten auch über weite Strecken beim richtigen Empfänger ankommen und zugestellt werden können.
- Schicht 4, die *Transportschicht*, ist für die Daten der Applikation verantwortlich und stellt sicher, dass die Daten in der richtigen Reihenfolge ankommen und nicht verloren gehen. Die Sicherungsschicht ist

nur dafür verantwortlich, dass die ankommenden Daten korrekt sind. Gegen das „Verlieren“ von Daten schützt die *Transportschicht*.

- Schicht 5 schließlich ist die Datenverarbeitung durch die Applikation selbst.

Damit jede der Schichten die ihr zugeteilte Aufgabe erfüllen kann, müssen zusätzliche Informationen der jeweiligen Schicht im Datenpaket im *Header*, dem Kopf des Datenpakets, gespeichert werden. Jede der Schichten fügt einen kleinen Datenblock, den sog. „Protokollkopf“ (engl. *Protocol header*), an das im Entstehen begriffene Paket vorne dran. Schauen wir uns also einmal ein beliebiges TCP/IP-Datenpaket an, das auf einem Ethernetkabel unterwegs ist, so setzt sich dieses wie in Bild 14.2 abgebildet zusammen.

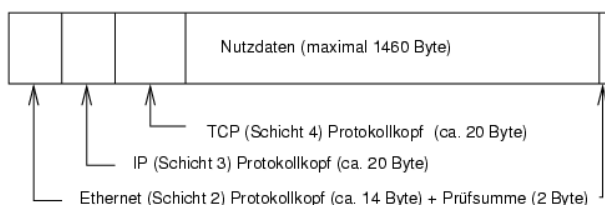


Abbildung 14.2: TCP/IP Paket im Ethernet

Wie Sie sehen, ist die Welt nicht perfekt und ohne Ausnahme. Die Prüfsumme der Sicherungsschicht befindet sich am Ende des Pakets und nicht am Anfang. Dies bringt aber für die Netzwerkhardware eine Vereinfachung. Die maximal mögliche Menge der Nutzdaten in einem Paket beträgt im Ethernet-Netzwerk 1460 Byte.

Möchte eine Applikation also Daten über das Netzwerk verschicken, durchlaufen die Daten die einzelnen Schichtebenen, die alle im Linuxkernel (Ausnahme Schicht 1: Netzwerkkarte) implementiert sind. Jede der Schichten ist dafür verantwortlich, die Daten so aufzubereiten, dass sie an die jeweils darunter liegende Schicht weitergereicht werden können. Die unterste Schicht ist schließlich für den eigentlichen Datenversand zuständig. Beim Empfang läuft das ganze nun umgekehrt ab. Wie bei den Schalen einer Zwiebel werden von jeder Schicht die Protokollköpfe von den Nutzdaten entfernt. Schicht 4 ist dann letztendlich dafür verantwortlich, die Daten für die Applikation auf dem Zielrechner bereitzustellen. Dabei kommuniziert eine Schicht immer nur mit der Schicht direkt über oder unter ihr. Für eine Applikation ist es also irrelevant, ob die Daten über ein

100-MBit/s-FDDI-Netzwerk oder über eine 56-KBit/s-Modemleitung übertragen werden. Umgekehrt ist es für die Datenübertragungsleitung egal, welche Daten eigentlich verschickt werden, solange sie richtig verpackt sind.

14.1.2 IP-Adressen und Routing

Hinweis

Die folgenden Abschnitte beschreiben IPv4-Netzwerke. Informationen zu seinem Nachfolgeprotokoll IPv6 bekommen Sie Abschnitt 14.2 auf Seite 353.

Hinweis

IP-Adressen

Jeder Computer im Internet hat eine eindeutige 32-Bit-Adresse. Diese 32 Bit bzw. 4 Byte werden normalerweise wie in Beispiel 14.1 in der zweiten Zeile abgebildet geschrieben.

Beispiel 14.1: Schreibweise einer IP-Adresse

```
IP-Adresse (binär):   11000000 10101000 00000000 00010100
IP-Adresse (dezimal):    192.      168.      0.      20
```

Die vier Bytes werden also im dezimalen Zahlensystem durch einen Punkt getrennt nebeneinander geschrieben. Die IP-Adresse ist einem Rechner bzw. einer Netzwerkschnittstelle zugeordnet, sie kann also nicht woanders auf der Welt nochmals verwendet werden. Ausnahmen von diesen Regeln gibt es zwar, spielen aber bei der folgenden Betrachtung erst einmal keine Rolle.

Auch die Ethernetkarte besitzt selbst eine eindeutige Adresse, die so genannte *MAC* (engl. *Media access control*) Adresse. Diese ist 48 Bit lang, weltweit eindeutig und wird vom Hersteller der Netzwerkkarte fest in der Hardware gespeichert. Durch die Vergabe der Adresse vom Hersteller ergibt sich aber ein fataler Nachteil: Die MAC-Adressen bilden kein hierarchisches System, sondern sind mehr oder weniger zufällig verteilt. Sie können daher nicht zur Adressierung eines weit entfernten Rechners verwendet

werden. Die MAC-Adresse spielt aber bei der Kommunikation von Rechnern in einem lokalen Netz eine entscheidende Rolle (und ist der Hauptbestandteil des Protokollkopfes von Schicht 2).

Zurück zu den IP-Adressen: Die Punkte deuten schon an, dass die IP-Adressen ein hierarchisches System bilden. Bis Mitte der 90er Jahre waren die IP-Adressen fest in Klassen eingeteilt. Dieses System erwies sich aber als zu unflexibel und daher wurde diese Aufteilung aufgegeben. Man verwendet nun „klassenloses Routing“ (CIDR (engl. *classless inter domain routing*)).

Netzmasken und Routing

Da der Rechner mit der IP-Adresse 192.168.0.0 erst einmal nicht wissen kann, wo sich der Rechner mit der IP-Adresse 192.168.0.20 befindet, wurden die Netzmasken erdacht.

Vereinfacht gesagt definiert die (Sub-)Netzmaske auf einem Rechner mit IP-Adresse, was „drinnen“ und was „draußen“ ist. Rechner, die sich „drinnen“ (Profis sagen: „im gleichen Subnetz“) befinden, können direkt angesprochen werden. Rechner, die sich „draußen“ („nicht im gleichen Subnetz“) befinden, müssen über ein so genanntes Gateway oder Router angesprochen werden. Da jedes Netzwerkinterface eine eigene IP-Adresse bekommen kann, ahnen Sie schon, dass es schnell beliebig kompliziert wird.

Bevor ein Netzwerkpaket auf die Reise geschickt wird, läuft folgendes im Rechner ab: Die Zieladresse wird mit der Netzmaske bitweise UND verknüpft. Daraufhin wird auch die Absendeadresse bitweise mit der Netzmaske UND verknüpft (siehe Tabelle 14.2). Stehen mehrere Netzwerkinterfaces zur Verfügung, werden in der Regel alle möglichen Absendeadressen überprüft.

Die Ergebnisse der UND-Verknüpfungen werden verglichen. Ergibt sich zwischen den Ergebnissen eine exakte Übereinstimmung, so befindet sich der Zielrechner im gleichen Subnetz. Ansonsten muss er über ein Gateway angesprochen werden. Das heißt, je mehr „1“ Bits sich in der Netzmaske befinden, desto weniger Rechner können direkt, sondern nur über ein Gateway angesprochen werden. Zur Veranschaulichung sind in Beispiel 14.2 mehrere Beispiele aufgeführt.

Beispiel 14.2: Verknüpfungen der IP-Adressen mit der Netzmaske

```
IP-Adresse (192.168.0.20):  11000000 10101000 00000000 00010100
Netzmaske (255.255.255.0): 11111111 11111111 11111111 00000000
```

```

Ergebnis (binär):          11000000 10101000 00000000 00000000
Ergebnis (dezimal):        192.      168.      0.      0

IP-Adresse (213.95.15.200): 11010101 10111111 00001111 11001000
Netzmaske (255.255.255.0): 11111111 11111111 11111111 00000000
-----
Ergebnis (binär):          11010101 10111111 00001111 00000000
Ergebnis (dezimal):        213.      95.      15.      0

```

Die Netzmaske wird wieder – wie schon die IP-Adresse – in Form von durch Punkte getrennten Dezimalzahlen geschrieben. Da die Netzmaske auch ein 32-Bit-Wert ist, werden vier Zahlenwerte nebeneinander geschrieben. Welche Rechner Gateway sind oder welche Adressbereiche über welche Netzwerkschnittstelle erreichbar sind, muss vom Benutzer konfiguriert werden.

Um wieder ein Beispiel zu geben: Alle Rechner, die am gleichen Ethernetkabel angeschlossen sind, befinden sich in der Regel *im gleichen Subnetz* und sind direkt erreichbar. Auch wenn das Ethernet über Switches oder Bridges unterteilt ist, sind diese Rechner immer noch direkt erreichbar.

Wollen Sie eine längere Strecke überbrücken, ist das preiswerte Ethernet dafür nicht mehr geeignet. Sie müssen dann die IP-Pakete auf andere Hardware (zum Beispiel FDDI oder ISDN) weiterleiten. Solche Geräte heißen Router bzw. Gateway. Ein Linuxrechner kann diese Aufgabe selbstverständlich auch erledigen, die entsprechende Option wird mit `ip_forwarding` bezeichnet.

Ist ein Gateway konfiguriert, wird das IP-Paket an das passende Gateway geschickt. Dieses versucht, das Paket dann wiederum nach dem gleichen Schema weiterzuleiten. Das wiederholt sich auf jedem weiteren Rechner so oft, bis das Paket entweder den Zielrechner erreicht hat oder die „Lebenszeit“ TTL (engl. *time to live*) des Paketes verbraucht ist.

Tabelle 14.2: Spezielle Adressen

Adressart	Beschreibung
Netzwerkbasisisadresse	Das ist die Netzmaske UND eine beliebige Adresse aus dem Netz, also das was in Beispiel 14.2 auf der vorherigen Seite unter Ergebnis abgebildet ist. Diese Adresse kann keinem Rechner zugewiesen werden.

Broadcastadresse	Sie heißt soviel wie: „Sprich alle Rechner in diesem Subnetz an“. Um sie zu erzeugen wird die Netzmaske binär invertiert und mit der Netzwerkbasissadresse ODER verknüpft. Obiges Beispiel ergibt also 192.168.0.255. Natürlich kann auch diese Adresse keinem Rechner zugewiesen werden.
Localhost	Die Adresse 127.0.0.1 ist auf jedem Rechner fest dem so genannten „Loopbackdevice“ zugewiesen. Über diese Adresse kann man eine Verbindung auf den eigenen Rechner aufbauen.

Da die IP-Adressen aber weltweit eindeutig sein müssen, können Sie natürlich nicht beliebige Adressen erfinden. Damit Sie aber trotzdem ein auf IP basierendes Netzwerk aufbauen können gibt es drei Adressbereiche, die Sie ohne weiteres verwenden können. Mit diesen können Sie allerdings nicht so ohne weiteres Verbindungen in das Internet aufbauen, da diese Adressen im Internet nicht weitergeleitet werden.

Dabei handelt es sich um diese Adressbereiche die in RFC 1597 definiert sind:

Tabelle 14.3: Private IP-Adressbereiche

Netzwerk/Netzmaske	Bereich
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

14.1.3 Domain Name System – DNS

DNS sorgt dafür, dass Sie sich nicht zwingend irgendwelche IP-Adressen merken müssen: Mit Hilfe von DNS kann eine IP-Adresse einem oder sogar mehreren Namen zugeordnet werden und umgekehrt auch eine Name einer IP-Adresse. Unter Linux erfolgt diese Umwandlung üblicherweise von einer speziellen Software namens `bind`. Der Rechner, der diese Umwandlung dann erledigt, nennt sich *Nameserver*. Dabei bilden die Namen wieder ein hierarchisches System, in dem die einzelnen Namensbestandtei-

le durch Punkte getrennt sind. Die Namenshierarchie ist aber unabhängig von der oben beschriebenen Hierarchie der IP-Adressen.

Schauen wir uns einmal einen vollständigen Namen an, zum Beispiel `laurent.suse.de` geschrieben im Format `Rechnername.Domain`. Ein vollständiger Name – Experten sagen „fully qualified domain name“ oder kurz *FQDN* dazu – besteht aus einem Rechnernamen und einem Domain-Teil. Dabei wird der Domain-Teil aus einem frei wählbaren Anteil – im obigen Beispiel `suse` – und der so genannten *Top level domain*, *TLD* gebildet.

Aus historischen Gründen ist die Zuteilung der TLDs etwas verwirrend. So werden in den USA traditionell dreibuchstabige TLDs verwendet, woanders aber immer die aus zwei Buchstaben bestehenden ISO-Länderbezeichnungen; seit 2000 stehen zusätzliche TLDs für spezielle Sachgebiete mit zum Teil mehr als drei Buchstaben zur Verfügung (zum Beispiel `.info`, `.name`, `.museum` usw.).

In der Frühzeit des Internets (vor 1990) gab es hierzu eine Datei `/etc/hosts`, in der die Namen aller im Internet vertretenen Rechner gespeichert waren. Dies erwies sich bei der schnell wachsenden Menge von am Internet angeschlossener Rechner als unpraktikabel. Deshalb wurde eine dezentralisierte Datenbank entworfen, die die Rechnernamen verteilt speichern kann. Diese Datenbank, eben jener oben erwähnte Nameserver, hält also nicht die Daten aller Rechner im Internet vorrätig, sondern kann Anfragen an ihm nachgeschaltete, andere Nameserver weiterdelegieren.

An der Spitze der Hierarchie befinden sich die „Root-Nameserver“, die die Top level domains verwalten. Die Root-Nameserver werden vom Network Information Center (NIC) verwaltet. Der Root-Nameserver kennt die jeweils für eine Top level domain zuständigen Nameserver. Im Falle der deutschen Top level domain `de` ist das DE-NIC für die Domains zuständig, die mit der TLD `de` aufhören. Mehr Informationen zum DE-NIC erhalten Sie auf der Website <http://www.denic.de>, mehr Informationen zum Top level domain NIC erfahren Sie unter <http://www.internic.net>.

Damit auch Ihr Rechner einen Namen in eine IP-Adresse auflösen kann, muss ihm mindestens ein Nameserver mit einer IP-Adresse bekannt sein. Die Konfiguration eines Nameservers erledigen Sie komfortabel mit Hilfe von YaST. Falls Sie eine Einwahl über Modem vornehmen, kann es sein, dass das zur Einwahl verwendete Protokoll die Adresse des Nameservers während der Einwahl mitliefert.

Aber nicht nur Rechnernamen können über DNS aufgelöst werden, DNS kann noch mehr. Zum Beispiel „weiß“ der Nameserver auch, welcher Rechner für eine ganze Domain E-Mails annimmt, der so genannte *Mail exchanger* (MX).

Die Konfiguration des Nameserverzugriffs unter SUSE LINUX ist im Abschnitt 14.6 auf Seite 375 beschrieben.

whois

Eng verwandt mit DNS ist das Protokoll `whois`. Mit dem gleichnamigen Programm `whois` können Sie schnell herauskriegen, wer für eine bestimmte Domain verantwortlich ist.

14.2 IPv6 – Internet der nächsten Generation

Bedingt durch die Erfindung des WWW (engl. *World Wide Web*) ist das Internet und damit die Anzahl der Rechner, die TCP/IP „sprechen“, in den letzten zehn Jahren explosionsartig gewachsen. Seit der Erfindung des WWW durch Tim Berners-Lee 1990 am CERN (<http://public.web.cern.ch/>) ist die Zahl der Internet-Hosts von wenigen tausend auf mittlerweile ca. 100 Millionen angewachsen.

Wie Sie wissen, besteht eine IP-Adresse „nur“ aus 32 Bit. Viele IP-Adressen können durch organisatorische Bedingtheiten gar nicht verwendet werden, sie gehen verloren. Zur Erinnerung: Das Internet wird in Subnetze, also Teilnetze unterteilt. Diese bestehen immer aus einer Zweierpotenz minus zwei nutzbaren IP-Adressen. Ein Subnetz besteht also beispielsweise aus 2, 6, 14, 30 usw. IP-Adressen. Möchten Sie beispielsweise 128 Rechner an das Internet anbinden, so benötigen Sie ein Subnetz mit 256 IP-Adressen, von denen nur 254 nutzbar sind. Wie Sie oben gesehen haben, entfallen zwei der IP-Adressen aus einem Subnetz, nämlich die Broadcastadresse und die Netzwerkbasissadresse.

Um die absehbare Adressknappheit zu entschärfen, verwendet man unter dem momentan eingesetzten IPv4 Mechanismen wie DHCP oder NAT (engl. *Network Address Translation*). Beide Verfahren mildern zusammen mit der Konvention von öffentlichen und privaten Netzwerkadressbereichen die Adressnot im Internet. Nachteil dieser Methoden ist die teilweise sehr umständliche und wartungsintensive Konfiguration. Sie benötigen zum korrekten Aufsetzen eines Rechners im IPv4-Netzwerk zahlreiche Informationen wie die eigene IP-Adresse, Subnetzmaske, Gatewayadresse und unter Umständen einen Nameserver. Alle diese Angaben müssen Sie „wissen“ und können Sie nirgendwoher ableiten.

Mit IPv6 gehören Adressknappheit und komplizierte Konfigurationen der Vergangenheit an. In den folgenden Abschnitten erfahren Sie mehr zu den Neuerungen und Vorteilen von IPv6 und über den Übergang von altem zum neuen Protokoll.

14.2.1 Vorteile von IPv6

Der wichtigste und augenfälligste Vorteil des neuen Protokolls ist die enorme Vergrößerung des verfügbaren Adressraums. Eine IPv6-Adresse enthält 128 Bit anstelle der traditionellen 32 Bit. Somit stehen viele Billiarden (!) IP-Adressen zur Verfügung.

IPv6-Adressen unterscheiden sich von ihren Vorgängern nicht nur in der Länge, auch ihre innere Struktur ist anders und erlaubt es, speziellere Informationen über das zugehörige System und sein Netzwerk zu kodieren. Mehr dazu unter Abschnitt 14.2.2 auf der nächsten Seite.

Weitere wichtige Vorteile des neuen Protokolls in Kurzform:

Autokonfiguration IPv6 setzt das „Plug and Play“-Prinzip im Netzwerk um. Ein frisch installiertes System integriert sich ohne weiteren Konfigurationsaufwand ins (lokale) Netz. Der Autokonfigurationsmechanismus des Terminals leitet die eigene Adresse aus den Informationen ab, die ihm über das „Neighbor Discovery Protocol“ (ND) von den benachbarten Routern zugespielt werden. Dieses Verfahren erfordert keinerlei Eingriff von Seiten des Administrators und hat gegenüber dem unter IPv4 genutzten Adressverteiler DHCP den weiteren Vorteil, dass die Wartung eines zentralen Servers mit den verfügbaren Adressen entfällt.

Mobilität IPv6 erlaubt es, dass einer Netzwerkschnittstelle gleichzeitig mehrere Adressen zugeordnet werden. Somit haben Sie als Benutzer eines Systems einfach und ohne Zusatzaufwand Zugang zu mehreren verschiedenen Netzen. Sie können dies mit dem „Roaming“ in Mobilfunknetzen vergleichen: Befinden Sie sich mitsamt Ihrem Mobiltelefon im Ausland, bucht sich das Handy automatisch in das fremde Netz ein. Egal, wo Sie sind, Ihre Erreichbarkeit unter Ihrer normalen Telefonnummer ist gewährleistet und Sie telefonieren im fremden Netz, als wäre es Ihr Heimatnetz.

Sichere Kommunikation Während sichere Kommunikation unter IPv4 nur als Zusatzfunktion zu realisieren war, ist IPSec und damit die sichere Kommunikation zwischen zwei Systemen über einen Tunnel durch das unsichere Internet in IPv6 bereits enthalten.

Kompatibilität zum Vorgänger Ein schneller Umstieg des gesamten Internets von IPv4 auf IPv6 ist nicht realistisch. Deshalb ist es wichtig, dass beide Versionen im Internet und sogar auf einem System koexistieren können. Die Koexistenz beider im Internet ist durch die Verwendung kompatibler Adressen (IPv4-Adressen lassen sich einfach in IPv6-Adressen umsetzen) und die Verwendung verschiedener „Tunnel“ gesichert (siehe Abschnitt 14.2.3 auf Seite 360). Über „Dual-Stack-IP“ ist die Unterstützung beider Protokolle auf dem einzelnen System möglich. Jedes der beiden Protokolle verwendet einen eigenen Netzwerkstack, so dass sich die beiden Protokollversionen nicht gegenseitig in die Quere kommen.

Multicasting – maßgeschneidertes Dienstangebot

Während unter IPv4 einige Dienste (zum Beispiel SMB) ihre Pakete per Broadcast an alle Teilnehmer des lokalen Netzes senden mussten, ist unter IPv6 ein viel differenzierteres Vorgehen möglich. Mit Hilfe von Multicast kann eine Gruppe von Rechnern auf einmal angesprochen werden, also nicht alle auf einmal („broadcast“), oder nur einer („unicast“), sondern eben ein paar. Welche das sind, hängt von der Anwendung ab. Es gibt aber auch ein paar wohldefinierte Multicastgruppen, beispielsweise „alle Nameserver“ (engl. *all nameservers multicast group*), oder „alle Router“ (engl. *all routers multicast group*).

14.2.2 Das Adresssystem von IPv6

Wie bereits erwähnt, hat das bisher verwendete IP-Protokoll zwei schwerwiegende Nachteile. Zum einen gehen die verfügbaren IP-Adressen langsam aus und zum anderen ist die Netzwerkkonfiguration und das Verwalten von Routingtabellen immer komplizierter und wartungsintensiver. Dem ersten Problem begegnet IPv6 mit der Erweiterung des Adressraums auf 128 Bit. Die Lösung für das zweite Problem liegt der hierarchischen Adressstruktur, ausgeklügelten Mechanismen zur Adresszuweisung im Netz und der Möglichkeit des „Multi-Homings“ (mehrere Adressen pro Schnittstelle mit Zugang zu verschiedenen Netzwerken).

In Zusammenhang mit IPv6 sollten Sie folgende drei Adresstypen unterscheiden können:

unicast Adressen dieses Typs gehören zu genau einer Netzwerkschnittstelle. Pakete mit einer Adresse dieses Typs werden an genau einen Empfänger ausgeliefert. Unicast-Adressen werden verwendet, um einzelne Rechner im lokalen Netz oder Internet anzusprechen.

multicast Adressen dieses Typs weisen auf eine Gruppe von Schnittstellen. Pakete mit einer Adresse dieses Typs werden an alle Empfänger zugestellt, die zu dieser Gruppe gehören. Multicast-Adressen werden vorwiegend von bestimmten Netzwerkdiensten benutzt, um gezielt bestimmte Gruppen von Rechnern zu adressieren.

anycast Adressen dieses Typs weisen auf eine Gruppe von Schnittstellen. Pakete mit einer Adresse dieses Typs werden an den Angehörigen der Gruppe ausgeliefert, der nach den Begriffen des verwendeten Routingprotokolls dem Absender am nächsten ist. Anycast-Adressen werden verwendet, um Terminal das Auffinden eines Servers mit einem bestimmten Dienstangebot in ihrem Netzbereich zu finden. Alle Server eines Typs erhalten die gleiche Anycast-Adresse. Fordert der Terminal einen Dienst an, antwortet derjenige Server, der nach Einschätzung des Routingprotokolls dem Host am nächsten liegt. Sollte dieser Server ausfallen, wird automatisch der zweitnächste verwendet

Aufbau einer IPv6-Adresse

Eine IPv6-Adresse setzt sich aus acht Blöcken zu je 16 Bit zusammen, die durch : (Doppelpunkt) getrennt werden und in Hexadezimalschreibweise dargestellt werden. Führende Null-Bytes in einer Gruppe dürfen weggelassen werden, nicht aber inmitten oder am Ende einer Gruppe. Mehr als vier Null-Bytes direkt hintereinander kann man durch das Auslassungszeichen :: überspringen. Allerdings ist nur ein Auslassungszeichen in einer Adresse erlaubt. Dieser Vorgang des Auslassens wird in Englisch mit „collapsing“ bezeichnet. In Ausgabe 14.3 ist dieser Vorgang anhand dreier äquivalenter Schreibweisen ein und derselben Adresse dargestellt.

Beispiel 14.3: Beispiel einer IPv6-Adresse

```
fe80 : 0000 : 0000 : 0000 : 0000 : 10 : 1000 : 1a4
fe80 :    0 :    0 :    0 :    0 : 10 : 1000 : 1a4
fe80 :                                     : 10 : 1000 : 1a4
```

Jeder Teil einer IPv6-Adresse hat eine definierte Bedeutung. Die ersten Bytes bilden einen Präfix und geben den Typ der Adresse an. Der Mittelteil adressiert ein Netzwerk oder ist bedeutungslos und den Schluss der Adresse bildet der Hostteil. Netzmasken definieren sich unter IPv6 über die Länge des Präfix, die per / am Ende der Adresse mit angegeben wird.

Eine Adressdarstellung wie in Ausgabe 14.4 besagt, dass die letzten 64 Bit den Hostteil und die vorderen 64 Bit den Netzwerkteil der Adresse bilden. Anders gesagt bedeutet die 64, dass von links her die Netzmaske mit 1 Bits aufgefüllt wird. Es gibt in der Netzmaske also 64 1 Bits. Wie bei IPv4 wird durch eine UND-Verknüpfung der Netzmaske mit der IP-Adresse bestimmt, ob sich ein Rechner im gleichen oder in einem anderen Subnetz befindet.

Beispiel 14.4: IPv6-Adresse mit Präfixangabe

```
fe80::10:1000:1a4/64
```

IPv6 kennt verschiedene Präfixe mit definierter Bedeutung (siehe Tabelle 14.4).

Tabelle 14.4: verschiedene IPv6-Präfixe

Präfix (hexadez.)	Verwendung
00	IPv4 Adressen und IPv4 über IPv6-Kompatibilitätsadressen. Es handelt sich um eine zu IPv4 kompatible Adresse. Ein geeigneter Router muss das IPv6-Paket noch in IPv4 verwandeln. Weitere Spezialadressen (zum Beispiel Loopback Device) sind ebenfalls mit diesem Präfix ausgestattet.
erste Ziffer 2 oder 3	(engl. <i>Aggregatable Global Unicast Address</i>). Wie bisher auch können Sie bei IPv6 Teilnetze zugewiesen bekommen. Aktuell gibt es folgende Adressräume: 2001::/16 (<i>production quality address space</i>) und 2002::/16 (<i>6to4 address space</i>).
fe80::/10	(engl. <i>link-local</i>) Adressen mit diesem Präfix dürfen nicht geroutet werden und können daher nur im gleichen Subnetz erreicht werden.
fec0::/10	(engl. <i>site-local</i>) Diese Adressen dürfen zwar geroutet werden, aber nur innerhalb einer Organisation. Damit entsprechen diese Adressen den bisherigen „privaten“ Netzen (beispielsweise 10.x.x.x).
ff	(engl. <i>multicast</i>) IPv6-Adressen, die mit ff anfangen, sind Multicastadressen.

Unicastadressen folgen einem dreigeteilten Aufbauprinzip:

Public Topology Der erste Teil, der unter anderem auch eines der oben erwähnten Präfixes enthält, dient dem Routing des Pakets im öffentlichen Internet. Hier sind Informationen zum Provider oder der Institution kodiert, die den Netzwerkzugang bereitstellen.

Site Topology Der zweite Teil enthält Routinginformationen über das Subnetz, in dem das Paket zugestellt werden soll.

Interface ID Der dritte Teil identifiziert eindeutig die Schnittstelle, an die das Paket gerichtet ist. Dies erlaubt, die MAC-Adresse als Adressbestandteil zu verwenden. Da diese weltweit nur einmal vorhanden und zugleich vom Hardwarehersteller fest vorgegeben ist, vereinfacht sich die Konfiguration der Rechner sehr. In Wirklichkeit werden sogar die ersten 64 Bit zu einem so genannten EUI-64-Token zusammengefasst. Dabei werden die letzten 48 Bit der MAC-Adresse entnommen, die restlichen 24 Bit enthalten spezielle Informationen, die etwas über den Typ des Tokens aussagen. Das ermöglicht dann auch, Geräten ohne MAC-Adresse (PPP- und ISDN-Verbindungen!) ein EUI-64-Token zuzuweisen.

Abgeleitet aus diesem Grundaufbau werden fünf verschiedene Typen von Unicastadressen unterschieden:

:: (unspecified) diese Adresse verwendet ein Rechner als Quelladresse, wenn seine Netzwerkschnittstelle zum ersten Mal initialisiert wird und noch keine Informationen über die eigene Adresse hat.

:::1 (loopback) Adresse des Loopback-Devices.

IPv4 kompatible Adresse Die IPv6-Adresse wird aus der IPv4-Adresse und einem Präfix von 96 0-Bits am Beginn der Adresse zusammengestellt. Dieser Typ der Kompatibilitätsadressen wird beim Tunneling verwendet (siehe Abschnitt 14.2.3 auf Seite 360). IPv4/IPv6-Hosts können so mit anderen kommunizieren, die sich im reinen IPv4-Netz befinden.

IPv6 gemappte IPv4-Adresse Dieser Adresstyp gibt die IPv6-Adresse eines reinen IPv4-Rechners an.

Lokale Adressen Es gibt zwei Typen von Adressen zum rein lokalen Gebrauch:

link-local Dieser Adresstyp ist ausschließlich für den Gebrauch im lokalen Subnetz. Router dürfen Pakete mit solcher Ziel- oder Quelladresse nicht an das Internet oder andere Subnetze weiterreichen. Diese Adressen zeichnen sich durch einen speziellen Präfix ($\text{fe80}::/10$) und die Interface-ID der Netzwerkkarte aus. Der Mittelteil der Adresse besteht aus aussagefreien Nullbytes. Diese Art von Adresse wird von den Autokonfigurationsmethoden verwendet, um Rechner im gleichen Subnetz anzusprechen.

site-local Dieser Adresstyp darf zwischen einzelnen Subnetzen geroutet werden, aber nicht außerhalb einer Organisation (engl. *site*) ins Internet gelangen. Solche Adressen werden für Intranets eingesetzt und sind ein Äquivalent zu den privaten Adressen des IPv4. Neben einem definierten Präfix ($\text{fec0}::/10$) und der Interface-ID enthalten diese Adressen ein 16 Bit-Feld, in dem die Subnetz-ID kodiert ist. Der Rest wird wieder mit Null-Bytes aufgefüllt.

Zusätzlich gibt es in IPv6 eine neue Erfindung: Einer Netzwerkschnittstelle werden üblicherweise mehrere IP-Adressen zugewiesen. Das hat den Vorteil, dass mehrere verschiedene Netze zur Verfügung stehen. Eines davon kann mit Hilfe der MAC-Adresse und einem bekannten Präfix zu einem vollautomatisch konfigurierten Netz zusammengestellt werden, und ohne weitere Konfigurationsarbeiten sind damit direkt nach dem Starten von IPv6 alle Rechner im lokalen Netz erreichbar (sog. „Link-local-Adresse“). Die MAC-Adresse als Bestandteil der IP-Adresse macht jede dieser Adressen global unterscheidbar. Einzig die Teile der „Site Topology“ oder „Public Topology“ können variieren, je nachdem in welchem Netz dieser Rechner aktuell zu erreichen ist.

„Bewegt“ sich ein Rechner zwischen mehreren Netzen hin und her, braucht er mindestens zwei Adressen. Die eine, seine „Home Address“ beinhaltet neben seiner Interface-ID die Informationen zu seinem Heimatnetz, in dem er normalerweise betrieben wird und das entsprechende Präfix. Die „Home Address“ ist statisch und wird nicht verändert. Alle Pakete, die für diesen Rechner bestimmt sind, werden ihm sowohl im eigenen als auch in fremden Netzen zugestellt. Möglich wird die Zustellung im Fremdnetz über wesentliche Neuerungen des IPv6-Protokolls, über *Stateless Autoconfiguration* und *Neighbor Discovery*. Der mobile Rechner hat neben seiner „Home Address“ eine oder mehrere weitere Adressen, die in die fremden Netze gehören, in denen er sich bewegt. Diese Adressen heißen „Care-of Address“. Im Heimatnetz des mobilen Rechners muss eine Instanz vorhanden

sein, die an seine „Home Address“ gerichtete „nachsendet“, sollte er sich in einem anderen Netz befinden. Diese Funktion wird in einem IPv6-Szenario vom „Home Agent“ übernommen. Er stellt alle Pakete, die an die Heimatadresse des mobilen Rechners gerichtet sind, über einen Tunnel zu. Pakete, die als Zieladresse die „Care-of Address“ tragen, können ohne Umweg über den Home Agent zugestellt werden.

14.2.3 IPv4 versus IPv6 – Wandern zwischen den Welten

Der Umstieg aller Rechner im Internet von IPv4 auf IPv6 wird nicht auf einen Schlag geschehen. Vielmehr werden altes und neues Protokoll noch eine ganze Weile nebeneinanderher existieren. Die Koexistenz auf einem Rechner ist per „Dual Stack“ gelöst, es bleibt aber die Frage, wie IPv6-Rechner mit IPv4-Rechnern kommunizieren können und wie IPv6 über die momentan noch vorherrschenden IPv4-Netze transportiert werden sollen. Tunneling und die Verwendung von Kompatibilitätsadressen (siehe Abschnitt 14.2.2 auf Seite 356) sind hier die Methoden der Wahl.

Einzelne IPv6-Inseln im (weltweiten) IPv4-Netz tauschen ihre Daten über Tunnel aus. Beim Tunneling werden IPv6-Pakete in IPv4-Pakete verpackt, um sie über ein reines IPv4-Netzwerk transportieren zu können. Ein Tunnel ist definiert als die Verbindung zwischen zwei IPv4-Endpunkten. Hierbei muss die IPv6-Zieladresse (oder das entsprechende Präfix) angegeben werden, an die die verkappten IPv6-Pakete gerichtet sind und die entfernte IPv4-Adresse, an der die getunnelten Pakete in Empfang genommen werden sollen. Im einfachsten Fall konfigurieren Administratoren solche Tunnel zwischen ihren Netzwerken *manuell* und nach Absprache. Solches Tunneling wird *statisches* Tunneling genannt.

Trotzdem reicht manuelles Tunneling oft nicht aus, um die Menge der zum täglichen vernetzten Arbeiten nötigen Tunnel aufzubauen und zu verwalten. Aus diesem Grund wurden drei verschiedene Verfahren entwickelt, die *dynamisches* Tunneling erlauben:

6over4 IPv6-Pakete werden automatisch in IPv4-Pakete verpackt und über ein IPv4-Netzwerk versandt, in dem Multicasting aktiviert ist. IPv6 wird vorgespiegelt, das gesamte Netzwerk (Internet) sei ein einziges, riesiges LAN (engl. *Local Area Network*). So wird der IPv4-Endpunkt des Tunnel automatisch ermittelt. Nachteil dieser Methode sind die schlechte Skalierbarkeit und die Tatsache, dass IP-Multicasting keineswegs im gesamten Internet verfügbar ist. Diese Lösung eignet sich für kleinere Firmen- oder Institutsnetzwerke, die

die Möglichkeit von IP-Multicasting bieten. Das zugrundeliegende RFC ist RFC2529.

6to4 Bei dieser Methode werden automatisch IPv4-Adressen aus IPv6-Adressen generiert. So können IPv6-Inseln über ein IPv4-Netz miteinander kommunizieren. Allerdings gibt es betreffend der Kommunikation zwischen IPv6-Inseln und dem Internet einige Probleme. Das zugrundeliegende RFC ist RFC3056.

IPv6 Tunnel Broker Dieser Ansatz sieht spezielle Server vor, die für den Benutzer automatisch Tunnel anlegen. Das zugrundeliegende RFC ist RFC3053.

Hinweis

Die 6Bone Initiative

Mitten im „altmodischen“ Internet existiert mit *6Bone* (www.6bone.net) ein weltweit verteiltes Netzwerk von IPv6-Subnetzen, die über Tunnel miteinander verbunden sind. Innerhalb des 6Bone-Netzes wird IPv6 getestet. Softwareentwickler und Provider, die IPv6-Dienste entwickeln oder anbieten, können diese Testumgebung nutzen, um wichtige Erfahrungen mit dem neuen Protokoll zu bekommen. Weitere Informationen finden Sie auf den Projektseiten von 6Bone.

Hinweis

14.2.4 Weiterführende Literatur und Links zu IPv6

Natürlich kann und will der obige Überblick keine vollständige Einführung zum sehr umfangreichen Thema IPv6 sein. Zum tieferen Einstieg in IPv6 können Sie die folgende Onlineliteratur und Bücher zu Rate ziehen:

<http://www.ngnet.it/e/cosa-ipv6.php>

Artikelserie mit sehr guten Beschreibungen zu den Grundlagen von IPv6. Gut geeignet für einen Einstieg ins Thema.

<http://www.bieringer.de/linux/IPv6/>

Linux-IPv6-HOWTO und viele Links.

<http://www.6bone.de/> Anschluss an das IPv6 über einen Tunnel bekommen.

<http://www.ipv6.org/> Alles rund um IPv6.

RFC 2640 Das einführende RFC zum Thema IPv6.

IPv6 Essentials Englischsprachiger Überblick zum Thema IPv6. Hagen, Silvia: *IPv6 Essentials*. O'Reilly & Associates, 2002. - (ISBN 0-596-00125-8).

14.3 Manuelle Netzwerkkonfiguration

Die manuelle Konfiguration der Netzwerksoftware sollte stets die zweite Wahl sein. Wir empfehlen, YaST zu benutzen. Wesentlich ist, dass alle Netzwerk-Interfaces mit dem Skript `/sbin/ifup` aufgesetzt werden. Zum Anhalten oder Prüfen eines Interfaces gibt es `ifdown` und `ifstatus`.

Wenn Sie nur fest eingebaute Netzwerkkarten haben, genügt es die Interfaces über ihren Namen zu konfigurieren. Mit `ifup eth0`, `ifstatus eth0` und `ifdown eth0` starten, prüfen und stoppen Sie das Netzwerkinterface `eth0`. Die verwendeten Konfigurationsdaten liegen unter `/etc/sysconfig/network/ifcfg-eth0`. `eth0` ist hier sowohl der Interface-Name als auch der Name für die Netzwerkkonfiguration.

Die Netzwerkkonfiguration kann alternativ auch der Hardware-Adresse (MAC-Adresse) einer Netzwerkkarte zugeordnet werden. Dazu wird eine Konfigurationsdatei `ifcfg-<HardwareadresseohneDoppelpunkte>` verwendet. Die Buchstaben der Hardware-Adresse müssen hier klein geschrieben werden, so wie sie von `ip link` ausgegeben wird (`ifconfig` verwendet große Buchstaben). Wenn `ifup` eine Konfigurationsdatei passend zur Hardware-Adresse findet, wird ein möglicherweise auch vorhandenes `ifcfg-eth0` ignoriert.

Mit hotplugfähigen Netzwerkkarten wird es ein wenig komplizierter. Wenn Sie keine solche Karte besitzen, können Sie beim Abschnitt 14.3.1 auf der nächsten Seite weiterlesen.

Da bei hotplugfähigen Netzwerkkarten die Zuordnung des Interface-Namen zur Karte eher zufällig ist, werden die Konfigurationen für eine solche Karte nicht unter dem Interface-Namen abgelegt, sondern unter einem Bezeichner, der die Art der verwendeten Hardware und den Anschlusspunkt beschreibt, im Folgenden Hardwarebeschreibung genannt. `ifup` muss in diesem Fall mit zwei Argumenten aufgerufen werden, der genauen Hardwarebeschreibung und dem gegenwärtigen Interface-Namen. Anschließend wird von `ifup` die Konfiguration ermittelt, die möglichst genau auf die Hardwarebeschreibung passt.

Als Beispiel wollen wir einen Laptop mit zwei PCMCIA-Steckplätzen und einer PCMCIA Ethernet Netzwerkkarte annehmen. Außerdem gibt es in

diesem Gerät noch eine festeingebaute Netzwerkkarte, die als Interfacenamen `eth0` erhält. Wenn die PCMCIA-Karte im Steckplatz 0 steckt, lautet ihre Hardwarebeschreibung `eth-pcmcia-0`. Der `cardmgr` oder das Hotplug-Netzwerkskript rufen nun `ifup eth-pcmcia-0 eth1` auf. Nun sucht `ifup`, ob es unter `/etc/sysconfig/network/` eine Datei `ifcfg-eth-pcmcia-0` gibt. Wenn nicht wird weiter nach `ifcfg-eth-pcmcia`, `ifcfg-pcmcia-0`, `ifcfg-pcmcia`, `ifcfg-eth1` und `ifcfg-eth` gesucht. Die zuerst gefundene Datei wird zur Konfiguration verwendet. Wenn also eine Netzwerkkonfiguration angelegt werden soll, die für alle PCMCIA-Netzwerkkarten (in allen Steckplätzen) gelten soll, muss diese `ifcfg-pcmcia` heißen. Diese würde dann für `eth-pcmcia-0` genauso wie für eine Tokenringkarte in Steckplatz 1 `tr-pcmcia-1` verwendet.

Auch hier hat wieder eine Konfiguration nach Hardwareadresse absoluten Vorrang. Dies wurde nur aus Gründen der Übersichtlichkeit aus dem Beispiel weggelassen.

YaST geht bei der Konfiguration von hotplugfähigen Karten einen Umweg. Dort werden Konfigurationen für solche Karten durchnummeriert. Deshalb schreibt YaST die Einstellungen für eine PCMCIA-Karte immer nach `ifcfg-eth-pcmcia-<laufendeNummer>`. Damit diese Konfiguration dann trotzdem für alle Steckplätze funktioniert, wird noch ein Link `ifcfg-eth-pcmcia` auf diese Datei angelegt. Dies sollten Sie beachten, wenn Sie teilweise mit und teilweise ohne YaST konfigurieren.

14.3.1 Konfigurationsdateien

Dieser Abschnitt gibt eine Übersicht über die Netzwerkkonfigurationsdateien und erklärt ihre Funktion sowie das verwendete Format.

`/etc/sysconfig/network/ifcfg-*`

Diese Dateien enthalten die Daten, die spezifisch für ein Netzwerk-Interface sind. Sie können nach dem Interface-Namen benannt sein (`ifcfg-eth2`), nach der Hardware-Adresse einer Netzwerkkarte (`ifcfg-000086386be3`) oder nach einer Hardware-Beschreibung für eine Karte (`ifcfg-usb`). Sollen Netzwerkaliasse verwendet werden, heißen die dazu nötigen Dateien einfach `ifcfg-eth2:1` oder `ifcfg-usb:1`. Das Skript `ifup` bekommt neben dem Interface-Namen bei Bedarf auch eine genaue Hardwarebeschreibung und sucht dann die am besten passende Datei zur Konfiguration aus.

Die Dateien enthalten die IP-Adresse (`BOOTPROTO=static`, `IPADDR=10.10.11.214`) oder die Anweisung, DHCP zu verwenden

(BOOTPROTO="dhcp"). Die IP-Adresse sollte die Netzmaske bereits enthalten (IPADDR="10.10.11.214/16"). Die vollständige Liste von Variablen enthält die Manpage zu `ifup`. Es können außerdem alle Variablen aus den Dateien `dhcp`, `wireless` und `config` in den `ifcfg-*`-Dateien verwendet werden, wenn eine sonst allgemeine Einstellung nur für ein Interface verwendet werden soll.

/etc/sysconfig/network/config,dhcp,wireless

Die Datei `config` enthält allgemeine Einstellungen zum Verhalten von `ifup`, `ifdown` und `ifstatus`. Sie ist vollständig kommentiert. Ebenso gibt es Kommentare in `dhcp` und `wireless`, wo allgemeine Einstellungen zu DHCP und Funknetzwerkkarten Platz finden. Alle Variablen aus diesen Dateien können auch in `ifcfg-*` verwendet werden und haben dort natürlich Vorrang.

/etc/resolv.conf

Wie bereits die Datei `/etc/host.conf`, so spielt auch diese Datei in Bezug auf Auflösung von Rechnernamen durch die *resolver*-Bibliothek eine Rolle.

In dieser Datei wird angegeben, welcher Domain der Rechner angehört (Schlüsselwort `search`) und wie die Adresse des Nameservers ist (Schlüsselwort `nameserver`), der angesprochen werden soll. Es können mehrere Domainnamen angegeben werden. Beim Auflösen eines nicht voll qualifizierten Namens wird versucht, durch Anhängen der einzelnen Einträge in `search` einen gültigen, voll qualifizierten Namen zu erzeugen. Mehrere Nameserver können durch mehrere Zeilen, die mit `nameserver` beginnen, bekannt gemacht werden. Kommentare werden wieder mit `#` eingeleitet.

Ein Beispiel für `/etc/resolv.conf` zeigt Datei 14.5.

Beispiel 14.5: /etc/resolv.conf

```
# Our domain
search example.com
#
# We use sun (192.168.0.20) as nameserver
nameserver 192.168.0.20
```

YaST trägt hier den angegebenen Nameserver ein!

Einige Dienste wie `pppd` (`wvdial`), `ipppd` (`isdn`), `dhcp` (`dhcpcd` und `dhclient`), `pcmcia` und `hotplug` modifizieren die Datei `/etc/resolv.conf` über das Skript `modify_resolvconf`.

Wenn die Datei `/etc/resolv.conf` durch dieses Skript vorübergehend modifiziert wurde, enthält sie einen definierten Kommentar, der Auskunft darüber gibt, welcher Dienst sie modifiziert hat, wo die ursprüngliche Datei gesichert ist und wie man die automatischen Modifikationen abstellen kann.

Wenn `/etc/resolv.conf` mehrmals modifiziert wird, wird diese Verschachtelung von Modifikationen auch dann wieder sauber abgebaut, wenn sie in einer anderen Reihenfolge zurückgenommen werden; dies kann bei `isdn`, `pcmcia` und `hotplug` durchaus vorkommen.

Wenn ein Dienst nicht sauber beendet wurde, kann mit Hilfe des Skripts `modify_resolvconf` der Ursprungszustand wiederhergestellt werden. Beim Booten wird geprüft, ob eine modifizierte `resolv.conf` stehen geblieben ist (z. B. wegen Systemabsturz). Dann wird die ursprüngliche (unmodifizierte) `resolv.conf` wiederhergestellt.

YaST findet mittels `modify_resolvconf check` heraus, ob `resolv.conf` modifiziert wurde, und dann den Benutzer warnen, dass seine Änderungen nach der Restauration wieder verloren sein werden. Ansonsten verwendet YaST `modify_resolvconf` nicht, das heißt eine Änderung der Datei `resolv.conf` mittels YaST und eine manuelle Änderung sind äquivalent. Beides entspricht einer gezielten und dauerhaften Änderung, während eine Änderung durch einen der genannten Dienste nur vorübergehend ist.

/etc/hosts

In dieser Datei (siehe Datei 14.6) werden Rechnernamen IP-Adressen zugeordnet. Wird kein Nameserver verwendet, müssen hier alle Rechner aufgeführt werden, zu denen eine IP-Verbindung aufgebaut werden soll. Je Rechner wird eine Zeile bestehend aus IP-Adresse, dem voll qualifizierten Hostnamen und dem Rechnernamen (zum Beispiel `earth`) in die Datei eingetragen. Die IP-Adresse muss am Anfang der Zeile stehen, die Einträge werden durch Leerzeichen bzw. Tabulatoren getrennt. Kommentare werden durch `#` eingeleitet.

Beispiel 14.6: /etc/hosts

```
127.0.0.1 localhost
192.168.0.20 sunexample.com sun
192.168.0.0 earthexample.com earth
```

/etc/networks

Hier werden Netzwerknamen in Netzwerkadressen umgesetzt. Das Format ähnelt dem der `hosts`-Datei, jedoch stehen hier die Netzwerknamen vor den Adressen (siehe Datei 14.7).

Beispiel 14.7: */etc/networks*

```
loopback      127.0.0.0
localnet      192.168.0.0
```

/etc/host.conf

Das Auflösen von Namen – das heisst das Übersetzen von Rechner- bzw. Netzwerknamen über die *resolver*-Bibliothek – wird durch diese Datei gesteuert. Diese Datei wird nur für Programme verwendet, die gegen die `libc4` oder die `libc5` gelinkt sind; für aktuelle `glibc`-Programme vgl. die Einstellungen in `/etc/nsswitch.conf`! Ein Parameter muss in einer eigenen Zeile stehen, Kommentare werden durch `#` eingeleitet. Die möglichen Parameter zeigt Tabelle 14.5.

Tabelle 14.5: *Parameter für /etc/host.conf*

<i>order hosts, bind</i>	Legt fest, in welcher Reihenfolge die Dienste zum Auflösen eines Namens angesprochen werden sollen. Mögliche Argumente sind (durch Leerzeichen oder Kommata voneinander getrennt): <i>hosts</i> : Durchsuchen der Datei <code>/etc/hosts</code> <i>bind</i> : Ansprechen eines Nameservers <i>nis</i> : Über NIS
<i>multi on/off</i>	Bestimmt, ob ein in <code>/etc/hosts</code> eingetragener Rechner mehrere IP-Adressen haben darf.
<i>nospoof on spoofalert on/off</i>	Diese Parameter beeinflussen das <i>spoofing</i> des Nameservers, haben aber weiter keinen Einfluss auf die Netzwerkkonfiguration.

`trim domainname`

Der angegebene Domainname wird vor dem Auflösen des Rechnernamens von diesem abgeschnitten (insofern der Rechnername diesen Domainnamen enthält). Diese Option ist dann von Nutzen, wenn in der Datei `/etc/hosts` nur Namen aus der lokalen Domain stehen, diese aber auch mit angehängtem Domainnamen erkannt werden sollen.

Ein Beispiel für `/etc/hosts.conf` zeigt Datei 14.8.

Beispiel 14.8: `/etc/hosts.conf`

```
# We have named running
order hosts bind
# Allow multiple addrs
multi on
```

`/etc/nsswitch.conf`

Mit der GNU C Library 2.0 hat der Name Service Switch (NSS) Einzug gehalten (vgl. die Manpage von `man 5 nsswitch.conf`, sowie ausführlicher *The GNU C Library Reference Manual*, Kapitel „System Databases and Name Service Switch“ >).

In der Datei `/etc/nsswitch.conf` wird festgelegt, in welcher Reihenfolge bestimmte Informationen abgefragt werden. Ein Beispiel für `nsswitch.conf` zeigt Datei 14.9. Kommentare werden durch `#` eingeleitet. Dort bedeutet zum Beispiel der Eintrag bei der Datenbank `hosts`, dass nach `/etc/hosts` (files) eine Anfrage über DNS (vgl. Abschnitt 14.6 auf Seite 375) losgeschickt wird.

Beispiel 14.9: `/etc/nsswitch.conf`

```
passwd:      compat
group:       compat

hosts:       files dns
networks:    files dns

services:    db files
```

```
protocols:  db files

netgroup:   files
automount:  files nis
```

Die über NSS verfügbaren Datenbanken sind in Tabelle 14.6 genannt. Zusätzlich sind in Zukunft automount, bootparams, netmasks und publickey zu erwarten.

Tabelle 14.6: Über /etc/nsswitch.conf verfügbare Datenbanken

aliases	Mail-Aliase, von sendmail verwendet; vgl. die Manpage man 5 aliases.
ethers	Ethernet-Adressen.
group	Für Benutzergruppen, von getgrent verwendet; vgl. die Manpage man 5 group.
hosts	Für Hostnamen und IP-Adressen, von gethostbyname und ähnlichen Funktionen verwendet.
netgroup	Im Netzwerk gültige Liste von Hosts und Benutzern, um Zugriffsrechte zu steuern; vgl. die Manpage man 5 netgroup.
networks	Netzwerknamen und -adressen, von getnetent verwendet.
passwd	Benutzerpasswörter, von getpwent verwendet; vgl. die Manpage man 5 passwd.
protocols	Netzwerk-Protokolle, von getprotoent verwendet; vgl. die Manpage man 5 protocols.
rpc	Remote Procedure Call-Namen und -Adressen, von getrpcbyname und ähnlichen Funktionen verwendet.
services	Netzwerkdienste, von getservent verwendet.
shadow	Shadow-Passwörter der Benutzer, von getspnam verwendet; vgl. die Manpage man 5 shadow.

Die Konfigurationsmöglichkeiten der NSS-Datenbanken stehen in Tabelle 14.7 auf der nächsten Seite.

Tabelle 14.7: Konfigurationsmöglichkeiten der NSS-Datenbanken

<code>files</code>	direkt auf Dateien zugreifen, zum Beispiel auf <code>/etc/aliases</code> .
<code>db</code>	über eine Datenbank zugreifen.
<code>nis</code>	NIS, vgl. Abschnitt 14.8 auf Seite 404.
<code>nisplus</code>	
<code>dns</code>	Nur bei <code>hosts</code> und <code>networks</code> als Erweiterung verwendbar.
<code>compat</code>	Nur bei <code>passwd</code> , <code>shadow</code> und <code>group</code> als Erweiterung verwendbar.

Zusätzlich ist es möglich, unterschiedliche Reaktionen bei bestimmten Lookup-Ergebnissen auszulösen; Details sind der Manpage `man 5 nsswitch.conf` zu entnehmen.

`/etc/nscd.conf`

Über diese Datei wird der `nscd` *Name Service Cache Daemon* konfiguriert (vgl. `man 8 nscd` und die `man 5 nscd.conf`). Per default werden die Einträge von `passwd` und `groups` gecached. `hosts` wird normalerweise nicht gecached, da sich der Rechner dann nicht mehr auf „forward/reverse lookups“ dieses Namensdienstes verlassen kann. Statt dem `nscd` diese Aufgabe zu übertragen, sollten sie einen „caching“ Nameserver einrichten.

Wenn beispielsweise das Caching für `passwd` aktiviert ist, dauert es in der Regel 15 Sekunden, bis ein neu angelegter lokaler Benutzer dem System bekannt ist. Durch das Neustarten des `nscd` mit dem Befehl `rcnscd restart` kann diese Wartezeit verkürzt werden.

`/etc/HOSTNAME`

Hier steht der Name des Rechners, also nur der Hostname ohne den Domainnamen. Diese Datei wird von verschiedenen Skripten während des Starts des Rechners gelesen. Sie darf nur eine Zeile enthalten, in der der Rechnername steht!

14.3.2 Startup-Skripten

Neben den beschriebenen Konfigurationsdateien gibt es noch verschiedene Skripten, die während des Hochfahrens des Rechners die Netzwerkprogramme starten. Diese werden gestartet, sobald das System in einen der *Multiuser-Runlevel* übergeht (vgl. Tabelle 14.8).

Tabelle 14.8: Einige Startup-Skripten der Netzwerkprogramme

<code>/etc/init.d/network</code>	Dieses Skript übernimmt die Konfiguration der Netzwerk Hard- und Software während der Startphase des Systems.
<code>/etc/init.d/xinetd</code>	Startet den <code>xinetd</code> . Der <code>xinetd</code> kann verwendet werden, um bei Bedarf Serverdienste auf dem System zur Verfügung zu stellen. Beispielsweise kann er den <code>vsftpd</code> starten, sobald eine FTP-Verbindung initiiert wird.
<code>/etc/init.d/portmap</code>	Startet den Portmapper, der benötigt wird, um RPC-Server verwenden zu können, wie zum Beispiel einen NFS-Server.
<code>/etc/init.d/nfsserver</code>	Startet den NFS-Server.
<code>/etc/init.d/postfix</code>	Kontrolliert den postfix-Prozess.
<code>/etc/init.d/ypserv</code>	Startet den NIS-Server.
<code>/etc/init.d/ypbind</code>	Startet den NIS-Client.

14.4 Die Einbindung ins Netzwerk

TCP/IP ist inzwischen das Standard-Netzwerkprotokoll, über das alle modernen Betriebssysteme mit TCP/IP kommunizieren können. Dennoch unterstützt Linux auch noch andere Netzwerkprotokolle, beispielsweise das (früher) von Novell Netware verwendete IPX oder das von Macintosh-Rechnern verwendete Appletalk. In diesem Rahmen besprechen wir nur die Integration eines Linux-Rechners in ein TCP/IP-Netzwerk. Wenn Sie exotische Arcnet, Token-Ring oder FDDI-Netzwerkkarten einbinden wollen, finden Sie weiterführende Hilfe hierzu in den Kernelquellen `/usr/src/linux/Documentation`, die Sie separat mit dem Paket `kernel-source` installieren.

14.4.1 Vorbereitungen

Der Rechner muss über eine unterstützte Netzwerkkarte verfügen. Üblicherweise wird die Netzwerkkarte schon bei der Installation erkannt und der passende Treiber eingebunden. Ob Ihre Karte korrekt eingebunden wurde, können Sie unter anderem daran sehen, dass die Ausgabe des Kommandos `ifstatus eth0` das Netzwerk-Device `eth0` anzeigt.

Wenn der Kernel-Support für die Netzwerkkarte als Modul realisiert wird – so wie es beim SUSE-Kernel standardmäßig der Fall ist –, dann muss der Name des Moduls als Alias in der `/etc/modules.conf` eingetragen werden. Für die erste Ethernet-Karte zum Beispiel in der Art `alias eth0 tulip`. Dies geschieht automatisch, wenn im `linuxrc` während der Erstinstallation der Treiber-Support für die Netzwerkkarte geladen wird. Nachträglich lässt sich diese Aufgabe von YaST aus erledigen.

Bei hotplugfähigen Netzwerkkarten (zum Beispiel PCMCIA oder USB) werden die Treiber beim Einstecken automatisch ermittelt; es muss dazu nichts konfiguriert werden.

14.4.2 Konfiguration mit YaST

Die Konfiguration der Netzwerkkarte lässt sich mit YaST schnell durchführen. Wählen Sie im YaST-Kontrollzentrum den Punkt ‘Netzwerkgeräte’ und anschließend ‘Netzwerkkarte’. In diesem Dialog integrieren Sie mit ‘Hinzufügen’ eine Netzwerkkarte, mit ‘Entfernen’ wird die entsprechende Karte aus der Konfiguration gelöscht und mit ‘Bearbeiten’ können die Einstellungen zu einer Netzwerkkarte geändert werden.

Aktivieren Sie den Punkt ‘Hardwaredetails’, um die Hardwaredaten einer schon eingerichteten Netzwerkkarte zu verändern. Sie gelangen in das Menü zur Konfiguration der Hardwaredaten Ihrer Netzwerkkarte; vgl. Abbildung 14.3 auf der nächsten Seite.

Üblicherweise wird der richtige Treiber für Ihre Netzwerkkarte schon während der Installation von YaST konfiguriert und die Netzwerkkarte aktiviert. Daher sind manuelle Einstellungen der Hardwareparameter nur nötig, wenn Sie mehr als eine Netzwerkkarte einsetzen oder die Netzwerkkarte nicht automatisch erkannt wird. In diesem Fall müssen Sie den Punkt ‘Hinzufügen’ anwählen, damit ein neues Treibermodul ausgewählt werden kann.

In diesem Dialog können Sie den Typ der Netzwerkkarte und im Falle von ISA-Karten auch den zu verwendenden Interrupt und die IO-Adresse einstellen. Manchen Netzwerktreibern können Sie auch spezielle Parameter

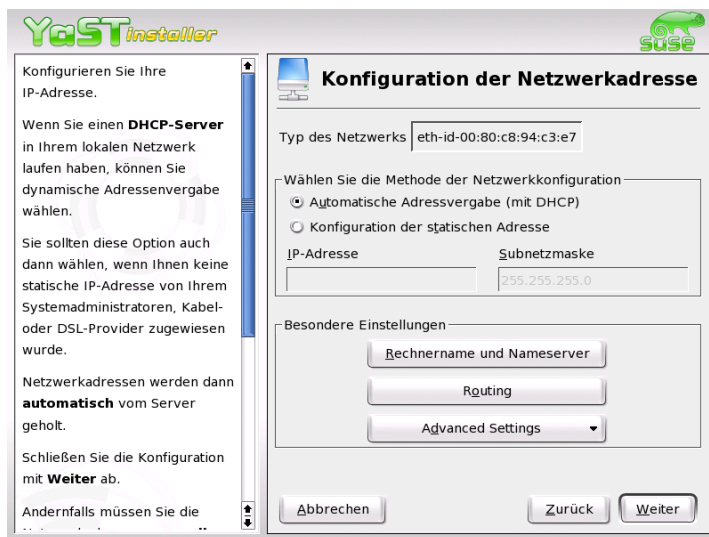


Abbildung 14.3: Konfiguration der Hardwareparameter

wie die zu verwendende Schnittstelle mitgeben, ob Sie beispielsweise den RJ-45- oder BNC-Anschluss auf der Karte verwenden wollen. Beachten Sie hierzu die Dokumentation des Treibermoduls. Für PCMCIA und USB genügt es die entsprechenden Kästchen zu aktivieren.

Nach der Eingabe der Hardwareparameter konfigurieren Sie die weiteren Daten der Netzwerkschnittstelle. Wählen Sie im Dialog 'Manuelle Konfiguration der Netzwerkkarte' den Punkt 'Netzwerkschnittstelle' aus, um die soeben einrichtete Netzwerkkarte zu aktivieren und dieser Netzwerkkarte eine IP-Adresse zuzuweisen. Wählen Sie dann die Kartenummer aus und klicken Sie auf 'Weiter'. Es erscheint ein neuer Dialog, in dem Sie die IP-Adresse und die weiteren Daten des IP-Netzwerks auswählen können. Falls Sie selbst ein eigenes Netzwerk aufbauen, können Sie sich bei der Vergabe der Adressen am Abschnitt 14.1 auf Seite 344 bzw. der Tabelle 14.3 auf Seite 351 orientieren. Ansonsten tragen Sie bitte die von Ihrem Netzwerkadministrator zugewiesenen Adressen in die vorgesehenen Felder ein.

Vergessen Sie nicht, einen Nameserver unter 'Rechnername und Nameserver' einzustellen, damit die Namensauflösung wie in Abschnitt 14.6 auf Seite 375 beschrieben funktionieren kann. Über den Punkt 'Routing' kön-

nen Sie das Routing einstellen. Wählen Sie den Punkt 'Konfiguration für Experten', um fortgeschrittene Einstellungen vorzunehmen.

Falls Sie Funknetzwerkarten verwenden, aktivieren Sie bitte das Kästchen 'Einstellungen für Funkverbindungen'. Sie können dann die wichtigsten Einstellungen hierzu in einem eigenen Dialog vornehmen. Im Wesentlichen sind das der Betriebsmodus, Netzwerknamen und ein Schlüssel für die verschlüsselte Datenübertragung.

Damit ist die Netzwerkkonfiguration abgeschlossen. YaST ruft abschließend SuSEconfig auf und trägt Ihre Angaben in die entsprechenden Dateien ein. Damit die Einstellungen wirksam werden, müssen die betroffenen Programme neu konfiguriert und die entsprechenden Daemonen neu gestartet werden. Dies erreichen Sie, indem Sie als Benutzer `root` den Befehl `rcnetwork restart` eingeben.

14.4.3 Hotplug/PCMCIA

Eine Sonderstellung nehmen hotplugfähige Netzwerkkarten ein wie zum Beispiel PCMCIA oder USB-Geräte. Im Gegensatz zu fest eingebauten Netzwerkkarten, die eine gleich bleibende Gerätebezeichnung erhalten, beispielsweise `eth0`, wird solchen Karten dynamisch bei Bedarf eine freie Gerätebezeichnung zugewiesen. Um Konflikte mit eventuell fest eingebauten Karten zu vermeiden wird PCMCIA und Hotplug beim Booten erst nach dem Netzwerk gestartet.

Diese Karten werden automatisch eingerichtet sobald sie eingesetzt bzw. beim Booten erkannt werden. Deshalb ist es nicht notwendig, dass PCMCIA vor dem Netzwerk gestartet wird. Im Gegenteil: Wenn diese Karten nur vom Netzwerkstartscript beim Booten behandelt würden, ginge die Möglichkeit, sie zur Laufzeit des Systems zu tauschen, verloren.

14.4.4 Konfiguration von IPv6

Falls Sie die Verwendung von IPv6 konfigurieren möchten, müssen Sie in der Regel keine Konfiguration auf den Arbeitsstationen durchführen. Allerdings muss die IPv6-Unterstützung geladen werden. Rufen Sie als Benutzer `root` den Befehl `modprobe ipv6` auf.

Aufgrund der Autokonfigurationsphilosophie von IPv6 wird dann der Netzwerkkarte eine Adresse im `link-local` Netz zugewiesen. Normalerweise wird auf einer Arbeitsstation keine Routingtabelle gepflegt. Die Router im Netz können über das Router Advertisement Protocol von der Arbeitsstation darüber befragt werden, welches Präfix und welche Gateways

zu verwenden sind. Um einen IPv6-Router aufzusetzen, können Sie das Programm `radvd` aus `radvd` verwenden. Dieses Programm teilt den Arbeitsstationen das zu verwendende Präfix für IPv6-Adressen und den/die Router mit. Das Programm `zebra` kann ebenfalls zur Autokonfiguration von Adressen und für Routingkonfiguration eingesetzt werden.

Um einer Arbeitsstation eine IPv6-Adresse zuweisen zu können, ist es ratsam, einen Router mit dem Programm `radvd` oder `zebra` zu installieren und zu konfigurieren. Die Arbeitsstationen bekommen die IPv6-Adresse dann automatisch zugewiesen.

Zur Einrichtung verschiedener Tunnel mit Hilfe der Dateien unter `/etc/sysconfig/network` finden Sie wichtige Informationen in der Manualpage von `ifup` (man `ifup`).

14.5 Routing unter SuSE Linux

Ab SUSE LINUX 8.0 wird die Routing-Tabelle in den Konfigurationsdateien `/etc/sysconfig/network/routes` und `/etc/sysconfig/network/ifroute-*` eingestellt.

In der Datei `/etc/sysconfig/network/routes` können alle statischen Routen eingetragen werden, die für die verschiedenen Aufgaben eines Systems benötigt werden könnten: Route zu einem Rechner, Route zu einem Rechner über ein Gateway und Route zu einem Netzwerk. Hier wird z.B. der Default Gateway bei statischen Routen konfiguriert:

```
default GATEWAY - -
```

wobei GATEWAY die IP-Adresse des Gateways ist.

Für alle Interfaces, die individuelles Routing benötigen, kann dies jeweils in einer eigenen Datei pro Interface definiert werden: `/etc/sysconfig/network/ifroute-*`. Für das Zeichen `*` muss die Interface-Bezeichnung eingesetzt werden. Die Einträge können folgendermaßen aussehen:

DESTINATION	GATEWAY	NETMASK	INTERFACE	[TYPE]	[OPTIONS]
DESTINATION	GATEWAY	PREFIXLEN	INTERFACE	[TYPE]	[OPTIONS]
DESTINATION/PREFIXLEN	GATEWAY	-	INTERFACE	[TYPE]	[OPTIONS]

Falls GATEWAY, NETMASK, PREFIXLEN oder INTERFACE nicht angegeben werden, muss an ihrer Stelle das Zeichen – gesetzt werden. Die Einträge TYPE und OPTIONS können schlicht entfallen.

- In der ersten Spalte steht das Ziel einer Route. Dabei kann dort die IP-Adresse eines Netzes oder Rechners oder bei *erreichbaren* Nameservern auch der voll qualifizierte Name eines Netzes oder eines Rechners stehen.
- Die zweite Spalte enthält entweder das Default-Gateway oder ein Gateway, hinter dem ein Rechner oder Netzwerk erreichbar ist.
- Die dritte Spalte enthält die Netzmaske für Netzwerke oder Rechner hinter einem Gateway. Für Rechner hinter einem Gateway lautet die Maske zum Beispiel 255 . 255 . 255 . 255.
- Die letzte Spalte ist nur für die am lokalen Rechner angeschlossenen Netzwerke (Loopback, Ethernet, ISDN, PPP, ...) wichtig. Hier muss der Name des Devices eingetragen werden.

14.6 DNS – Domain Name System

DNS (engl. *Domain Name System*) wird benötigt, um die Domain- und Rechnernamen in IP-Adressen aufzulösen. Bevor Sie einen eigenen Nameserver einrichten, sollten Sie die allgemeinen Informationen zu DNS im Abschnitt 14.1.3 auf Seite 351 lesen.

Die folgenden Konfigurationsbeispiele beziehen sich auf BIND 9, der jetzt Standard bei SUSE LINUX ist.

14.6.1 Nameserver BIND starten

Der Nameserver BIND (*Berkeley Internet Name Domain*) ist auf SUSE LINUX bereits soweit vorkonfiguriert, dass man ihn problemlos sofort nach der Installation starten kann. Hat man bereits eine funktionsfähige Internetverbindung und trägt in der `/etc/resolv.conf` als Nameserver 127.0.0.1 für localhost ein, hat man in der Regel schon eine funktionierende Namensauflösung, ohne dass man den DNS des Providers kennt. BIND führt so die Namensauflösung über die Root-Nameserver durch, was aber merklich langsamer ist. Normalerweise sollte man den DNS des Providers mit seiner IP-Adresse in der Konfigurationsdatei `/etc/named.conf`

unter `forwarders` eintragen, um eine effektive und sichere Namensauflösung zu erhalten. Funktioniert das soweit, läuft der Nameserver als reiner „Caching-only“-Nameserver. Erst wenn man ihm eigene Zonen bereitstellt, wird er ein richtiger DNS werden. Ein einfaches Beispiel dafür, findet man im Dokumentations-Verzeichnis: `/usr/share/doc/packages/bind9/sample-config`.

Man sollte allerdings keine offizielle Domain aufsetzen, solange man diese nicht von der zuständigen Institution — für `.de` ist das die DENIC eG — zugewiesen bekommen hat. Auch wenn man eine eigene Domain hat, diese aber vom Provider verwaltet wird, sollte man diese besser nicht verwenden, da BIND sonst keine Anfragen für diese Domain mehr forwarden (weiterleiten) würde und so zum Beispiel der Webserver beim Provider für die eigene Domain nicht mehr erreichbar wäre.

Um den Nameserver zu starten, gibt man auf der Kommandozeile als `root` ein:

```
rcnamed start
```

Erscheint rechts in grün „done“, ist der `named`, so heißt der Nameserver-Prozess, erfolgreich gestartet. Auf dem lokalen System kann man die Funktionsfähigkeit des Nameservers sofort testen, indem man die Programme `host` oder `dig` verwendet. Als Default-Server muss `localhost` mit der Adresse `127.0.0.1` angezeigt werden. Sollte das nicht der Fall sein, steht wahrscheinlich in der `/etc/resolv.conf` ein falscher Nameserver oder diese Datei existiert gar nicht. Für einen ersten Test gibt man `host 127.0.0.1` ein, das sollte immer funktionieren; erhält man eine Fehlermeldung, sollte man mit folgendem Kommando überprüfen, ob der `named` überhaupt läuft

```
rcnamed status
```

Falls der Nameserver nicht startet oder ein fehlerhaftes Verhalten zeigt, findet man die Ursache in den meisten Fällen in `/var/log/messages` protokolliert.

Um den Nameserver des Providers oder um einen eigenen, der bereits im lokalen Netz läuft, als „Forwarder“ zu verwenden, trägt man diesen oder auch mehrere, im Abschnitt `options` unter `forwarders` ein; die in Datei 14.10 auf der nächsten Seite verwendeten IP-Adressen sind willkürlich gewählt und müssen entsprechend den eigenen Gegebenheiten angepasst werden.

Beispiel 14.10: *Forwarding-Optionen in named.conf*

```
options {
    directory "/var/lib/named";
    forwarders { 10.11.12.13; 10.11.12.14; };
    listen-on { 127.0.0.1; 192.168.0.99; };
    allow-query { 127/8; 192.168.0/24; };
    notify no;
};
```

Nach den `options` folgen die Einträge für die Zonen, die Einträge für `localhost`, `0.0.127.in-addr.arpa`, sowie `.` vom `type hint` sollten immer vorhanden sein. Die zugehörigen Dateien müssen nicht verändert werden, da sie so funktionieren wie sie sind. Beachten muss man auch, dass nach jedem Eintrag ein `;` steht und die geschweiften Klammern korrekt gesetzt sind. Hat man nun Änderungen an der Konfigurationsdatei `/etc/named.conf` oder an den Zonen-Dateien vorgenommen, muss man BIND mit dem Kommando `rndc reload` dazu veranlassen, diese neu einzulesen. Alternativ kann man den Nameserver auch komplett mit dem Befehl `rndc restart` neu starten. Mit dem Kommando `rndc stop` kann man den Nameserver jederzeit komplett beenden.

14.6.2 Die Konfigurationsdatei `/etc/named.conf`

Alle Einstellungen zum Nameserver BIND sind in der Datei `/etc/named.conf` vorzunehmen. Die Zonendaten selbst, die Rechnernamen, IP-Adressen usw. für die zu verwaltenden Domains, sind in separaten Dateien im Verzeichnis `/var/lib/named` abzulegen, dazu aber unten mehr.

Die `/etc/named.conf` unterteilt sich grob in zwei Bereiche, zum einen der Abschnitt `options` für allgemeine Einstellungen und zum anderen die `zone`-Einträge für die einzelnen Domains. Außerdem kann man noch einen Bereich `logging`, sowie Einträge vom Typ `acl` (engl. *Access Control List*) definieren. Kommentarzeilen beginnen mit einem `#`-Zeichen, alternativ ist `//` auch erlaubt.

Eine minimalistische `/etc/named.conf` stellt Datei 14.11 dar.

Beispiel 14.11: *Minimalistische Datei `/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";
    };

```

14.6.3 Die wichtigsten Konfigurationsoptionen im Abschnitt options

directory "/var/lib/named"; gibt das Verzeichnis an, in dem der BIND die Dateien mit den Zonendaten findet.

forwarders 10.0.0.1;; verwendet man, um den oder die Nameserver (meist des Providers) anzugeben, an den oder die die DNS-Anfragen weitergereicht werden, die nicht direkt beantwortet werden können.

forward first; bewirkt, dass die DNS-Anfragen zu erst geforwarded werden, bevor versucht wird diese über die Root-Nameserver aufzulösen. Anstelle von `forward first` kann man auch `forward only` schreiben, dann werden alle Anfragen weitergeleitet und die Root-Nameserver werden gar nicht mehr angesprochen. Das kann für Firewall-Konfigurationen sinnvoll sein.

listen-on port 53 127.0.0.1; 192.168.0.1;; sagt dem BIND, auf welchen Netzwerkinterfaces und welchem Port er auf Anfragen der Clients horcht. Die Angabe `port 53` kann man sich dabei sparen, da 53 ohnehin der Standardport ist. Lässt man diesen Eintrag komplett weg, werden standardmäßig alle Interfaces verwendet.

listen-on-v6 port 53 any;; sagt dem BIND, auf welchem Port er auf Anfragen der Clients horcht, die IPv6 verwenden. Außer any ist alternativ nur noch none erlaubt, da der Server stets auf der IPv6-Wildcard-Adresse horcht.

query-source address * port 53; Dieser Eintrag kann notwendig sein, wenn eine Firewall die externen DNS-Abfragen blockiert. So wird BIND dazu gebracht, Anfragen nach außen von Port 53 aus und nicht von den hohen Ports > 1024 zu stellen.

query-source-v6 address * port 53; Dieser Eintrag muss für Anfragen über IPv6 verwendet werden.

allow-query 127.0.0.1; 192.168.1/24;; bestimmt die Netze, aus denen Clients DNS-Anfragen stellen dürfen. Das /24 ist dabei eine Kurzschreibweise für die Anzahl der Bits in der Netzmaske, in diesem Fall 255 . 255 . 255 . 0.

allow-transfer !*;; regelt, welche Rechner Zonentransfers anfordern dürfen, dieses Beispiel unterbindet sie, aufgrund des ! * komplett. Ohne diesen Eintrag können Zonentransfers ohne Einschränkungen von überall angefordert werden.

statistics-interval 0; Ohne diesen Eintrag produziert BIND stündlich mehrere Zeilen Statistikmeldungen in /var/log/messages. Die Angabe von 0 bewirkt, dass diese komplett unterdrückt werden; hier kann man die Zeit in Minuten angeben.

cleaning-interval 720; Diese Option legt fest, in welchem Zeitabstand BIND seinen Cache aufräumt. Die Aktivität führt jedes Mal zu einem Eintrag in /var/log/messages. Die Zeitangabe erfolgt in Minuten. Voreingestellt sind 60 Minuten.

interface-interval 0; BIND durchsucht regelmäßig die Netzwerkschnittstellen nach neuen oder nicht mehr vorhandenen Interfaces. Setzt man diesen Wert auf 0, so wird darauf verzichtet und BIND lauscht nur auf den beim Start gefundenen Interfaces. Alternativ kann man das Intervall in Minuten angeben. Voreingestellt sind 60 Minuten.

notify no; Das no bewirkt, dass keine anderen Nameserver benachrichtigt werden, wenn an den Zonendaten Änderungen vorgenommen werden oder der Nameserver neu gestartet wird.

14.6.4 Der Konfigurationsabschnitt Logging

Was und wie wohin mitprotokolliert wird, kann man beim BIND recht vielseitig konfigurieren. Normalerweise sind die Voreinstellungen ausreichend. Datei 14.12 zeigt die einfachste Form eines solchen Eintrags und unterdrückt das „Logging“ komplett.

Beispiel 14.12: Logging wird unterdrückt

```
logging {  
    category default { null; };  
};
```

14.6.5 Aufbau der Zonen-Einträge

Nach zone wird der Name der zu verwaltenden Domain angegeben, hier willkürlich `meine-domain.de` gefolgt von einem `in` und einem in geschweiften Klammern gesetzten Block zugehöriger Optionen; vgl. 14.13.

Beispiel 14.13: Zone-Eintrag für meine-domain.de

```
zone "meine-domain.de" in {  
    type master;  
    file "meine-domain.zone";  
    notify no;  
};
```

Will man eine „Slave-Zone“ definieren, ändert sich nur der `type` auf `slave` und es muss ein Nameserver angegeben werden, der diese Zone als master verwaltet – das kann aber auch ein „slave“ sein; vgl. Datei 14.14.

Beispiel 14.14: Zone-Eintrag für andere-domain.de

```
zone "andere-domain.de" in {  
    type slave;  
    file "slave/andere-domain.zone";  
    masters { 10.0.0.1; };  
};
```

Die Zonen-Optionen:

type master; Das master legt fest, dass diese Zone auf diesem Nameserver verwaltet wird. Das setzt eine korrekt erstellte Zonendatei voraus.

type slave; Diese Zone wird von einem anderen Nameserver transferiert. Muss zusammen mit masters verwendet werden.

type hint; Die Zone . vom Typ hint wird für die Angabe der Root-Nameserver verwendet. Diese Zonendefinition kann man unverändert lassen.

file "meine-domain.zone" oder file "slave/andere-domain.zone";

Dieser Eintrag gibt die Datei an, in der die Zonendaten für die Domain eingetragen sind. Bei einem slave braucht die Datei nicht zu existieren, da ihr Inhalt von einem anderen Nameserver geholt wird. Um Master- und Slave-Dateien auseinander zu halten, gibt man für die Slave-Dateien das Verzeichnis slave an.

masters 10.0.0.1;; Diesen Eintrag braucht man nur für Slave-Zonen und er gibt an, von welchem Nameserver die Zonendatei transferiert werden soll.

allow-update !*;; diese Option regelt den Schreibzugriff von extern auf die Zonendaten. Damit wäre es Clients möglich, sich selbst im DNS einzutragen, was aus Sicherheitsgründen nicht wünschenswert ist. Ohne diesen Eintrag, sind Zonen-Updates generell untersagt, dieses Beispiel würde daran auch nichts ändern, da ! * ebenfalls alles verbietet.

14.6.6 Aufbau der Zonendateien

Man benötigt zwei Arten von Zonen-Dateien, die einen dienen dazu, einem Rechnernamen die IP-Adresse zu zuordnen und die anderen gehen den umgekehrten Weg und liefern zu einer gegebenen IP-Adresse den Rechnernamen.

Eine wichtige Bedeutung hat der . in den Zonendateien. Werden Rechnernamen, ohne abschließenden . angegeben, wird immer die Zone ergänzt. Man muss also komplette Rechnernamen, die bereits mit vollständiger Domain angegeben wurden, mit einem . abschließen, damit die Domain nicht noch einmal dran gehängt wird. Ein fehlender Punkt oder einer an der falschen Stelle, dürfte die häufigste Fehlerursache bei der Konfiguration von Nameservern sein.

Den ersten Fall betrachten wir die Zonen-Datei `welt.zone`, die für die Domain `welt.all` zuständig ist; vgl. Datei 14.15 auf der nächsten Seite.

Beispiel 14.15: Datei */var/lib/named/welt.zone*

```
1 $TTL 2D
2 welt.all.      IN SOA      gateway root.welt.all. (
3                2003072441  ; serial
4                1D          ; refresh
5                2H          ; retry
6                1W          ; expiry
7                2D )        ; minimum
8
9                IN NS      gateway
10               IN MX      10 sonne
11
12 gateway       IN A        192.168.0.1
13               IN A        192.168.1.1
14 sonne         IN A        192.168.0.2
15 mond         IN A        192.168.0.3
16 erde         IN A        192.168.1.2
17 mars         IN A        192.168.1.3
18 www          IN CNAME     mond
```

Zeile 1: \$TTL definiert die Standard-TTL (engl. *Time To Live*), also zu deutsch Gültigkeitsdauer, die für alle Einträge in dieser Datei gilt: hier 2 Tage (2D = 2 days).

Zeile 2: Hier beginnt der SOA control record:

- An erster Stelle steht hier der Name der zu verwaltenden Domain `welt.all`, diese ist mit einem `.` abgeschlossen, da ansonsten die Zone noch einmal angehängt würde. Alternativ kann man hier ein `@` schreiben, dann wird die Zone dem zugehörigen Eintrag in der `/etc/named.conf` entnommen.
- Nach dem `IN SOA` steht der Name des Nameservers, der als Master für diese Zone zuständig ist. In diesem Fall wird der Name `gateway` zu `gateway.welt.all` ergänzt, da er nicht mit einem `.` abgeschlossen ist.
- Danach folgt eine E-Mail-Adresse, der für diesen Nameserver zuständigen Person. Da das `@`-Zeichen bereits eine besondere Bedeutung hat, ist hier stattdessen einfach ein `.` zu setzen, für `root@welt.all` trägt man hier folglich `root.welt.all.` ein. Den `.` am Ende darf man hier nicht vergessen, da sonst die Zone noch angehängt würde.

- Am Ende folgt eine (, um die folgenden Zeilen, bis zur) mit in den SOA-Record einzuschließen.

Zeile 3: Die `serial number` ist eine willkürliche Zahl, die bei jeder Änderung an dieser Datei erhöht werden sollte. Sie wird benötigt, um sekundäre Nameserver (Slave-Server) über Änderungen zu informieren. Eingebürgert hat sich dafür eine zehnstellige Zahl aus Datum und fortlaufender Nummer in der Form `JJJJMMTTNN`.

Zeile 4: Die `refresh rate` gibt das Zeitintervall an, in dem Sekundär-Nameserver die `serial number` der Zone überprüfen. In diesem Fall 1 Tag (1D = 1 day).

Zeile 5: Die `retry rate` gibt den Zeitabstand an, in dem ein sekundärer Nameserver, im Fehlerfall versucht den primären Server erneut zu kontaktieren. Hier 2 Stunden (2H = 2 hours).

Zeile 6: Die `expiration time` gibt den Zeitraum an, nachdem ein sekundärer Nameserver die gecachten Daten verwirft, wenn er keinen Kontakt zum primären Server mehr bekommen hat. Hier ist das eine Woche (1W = 1 week).

Zeile 7: Der letzte Eintrag im SOA ist die `negative caching TTL`. Er sagt aus, wie lange die Ergebnisse von DNS-Anfragen von anderen Servern gecached werden dürfen, die nicht aufgelöst werden konnten.

Zeile 9: Das `IN NS` gibt den Nameserver an, der für diese Domain zuständig ist. Auch hier gilt, dass `gateway` wieder zu `gateway.welt.all` ergänzt wird, weil es nicht mit einem `.` abgeschlossen ist. Es kann mehrere Zeilen dieser Art geben, eine für den primären und jeweils eine für jeden sekundären Nameserver. Ist für diese Zone `notify` in der `/etc/named.conf` nicht auf `no` gesetzt, werden alle hier aufgeführten Nameserver über Änderungen der Zonendaten informiert.

Zeile 10: Der MX-Record gibt den Mailserver an, der für die Domain `welt.all` die Mails annimmt und weiterverarbeitet oder weiterleitet. In diesem Beispiel ist das der Rechner `sonne.welt.all`. Die Zahl vor dem Rechnernamen ist der Präferenz-Wert, gibt es mehrere MX-Einträge, wird zuerst der Mailserver mit dem kleinsten Wert genommen und falls die Auslieferung an diesen scheitert, wird der mit dem nächst höheren Wert versucht.

Zeile 12-17: Das sind jetzt die eigentlichen Adressen-Einträge (engl. *Address Records*), in denen den Rechnernamen eine oder mehrere IP-Adressen zugeordnet werden. Die Namen stehen hier ohne abschließenden `.`, da sie ohne angehängte Domain eingetragen sind und alle um `welt.all` ergänzt werden dürfen. Dem Rechner `gateway` sind zwei IP-Adressen zugeordnet, da er über zwei Netzwerkkarten verfügt. Das `A` steht jeweils für eine traditionelle Rechner-Adresse; mit `A6` trägt man IPv6-Adressen ein und `AAAA` ist das obsoleete Format für IPv6-Adressen.

Zeile 18: Mit dem Alias `www` kann auch `mond` (`CNAME = canonical name`) angesprochen werden.

Für die Rückwärts-Auflösung (engl. *reverse lookup*) von IP-Adressen in Rechnernamen wird die Pseudo-Domain `in-addr.arpa` zu Hilfe genommen. Diese wird dazu an den in umgekehrter Reihenfolge geschriebenen Netzanteil angehängt. Aus `192.168.1` wird dann `1.168.192.in-addr.arpa`.

Beispiel 14.16: Umgekehrte Adress-Auflösung

```

1
2 $TTL 2D
3 1.168.192.in-addr.arpa. IN SOA gateway.welt.all. root.welt.all. (
4                           2003072441      ; serial
5                           1D                ; refresh
6                           2H                ; retry
7                           1W                ; expiry
8                           2D )              ; minimum
9
10                          IN NS            gateway.welt.all.
11
12 1                          IN PTR          gateway.welt.all.
13 2                          IN PTR          erde.welt.all.
14 3                          IN PTR          mars.welt.all.
```

Zeile 1: `$TTL` definiert die Standard-TTL, die hier für alle Einträge gilt.

Zeile 2: Der „Reverse Lookup“ soll mit dieser Datei für das Netz `192.168.1.0` ermöglicht werden. Da die Zone hier `1.168.192.in-addr.arpa` heißt, will man dies natürlich nicht an die Rechnernamen anhängen, deshalb sind diese alle komplett mit Domain und abschließendem `.` eingetragen. Der Rest entspricht dem, was im vorangegangenen Beispiel für `welt.all`, bereits beschrieben wurde.

Zeile 3-7: Siehe vorangegangenes Beispiel für `welt.all`.

Zeile 9: Diese Zeile gibt auch hier wieder den Nameserver an, der für diese Zone zuständig ist, diesmal wird aber der Name komplett mit Domain und abschließendem `.` hier eingetragen.

Zeile 11-13: Das sind die Pointer-Records, die zu einer IP-Adresse auf den zugehörigen Rechnernamen zeigen. Hier steht am Anfang der Zeile nur die letzte Stelle der IP-Adresse, ohne abschließenden `.` Wird jetzt die Zone daran angehängt und man denkt sich das `.in-addr.arpa` weg, hat man die komplette IP-Adresse in umgekehrter Reihenfolge.

Zonentransfers zwischen den verschiedenen Versionen von BIND sollten normalerweise kein Problem darstellen.

14.6.7 Sichere Transaktionen

Sichere Transaktionen kann man mithilfe der „Transaction SIGnatures“ (TSIG) verwirklichen. Dafür kommen Transaktionsschlüssel (engl. *Transaction Keys*) und -signaturen (engl. *Transaction Signatures*) zum Einsatz, deren Erzeugung und Verwendung in diesem Abschnitt beschrieben wird.

Benötigt werden sichere Transaktionen bei der Kommunikation von Server zu Server und für dynamische Aktualisierungen der Zonendaten. Eine auf Schlüsseln basierende Zugriffskontrolle bietet dafür eine weit größere Sicherheit als eine Kontrolle, die auf IP-Adressen basiert.

Ein Transaktionsschlüssel kann mit folgendem Kommando erzeugt werden (für mehr Informationen vgl. die Manualpage von `dnssec-keygen`):

```
dnssec-keygen -a hmac-md5 -b 128 -n HOST host1-host2
```

Es entstehen dadurch zwei Dateien mit beispielsweise folgenden Namen:

```
Khost1-host2.+157+34265.private
Khost1-host2.+157+34265.key
```

Der Schlüssel ist in beiden Dateien enthalten (z.B. `eJIKuCyyGJwwuN3xAteKgg=`). Zur weiteren Verwendung sollte `Khost1-host2.+157+34265.key` auf sicherem Wege (zum Beispiel mit `scp`) auf den entfernten Rechner übertragen und dort in der `/etc/named.conf` eingetragen werden, um eine sichere Kommunikation zwischen `host1` und `host2` zu bewirken:

```
key host1-host2. {
    algorithm hmac-md5;
    secret "ejIkuCyyGJwwuN3xAteKgg==";
};
```

Achtung

Achten Sie darauf, dass die Zugriffsrechte auf `/etc/named.conf` eingeschränkt bleiben; die Vorgabe ist 0640 für root und die Gruppe named; alternativ kann man die Schlüssel auch in eine eigene geschützte Datei auslagern und diese dann includieren.

Achtung

Damit auf dem Server `host1` der Schlüssel für `host2` mit der Beispielsadresse `192.168.2.3` verwendet wird, muss auf dem Server in der `/etc/named.conf` eingetragen werden:

```
server 192.168.2.3 {
    keys { host1-host2. ; };
};
```

In den Konfigurationsdateien von `host2` müssen entsprechende Einträge vorgenommen werden.

Zusätzlich zu den ACLs auf Basis von IP-Adressen und Adress-Bereichen, soll man, um sichere Transaktionen auszuführen, TSIG-Schlüssel hinzufügen; ein Beispiel dafür kann so aussehen:

```
allow-update { key host1-host2. ;};
```

Mehr dazu findet man im *BIND Administrator Reference Manual* zu `update-policy`.

14.6.8 Zonendaten dynamisch aktualisieren

Dynamische Aktualisierungen (engl. *Dynamic Update*) ist der Terminus, der das Hinzufügen, Ändern oder Löschen von Einträgen in den Zonen-Dateien eines Masters bezeichnet. Beschrieben ist dieser Mechanismus im RFC 2136.

Dynamische Aktualisierungen werden je Zone mit den Optionen `allow-update` oder `update-policy` bei den Zonen-Einträgen konfiguriert. Zonen, die dynamisch aktualisiert werden, sollten nicht von Hand bearbeitet werden.

Mit `nsupdate` werden die zu aktualisierenden Einträge an den Server übertragen; zur genauen Syntax vgl. die Manualpage von `nsupdate`. Die Aktualisierung sollte aus Sicherheitsüberlegungen heraus unbedingt über sichere Transaktionen (TSIG) geschehen; vgl. Abschnitt 14.6.7 auf Seite 385.

14.6.9 DNSSEC

DNSSEC (engl. *DNS Security*) ist im RFC 2535 beschrieben; welche Tools für den Einsatz von DNSSEC zur Verfügung stehen, ist im BIND-Manual beschrieben.

Eine sichere Zone muss einen oder mehrere Zonen-Schlüssel haben; diese werden, wie die Host-Schlüssel, auch mit `dnssec-keygen` erzeugt. Zur Verschlüsselung wählt man momentan DSA.

Die öffentlichen Schlüssel (engl. *public keys*) sollten in die Zonen-Dateien mit `$INCLUDE` eingebunden werden.

Alle Schlüssel werden mit `dnssec-makekeyset` zu einem Set zusammengefasst, das auf sicherem Wege an die übergeordnete Zone (engl. *Parent Zone*) zu übertragen ist, um dort mit (engl. *dnssec-signkey*) signiert zu werden. Die bei der Signierung erzeugten Dateien müssen zum Signieren von Zonen mit `dnssec-signzone` verwendet werden und die dabei entstandenen Dateien sind schließlich in `/etc/named.conf` für die jeweilige Zone einzubinden.

14.6.10 Weitere Informationen

Hinzuweisen ist insbesondere auf das *BIND Administrator Reference Manual*, das online in `/usr/share/doc/packages/bind9/` zu finden ist, sowie auf die dort genannten RFCs und die mit BIND 9 mitgelieferten Manual-Pages.

14.7 LDAP – Ein Verzeichnisdienst

Innerhalb einer vernetzten Arbeitsumgebung ist es entscheidend, wichtige Informationen strukturiert und schnell abrufbar bereitzuhalten. Datenchaos droht nicht erst beim Benutzen des Internets. Ebenso schnell kann die Suche nach wichtigen Daten im betriebsinternen Netz ausarten: Was ist die Telefondurchwahl meines Kollegen XY? Wie lautet seine E-Mailadresse?

Dieses Problem löst ein Verzeichnisdienst, der ähnlich den Gelben Seiten (engl. *Yellow Pages*) im normalen Alltagsleben die gesuchten Informationen in gut strukturierter, schnell durchsuch- und abrufbarer Form bereithält.

Im Idealfall existiert ein zentraler Server, der die Daten in einem Verzeichnis vorhält und über ein bestimmtes Protokoll an alle Clients im Netzwerk verteilt. Die Daten sollten derart strukturiert sein, dass ein möglichst breites Spektrum von Anwendungen darauf zugreifen kann. So muss nicht jedes Kalendertool oder jeder E-Mailclient seine eigenen Datenbanken vorhalten, sondern kann auf den zentralen Bestand zurückgreifen. Dies verringert den Verwaltungsaufwand für die betreffenden Informationen beträchtlich. Die Verwendung eines offenen und standardisierten Protokolls wie LDAP stellt sicher, dass möglichst viele Clientapplikationen auf solche Informationen zugreifen können.

Ein Verzeichnis in diesem Kontext ist eine Art von Datenbank, die daraufhin optimiert ist, besonders gut und schnell les- und durchsuchbar zu sein:

- Um zahlreiche (gleichzeitige) Lesezugriffe zu ermöglichen, wird der Schreibzugriff auf einige wenige Aktualisierungen seitens des Administrators begrenzt. Herkömmliche Datenbanken sind daraufhin optimiert, in kurzer Zeit ein möglichst großes Datenvolumen aufzunehmen.
- Da Schreibzugriffe nur sehr eingeschränkt ausgeführt werden sollen, verwaltet man über einen Verzeichnisdienst möglichst unveränderliche, *statische* Informationen. Die Daten innerhalb einer konventionellen Datenbank ändern sich typischerweise sehr häufig (*dynamische* Daten). Telefonnummern in einem Mitarbeiterverzeichnis ändern sich nicht annähernd so häufig wie zum Beispiel die Zahlen, die in der Buchhaltung verarbeitet werden.
- Werden statische Daten verwaltet, sind Updates der bestehenden Datensätze sehr selten. Bei der Arbeit mit dynamischen Daten, besonders wenn es um Datensätze wie Bankkonten und Buchhaltung geht, steht die Konsistenz der Daten im Vordergrund. Soll eine Summe an einer Stelle abgebucht werden, um sie an anderer Stelle hinzuzufügen, müssen beide Operationen gleichzeitig – innerhalb einer „Transaktion“ ausgeführt werden, um die Ausgeglichenheit des gesamten Datenbestandes sicherzustellen. Datenbanken unterstützen solche Transaktionen, Verzeichnisse nicht. Kurzfristige Inkonsistenzen der Daten sind bei Verzeichnissen durchaus akzeptabel.

Das Design eines Verzeichnisdienstes wie LDAP ist nicht dazu ausgelegt, komplexe Update- oder Abfragemechanismen zu unterstützen. Alle auf diesen Dienst zugreifende Anwendungen sollen möglichst leicht und schnell Zugriff haben.

Verzeichnisdienste gab und gibt es, nicht nur in der Unix-Welt, viele. Novells NDS, Microsofts ADS, Banyans Street Talk und den OSI-Standard X.500.

LDAP war ursprünglich als eine schlanke Variante des DAP (engl. *Directory Access Protocol*) geplant, das für den Zugriff auf X.500 entwickelt wurde. Der X.500-Standard regelt die hierarchische Organisation von Verzeichniseinträgen.

LDAP ist um einige Funktionen des DAP erleichtert und kann plattformübergreifend und vor allem ressourcenschonend eingesetzt werden, ohne dass man auf die in X.500 definierten Eintragshierarchien verzichten müsste. Durch die Verwendung von TCP/IP ist es wesentlich einfacher, Schnittstellen zwischen aufsetzender Applikation und LDAP-Dienst zu realisieren.

Mittlerweile hat sich LDAP weiterentwickelt und kommt immer häufiger als Stand-alone-Lösung ohne X.500-Unterstützung zum Einsatz. Mit LDAPv3 (der Protokollversion, die Sie mit dem installierten Paket `openldap2` vorliegen haben) unterstützt LDAP so genannte *Referrals*, mit deren Hilfe sich verteilte Datenbanken realisieren lassen. Ebenfalls neu ist die Nutzung von SASL (engl. *Simple Authentication and Security Layer*) als Authentifizierungs- und Sicherungsschicht.

LDAP kann nicht nur zur Datenabfrage von X.500-Servern eingesetzt werden, wie ursprünglich geplant war. Es gibt mit `slapd` einen Open Source Server, mit dem Objektinformationen in einer lokalen Datenbank gespeichert werden können. Ergänzt wird er durch `slurpd`, der für die Replikation mehrerer LDAP-Server zuständig ist.

Das Paket `openldap2` besteht im Wesentlichen aus zwei Programmen.

slapd Ein Stand-alone-LDAPv3-Server, der Objektinformationen in einer BerkeleyDB-basierten Datenbank verwaltet.

slurpd Dieses Programm ermöglicht es, Änderungen an den Daten des lokalen LDAP-Servers an andere im Netz installierte LDAP-Server zu replizieren.

Zusätzliche Tools zur Systempflege

`slapcat`, `slapadd`, `slapindex`

14.7.1 LDAP versus NIS

Traditionell verwendet der Unix-Systemadministrator zur Namensauflösung und Datenverteilung im Netzwerk den NIS-Dienst. Auf einem zentralen Server werden die Konfigurationsdaten aus den `/etc/`-Dateien und Verzeichnissen `group/`, `hosts/`, `mail/`, `netgroup/`, `networks/`, `passwd/`, `printcap/`, `protocols/`, `rpc/` und `services/` über die Clients im Netz verteilt. Als bloße Textdateien sind diese Dateien ohne größeren Aufwand wartbar. Allerdings wird die Verwaltung größerer Datenmengen aufgrund mangelnder Strukturierung schwierig. NIS ist nur für Unix-Plattformen ausgelegt, was einen Einsatz als zentrale Datenverwaltung im heterogenen Netz unmöglich macht.

Das Einsatzgebiet des LDAP-Dienstes ist im Gegensatz zu NIS nicht auf reine Unix-Netze beschränkt. Windows Server (ab 2000) unterstützen LDAP als Verzeichnisdienst. Ebenso bietet auch Novell einen LDAP-Dienst an. Zudem ist er nicht auf die oben genannten Aufgabengebiete beschränkt.

Das LDAP-Prinzip kann für beliebige Datenstrukturen verwendet werden, die zentral verwaltet werden sollen. Einige Anwendungsbeispiele wären zum Beispiel:

- Einsatz anstelle eines NIS-Servers
- Mailrouting (postfix, sendmail)
- Adressbücher für Mailclients wie Mozilla, Evolution, Outlook, ...
- Verwaltung von Zonenbeschreibungen für einen BIND9-Nameserver

Diese Aufzählung kann beliebig fortgesetzt werden, da LDAP im Gegensatz zu NIS erweiterbar ist. Die klar definierte hierarchische Struktur der Daten hilft bei der Verwaltung sehr großer Datenmengen, da sie besser durchsuchbar ist.

14.7.2 Aufbau eines LDAP-Verzeichnisbaums

Ein LDAP-Verzeichnis hat eine baumartige Struktur. Alle Einträge (Objekte genannt) im Verzeichnis haben eine definierte Position innerhalb dieser Hierarchie. Diese Hierarchie wird als *Directory Information Tree* oder kurz DIT bezeichnet. Der komplette Pfad zum gewünschten Eintrag, der ihn eindeutig identifiziert, wird *Distinguished Name* oder DN genannt. Die einzelnen Knoten auf dem Weg zu diesem Eintrag werden *Relative Distinguished Name* oder RDN genannt. Objekte können generell zwei verschiedenen Typen zugeordnet werden:

Container Diese Objekte können wieder andere Objekte enthalten. Solche Objektklassen sind `Root` (Wurzelement des Verzeichnisbaums, das nicht real existiert), `c` (engl. *country*), `ou` (engl. *OrganizationalUnit*), und `dc` (engl. *domainComponent*). Vergleichbar ist dieses Modell auch mit Verzeichnissen (Ordern) im Dateisystem.

Blatt Diese Objekte sitzen am Ende eines Astes. Ihnen sind keine anderen Objekte untergeordnet. Beispiele sind `Person`, `InetOrgPerson` oder `groupofNames`.

An der Spitze der Verzeichnishierarchie liegt ein Wurzelement `Root`. Diesem können in der nächsten Ebene entweder `c` *country*, `dc` *domainComponent* oder `o` *organization* untergeordnet werden.

Die Beziehungen innerhalb eines LDAP-Verzeichnisbaums werden am folgenden Beispiel (siehe Abbildung 14.4) deutlich.

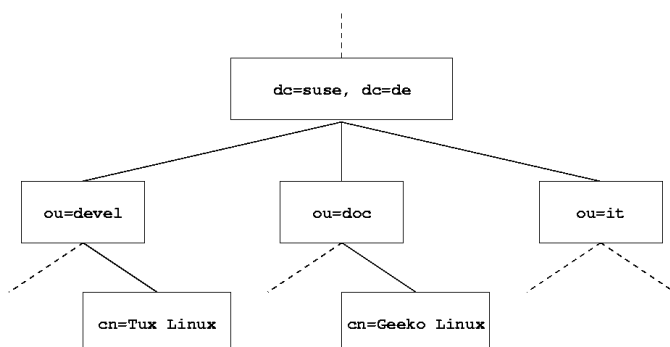


Abbildung 14.4: Aufbau eines LDAP-Verzeichnisses

Die gesamte Abbildung umfasst einen fiktiven *Directory Information Tree*. Abgebildet sind die Einträge (engl. *entries*) auf drei Ebenen. Jeder Eintrag entspricht in der Abbildung einem Kästchen. Der vollständige gültige *Distinguished Name* für den fiktiven SuSE-Mitarbeiter `Geeko Linux` ist in diesem Fall `cn=Geeko Linux,ou=doc,dc=suse,dc=de`. Er setzt sich zusammen, indem der RDN `cn=Geeko Linux` zum DN des Vorgängereintrags `ou=doc,dc=suse,dc=de` hinzugefügt wird.

Die globale Festlegung, welche Typen von Objekten im DIT gespeichert werden sollen, geschieht über ein *Schema*. Der Typ eines Objekts wird durch die *Objektklasse* festgelegt. Die Objektklasse bestimmt, welche Attribute dem betreffenden Objekt zugeordnet werden müssen bzw. können.

Ein Schema muss demnach Definitionen aller Objektklassen und Attribute enthalten, die im gewünschten Einsatzszenario verwendet werden. Es existieren einige allgemein gebräuchliche Schemata (siehe RFC 2252 und 2256). Allerdings können auch benutzerdefinierte Schemata geschaffen werden oder mehrere Schemata ergänzend zueinander verwendet werden, wenn es die Umgebung erfordert, in der der LDAP-Server betrieben werden soll.

Tabelle 14.9 gibt einen kleinen Überblick über die im Beispiel verwendeten Objektklassen aus `core.schema` und `inetorgperson.schema` samt zwingend erforderlicher Attribute und den passender Attributwerte.

Tabelle 14.9: Häufig verwendete Objektklassen und Attribute

Objektklasse	Bedeutung	Beispieleintrag	erforderl. Attribute
dcObject	<i>domainComponent</i> (Namensbestandteile der Domain)	suse	dc
organizationalUnit	<i>organizationalUnit</i> (Organisationseinheit)	doc	ou
inetOrgPerson	<i>inetOrgPerson</i> (Personenbezogene Daten für Intra-/Internet)	Geeko Linux	sn und cn

In Beispiel 14.17 sehen Sie einen Auszug aus einer Schema-Anweisung mit Erklärungen, der Ihnen beim Verstehen der Syntax neuer Schemata hilft.

Beispiel 14.17: Auszug aus `schema.core` (Zeilennummerierung aus Verständnisgründen)

```
...
#1 attributetype ( 2.5.4.11 NAME ( 'ou' 'organizationalUnitName' )
#2     DESC 'RFC2256: organizational unit this object belongs to'
#3     SUP name )

...
#4 objectclass ( 2.5.6.5 NAME 'organizationalUnit'
#5     DESC 'RFC2256: an organizational unit'
#6     SUP top STRUCTURAL
#7     MUST ou
#8     MAY ( userPassword $ searchGuide $ seeAlso $ businessCategory $
```

```

xl21Address $ registeredAddress $ destinationIndicator $
preferredDeliveryMethod $ telexNumber $
teletexTerminalIdentifier $ telephoneNumber $
internationalISDNNumber $ facsimileTelephoneNumber $
street $ postOfficeBox $ postalCode $ postalAddress $
physicalDeliveryOfficeName $ st $ l $ description) )

```

...

Als Beispiel dient der Attributtyp `organizationalUnitName` und die zugehörige Objektklasse `organizationalUnit`. In Zeile 1 wird der Name des Attributs, sein eindeutiger OID (*Object Identifier*) (numerisch) sowie das Kürzel des Attributs gelistet. In Zeile 2 wird mit `DESC` eine kurze Beschreibung des Attributs eingeleitet. Hier ist auch der zugehörige RFC genannt, auf den die Definition zurückgeht. `SUP` in Zeile 3 weist auf einen übergeordneten Attributtyp hin, zu dem dieses Attribut gehört.

Die Definition der Objektklasse `organizationalUnit` beginnt in Zeile 4 wie bei der Attributsdefinition mit einem OID und dem Namen der Objektklasse. In Zeile 5 lesen Sie eine Kurzbeschreibung der Objektklasse. Zeile 6 mit dem Eintrag `SUP top` besagt, dass diese Objektklasse keine Unterklasse einer anderen Objektklasse ist. Zeile 7, beginnend mit `MUST`, führt alle Attributtypen auf, die zwingend in einem Objekt vom Typ `organizationalUnit` verwendet werden *müssen*. In Zeile 8 sind nach `MAY` alle Attributtypen gelistet, die in Zusammenhang mit dieser Objektklasse verwendet werden *können*.

Eine sehr gute Einführung in den Umgang mit Schemata finden Sie in der Dokumentation zu OpenLDAP in Ihrem installierten System unter `/usr/share/doc/packages/openldap2/admin-guide/index.html`.

14.7.3 Serverkonfiguration mit `slapd.conf`

Ihr installiertes System enthält unter `/etc/openldap/slapd.conf` eine vollständige Konfigurationsdatei für Ihren LDAP-Server. Im Folgenden werden die einzelnen Einträge kurz beleuchtet und notwendige Anpassungen erklärt. Beachten Sie, dass Einträge mit führendem `#` inaktiv sind. Um solche Einträge zu aktivieren, entfernen Sie dieses Kommentarzeichen.

Globale Anweisungen in slapd.conf

Beispiel 14.18: slapd.conf: Include-Anweisung für Schemata

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/inetorgperson.schema
```

Mit dieser ersten Anweisung in slapd.conf wird das Schema spezifiziert, nach dem Ihr LDAP-Verzeichnis organisiert ist (siehe Beispiel 14.18). Der Eintrag core.schema ist zwingend erforderlich. Sollten Sie weitere Schemata benötigen, fügen Sie sie hinter dieser Anweisung ein (als Beispiel wurde hier inetorgperson.schema hinzugefügt). Weitere verfügbare Schemata finden Sie im Verzeichnis /etc/openldap/schema/. Soll NIS durch einen analogen LDAP-Dienst ersetzt werden, binden Sie hier die Schemata cosine.schema und rfc2307bis.schema ein. Informationen zu dieser Problematik entnehmen Sie der mitgelieferten OpenLDAP-Dokumentation.

Beispiel 14.19: slapd.conf: pidfile und argsfile

```
pidfile /var/run/slapd.pid
argsfile /var/run/slapd.args
```

Diese zwei Dateien enthalten die PID (engl. *process id*) und einige Argumente, mit denen der slapd Prozess gestartet wird. An dieser Stelle ist keine Änderung erforderlich.

Beispiel 14.20: slapd.conf: Zugangskontrolle

```
# Sample Access Control
#   Allow read access of root DSE
#   Allow self write access
#   Allow authenticated users read access
#   Allow anonymous users to authenticate
#
access to dn="" by * read
access to *
    by self write
    by users read
    by anonymous auth
#
# if no access controls are present, the default is:
#   Allow read by all
#
# rootdn can always write!
```

Beispiel 14.20 auf der vorherigen Seite ist der Ausschnitt aus `slapd.conf`, der die Zugangskontrolle zum LDAP-Verzeichnis auf dem Server regelt. Die Einstellungen, die hier im globalen Abschnitt der `slapd.conf` gemacht werden, gelten, soweit nicht im datenbankspezifischen Abschnitt eigene Zugangsregeln aufgestellt werden, die sie überschreiben. So wie hier wiedergegeben, können alle Benutzer lesend auf das Verzeichnis zugreifen, aber nur der Administrator (`rootdn`) kann auf diesem Verzeichnis schreiben. Das Regeln der Zugriffsrechte unter LDAP ist ein sehr komplexer Prozess. Daher hier einige Grundregeln, die Ihnen helfen, diesen Vorgang nachzuvollziehen.

- Jede Zugangsregel ist folgendermaßen aufgebaut:

```
access to <what> by <who> <access>
```

- *<what>* steht für das Objekt oder Attribut, zu dem Sie Zugang gewähren. Sie können einzelne Verzeichnisäste explizit durch separate Regeln schützen oder aber mit Hilfe regulärer Ausdrücke ganze Regionen des Verzeichnisbaums mit einer Regel abarbeiten. `slapd` wird alle Regeln in der Reihenfolge evaluieren, in der diese in der Konfigurationsdatei eingeführt wurden. Demnach führen Sie allgemeinere Regeln immer hinter spezifischeren auf. Die erste Regel, die `slapd` als zutreffend bewertet, wird ausgewertet und alle folgenden Einträge ignoriert.
- *<who>* legt fest, wer Zugriff auf die unter *<what>* festgelegten Bereiche erhalten soll. Auch hier können Sie durch die Verwendung passender regulärer Ausdrücke viel Aufwand sparen. Wiederum wird `slapd` nach dem ersten „Treffer“ mit der Auswertung von *<who>* abbrechen, d.h. spezifischere Regeln sollten wieder vor den allgemeineren aufgeführt werden. Folgende Einträge sind möglich (siehe Tabelle 14.10):

Tabelle 14.10: Zugangsberechtigte Benutzergruppen

Bezeichner	Bedeutung
*	ausnahmslos alle Benutzer
anonymous	nicht authentifizierte („anonyme“) Benutzer
users	authentifizierte Benutzer
self	Benutzer, die mit dem Zielobjekt verbunden sind

dn=<regex> Alle Benutzer, auf die dieser reguläre Ausdruck zutrifft

- `<access>` spezifiziert die Art des Zugriffs. Es wird hier unterschieden zwischen den in Tabelle 14.11 aufgeführten Möglichkeiten:

Tabelle 14.11: Zugriffsarten

Bezeichner	Bedeutung
none	Zutritt verboten
auth	zur Kontaktaufnahme mit dem Server
compare	zum vergleichenden Zugriff auf Objekte
search	zur Anwendung von Suchfiltern
read	Leserecht
write	Schreibrecht

slapd vergleicht die vom Client angeforderte Berechtigung mit der in `slapd.conf` gewährten. Werden dort höhere oder gleiche Rechte gewährt als der Client anfordert, wird dem Client der Zugang erlaubt. Fordert der Client höhere Rechte als dort angegeben, erhält er keinen Zugang.

Beispiel 14.21 zeigt ein einfaches Beispiel für eine Zugangskontrolle, die Sie durch Einsatz regulärer Ausdrücke beliebig ausgestalten können.

Beispiel 14.21: `slapd.conf`: Beispiel für Zugangskontrolle

```
access to dn.regex="ou=([^\,]+),dc=suse,dc=de"
by cn=administrator,ou=$1,dc=suse,dc=de write
by user read
by * none
```

Diese Regel besagt, dass zu allen `ou`-Einträgen nur der jeweilige Administrator schreibenden Zugang hat. Die übrigen authentifizierten Benutzer sind leseberechtigt und der Rest der Welt erhält keinen Zugang.

Hinweis

Aufstellen von Access Regeln

Falls es keine `access` to Regel oder keine `by <who>` Anweisung greift, ist der Zugriff verboten. Nur explizit angegebene Zugriffsrechte werden gewährt. Für den Fall, dass keine einzige Regel aufgestellt wird, gilt das Standardprinzip: Schreibrecht für den Administrator und Leserecht für die übrige Welt.

Hinweis

Detailinformationen und eine Beispielkonfiguration für LDAP-Zugriffsrechte finden Sie in der Online-Dokumentation des installierten `openldap2`-Pakets. Neben der Möglichkeit, Zugriffskontrollen über die zentrale Serverkonfigurationsdatei (`slapd.conf`) zu verwalten, gibt es den Weg über ACIs (engl. *Access Control Information*). Mittels ACIs können die Zugangsinformationen zu einzelnen Objekten im LDAP-Baum selbst abgespeichert werden. Da diese Art der Zugangskontrolle noch nicht sehr verbreitet ist und von den Entwicklern als experimentell eingestuft wird, verweisen wir an dieser Stelle auf die entsprechende Dokumentation auf den Seiten des OpenLDAP-Projekts: <http://www.openldap.org/faq/data/cache/758.html>.

Datenbankspezifische Anweisungen in `slapd.conf`

Beispiel 14.22: `slapd.conf`: Datenbankspezifische Anweisungen

```
database ldbm
suffix "dc=suse,dc=de"
rootdn "cn=admin,dc=suse,dc=de"
# Cleartext passwords, especially for the rootdn, should
# be avoided. See slapd.conf(5) and slapd.conf(5) for details.
# Use of strong authentication encouraged. rootpw secret
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd/tools. Mode 700 recommended.
directory /var/lib/ldap
# Indices to maintain
index objectClass eq
```

In der ersten Zeile dieses Abschnitts (siehe Beispiel 14.22) wird der Datenbanktyp festgelegt, hier LDBM. Über `suffix` in der zweiten Zeile wird festgelegt, für welchen Teil des LDAP-Verzeichnisbaumes dieser Server

verantwortlich sein soll. Das folgende `rootdn` legt fest, wer Administratorzugriff auf diesen Server besitzt. Der hier angegebene Benutzer muss keinen LDAP-Eintrag besitzen oder als „normaler“ Benutzer existieren. Mit der `rootpw` Anweisung wird das Administratorpasswort gesetzt. Sie können hier statt `secret` auch den mit `slappasswd` erzeugten Hash des Administratorpassworts eintragen. Die `directory` Anweisung gibt das Verzeichnis an, in dem die Datenbankverzeichnisse auf dem Server abgelegt sind. Die letzte Anweisung, `index objectClass eq`, bewirkt, dass ein Index über die Objektklassen gepflegt wird. Ergänzen Sie hier unter Umständen einige Attribute, nach denen Ihrer Erfahrung nach am häufigsten gesucht wird. Wenn nachgestellt für die Datenbank eigene Access Regeln definiert werden, werden diese statt der globalen Access Regeln angewendet.

Start und Stopp des Servers

Ist der LDAP-Server fertig konfiguriert und sind alle gewünschten Einträge im LDAP-Verzeichnis nach dem unten beschriebenen Muster (siehe Abschnitt 14.7.4) erfolgt, starten Sie den LDAP-Server als Benutzer `root` durch Eingabe des folgenden Befehls:

```
rcldap start
```

Möchten Sie den Server manuell wieder stoppen, geben Sie entsprechend `rcldap stop` ein. Die Statusabfrage über den Laufzustand des LDAP-Servers nehmen Sie mit `rcldap status` vor. Um Start und Stopp des Servers beim Starten bzw. Herunterfahren des betreffenden Rechners zu automatisieren, nutzen Sie den YaST Runlevel-Editor (vergleiche Abschnitt 13.5 auf Seite 334) oder Sie legen die entsprechenden Links der Start- und Stoppskripten mittels `insserv` auf der Kommandozeile selbst an (siehe Abschnitt 13.4.1 auf Seite 332).

14.7.4 Handhabung von Daten im LDAP-Verzeichnis

OpenLDAP gibt Ihnen als Administrator eine Reihe von Programmen an die Hand, mit denen Sie die Daten im LDAP-Verzeichnis verwalten können. Im Folgenden werden die vier wichtigsten von ihnen zum Hinzufügen, Löschen, Durchsuchen und Verändern des Datenbestandes kurz behandelt.

Daten in ein LDAP-Verzeichnis eintragen

Vorausgesetzt, die Konfiguration Ihres LDAP-Servers in `/etc/openldap/slapd.conf` ist korrekt und einsatzfähig, d.h. sie enthält die passenden Angaben für `suffix`, `directory`, `rootdn`, `rootpw` und `index`, können Sie nun mit der Aufnahme von Einträgen beginnen. OpenLDAP bietet hierfür den Befehl `ldapadd`. Aus praktischen Gründen sollten Sie Objekte nach Möglichkeit gebündelt zur Datenbank hinzufügen. Zu diesem Zweck kennt LDAP das so genannte LDIF-Format (engl. *LDAP Data Interchange Format*). Eine LDIF-Datei ist eine einfache Textdatei, die aus beliebig vielen Attribut-Wert-Paaren bestehen kann. Für die zur Verfügung stehenden Objektklassen und Attribute schauen Sie in den in `slapd.conf` angegebenen Schemadateien nach. Die LDIF-Datei zum Anlegen eines groben Gerüsts für das Beispiel aus Abbildung 14.4 auf Seite 391 sähe folgendermaßen aus (siehe Beispiel 14.23):

Beispiel 14.23: Beispiel für eine LDIF-Datei

```
# Die Organisation SUSE
dn: dc=suse,dc=de
objectClass: dcObject
objectClass: organization
o: SUSE AG
dc: suse

# Die Organisationseinheit Entwicklung (devel)
dn: ou=devel,dc=suse,dc=de
objectClass: organizationalUnit
ou: devel

# Die Organisationseinheit Dokumentation (doc)
dn: ou=doc,dc=suse,dc=de
objectClass: organizationalUnit
ou: doc

# Die Organisationseinheit Interne EDV (it)
dn: ou=it,dc=suse,dc=de
objectClass: organizationalUnit
ou: it
```

Hinweis

Kodierung der LDIF-Dateien

LDAP arbeitet mit UTF-8 (Unicode). Umlaute müssen demnach bei der Eingabe korrekt kodiert werden. Verwenden Sie einen Editor, der UTF-8 unterstützt (Kate oder neuere Versionen des Emacs). Andernfalls müssten Sie auf die Eingabe von Umlauten verzichten oder recode zum Umkodieren Ihrer Eingaben nach UTF-8 verwenden.

Hinweis

Speichern Sie die Datei unter `<datei>.ldif` ab und übergeben Sie sie mit folgendem Befehl an den Server:

```
ldapadd -x -D <dn des Administrators> -W -f <datei>.ldif
```

Die erste Option `-x` gibt an, dass in diesem Fall auf Authentifizierung über SASL verzichtet wird. `-D` kennzeichnet den Benutzer, der diese Operation vornimmt; hier geben Sie den gültigen DN des Administrators an, wie sie in `slapd.conf` konfiguriert wurde. Im konkreten Beispiel wäre dies `cn=admin,dc=suse,dc=de`. Mit `-W` umgehen Sie die Eingabe des Passworts auf der Kommandozeile (Klartext) und aktivieren eine separate Passwortabfrage. Das betreffende Passwort wurde vorher in `slapd.conf` unter `rootpw` eingerichtet. `-f` übergibt die Datei. In Beispiel 14.24 sehen Sie Aufruf von `ldapadd` im Detail.

Beispiel 14.24: ldapadd von beispiel.ldif

```
ldapadd -x -D cn=admin,dc=suse,dc=de -W -f beispiel.ldif
```

```
Enter LDAP password:
adding new entry "dc=suse,dc=de"
adding new entry "ou=devel,dc=suse,dc=de"
adding new entry "ou=doc,dc=suse,dc=de"
adding new entry "ou=it,dc=suse,dc=de"
```

Die Benutzerdaten der einzelnen Mitarbeiter können Sie in separaten LDIF-Dateien angeben. Mit dem folgenden Beispiel `tux.ldif` (siehe Beispiel 14.25 auf der nächsten Seite) wird der Mitarbeiter Tux dem neuen LDAP-Verzeichnis hinzugefügt:

Beispiel 14.25: LDIF-Datei für Tux

```
# Der Mitarbeiter Tux
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
objectClass: inetOrgPerson
cn: Tux Linux
givenName: Tux
sn: Linux
mail: tux@suse.de
uid: tux
telephoneNumber: +49 1234 567-8
```

Eine LDIF-Datei kann beliebig viele Objekte enthalten. Sie können ganze Verzeichnisbäume am Stück an den Server übergeben oder auch nur Teile davon wie zum Beispiel einzelne Objekte. Wenn Sie Ihre Daten relativ häufig ändern müssen, empfiehlt sich eine feine Stückelung in einzelne Objekte, da Ihnen dann das mühsame Suchen nach dem zu ändernden Objekt in einer großen Datei erspart bleibt.

Daten im LDAP-Verzeichnis ändern

Stehen in Ihrem Datensatz Änderungen an, verwenden Sie das Tool `ldapmodify`. Am einfachsten ändern Sie zuerst die betreffende LDIF-Datei und übergeben anschließend die geänderte Datei wieder an den LDAP-Server. Um zum Beispiel die Telefonnummer des Mitarbeiters Tux von +49 1234 567-8 auf +49 1234 567-10 zu ändern, editieren Sie die LDIF-Datei wie in Beispiel 14.26 gezeigt.

Beispiel 14.26: Geänderte LDIF Datei *tux.ldif*

```
# Der Mitarbeiter Tux
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

Die geänderte Datei importieren Sie mit dem folgenden Befehl in das LDAP-Verzeichnis:

```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W -f tux.ldif
```

Alternativ können Sie `ldapmodify` auch direkt die zu ändernden Attribute auf der Kommandozeile angeben. Hierbei gehen Sie wie folgt vor:

- Rufen Sie `ldapmodify` auf und geben Sie Ihr Passwort ein:

```
ldapmodify -x -D cn=admin,dc=suse,dc=de -W
```

Enter LDAP password:

- Geben Sie Ihre Änderungen nach der folgenden Syntax in genau dieser Reihenfolge an:

```
dn: cn=Tux Linux,ou=devel,dc=suse,dc=de
changetype: modify
replace: telephoneNumber
telephoneNumber: +49 1234 567-10
```

Detaillierte Informationen zu `ldapmodify` und zur Syntax lesen Sie in der Manualpage von `ldapmodify` nach.

Daten aus einem LDAP-Verzeichnis suchen oder auslesen

OpenLDAP bietet mit `ldapsearch` ein Kommandozeilenwerkzeug zum Durchsuchen und Auslesen von Daten im LDAP-Verzeichnis. Ein einfaches Suchkommando hätte folgende Syntax:

```
ldapsearch -x -b dc=suse,dc=de "(objectClass=*)"
```

Die Option `-b` legt die Suchbasis, d.h. den Baumbereich, in dem gesucht werden soll, fest. In diesem Fall ist dies `dc=suse,dc=de`. Möchten Sie eine verfeinerte Suche auf bestimmten Unterbereichen des LDAP-Verzeichnisses ausführen (z.B. nur über die Abteilung `devel`), übergeben Sie diesen Bereich mittels `-b` an. `ldapsearch -x` legt die Verwendung einfacher Authentifizierung fest. Mit `(objectClass=*)` legen Sie fest, dass Sie alle in Ihrem Verzeichnis enthaltenen Objekte auslesen wollen. Verwenden Sie dieses Kommando nach dem Aufbau eines neuen Verzeichnisbaumes, um zu überprüfen, ob alle Ihre Einträge korrekt übernommen wurden und der Server in der gewünschten Form antwortet. Weitere Informationen zum Gebrauch von `ldapsearch` finden Sie in entsprechenden Manualpage (`man ldapsearch`).

Daten aus einem LDAP-Verzeichnis löschen

Löschen Sie nicht mehr erwünschte Einträge mittels `ldapdelete`. Die Syntax ähnelt der der oben beschriebenen Kommandos. Um beispielsweise den Eintrag von Tux Linux im Ganzen zu löschen geben Sie folgendes Kommando ein:

```
ldapdelete -x -D cn=admin,dc=suse,dc=de -W cn=Tux \
Linux,ou=devel,dc=suse,dc=de
```

14.7.5 Weitere Informationen

Komplexere Themen wie die SASL-Konfiguration oder das Aufsetzen eines replizierenden LDAP-Servers, der sich die Arbeit mit mehreren „slaves“ teilt, wurden in diesem Kapitel bewusst ausgeklammert. Detaillierte Informationen zu beiden Themen finden Sie im *OpenLDAP 2.1 Administrator's Guide* (Links siehe unten).

Auf den Webseiten des OpenLDAP-Projekts stehen ausführliche Dokumentationen für Anfänger und fortgeschrittene LDAP-Benutzer bereit:

OpenLDAP Faq-O-Matic Eine sehr ergiebige Frage- und Antwortsammlung rund um Installation, Konfiguration und Benutzung von OpenLDAP: <http://www.openldap.org/faq/data/cache/1.html>

Quick Start Guide Eine knappe Schritt-für-Schritt-Anleitung zum ersten eigenen LDAP-Server: <http://www.openldap.org/doc/admin21/quickstart.html> oder im installierten System unter `/usr/share/doc/packages/openldap2/admin-guide/quickstart.html`.

OpenLDAP 2.1 Administrator's Guide

Eine ausführliche Einführung in alle wichtigen Bereiche der LDAP-Konfiguration inkl. Access Controls und Verschlüsselung: <http://www.openldap.org/doc/admin21/> oder im installierten System unter `/usr/share/doc/packages/openldap2/admin-guide/index.html`

Weiterhin beschäftigen sich folgende Redbooks von IBM mit dem Thema LDAP:

Understanding LDAP Eine sehr ausführliche, allgemeine Einführung in die Grundprinzipien von LDAP: <http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf>

LDAP Implementation Cookbook

Zielgruppe sind speziell Administratoren von *IBM SecureWay Directory*. Jedoch sind auch wichtige allgemeine Informationen zum Thema LDAP enthalten: <http://www.redbooks.ibm.com/redbooks/pdfs/sg245110.pdf>

Gedruckte, englischsprachige Literatur zu LDAP:

- Howes, Smith & Good: *Understanding and Deploying LDAP Directory Services*. Addison-Wesley, 2. Aufl., 2003. - (ISBN 0-672-32316-8)
- Hodges: *LDAP System Administration*. O'Reilly & Associates, 2003. - (ISBN 1-56592-491-6)

Ultimative Nachschlagewerke zum Thema LDAP sind die entsprechenden RFCs (engl. *Request for comments*) 2251 bis 2256.

14.8 NIS – Network Information Service

Sobald mehrere Unix-Systeme in einem Netzwerk auf gemeinsame Ressourcen zugreifen wollen, muss sichergestellt sein, dass Benutzer- und Gruppenkennungen auf allen Rechnern miteinander harmonieren. Das Netzwerk soll für den Anwender transparent sein. Egal welcher Rechner, der Anwender findet immer die gleiche Umgebung vor. Möglich wird dies durch die Dienste NIS und NFS. NFS dient der Verteilung von Dateisystemen im Netz und wird in Abschnitt 14.9 auf Seite 408 beschrieben.

NIS (engl. *Network Information Service*) kann als Datenbankdienst verstanden werden, der Zugriff auf Informationen aus den Dateien */etc/passwd*, */etc/shadow* oder */etc/group* netzwerkweit ermöglicht. NIS kann auch für weitergehende Aufgaben eingesetzt werden (zum Beispiel für */etc/hosts* oder */etc/services*). Darauf soll hier jedoch nicht im Detail eingegangen werden. Für NIS wird vielfach synonym der Begriff *YP* verwendet. Dieser leitet sich ab von den *yellow pages*, also den *gelben Seiten* im Netz.

14.8.1 NIS Master und Slave Server

Zur Konfiguration wählen Sie in YaST 'Netzwerkdienste' und dort 'NIS-Server'. Wenn in Ihrem Netzwerk bisher noch kein NIS-Server existiert,

müssen Sie in der nächsten Maske den Punkt 'NIS Master Server installieren und einrichten' aktivieren. Falls Sie schon einen NIS-Server (also einen „Master“) haben, können Sie (beispielsweise wenn Sie ein neues Subnetz einrichten) einen NIS Slave-Server hinzufügen. Zunächst wird die Konfiguration des Master-Servers erläutert. Falls nicht alle nötigen Pakete installiert sind, wird YaST Sie auffordern, die entsprechende CD oder DVD einzulegen, damit die Pakete automatisch nachinstalliert werden. In der ersten Konfigurationsmaske (Abbildung 14.5) geben Sie oben den Domainnamen ein. In der Checkbox darunter können Sie festlegen, ob der Rechner auch ein NIS-Client werden soll, also ob sich darauf auch Benutzer einloggen können, die dann ebenfalls die Daten vom NIS-Server erhalten.

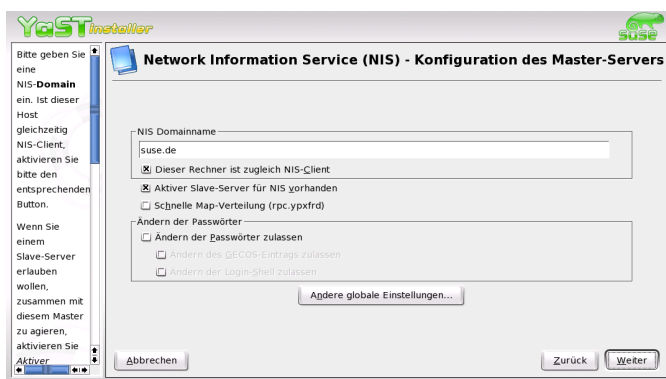


Abbildung 14.5: YaST: NIS-Server Konfigurationstool

Wollen Sie zusätzliche NIS-Server („Slave-Server“) in Ihrem Netzwerk einrichten, müssen Sie die Box 'Aktiver Slave-Server für NIS vorhanden' aktivieren. Zusätzlich sollten Sie dann auch die 'Schnelle Map-Verteilung' aktivieren, die bewirkt, dass die Datenbankeinträge sehr schnell vom Master auf die Slave-Server übertragen werden.

Wollen Sie den Nutzern in Ihrem Netzwerk erlauben, dass sie ihre Passwörter ändern können (mit dem Befehl `yppasswd`, also nicht nur die lokalen, sondern die, die auf dem NIS-Server abgelegt sind), können Sie das hier ebenfalls aktivieren. Dann werden auch die Checkboxes 'Ändern des GECOS-Eintrags zulassen' und 'Ändern des SHELL-Eintrags zulassen' aktiv. „GECOS“ bedeutet, der User kann auch seine Namens- und Adresseinstellungen ändern (mit dem Befehl `ypchfn`). „SHELL“ heisst, er darf auch seine standardmäßig eingetragene Shell ändern (mit dem Befehl `ypchsh`,

zum Beispiel von `bash` zu `sh`).

Durch Klick auf 'Andere globale Einstellungen...' gelangen Sie in einen Dialog (Abb. 14.6), in dem man das Quellverzeichnis des NIS-Servers (standardmäßig `/etc/`) ändern kann. Zusätzlich kann man hier noch Passwörter und Gruppen zusammenführen. Die Einstellung sollte man auf 'Ja' belassen, damit die jeweiligen Dateien (`/etc/passwd` und `/etc/shadow` bzw. `/etc/group`) aufeinander abgestimmt werden. Zusätzlich kann noch die jeweils kleinste Benutzer- und Gruppenkennung festgelegt werden. Mit 'OK' bestätigen Sie Ihre Eingaben und gelangen wieder in die vorige Maske zurück. Klicken Sie hier auf 'Weiter'.

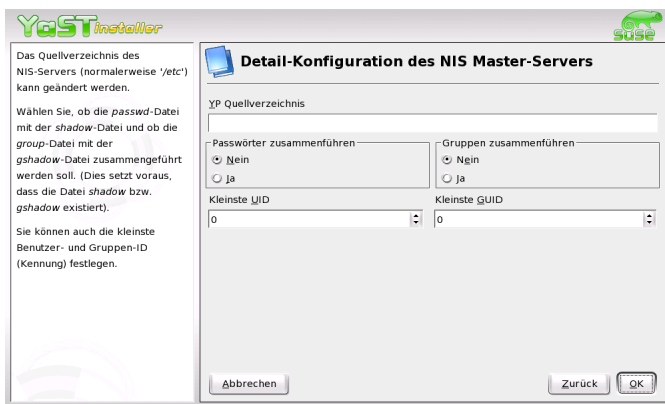


Abbildung 14.6: YaST: NIS-Server: Verzeichnis ändern und Dateien synchronisieren

Haben Sie vorher 'Aktiver Slave-Server für NIS vorhanden' aktiviert, müssen Sie nun die Namen der Rechner angeben, die als Slaves fungieren sollen. Anschließend klicken Sie auf 'Weiter'. Werden keine Slave-Server benutzt, belangen Sie direkt zum Dialog für die Datenbank-Einstellungen. Hier geben Sie die „Maps“ an, das heißt die Teildatenbanken, die vom NIS-Server auf den jeweiligen Client übertragen werden sollen. Die Voreinstellungen hier sind für die meisten Fälle sehr sinnvoll. Daher sollten Sie im Normalfall nichts ändern.

Mit 'Weiter' gelangen Sie in den letzten Dialog. Legen Sie fest, aus welchen Netzwerken Anfragen an den NIS-Server gestellt werden dürfen (siehe Abb. 14.7 auf der nächsten Seite). Normalerweise wird das Ihr Firmennetzwerk sein. Dann sollten die folgenden beiden Einträge hier stehen:

```
255.0.0.0 127.0.0.0
0.0.0.0   0.0.0.0
```

Der erste erlaubt Verbindungen vom eigenen Rechner, der zweite ermöglicht allen Rechnern, die Zugriff auf das Netzwerk haben, Anfragen an den Server.

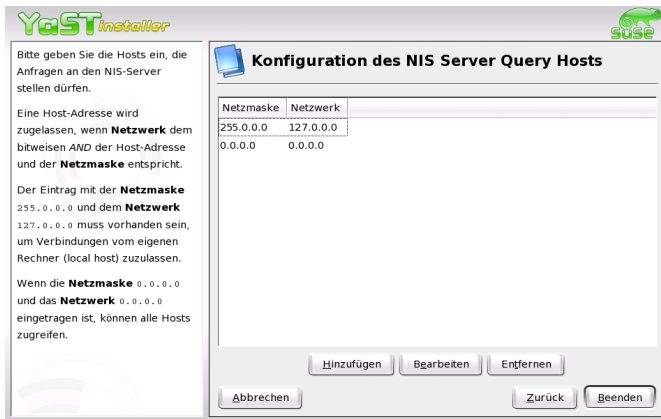


Abbildung 14.7: YaST: NIS-Server: Festlegen der Anfrage-Erlaubnis

14.8.2 Das NIS-Client-Modul in YaST

Mit diesem Modul können Sie sehr einfach den NIS-Client konfigurieren. Nachdem Sie sich in der Startmaske für die Verwendung von NIS und unter Umständen des Automounters entschieden haben, gelangen Sie in die nächste Maske. Geben Sie hier an, ob der NIS-Client eine statische IP-Adresse hat oder ob er diese über DHCP erhalten soll. In letzterem Fall können Sie keine NIS-Domain oder IP-Adresse des Servers angeben, da diese Daten ebenfalls über DHCP zugewiesen werden. Weitere Information zu DHCP finden Sie im Abschnitt 14.10 auf Seite 413. Falls der Client über eine feste IP-Adresse verfügt, müssen NIS-Domain und -Server manuell eingetragen werden (siehe Abbildung 14.8 auf der nächsten Seite). Über den Button 'Suchen' kann YaST nach einem aktiven NIS-Server in Ihrem Netz suchen.

Sie haben auch die Möglichkeit, multiple Domains mit einer Default-Domain anzugeben. Für die einzelnen Domains können Sie wiederum mit

‘Hinzufügen’ mehrere Server einschließlich Broadcast-Funktion angeben. In den Experten-Einstellungen können Sie verhindern, dass ein anderer Rechner im Netz abfragen kann, welchen Server Ihr Client benutzt. Wenn Sie ‘Broken Server’ aktivieren, werden auch Antworten von einem Server auf einem unprivilegierten Port akzeptiert. Details dazu finden Sie in der Manual-Page von ypbind.



Abbildung 14.8: Angabe von Domain und Adresse des NIS-Servers

14.9 NFS – verteilte Dateisysteme

Wie bereits in Abschnitt 14.8 auf Seite 404 erwähnt, dient NFS neben NIS dazu, ein Netzwerk für Anwender transparent zu machen. Durch NFS lassen sich Dateisysteme im Netz verteilen. Unabhängig davon, an welchem Rechner im Netz ein Anwender arbeitet, findet er so stets die gleiche Umgebung vor.

Wie NIS ist auch NFS ein asymmetrischer Dienst. Es gibt NFS-Server und NFS-Clients. Allerdings kann ein Rechner beides sein, d.h. gleichzeitig Dateisysteme dem Netz zur Verfügung stellen („exportieren“) und Dateisysteme anderer Rechner mounten („importieren“). Im Regelfall jedoch

benutzt man dafür Server mit großer Festplattenkapazität, deren Dateisysteme von Clients gemountet werden.

14.9.1 Importieren von Dateisystemen mit YaST

Jeder Benutzer (der die Rechte dazu erteilt bekommt), kann NFS-Verzeichnisse von NFS-Servern in seinen eigenen Dateibaum einhängen. Dies lässt sich am einfachsten mit dem Modul 'NFS-Client' in YaST erledigen. Dort muss lediglich der Hostname des als NFS-Server fungierenden Rechners eingetragen werden, das Verzeichnis, das von dem Server exportiert wird und den Mountpunkt, unter dem es auf dem eigenen Computer eingehängt werden soll. Wählen Sie dazu im ersten Dialogfenster 'Hinzufügen' und tragen Sie dann die genannten Angaben ein (s. Abb. 14.9).



Abbildung 14.9: Konfiguration des NFS-Clients

14.9.2 Manuelles Importieren von Dateisystemen

Dateisysteme von einem NFS-Server manuell zu importieren, ist sehr einfach. Einzige Voraussetzung ist, dass der RPC-Portmapper läuft. Das Starten erledigen Sie durch Aufruf des Befehls `rpcportmap start` als Benutzer `root`. Ist diese Voraussetzung erfüllt, können fremde Dateisysteme, sofern sie von den entsprechenden Maschinen exportiert werden, analog zu lokalen Platten mit dem Befehl `mount` in das Dateisystem eingebunden werden. Die Syntax ist wie folgt:

```
mount Rechner:Remote-Pfad Lokaler-Pfad
```

Sollen also z.B. die Benutzerverzeichnisse vom Rechner `sun` importiert werden, so kann dies mit folgendem Befehl erreicht werden:

```
mount sun:/home /home
```

14.9.3 Exportieren von Dateisystemen mit YaST

Mit YaST können Sie einen Rechner Ihres Netzwerks zu einem NFS-Server machen. Das ist ein Server, der Verzeichnisse und Dateien für alle Rechner, denen Sie Zugang gewähren, bereitstellt. Viele Anwendungsprogramme können so z.B. für Mitarbeiter zur Verfügung gestellt werden, ohne dass sie lokal auf deren Rechnern installiert werden müssen.

Zur Installation wählen Sie in YaST 'Netzwerkdienste' und dort 'NFS-Server' (Abb. 14.10).

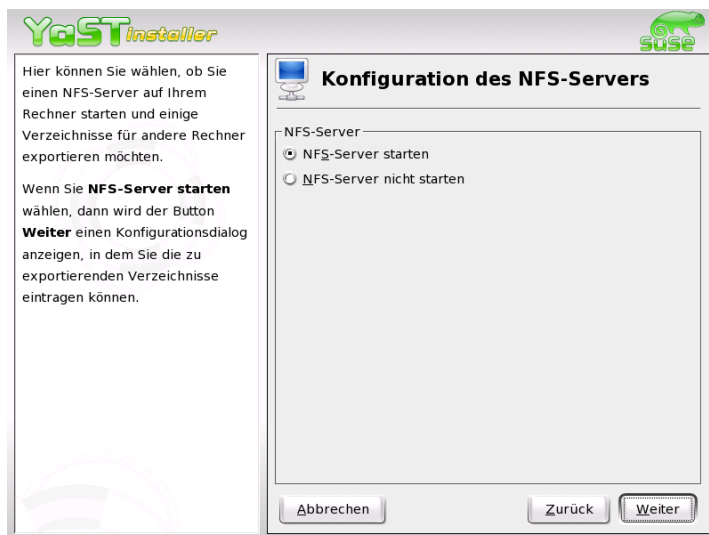


Abbildung 14.10: NFS-Server Konfigurationstool

Im nächsten Schritt aktivieren Sie 'NFS-Server starten' und klicken auf 'Weiter'. Jetzt ist nur noch ein Schritt zu tun: Sie müssen im oberen Feld die Verzeichnisse eintragen, die exportiert werden sollen und im unteren die Rechner Ihres Netzwerks, die darauf Zugriff erhalten (Abb. 14.11 auf der nächsten Seite). Zu den Rechnern sind jeweils vier Optionen einstellbar, single host, netgroups, wildcards und IP networks. Nähere Erläuterungen zu diesen Optionen finden Sie in den Man-Pages zu exports. Mit 'Beenden' schließen Sie die Konfiguration ab.

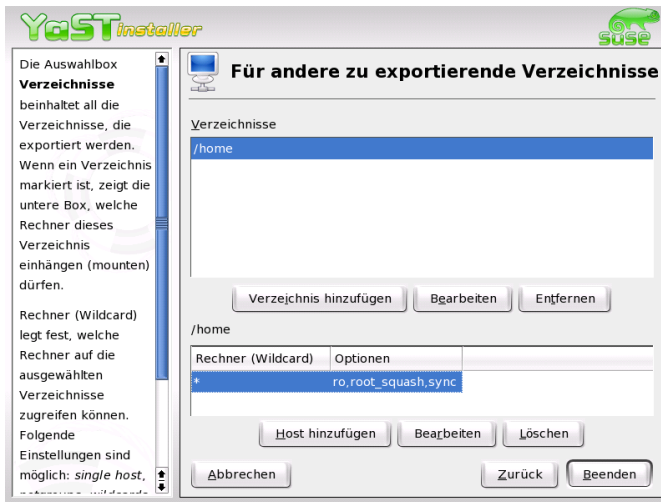


Abbildung 14.11: NFS-Server: Exportverzeichnisse und Hosts eintragen

14.9.4 Manuelles Exportieren von Dateisystemen

Wenn Sie auf die Unterstützung durch YaST verzichten, müssen Sie dafür sorgen, dass die folgenden Dienste auf dem NFS-Server laufen:

- RPC-Portmapper (portmap)
- RPC-Mount-Daemon (rpc.mountd)
- RPC-NFS-Daemon (rpc.nfsd)

Damit diese beim Hochfahren des Systems von den Skripten `/etc/init.d/portmap` und `/etc/init.d/nfsserver` gestartet werden, geben Sie bitte die Befehle `insserv /etc/init.d/nfsserver` und `insserv /etc/init.d/portmap` ein.

Neben dem Start dieser Daemons muss noch festgelegt werden, welche Dateisysteme an welche Rechner exportiert werden sollen. Dies geschieht in der Datei `/etc/exports`.

Je Verzeichnis, das exportiert werden soll, wird eine Zeile für die Information benötigt, welche Rechner wie darauf zugreifen dürfen. Alle Unterverzeichnisse eines exportierten Verzeichnisses werden ebenfalls automatisch

exportiert. Die berechtigten Rechner werden üblicherweise mit ihren Namen (inklusive Domainname) angegeben, es ist aber auch möglich, mit den Jokerzeichen * und ? zu arbeiten, die die aus der `bash` bekannte Funktion haben. Wird kein Rechnername angegeben, hat jeder Rechner die Erlaubnis, auf dieses Verzeichnis (mit den angegebenen Rechten) zuzugreifen.

Die Rechte, mit denen das Verzeichnis exportiert wird, werden in einer von Klammern umgebenen Liste nach dem Rechnernamen angegeben. Die wichtigsten Optionen für die Zugriffsrechte sind in der folgenden Tabelle beschrieben.

***Tabelle 14.12:** Zugriffsrechte für exportierte Verzeichnisse*

Optionen	Bedeutung
<code>ro</code>	Dateisystem wird nur mit Leserechten exportiert (Vorgabe).
<code>rw</code>	Dateisystem wird mit Schreib- und Leserechten exportiert.
<code>root_squash</code>	Diese Option bewirkt, dass der Benutzer <code>root</code> des angegebenen Rechners keine für <code>root</code> typischen Sonderrechte auf diesem Dateisystem hat. Erreicht wird dies, indem Zugriffe mit der User-ID 0 auf die User-ID 65534 (-2) umgesetzt werden. Diese User-ID sollte dem Benutzer <code>nobody</code> zugewiesen sein (Vorgabe).
<code>no_root_squash</code>	Rootzugriffe nicht umsetzen; Root-rechte bleiben also erhalten.
<code>link_relative</code>	Umsetzen von absoluten, symbolischen Links (solche, die mit / beginnen) in eine entsprechende Folge von . . /. Diese Option ist nur dann sinnvoll, wenn das gesamte Dateisystem eines Rechners gemountet wird (Vorgabe).
<code>link_absolute</code>	Symbolische Links bleiben unverändert.
<code>map_identity</code>	Auf dem Client werden die gleichen User-IDs wie auf dem Server verwendet (Vorgabe).
<code>map_daemon</code>	Client und Server haben keine übereinstimmenden User-IDs. Durch diese Option wird der <code>nfsd</code> angewiesen, eine Umsetztabelle für die User-IDs zu erstellen. Voraussetzung dafür ist jedoch die Aktivierung des Daemons ugidd .

Die exports-Datei kann beispielsweise aussehen wie Datei 14.27.

Beispiel 14.27: /etc/exports

```
#
# /etc/exports
#
/home          sonne(rw)   venus(rw)
/usr/X11       sonne(ro)   venus(ro)
/usr/lib/texmf sonne(ro)   venus(rw)
/              erde(ro,root_squash)
/home/ftp      (ro)
# End of exports
```

Die Datei /etc/exports wird von mountd und nfsd gelesen. Wird also eine Änderung daran vorgenommen, so müssen mountd und nfsd neu gestartet werden, damit diese Änderung berücksichtigt wird! Erreicht wird dies am einfachsten mit dem Befehl:

```
rcnfsdserver restart
```

14.10 DHCP

14.10.1 Das DHCP-Protokoll

Das so genannte „Dynamic Host Configuration Protocol“ dient dazu, Einstellungen in einem Netzwerk zentral von einem Server aus zu vergeben, statt diese dezentral an einzelnen Arbeitsplatzrechnern zu konfigurieren. Ein mit DHCP konfigurierter Client verfügt selbst nicht über statische Adressen, sondern konfiguriert sich voll und ganz selbstständig nach den Vorgaben des DHCP-Servers.

Dabei ist es möglich, jeden Client anhand der Hardware-Adresse seiner Netzwerkkarte zu identifizieren und ständig mit denselben Einstellungen zu versorgen, sowie Adressen aus einem dafür bestimmten Pool „dynamisch“ an jeden „interessierten“ Rechner zu vergeben. In diesem Fall wird sich der DHCP-Server bemühen, jedem Client bei jeder Anforderung (auch über längere Zeiträume hinweg) dieselbe Adresse zuzuweisen — dies funktioniert natürlich nicht, wenn es mehr Rechner im Netz als Adressen gibt.

Ein Systemadministrator kann somit gleich in zweierlei Hinsicht von DHCP profitieren. Einerseits ist es möglich, selbst umfangreiche Änderungen der Netzwerk-Adressen oder der Konfiguration komfortabel in der Konfigurationsdatei des DHCP-Servers zentral vorzunehmen, ohne dass eine Vielzahl von Clients einzeln konfiguriert werden müssen. Andererseits können vor allem neue Rechner sehr einfach ins Netzwerk integriert werden, indem sie aus dem Adress-Pool eine IP-Nummer zugewiesen bekommen. Auch für Laptops, die regelmäßig in verschiedenen Netzen betrieben werden, ist die Möglichkeit, von einem DHCP-Server jeweils passende Netzwerkeinstellungen zu beziehen, sicherlich interessant.

Neben IP-Adresse und Netzmaske werden der Rechner- und Domain-Name, der zu verwendende Gateway und Nameserver-Adressen dem Client mitgeteilt. Im Übrigen können auch etliche andere Parameter zentral konfiguriert werden, zum Beispiel ein Timeserver, von dem die jeweils aktuelle Uhrzeit abrufbar ist oder ein Printserver. Im Folgenden möchten wir Ihnen nun einen kurzen Einblick in die Welt von DHCP geben. Wir möchten Ihnen anhand des DHCP-Servers `dhcpcd` zeigen, wie einfach auch in Ihrem Netzwerk die gesamte Netzwerkkonfiguration zentral per DHCP erledigt werden kann.

14.10.2 DHCP-Softwarepakete

Bei SUSE LINUX stehen Ihnen sowohl ein DHCP-Server-, als auch zwei Client-Pakete zur Verfügung. Der vom Internet Software Consortium herausgegebene DHCP-Server `dhcpcd` stellt die Server Dienste zur Verfügung, als Clients können sowohl der vom ISC herausgegebene `dhclient` als auch der so genannte „DHCP Client Daemon“ im Paket `dhcpcd` verwendet werden.

Der bei SUSE LINUX standardmäßig installierte `dhcpcd` ist sehr einfach zu handhaben und wird beim Starten des Rechners automatisch gestartet, um nach einem DHCP-Server zu suchen. Er kommt ohne eine Konfigurationsdatei aus und sollte im Normalfall ohne weitere Konfiguration funktionieren.

Für komplexere Situationen kann man auf den ISC `dhclient` zurückgreifen, der sich über die Konfigurationsdatei `/etc/dhclient.conf` steuern lässt.

14.10.3 Der DHCP-Server `dhcpcd`

Der *Dynamic Host Configuration Protocol Daemon* ist das Herz eines DHCP-Systems. Er „vermietet“ Adressen und wacht über deren Nutzung, wie in

der Konfigurationsdatei `/etc/dhcpd.conf` festgelegt. Über die dort definierten Parameter und Werte stehen dem Systemadministrator eine Vielzahl von Möglichkeiten zur Verfügung, das Verhalten des DHCP nach seinen Wünschen zu beeinflussen.

Ein Beispiel für eine einfache `/etc/dhcpd.conf`-Datei:

Beispiel 14.28: Die Konfigurationsdatei `/etc/dhcpd.conf`

```
default-lease-time 600;           # 10 minutes
max-lease-time 7200;             # 2  hours

option domain-name "kosmos.all";
option domain-name-servers 192.168.1.1, 192.168.1.2;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option subnet-mask 255.255.255.0;

subnet 192.168.1.0 netmask 255.255.255.0
{
    range 192.168.1.10 192.168.1.20;
    range 192.168.1.100 192.168.1.200;
}
```

Diese einfache Konfigurationsdatei reicht bereits aus, damit DHCP im Netzwerk IP-Adressen zuweisen kann. Bitte achten Sie insbesondere auf die Strichpunkte am Ende jeder Zeile, ohne die `dhcpd` nicht starten wird!

Wie Sie sehen, lässt sich obige Beispieldatei in drei Blöcke unterteilen. Im ersten Abschnitt wird definiert, wie viele Sekunden eine IP-Adresse standardmäßig an einen anfragenden Rechner „vermietet“ wird, bevor sich dieser um eine Verlängerung bemühen sollte (`default-lease-time`). Auch wird hier angegeben, wie lange ein Rechner maximal eine vom DHCP-Server vergebene IP-Nummer behalten darf, ohne für diese eine Verlängerung zu beantragen (`max-lease-time`).

Im zweiten Block werden nun einige grundsätzliche Netzwerk-Parameter global festgesetzt:

- Mit `option domain-name` wird die Default-Domain Ihres Netzwerks definiert.

- Bei `option domain-name-servers` können bis zu drei DNS-Server angegeben werden, die zur Auflösung von IP-Adressen in Hostnamen (und umgekehrt) verwendet werden sollen. Idealerweise sollte auf Ihrem System bzw. innerhalb Ihres Netzwerks ein Nameserver bereits in Betrieb sein, der auch für dynamische Adressen jeweils einen Hostnamen und umgekehrt bereit hält. Mehr über die Einrichtung eines eigenen Nameservers erfahren Sie in Abschnitt 14.6 auf Seite 375.
- `option broadcast-address` legt fest, welche Broadcast-Adresse der anfragende Rechner verwenden soll.
- `option routers` definiert, wohin Datenpakete geschickt werden können, die (aufgrund der Adresse von Quell- und Zielhost sowie Subnetz-Maske) nicht im lokalen Netz zugestellt werden können. Gerade bei kleineren Netzen ist dieser Router auch meist der Übergang zum Internet.
- `option subnet-mask` gibt die an den Client zu übergebende Netzmaske an.

Unterhalb dieser allgemeinen Einstellungen wird nun noch ein Netzwerk samt Subnet Mask definiert. Abschließend muss noch ein Bereich gewählt werden, aus dem der DHCP-Daemon Adressen an anfragende Clients vergeben darf. Im Beispiel stehen alle Adressen zwischen 192.168.1.10 und 192.168.1.20 bzw. 192.168.1.100 und 192.168.1.200 zur Verfügung.

Nach diesen wenigen Zeilen sollten Sie bereits in der Lage sein, den DHCP-Daemon mit dem Kommando `rcdhcpd start` zu aktivieren, der sogleich zur Verfügung steht.

Bei SUSE LINUX wird der DHCP-Daemon aus Sicherheitsgründen per default in einer chroot-Umgebung gestartet. Damit die Konfigurationsdateien gefunden werden, müssen diese mit in die neue Umgebung kopiert werden. Dies geschieht mit dem Befehl `rcdhcpd start` automatisch.

Auch können Sie mit `rcdhcpd check-syntax` eine kurze, formale Überprüfung der Konfigurationsdatei vornehmen lassen. Sollte wider Erwarten ein Problem mit der Konfiguration auftreten und der Server mit einem Fehler abbrechen und nicht mit einem „done“ starten, finden Sie in der zentralen Systemprotokolldatei `/var/log/messages` meist ebenso Informationen dazu wie auf Konsole 10 (**Strg** + **Alt** + **F10**).

14.10.4 Rechner mit fester IP-Adresse

Wie eingangs bereits erwähnt, kann mit DHCP auch an ein- und denselben Rechner bei jeder Anfrage eine ganz bestimmte, definierte Adresse vergeben werden.

Selbstverständlich haben solche expliziten Adresszuweisungen Vorrang vor solchen aus dem Pool der dynamischen Adressen. Im Gegensatz zu diesen verfallen die festen Adressinformationen in keinem Fall, wie es bei den dynamischen der Fall ist, wenn nicht mehr genügend freie Adressen zur Verfügung stehen und deshalb eine Neuverteilung erforderlich ist.

Zur Identifizierung eines mit einer *statischen* Adresse definierten Systems, bedient sich der `dhcpd` der so genannten Hardwareadresse. Dies ist eine weltweit i.d.R. einmalige, fest definierte Nummer aus sechs Oktettpaaren, über die jedes Netzwerkgerät verfügt, zum Beispiel `00:00:45:12:EE:F4`.

Wird nun die Konfigurationsdatei aus Datei 14.28 auf Seite 415 um einen entsprechenden Eintrag wie in Datei 14.29 ergänzt, wird `dhcpd` unter allen Umständen dieselben Daten an den entsprechenden Rechner ausliefern.

Beispiel 14.29: Ergänzungen zur Konfigurationsdatei

```
host earth {
hardware ethernet 00:00:45:12:EE:F4;
fixed-address 192.168.1.21;
}
```

Der Aufbau dieser Zeilen ist nahezu selbsterklärend:

Zuerst wird der Name des zu definierenden Rechners eingetragen (`host <hostname>`) und in der folgenden Zeile die MAC-Adresse angegeben. Diese Adresse kann bei Linux-Rechnern mit dem Befehl `ifstatus` plus Netzwerkdevice (zum Beispiel `eth0`) festgestellt werden. Gegebenenfalls müssen Sie zuvor die Karte aktivieren: `ifup eth0`. Sie erhalten dann eine Ausgabe wie:

```
link/ether 00:00:45:12:EE:F4
```

In unserem Beispiel wird also dem Rechner, dessen Netzwerkkarte die MAC-Adresse `00:00:45:12:EE:F4` hat, die IP-Adresse `192.168.1.21` sowie der Rechnername `earth` zugewiesen.

Als Hardware-Typ wird heutzutage in aller Regel `ethernet` zum Einsatz kommen, wobei durchaus auch das vor allem bei IBM-Systemen häufig zu findende `token-ring` unterstützt wird.

14.10.5 Besonderheiten bei SUSE Linux

Aus Sicherheitsgründen enthält die SUSE Version des ISC DHCP-Servers den 'non-root/chroot'-Patch von Ari Edelkind. Damit kann der dhcpd

- als User 'nobody' laufen
- in einer chroot-Umgebung laufen (`/var/lib/dhcp/`)

Die Konfigurationsdatei `/etc/dhcpd.conf` muss dafür in `/var/lib/dhcp/etc/` liegen, und wird vom Init-Skript beim Start automatisch dorthin kopiert.

Dieses Verhalten lässt sich in der Datei `/etc/sysconfig/dhcpd` steuern. Um den dhcpd weiterhin ohne chroot-Umgebung laufen zu lassen, setzen Sie in der Datei `/etc/sysconfig/dhcpd` die Variable `DHCPD_RUN_CHROOTED` auf „no“

Damit der dhcpd auch in der chroot-Umgebung Hostnamen auflösen kann, müssen einige weitere Konfigurationsdateien mit kopiert werden. Dies sind:

- `/etc/localtime`
- `/etc/host.conf`
- `/etc/hosts`
- `/etc/resolv.conf`

Beim Start des Init-Skriptes werden diese deshalb nach `/var/lib/dhcp/etc/` kopiert. Diese Dateien müssen auf dem Laufenden gehalten werden, wenn sie durch ein Skript wie `/etc/ppp/ip-up` dynamisch modifiziert werden. Wenn in der Konfigurationsdatei nur IP-Adressen anstelle von Hostnamen verwendet werden, sind keine Probleme zu erwarten.

Wenn in Ihrer Konfiguration weitere Dateien mit in die chroot-Umgebung kopiert werden müssen, so können Sie diese mit dem Parameter `DHCPD_CONF_INCLUDE_FILES` in der Datei `etc/sysconfig/dhcpd` angeben.

Damit der dhcpd-Daemon aus der chroot-Umgebung heraus weiter loggen kann, auch wenn der Syslog-Daemon neu gestartet wird, muss zu der Variable `SYSLOGD_PARAMS` in `/etc/sysconfig/syslog` "-a `/var/lib/dhcp/dev/log`" hinzugefügt werden.

14.10.6 Weitere Informationen

Wenn Sie an zusätzlichen Informationen interessiert sind, bietet sich zum Beispiel die Seite des *Internet Software Consortium* an, auf der detaillierte Informationen zu DHCP verfügbar sind: <http://www.isc.org/products/DHCP/>.

Auch die neue Version 3 des Protokolls, die sich im Moment im Beta-Test befindet, wird dort dokumentiert. Im Übrigen stehen Ihnen selbstverständlich auch die Manpages zur Verfügung, dies sind insbesondere `man dhcpd`, `man dhcpd.conf`, `man dhcpd.leases` und `man dhcp-options`.

Auf dem Markt sind einige Bücher erschienen, die sich umfassend mit den Möglichkeiten des *Dynamic Host Name Configuration Protocol* auseinander setzen.

Übrigens, `dhcpd` kann sogar anfragenden Rechnern eine in der Konfigurationsdatei mit dem `filename`-Parameter definierte Datei anbieten, die einen bootbaren Betriebssystemkern enthält. Damit lassen sich Clients aufbauen, die über keine Festplatte verfügen und sowohl ihr Betriebssystem wie auch ihre Daten ausschließlich über das Netzwerk laden (*diskless clients*). Dies kann sowohl aus Kosten- als auch aus Sicherheitsgründen interessant sein.

14.11 Zeitsynchronisation mit `xntp`

Bei vielen Abläufen in einem Computersystem spielt eine exakte Zeit eine wichtige Rolle. Zu diesem Zweck haben alle Rechner normalerweise eine Uhr eingebaut. Leider genügt diese oftmals nicht den Anforderungen, die von Applikationen wie Datenbanken gefordert werden. Hierzu bietet es sich an, zum einen die lokale Rechneruhr permanent nachzustellen, oder auch über ein Netzwerk immer wieder zu korrigieren. Optimalerweise sollte eine Rechneruhr niemals rückwärts gestellt werden, und die Schritte, in denen sie nach vorne gestellt wird sollten gewisse Zeitintervalle nicht überschreiten. Verhältnismäßig einfach ist es, die Rechneruhr mit `ntpdate` von Zeit zu Zeit nachzustellen. Dies bewirkt aber immer einen harten Sprung in der Zeit, der nicht von allen Anwendungen toleriert wird.

Einen interessanten Ansatz zur Lösung dieses Problems liefert `xntp`. Zum einen korrigiert `xntp` die lokale Rechneruhr laufend anhand von gesammelten Korrekturdaten. Zum anderen Korrigiert es permanent die lokale Zeit mit Hilfe von Zeitservern im Netz. Als dritte Möglichkeit bietet es die Verwaltung von lokalen Zeitnormalen, wie Funkuhren, an.

14.11.1 Konfiguration im Netzwerk

Unter SUSE LINUX wird `xntp` so voreingestellt, dass nur die lokale Rechneruhr als Zeitreferenz dient. Die einfachste Möglichkeit, einen Zeitserver im Netz zu verwenden ist die Angabe von sog. „server“ parametern. Steht im Netzwerk ein Zeitserver zur Verfügung, der zum Beispiel den Namen `ntp.example.com` hat, so können Sie diesen Server in der Datei `/etc/ntp.conf` folgendermaßen ergänzen: `server ntp.example.com`.

Weitere Zeitserver trägt man einfach ein, indem zusätzliche Zeilen mit den Schlüsselwort „server“ eingefügt werden. Nachdem der `xntpd` mit dem dem Befehl `rcxntpd start` initialisiert wurde, benötigt er eine Stunde, bis sich die Zeit stabilisiert und das „drift-File“ zur Korrektur der lokalen Rechneruhr angelegt wird. Das „drift-File“ hat langfristig den Vorteil, dass bereits nach dem Einschalten des Rechners bekannt ist, wie sich die Hardwareuhr im Laufe der Zeit verstellt. Die Korrektur wird dann sofort aktiv, wodurch eine hohe Stabilität der Rechnerzeit erreicht wird.

Sofern in Ihrem Netzwerk der Zeitserver auch über einen Broadcast erreichbar ist, benötigen Sie den Server Namen nicht. Tragen Sie in diesem Fall den Befehl `broadcastclient` in die Konfigurationsdatei `/etc/ntp.conf` ein. In diesem Fall sollten Sie jedoch die Authentifizierungsmechanismen einrichten, da sonst ein fehlerhafter Zeitserver im Netzwerk Ihre Rechnerzeit verändern würde.

Jeder `xntpd` kann im Netzwerk normalerweise auch als Zeitserver angesprochen werden. Wenn Sie den `xntpd` auch mit broadcasts betreiben möchten, können Sie dies mit der Option `broadcast` einrichten:

```
broadcast 192.168.0.255
```

Ändern Sie hierzu die Broadcastadresse auf Ihre Gegebenheiten ab. Hierbei sollten Sie jedoch sicherstellen, dass der Zeitserver wirklich die richtige Uhrzeit verwendet. Hierzu eignen sich zum Beispiel Zeitnormale.

14.11.2 Einrichten eines lokalen Zeitnormals

Das Programmpaket `xntp` enthält auch Treiber, die den Anschluss von lokalen Zeitnormalen erlauben. Die unterstützten Uhren finden Sie im Paket `xntp-doc` in der Datei `file:/usr/share/doc/packages/xntp-doc/html/refclock.htm`. Jedem Treiber ist hierbei eine Nummer zugeordnet. Die eigentliche Konfiguration geschieht bei `xntp` über sogenannte Pseudo IPs. Die Uhren werden in die Datei `/etc/ntp.conf` so eingetragen, als wären sie im Netzwerk verfügbare Uhren.

Hierzu bekommen sie spezielle IP Adressen, die alle folgende Form haben: 127.127.t.u. Den Wert von t bekommen Sie aus der oben genannten Datei mit der Liste der Referenzuhren. u ist die Gerätenummer, die nur dann von 0 abweicht, wenn Sie mehrere Uhren des gleichen Typs an Ihrem Rechner verwenden. Eine Type 8 Generic Reference Driver (PARSE) hat demnach die Pseudo IP Adresse 127.127.8.0.

Die einzelnen Treiber haben im Normalfall spezielle Parameter, die die Konfiguration näher beschreiben. In der Datei `file:/usr/share/doc/packages/xntp-doc/html/refclock.htm` finden Sie zu jedem Treiber einen Link zur jeweiligen Treiberseite, die diese Parameter beschreiben. Für die Uhr mit dem Typ 8 ist es zum Beispiel notwendig, einen zusätzlichen mode anzugeben, der die Uhr genauer spezifiziert. Beispielsweise hat das Modul Conrad DCF77 receiver module den mode 5. Damit diese Uhr von xntp als Referenz genommen wird, können sie zusätzlich das Schlüsselwort prefer angeben. Die vollständige server-Zeile eines „Conrad DCF77 receiver module“ lautet somit:

```
server 127.127.8.0 mode 5 prefer
```

Andere Uhren folgen dem gleichen Schema. Die Dokumentation zu xntp steht Ihnen nach der Installation des Paketes xntp-doc im Verzeichnis `/usr/share/doc/packages/xntp-doc/html` zur Verfügung.

Der Webserver Apache

Mit einem Anteil von über 60 Prozent (laut <http://www.netcraft.com>) ist Apache der weltweit am weitesten verbreitete Webserver. Für Web-Anwendungen wird Apache häufig mit Linux, der Datenbank MySQL und den Programmiersprachen PHP und Perl kombiniert. Für diese Kombination hat sich die Abkürzung *LAMP* eingebürgert.

15.1 Grundlagen	424
15.2 HTTP-Server mit YaST einrichten	425
15.3 Apache Module	426
15.4 Neuerungen mit Apache 2	427
15.5 Threads	428
15.6 Installation	429
15.7 Konfiguration	431
15.8 Apache im Einsatz	436
15.9 Aktive Inhalte	437
15.10 Virtual Hosts	443
15.11 Sicherheit	447
15.12 Troubleshooting	448
15.13 Weitere Dokumentation	449

15.1 Grundlagen

15.1.1 Webserver

Ein Webserver liefert auf Anfrage eines Clients HTML-Seiten an diesen aus. Diese Seiten können in einem Verzeichnis auf dem Server abgelegt sein (so genannte passive oder statische Seiten) oder als Antwort auf die Anfrage neu generiert werden (aktive Inhalte).

15.1.2 HTTP

Bei den Clients handelt es sich meist um Webbrowser wie Konqueror und Mozilla. Die Kommunikation zwischen Browser und Webserver findet über das *HyperText Transfer Protocol* (HTTP) statt. Die aktuelle Version HTTP 1.1 ist im RFC 2068 sowie im Update RFC 2616 dokumentiert, diese RFCs findet man unter der URL `http://www.w3.org`.

15.1.3 URLs

Ein Client fordert über eine URL eine Seite vom Server an, zum Beispiel `http://www.suse.de/index.html`. Eine URL besteht aus:

- Einem Protokoll. Häufig benutzte Protokolle sind
 - ▷ `http://` Das HTTP-Protokoll.
 - ▷ `https://` Sichere, verschlüsselte Version von HTTP.
 - ▷ `ftp://` File Transfer Protocol, zum Down- und Upload von Dateien.
- Einer Domain, in diesem Fall `www.suse.de`. Die Domain kann man nochmals unterteilen, der erste Teil `www` verweist auf einen Computer, der zweite Teil `suse.de` ist die eigentliche Domain. Beides zusammen wird auch als FQDN (Fully Qualified Domain Name) bezeichnet.
- Einer Resource, in diesem Fall `index.html`. Dieser Teil gibt den kompletten Pfad zur Resource an. Die Resource kann eine Datei sein, wie in diesem Fall. Es kann sich aber auch um ein CGI-Skript, eine Java Server Page etc. handeln.

Dabei wird die Weiterleitung der Anfrage an die Domain `www.suse.de` von den entsprechenden Mechanismen des Internet (zum Beispiel Domain Name System, DNS) übernommen, die den Zugriff auf eine Domain an einen oder mehrere dafür zuständige Rechner weiterleiten. Apache selbst liefert dann die Resource, in diesem Fall also einfach die Seite `index.html` aus seinem Dateiverzeichnis aus. In diesem Fall liegt die Datei auf der obersten Ebene des Verzeichnisses, sie kann aber auch in einem Unterverzeichnis liegen, zum Beispiel `www.suse.de/business/services/support/index.html`.

Der Pfad der Datei ist dabei relativ zur sogenannten DocumentRoot, diese kann in den Konfigurationsdateien geändert werden. Wie das geht, beschreibt der Abschnitt 15.7.2 auf Seite 432.

15.1.4 Automatische Ausgabe einer Standardseite

Die Angabe der Seite kann fehlen. Apache hängt dann automatisch einen der gebräuchlichen Namen für solche Seiten an die URL an. Der gebräuchlichste Name für eine solche Seite ist `index.html`. Ob Apache diesen Automatismus ausführt und welche Seitennamen dabei berücksichtigt werden, lässt sich einstellen, dies ist im Abschnitt 15.7.2 auf Seite 433 beschrieben. In diesem Fall reicht dann beispielsweise der Aufruf von `http://www.suse.de` um vom Server die Seite `http://www.suse.de/index.html` angezeigt zu bekommen.

15.2 HTTP-Server mit YaST einrichten

Apache 2 lässt sich einfach und schnell mit YaST einrichten. Allerdings sollten Sie über einige Kenntnisse verfügen, wenn Sie damit einen Webserver aufsetzen möchten. Wenn Sie im YaST-Kontrollzentrum auf 'Netzwerkdienste' -> 'HTTP-Server' klicken, werden Sie gegebenenfalls gefragt, ob YaST fehlende Pakete installieren soll. Ist alles installiert, gelangen Sie in den Konfigurationsdialog.

Aktivieren Sie hier zunächst den HTTP-Dienst. Standardmäßig sind drei Optionen vorgegeben: 'Servername', 'E-Mail des Serveradministrators' und 'Lauschen auf'. Bei letztere Option ist bereits Port 80 voreingestellt. Über die Schaltfläche 'Hinzufügen' wählen Sie weitere Optionen aus. Mit 'Bearbeiten' ändern Sie den Wert der selektierten Option, 'Löschen' entfernt die Option.

Über die Kombo-Box 'Erweitert' können Sie sich das 'Zugriffsprotokoll anzeigen', 'Fehlerprotokoll anzeigen' lassen und die vom Server zu ladenden 'Servermodule' konfigurieren. In dieser Maske aktivieren und deaktivieren Sie Module über die Schaltfläche 'Status wechseln' und nehmen zusätzliche Module über die Schaltfläche 'Modul hinzufügen' auf.

15.3 Apache Module

Apache kann über Module um viele Funktionen erweitert werden und dadurch mit Hilfe von Modulen CGI-Skripte in verschiedensten Programmiersprachen ausführen. Hier stehen nicht nur Perl und PHP zur Verfügung, sondern auch weitere Skriptsprachen wie Python oder Ruby. Zudem gibt es Module für die gesicherte Übertragung von Daten mit SSL (Secure Sockets Layer), die Authentifizierung von Benutzern, erweitertes Logging und vieles mehr ermöglichen.

Apache kann über selbstgeschriebene Module an alle ausgefallenen Anforderungen und Wünsche angepasst werden. Natürlich ist dazu ein gewisses Maß an Know-How nötig. Referenzen auf weiterführende Informationen finden Sie im Abschnitt 15.13.4 auf Seite 450

Wenn Apache eine Anfrage bearbeitet, können für die Bearbeitung dieser Anfrage einer oder mehrere Handler eingetragen sein. Dies geschieht über Anweisungen in der Konfigurationsdatei. Die Handler können Teil von Apache sein, es kann aber auch ein Modul für die Bearbeitung aufgerufen werden. Dadurch lässt sich dieser Vorgang sehr flexibel gestalten. Zudem besteht die Möglichkeit, eigene Module in Apache einzuklinken und so Einfluss auf die Bearbeitung der Anfragen zu nehmen.

Bei Apache 2 geht die Modularisierung recht weit, hier erfüllt der Server nur minimale Aufgaben, alles andere wird über Module realisiert. Das geht so weit, dass bei Apache 2 selbst die Bearbeitung von HTTP über Module realisiert ist. Apache 2 muss demnach nicht unbedingt ein Webserver sein, er kann über andere Module auch ganz andere Aufgaben übernehmen. So gibt es als Modul beispielsweise einen Proof-of-Concept Mailserver (POP3) auf Apache-Basis.

Apache unterstützt eine Reihe von nützlichen Features, die im folgenden beschrieben werden.

Virtuelle Hosts Über virtuelle Hosts können mit einer Instanz von Apache auf einem einzigen Rechner mehrere Webseiten betrieben werden, wobei der Webserver für den Endbenutzer wie mehrere unab-

hängige Webserver wirkt. Dabei können die virtuellen Hosts auf unterschiedlichen IP-Adressen oder namensbasiert konfiguriert sein. Dies erspart die Anschaffungskosten und den Administrationsaufwand für zusätzliche Rechner.

Flexibles Umschreiben von URLs Apache bietet eine Vielzahl von Möglichkeiten, URLs zu manipulieren und umzuschreiben (URL-Rewriting). Näheres dazu findet man in der Dokumentation zu Apache.

Content Negotiation Apache kann in Abhängigkeit von den Fähigkeiten des Client (Browser) eine für diesen Client maßgeschneiderte Seite ausliefern. So kann man für ältere Browser oder Browser, die nur im Textmodus arbeiten (wie Lynx), einfachere Versionen der Seiten ausliefern, die keine Frames verwenden. Auf diese Weise kann man auch die notorischen Inkompatibilitäten der verschiedenen Browser, was JavaScript angeht, umgehen, indem man jedem Browser eine passende Version der Seiten liefert (wenn man den Aufwand treiben will, für jeden dieser Browser den JavaScript-Code anzupassen).

Flexible Fehlerbehandlung Falls ein Fehler auftritt, beispielsweise eine Seite nicht verfügbar ist, kann man flexibel reagieren und eine angemessene Antwort zurückgeben. Diese kann auch aktiv zusammengesetzt sein, beispielsweise mit Hilfe von CGI.

15.4 Neuerungen mit Apache 2

Im Folgenden finden Sie eine Liste der Neuerungen mit Apache 2. Ausführliche Dokumentation zum Apache HTTP Server Version 2.0 finden Sie auf der Webseite: <http://httpd.apache.org/docs-2.0/de/>.

- Bei der Art und Weise, wie mehrere Anfragen gleichzeitig ausgeführt werden, hat man die Wahl zwischen Threads und Prozessen. Die Prozessverwaltung ist in ein eigenes Modul, das sogenannte Multi-Processing-Modul (MPM) ausgelagert worden. Je nach MPM reagiert Apache 2 verschieden auf Anfragen. Das hat vor allem Auswirkungen auf die Performance und auf die Verwendung von Modulen. Dies wird im Folgenden ausführlicher besprochen.
- Das Innenleben von Apache wurde gründlich aufgeräumt, Apache verwendet nun eine neue, eigene Basisbibliothek (Apache Portable Runtime, APR) als Schnittstelle zu Systemfunktionen und für die

Speicherverwaltung. Darüberhinaus sind wichtige und weitverbreitete Module wie `mod_gzip` (Nachfolger: `mod_deflate`) oder `mod_ssl`, die tief in die Request-Verarbeitung eingreifen, jetzt weit besser in den Apache integriert.

- Apache 2 beherrscht nun das kommende Internetprotokoll IPv6.
- Es gibt jetzt einen Mechanismus, mit dem die Hersteller von Modulen selbst Angaben über die gewünschte Ladereihenfolge der Module machen können, so dass sich der Anwender nicht mehr selbst darum kümmern muss. Die Reihenfolge, in der Module ausgeführt werden, ist oft wichtig und wurde früher über die Ladereihenfolge festgelegt. So muss ein Modul, das nur authentifizierten Benutzern Zugriff auf bestimmte Ressourcen erlaubt, als erstes aufgerufen werden, damit Benutzer, die keine Zugriffsrechte haben, die Seiten erst gar nicht zu sehen bekommen können.
- Anfragen an und Antworten von Apache können durch Filter bearbeitet werden.
- Unterstützung fuer Dateien größer 2 GiB (Large-File-Support, LFS), auf 32-bit-Systemen
- Einige neuere Module gibt es nur für Apache 2.
- Mehrsprachige Fehlermeldungen.

15.5 Threads

Ein Thread ist eine Art leichtgewichtiger Prozess, der im Vergleich zu einem richtigen Prozess wesentlich weniger Ressourcen verbraucht. Dadurch steigt bei der Verwendung von Threads statt Prozessen auch die Performance. Der Nachteil ist dabei, dass Anwendungen für die Ausführung in einer Thread-Umgebung thread-safe sein müssen. Dies bedeutet:

- Funktionen (bzw. bei objektorientierten Anwendungen die Methoden) müssen „reentrant“ sein, das heißt dass die Funktion mit dem gleichen Input immer das gleiche Ergebnis liefert, unabhängig davon ob sie gleichzeitig von anderen Threads ausgeführt wird. Funktionen müssen demnach so programmiert sein, dass sie von mehreren Threads gleichzeitig aufgerufen werden können.

- Der Zugriff auf Ressourcen (meistens Variablen) muss so geregelt sein, dass sich die gleichzeitig laufenden Threads dabei nicht in die Quere kommen.

Apache 2 kann Anfragen als eigene Prozesse oder in einem gemischten Modell mit Prozessen und Threads ausführen. Für die Ausführung als Prozess sorgt das MPM „prefork“, für die Ausführung als Thread das MPM „worker“. Bei der Installation (siehe Abschnitt 15.6) kann man auswählen, welches MPM verwendet werden soll. Der dritte Modus, „perchild“ ist noch nicht voll ausgereift und steht deswegen in SUSE LINUX bei der Installation (noch) nicht zur Verfügung.

15.6 Installation

15.6.1 Paketauswahl in YaST

Für einfache Anforderungen muss man lediglich das Apache-Paket `apache2` installieren. Installieren Sie zusätzlich eines der MPM-Pakete (Multiprocessing Module) wie das Paket `apache2-prefork` oder `apache2-worker`. Bei der Auswahl des richtigen MPM ist zu beachten, dass das mit Threads laufende Worker-MPM nicht mit Paket `mod_php4` zusammen verwendet werden kann, da noch nicht alle von diesem Paket verwendeten Bibliotheken threadsafe sind.

15.6.2 Apache aktivieren

Wenn Apache installiert ist, wird er nicht automatisch gestartet. Um Apache zu starten, muss man ihn im Runlevel-Editor aktivieren. Um ihn dauerhaft beim Booten des Systems zu starten, muss man im Runlevel-Editor für die Runlevel 3 und 5 ein Häkchen setzen. Ob Apache läuft, lässt sich feststellen, indem man die URL `http://localhost/` in einem Browser aufruft. Läuft Apache, kann dann eine Beispielseite sehen, sofern das Paket `apache2-example-pages` installiert ist.

15.6.3 Module für aktive Inhalte

Um aktive Inhalte mit Hilfe von Modulen zu nutzen, muss man noch die Module für die jeweiligen Programmiersprachen installieren. Dies sind das Paket `apache2-mod_perl` für Perl bzw. das Paket `apache2-mod_php4`

für PHP und schließlich das Paket `apache2-mod_python` für Python. Die Verwendung dieser Module ist im Abschnitt 15.9.5 auf Seite 440 beschrieben.

15.6.4 Zusätzliche empfehlenswerte Pakete

Zusätzlich empfiehlt es sich, die reichhaltige Dokumentation zu installieren, diese findet sich im Paket `apache2-doc`. Für diese Dokumentation existiert ein Alias (was das ist, wird im Abschnitt 15.7 auf der nächsten Seite beschrieben), so dass man sie nach der Installation direkt über die URL `http://localhost/manual` aufrufen kann.

Wer Module für Apache entwickeln oder Module von Drittanbietern kompilieren will, muss noch das Paket `apache2-devel` installieren, sowie entsprechende Entwicklungswerkzeuge. Diese enthalten unter anderem die `apxs`-Tools, die im Abschnitt 15.6.5 näher beschrieben sind.

15.6.5 Installation von Modulen mit `apxs`

Ein wichtiges Werkzeug für Modulentwickler ist `apxs2`. Mit diesem Programm lassen sich Module, die als Quelltext vorliegen, mit einem einzigen Befehl kompilieren und installieren, samt der notwendigen Änderungen an den Konfigurationsdateien. Außerdem kann man auch Module installieren, die bereits als Objektdatei (Endung `.o`) oder statische Library (Endung `.a`) vorliegen. `apxs2` erstellt aus den Quellen ein Dynamic Shared Object (DSO), das von Apache direkt als Modul verwendet wird.

Die Installation eines Moduls aus dem Quelltext bewirkt beispielsweise der Befehl `apxs2 -c -i -a mod_foo.c`. Andere Optionen von `apxs2` sind in der zugehörigen Manpage beschrieben.

Von `apxs2` gibt es mehrere Versionen: `apxs2`, `apxs2-prefork` und `apxs2-worker`. Während `apxs2` ein Modul so installiert, dass es für alle MPMs verwendbar ist, installieren die beiden anderen Programme Module so, dass sie nur für die jeweiligen MPMs (also `prefork` bzw. `worker`) verwendet werden. Während also `apxs2` ein Modul in `/usr/lib/apache2/` installiert, landet dieses Modul bei Verwendung von `apxs2-prefork` in `/usr/lib/apache2-prefork/`.

Die Option `-a` sollte mit Apache 2 nicht verwendet werden, da dann die Änderungen direkt in `/etc/apache2/httpd.conf` geschrieben werden. Stattdessen sollten Module über den Eintrag `APACHE_MODULES` in `/etc/sysconfig/apache2/` aktiviert werden, wie im Abschnitt 15.7.1 auf der nächsten Seite beschrieben.

15.7 Konfiguration

Wenn man Apache installiert hat, sind weitere Anpassungen nur nötig, wenn man spezielle Wünsche oder Anforderungen hat. Apache kann einerseits über SuSEconfig konfiguriert werden, andererseits kann man auch die Datei `/etc/apache2/httpd.conf` direkt editieren.

15.7.1 Konfiguration mit SuSEconfig

Die Einstellungen, die Sie in `/etc/sysconfig/apache2` vornehmen können, werden mittels SuSEconfig in die Konfigurationsdateien von Apache eingepflegt. Diese umfassen jene Konfigurationsmöglichkeiten, die für viele Fälle ausreichend sein dürften. Zu jeder Variable finden Sie erläuternde Kommentare in der Datei.

Eigene Konfigurationsdateien

Statt Änderungen in der Konfigurationsdatei `/etc/apache2/httpd.conf` direkt vorzunehmen, kann man mit Hilfe der Variablen `APACHE_CONF_INCLUDE_FILES` eine eigene Konfigurationsdatei angeben (beispielsweise `httpd.conf.local`), die dann in die Hauptkonfigurationsdatei eingelesen wird. Auf diese Weise bleiben auch eigene Änderungen an der Konfiguration erhalten, wenn die Datei `/etc/apache2/httpd.conf` bei einer Neuinstallation überschrieben wird.

Module

Module, die per YaST bereits installiert wurden, werden aktiviert, indem man den Namen des Moduls in die für die Variable `APACHE_MODULES` angegebene Liste aufnimmt. Diese Variable findet sich in der Datei `/etc/sysconfig/apache2`.

Flags

Mit der Variable `APACHE_SERVER_FLAGS` können Flags angegeben werden, die bestimmte Bereiche in der Konfigurationsdatei an- und ausschalten. Ist also in der Konfigurationsdatei ein Abschnitt in

```
<IfDefine someflag>
.
.
.
</IfDefine>
```

eingeschlossen, so wird dieser nur aktiviert, wenn bei in der Variable `ACTIVE_SERVER_FLAGS` das entsprechende Flag gesetzt ist: `ACTIVE_SERVER_FLAGS = ... someflag ...`. Auf diese Weise können größere Bereiche der Konfigurationsdatei zu Testzwecken einfach aktiviert oder deaktiviert werden.

15.7.2 Manuelle Konfiguration

Die Konfigurationsdatei

Die Konfigurationsdatei `/etc/apache2/httpd.conf` erlaubt Änderungen, die über Einstellungen in `/etc/sysconfig/apache2` nicht möglich sind. Es folgen einige der Parameter, die dort eingestellt werden können. Sie sind ungefähr in der Reihenfolge aufgelistet, in der sie in dieser Datei vorkommen.

DocumentRoot

Eine grundlegende Einstellung ist die sogenannte `DocumentRoot`, das ist das Verzeichnis, unter dem Apache Webseiten erwartet, die vom Server ausgeliefert werden sollen. Sie ist für den Default-Virtual Host auf `/srv/www/htdocs` eingestellt und muss normalerweise nicht geändert werden.

Timeout

Gibt die Zeit an, die der Server wartet, bis er einen Timeout für eine Anfrage meldet.

MaxClients

Die Anzahl der Clients, die Apache maximal gleichzeitig bedient. Die Voreinstellung ist 150, dieser Wert könnte für eine viel besuchte Website aber auch zu niedrig sein.

LoadModule

Die `LoadModule` Anweisungen geben an, welche Module geladen werden. In Apache 2 wird die Ladereihenfolge durch die Module selbst angegeben, siehe dazu auch den Abschnitt 15.4 auf Seite 427. Ausserdem geben diese Anweisungen an, welche Datei das Modul enthält.

Port

Gibt den Port an, auf dem Apache auf Anfragen wartet. Dies ist normalerweise immer Port 80, der Standardport für HTTP. Diese Einstellung zu ändern, ist im Normalfall nicht sinnvoll. Ein Grund, Apache auf einem anderen Port lauschen zu lassen, kann der Test einer neuen Version einer Website sein. Auf diese Weise ist die funktionierende Version der Website nach wie vor über den Standardport 80 erreichbar.

Ein weiterer Grund kann sein, dass man Seiten lediglich im Intranet zur Verfügung stellen will, weil sie Informationen enthalten, die nicht jeden etwas angehen. Dazu stellt man den Port beispielsweise auf den Wert 8080 ein und sperrt Zugriffe von aussen auf diesen Port in der Firewall. So ist der Server vor jedem Zugriff von ausserhalb abgesichert.

Directory

Mit dieser Direktive werden die Zugriffs- und andere Rechte für ein Verzeichnis gesetzt. Auch für die `DocumentRoot` existiert eine solche Direktive, der dort angegebene Verzeichnisname muss immer parallel mit `DocumentRoot` geändert werden.

DirectoryIndex

Hiermit kann eingestellt werden, nach welchen Dateien Apache sucht, um eine URL zu vervollständigen, bei der die Angabe der Datei fehlt. Die Voreinstellung ist `index.html`. Wird also beispielsweise vom Client die URL `http://www.xyz.com/foo/bar` aufgerufen und existiert unterhalb der `DocumentRoot` ein Verzeichnis `foo/bar`, das eine Datei namens `index.html` enthält, so liefert Apache diese Seite an den Client zurück.

AllowOverride

Jedes Verzeichnis, aus dem Apache Dokumente ausliefert, kann eine Datei enthalten, die global eingestellte Zugriffsrechte und andere Einstellungen für dieses Verzeichnis abändern kann. Diese Einstellungen gelten rekursiv für das aktuelle Verzeichnis und seine Unterverzeichnisse, bis sie in einem Unterverzeichnis von einer weiteren solchen Datei geändert werden. Dies bedeutet auch, dass solche Einstellungen, wenn sie in einer Datei in `DocumentRoot` angegeben sind, global gelten. Diese Dateien haben normalerweise den Namen `.htaccess`, man kann das jedoch ändern, siehe dazu den Abschnitt 15.7.2 auf der nächsten Seite.

Mit `AllowOverride` kann man einstellen, ob die in den lokalen Dateien angegebenen Einstellungen die globalen Einstellungen überschreiben können. Mögliche Werte sind `None`, `All` sowie jede mögliche Kombination von `Options`, `FileInfo`, `AuthConfig` und `Limit`. Die Bedeutung dieser Werte ist in der Dokumentation zu Apache ausführlich beschrieben. Die (sichere) Voreinstellung ist `None`.

Order

Diese Option beeinflusst, in welcher Reihenfolge die Einstellungen für die Zugriffsrechte `Allow`, `Deny` angewandt werden. Die Voreinstellung ist:

```
Order allow,deny
```

Es werden also zuerst die Zugriffsrechte für erlaubte Zugriffe und dann die für verbotene Zugriffe angewandt. Die zugrundeliegende Denkweise ist eine von zweien:

allow all jeden Zugriff erlauben, aber Ausnahmen definieren.

deny all jeden Zugriff verweigern, aber Ausnahmen definieren.

Beispiel für `deny all`:

```
Order deny,allow
Deny from all
Allow from example.com
Allow from 10.1.0.0/255.255.0.0
```

AccessFileName

Hier lässt sich der Name für die Dateien einstellen, die in von Apache ausgelieferten Verzeichnissen die globalen Einstellungen für Zugriffsrechte etc. überschreiben können (siehe dazu auch Abschnitt 15.7.2 auf der vorherigen Seite). Die Voreinstellung ist `.htaccess`.

ErrorLog

Gibt den Namen der Datei an, in der Apache Fehlermeldungen ausgibt. Die Voreinstellung ist `/var/log/httpd/errorlog`. Fehlermeldungen für Virtual Hosts (siehe Abschnitt 15.10 auf Seite 443) werden ebenfalls in diese Datei ausgegeben, wenn im `VirtualHost`-Abschnitt der Konfigurationsdatei keine eigene Log-Datei angegeben wurde.

LogLevel

Fehlermeldungen werden je nach Dringlichkeit in verschiedene Stufen eingeteilt. Diese Einstellung gibt an, ab welcher Dringlichkeitsstufe die Meldungen ausgegeben werden. Eine Einstellung auf einen Level gibt an, dass Meldungen dieser Stufe und dringendere Meldungen ausgegeben werden. Voreinstellung ist warn.

Alias

Mit einem Alias kann man einen Shortcut für ein Verzeichnis angeben, mit dem man dann direkt auf dieses Verzeichnis zugreifen kann. So kann man beispielsweise über das Alias `/manual/` auf das Verzeichnis `/srv/www/htdocs/manual` zugreifen, auch wenn die DocumentRoot auf ein anderes Verzeichnis als `/srv/www/htdocs` eingestellt ist. Solange die DocumentRoot diesen Wert hat, macht das keinen Unterschied. Im Falle dieses Alias kann man dann direkt mit `http://localhost/manual` auf das entsprechende Verzeichnis zugreifen. Eventuell kann es nötig sein, für das in einer Alias-Direktive angegebene Zielverzeichnis eine `Directory`-Direktive anzugeben, in der die Rechte für das Verzeichnis eingestellt werden (vgl. Abschnitt 15.7.2 auf Seite 433).

ScriptAlias

Diese Anweisung ähnelt der Alias-Anweisung. Sie gibt zusätzlich an, dass die Dateien im Zielverzeichnis als CGI-Skripte behandelt werden sollen.

Server Side Includes

Server Side Includes können aktiviert werden, indem man alle ausführbaren Dateien nach SSIs durchsuchen lässt. Dies geschieht mit der folgenden Anweisung:

```
<IfModule mod_include.c>  
XBitHack on  
</IfModule>
```

Um eine Datei nach Server Side Includes durchsuchen zu lassen, muss man sie dann lediglich mit `chmod +x <dateiname>` ausführbar machen. Alternativ kann man auch explizit den Typ der Dateien angeben, die nach SSIs durchsucht werden sollen. Dies geschieht mit

```
AddType text/html .shtml  
AddHandler server-parsed .shtml
```

Es ist keine gute Idee, hier einfach `.html` anzugeben, da Apache dann alle Seiten nach Server Side Includes durchsucht (auch solche, die mit Sicherheit keine solchen enthalten), was die Performance erheblich beeinträchtigt. Bei SUSE LINUX sind diese beiden Anweisungen bereits in der Konfigurationsdatei eingetragen, es sind also normalerweise keine Anpassungen nötig.

UserDir

Mit Hilfe des Moduls `mod_userdir` und der Direktive `UserDir` kann man ein Verzeichnis im Home-Verzeichnis des Anwenders angeben, in dem dieser seine Dateien über Apache veröffentlichen kann. Dies wird in SuSEconfig über die Variable `HTTPD_SEC_PUBLIC_HTML` eingestellt. Um Dateien veröffentlichen zu können, muss diese Variable auf den Wert `yes` gesetzt sein. Dies führt zu folgendem Eintrag in der Datei `/etc/httpd/suse_public_html.conf`, die von `/etc/apache2/httpd.conf` eingelesen wird.

```
<IfModule mod_userdir.c>
UserDir public_html
</IfModule>
```

15.8 Apache im Einsatz

Um mit Apache eigene (statische) Webseiten anzuzeigen, muss man lediglich die eigenen Dateien im richtigen Verzeichnis unterbringen. Unter SUSE LINUX ist das `/srv/www/htdocs`. Eventuell sind dort bereits ein paar kleine Beispielseiten installiert. Diese dienen lediglich dazu, nach der Installation zu testen, ob Apache korrekt installiert wurde und läuft, man kann sie problemlos überschreiben oder besser deinstallieren. Eigene CGI-Skripte werden unter `/srv/www/cgi-bin` installiert.

Apache schreibt im laufenden Betrieb Log-Meldungen in die Datei `/var/log/httpd/access_log` bzw. `/var/log/apache2/access_log`. Dort ist dokumentiert, welche Ressourcen zu welcher Zeit mit welcher Methode (GET, POST ...) angefragt und ausgeliefert wurden. Bei Fehlern finden sich entsprechende Hinweise unter `/var/log/apache2`.

15.9 Aktive Inhalte

Apache bietet mehrere Möglichkeiten, aktive Inhalte an Clients auszuliefern. Unter aktiven Inhalten versteht man HTML-Seiten, die aufgrund einer Verarbeitung aus variablen Eingabedaten des Clients erstellt wurden. Ein bekanntes Beispiel dafür sind Suchmaschinen, die auf die Eingabe eines oder mehrerer, eventuell durch logische Operatoren wie UND bzw. ODER verknüpfter Suchbegriffe eine Liste von Seiten zurückgeben, in denen diese Begriffe vorkommen.

Mit Apache gibt es drei Wege, um aktive Inhalte zu erstellen:

Server Side Includes (SSI) Dabei handelt es sich um Anweisungen, die mit Hilfe spezieller Kommentare in eine HTML-Seite eingebettet werden. Apache wertet den Inhalt der Kommentare aus und gibt das Ergebnis als Teil der HTML-Seite mit aus.

Common Gateway Interface (CGI)

Hierbei werden Programme ausgeführt, die innerhalb bestimmter Verzeichnisse liegen. Apache übergibt vom Client übertragene Parameter an diese Programme und gibt die Ausgabe des Programms an den Client zurück. Diese Art der Programmierung ist relativ einfach, zumal man existierende Kommandozeilenprogramme so umbauen kann, dass sie Eingaben von Apache annehmen und Ausgaben an ihn ausgeben.

Module Apache bietet Schnittstellen, um beliebige Module als Teil der Verarbeitung einer Anfrage ausführen zu können, und gewährt diesen Programmen zudem Zugriff auf wichtige Informationen, wie den Request oder die HTTP-Header. Dies macht es möglich, Programme in die Verarbeitung der Anfrage einzufügen, die nicht nur aktive Inhalte erzeugen können, sondern auch andere Funktionen (wie Authentifizierung) übernehmen können. Die Programmierung solcher Module erfordert etwas Geschick, als Vorteil wiegt eine hohe Performance sowie Möglichkeiten, die sowohl über SSI als auch über CGI weit hinausgehen.

Während CGI-Skripte einfach von Apache aufgerufen werden (unter der Benutzer-ID ihres Eigentümers), wird bei Verwendung von Modulen ein Interpreter in Apache eingebettet, der dann unter der ID des Webserverns permanent läuft. Der Interpreter ist „persistent“. Auf diese Weise muss nicht für jede Anfrage ein eigener Prozess gestartet und beendet werden

(was einen erheblichen Overhead für Prozessmanagement, Speicherverwaltung usw. nach sich zieht), stattdessen wird das Skript einfach an den bereits laufenden Interpreter übergeben.

Einen Nachteil hat die Sache allerdings: Während über CGI ausgeführte Skripte einigermaßen tolerant gegen nachlässige Programmierung sind, wirkt sich diese bei Verwendung von Modulen schnell nachteilig aus. Der Grund dafür ist, dass bei normalen CGI-Skripten Fehler wie das Nichtfreigeben von Ressourcen und Speicher nicht so sehr ins Gewicht fallen, da die Programme nach Bearbeitung der Anfrage wieder beendet werden und damit vom Programm aufgrund eines Programmierfehlers nicht freigegebener Speicher wieder verfügbar wird. Bei Verwendung von Modulen häufen sich die Auswirkungen von Programmierfehlern an, da der Interpreter permanent läuft. Wenn der Server nicht neu gestartet wird, kann der Interpreter ohne weiteres monatelang laufen, da machen sich nicht freigegebene Datenbankverbindungen oder ähnliches durchaus bemerkbar.

15.9.1 Server Side Includes: SSI

Server Side Includes sind in spezielle Kommentare eingebettete Anweisungen, die Apache ausführt. Das Ergebnis wird dann an Ort und Stelle in die Ausgabe eingebettet. Ein Beispiel: Das aktuelle Datum kann man mit `<!--#echo var="DATE_LOCAL" -->` ausgegeben werden. Hierbei ist das `#` dem Kommentaranfang `<!--` der Hinweis für Apache, dass es sich um eine SSI-Anweisung und nicht um einen gewöhnlichen Kommentar handelt.

SSIs können auf mehrere Arten aktiviert werden. Die einfache Variante ist, alle Dateien, deren Rechte auf ausführbar gesetzt sind, auf Server Side Includes zu untersuchen. Die andere Variante ist, für bestimmte Dateitypen festzulegen, dass sie auf SSIs untersucht werden sollen. Beide Einstellungen werden im Abschnitt 15.7.2 auf Seite 435 erläutert.

15.9.2 Common Gateway Interface: CGI

CGI ist eine Abkürzung für „Common Gateway Interface“. Mit CGI liefert der Server nicht einfach eine statische HTML-Seite aus, sondern führt ein Programm aus, das die Seite liefert. Auf diese Weise können Seiten erstellt werden, die das Ergebnis einer Berechnung sind, beispielsweise das Ergebnis einer Suche in einer Datenbank. An das ausgeführte Programm können Argumente übergeben werden, so kann es für jede Anfrage eine individuelle Antwort-Seite zurückliefern.

Der große Vorteil an CGI ist, dass es eine recht einfache Technik ist. Das Programm muss lediglich in einem bestimmten Verzeichnis liegen und wird dann vom Webserver genauso wie ein Programm auf der Kommandozeile ausgeführt. Die Ausgaben des Programms auf dem Standardausgabekanal (`stdout`) gibt der Server einfach an den Client aus.

15.9.3 GET und POST

Eingabeparameter können entweder mit `GET` oder mit `POST` an den Server übergeben werden. Je nachdem, welche Methode verwendet wird, gibt der Server die Parameter auf unterschiedliche Weise an das Skript weiter. Bei `POST` übergibt der Server die Parameter auf dem Standardeingabekanal (`stdin`) an das Programm. Genauso würde das Programm seinen Input erhalten, wenn es in einer Konsole gestartet wird.

Bei `GET` werden die Parameter vom Server in der Umgebungsvariablen `QUERY_STRING` an das Programm übergeben. Eine Umgebungsvariable ist eine Variable, die vom System überall verfügbar gemacht wird, ein klassisches Beispiel ist die Variable `PATH`, die eine Liste von Pfaden enthält, die das System nach ausführbaren Kommandos durchsucht, wenn der Anwender einen Befehl eingibt.

15.9.4 Sprachen für CGI

CGI-Programme können prinzipiell in jeder Programmiersprache geschrieben sein. Typischerweise werden Skriptprachen (interpretierte Sprachen) wie Perl oder PHP verwendet, für geschwindigkeitskritische CGIs kann im Einzelfall auch C oder C++ die erste Wahl sein.

Im einfachsten Fall erwartet Apache diese Programme in einem bestimmten Verzeichnis (`cgi-bin`). Dieses Verzeichnis lässt sich in der Konfigurationsdatei einstellen, siehe den Abschnitt 15.7 auf Seite 431.

Ausserdem lassen sich weitere Verzeichnisse freigeben, die Apache dann nach ausführbaren Programmen durchsucht. Dies stellt aber ein gewisses Sicherheitsrisiko dar, da dann jeder Anwender (evtl. böswillige) Programme von Apache ausführen lassen kann. Wenn man ausführbare Programme lediglich in `cgi-bin` zulässt, kann der Administrator leichter kontrollieren, wer welche Skripte und Programme dort ablegt und ob diese evtl. bösartiger Natur sind.

15.9.5 Aktive Inhalte mit Modulen erzeugen

Es gibt eine Reihe von Modulen für die Verwendung in Apache. Alle im Folgenden beschriebenen Module stehen als Pakete in SUSE LINUX zur Verfügung. Der Begriff Modul wird in zwei Bedeutungen verwendet. Zum einen gibt es Module, die in Apache eingebaut werden können und dort eine bestimmte Funktion übernehmen, wie zum Beispiel die im Folgenden vorgestellten Module zur Einbettung von Programmiersprachen in Apache.

Zum anderen spricht man in Programmiersprachen von Modulen, wenn man eine abgeschlossene Menge von Funktionen, Klassen und Variablen meint. Diese Module werden in ein Programm eingebunden, um eine bestimmte Funktionalität zur Verfügung zu stellen. Ein Beispiel sind die in allen Skriptsprachen vorhandenen CGI-Module, die das Programmieren von CGI-Anwendungen erleichtern, indem sie unter anderem Methoden zum Lesen der Request-Parameter und zur Ausgabe von HTML-Code zur Verfügung stellen.

15.9.6 mod_perl

Perl ist eine weitverbreitete und bewährte Skriptsprache. Für Perl gibt es eine riesige Menge an Modulen und Bibliotheken (unter anderem auch eine Bibliothek zur Erweiterung der Konfigurationsdatei von Apache). Die Homepage für Perl ist <http://www.perl.com/>. Eine grosse Auswahl an Libraries für Perl findet man im Comprehensive Perl Archive Network (CPAN): <http://www.cpan.org/>. Eine deutschsprachige Seite für Perl-Programmierer ist <http://www.perlunity.de/>.

mod_perl einrichten

Um mod_perl unter SUSE LINUX einzurichten, muss man lediglich das entsprechende Paket installieren (siehe dazu den Abschnitt 15.6 auf Seite 429). Die erforderlichen Einträge in der Konfigurationsdatei für Apache sind dann ebenfalls schon vorhanden, siehe `/etc/apache2/mod_perl-startup.pl`. Informationen über mod_perl finden sich vor allem hier: <http://perl.apache.org/>

mod_perl vs. CGI

Im einfachsten Fall kann man ein bisheriges CGI-Skript als mod_perl-Skript laufen lassen, indem man es unter einer anderen URL aufruft. Die

Konfigurationsdatei enthält Aliase, die auf das gleiche Verzeichnis verweisen und darin enthaltene Skripte entweder über CGI oder über `mod_perl` aufrufen. Alle diese Einträge sind in der Konfigurationsdatei bereits eingetragen. Der Alias-Eintrag für CGI lautet:

```
ScriptAlias /cgi-bin/ "/srv/www/cgi-bin/"
```

Die Einträge für `mod_perl` lauten wie folgt:

```
<IfModule mod_perl.c>
# Provide two aliases to the same cgi-bin directory,
# to see the effects of the 2 different mod_perl modes.
# for Apache::Registry Mode
ScriptAlias /perl/          "/srv/www/cgi-bin/"
# for Apache::Perlrun Mode
ScriptAlias /cgi-perl/      "/srv/www/cgi-bin/"
</IfModule>
```

Die folgenden Einträge sind für `mod_perl` ebenfalls nötig. Auch sie sind bereits in der Konfigurationsdatei eingetragen.

```
#
# If mod_perl is activated, load configuration information
#
<IfModule mod_perl.c>
PerlRequire /usr/include/apache/modules/perl/startup.perl
PerlModule Apache::Registry

#
# set Apache::Registry Mode for /perl Alias
#
<Location /perl>
SetHandler perl-script
PerlHandler Apache::Registry
Options ExecCGI
PerlSendHeader On
</Location>

#
# set Apache::PerlRun Mode for /cgi-perl Alias
#
<Location /cgi-perl>
SetHandler perl-script
PerlHandler Apache::PerlRun
```

```
Options ExecCGI
PerlSendHeader On
</Location>

</IfModule>
```

Diese Einträge legen Aliase für die Modi `Apache::Registry` und `Apache::PerlRun` an. Der Unterschied zwischen den beiden Modi gestaltet sich folgendermaßen:

Apache::Registry Alle Skripte werden kompiliert und dann in einem Cache gehalten. Jedes Skript wird als Inhalt einer Subroutine angelegt. Dies ist gut für die Performance, hat jedoch auch einen Nachteil: Die Skripte müssen sehr sauber programmiert sein, da Variablen und Subroutinen zwischen den Aufrufen erhalten bleiben. Das bedeutet, dass man Variablen selbst zurücksetzen muss, damit sie beim nächsten Aufruf erneut verwendet werden können. Speichert man beispielsweise in einem Skript für Online-Banking die Kreditkartennummer eines Kunden in einer Variable auf, so könnte diese Nummer wieder auftauchen, wenn der nächste Kunde die Anwendung benutzt und somit das gleiche Skript wieder aufruft.

Apache::PerlRun Die Skripte werden für jede Anfrage neu kompiliert, sodass Variablen und Subroutinen zwischen den Aufrufen aus dem Namespace verschwinden. Der Namespace ist die Gesamtheit aller Variablennamen und Routinennamen, die zu einem bestimmten Zeitpunkt während der Existenz eines Skripts definiert sind. Mit `Apache::PerlRun` muss man deswegen nicht so genau auf saubere Programmierung achten, da alle Variablen beim Start des Skripts frisch initialisiert sind und keine Werte aus vorangegangenen Aufrufen mehr enthalten können. Dies geht natürlich zu Lasten der Geschwindigkeit, ist aber immer noch deutlich schneller als CGI, da man sich den Aufruf eines eigenen Prozesses für den Interpreter spart. `Apache::PerlRun` verhält sich ähnlich wie CGI.

15.9.7 mod_php4

PHP ist eine Programmiersprache, die speziell für den Einsatz mit Webservern entworfen wurde. Im Unterschied zu anderen Sprachen, deren Befehle in eigenständigen Dateien (Skripten) abgelegt werden, sind bei PHP die Befehle (ähnlich wie bei SSI) in eine HTML-Seite eingebettet. Der PHP-Interpreter verarbeitet die PHP-Befehle und bettet das Ergebnis der Verarbeitung in die HTML-Seite ein.

Die Homepage für PHP findet man unter <http://www.php.net/>. Eine deutschsprachige PHP-Seite findet man unter <http://www.php-homepage.de/>.

Das Paket `mod_php4-core` muss in jedem Fall installiert sein. Für Apache 2 wird zusätzlich Paket `apache2-mod_php4`.

15.9.8 mod_python

Python ist eine objektorientierte Programmiersprache mit einer sehr klaren und leserlichen Syntax. Etwas ungewöhnlich, aber nach einer kurzen Eingewöhnungsphase recht angenehm ist, dass die Struktur des Programms von der Einrückung abhängt. Blocks werden nicht über geschweifte Klammern (wie in C und Perl) oder andere Begrenzer (wie `begin` und `end`) definiert, sondern darüber, wie tief sie eingerückt sind. Installieren Sie Paket `apache2-mod_python`.

Mehr über die Sprache findet man unter <http://www.python.org/>. Mehr Informationen über `mod_python` bietet <http://www.modpython.org/>.

15.9.9 mod_ruby

Ruby ist eine relativ junge objektorientierte High-Level-Programmiersprache, die sowohl Ähnlichkeit mit Perl als auch mit Python hat und die sich hervorragend für Skripte eignet. Mit Python verbindet sie die saubere, sehr übersichtliche Syntax, während sie von Perl die von vielen Programmierern geliebten (und von anderen verachteten) Kürzel wie zum Beispiel `$. r`, die Nummer der zuletzt aus der Eingabedatei gelesenen Zeile, übernommen hat. Von der grundlegenden Konzeption erinnert Ruby stark an Smalltalk.

Die Homepage von Ruby ist <http://www.ruby-lang.org/>. Auch für Ruby gibt es ein Apache-Modul, die Homepage findet sich unter <http://www.modruby.net/>.

15.10 Virtual Hosts

Mit Virtual Hosts ist es möglich, mehrere Domains mit einem einzigen Webserver ins Netz zu stellen. Auf diese Weise spart man sich die Kosten und den Administrationsaufwand für einen eigenen Server pro Domain.

Apache war einer der ersten Webserver, die dieses Feature geboten haben und er bietet mehrere Möglichkeiten für Virtual Hosts:

- Namensbasierte Virtual Hosts
- IP-basierte Virtual Hosts
- Mehrere Instanzen von Apache auf einem Rechner laufen lassen.

15.10.1 Namensbasierte Virtual Hosts

Bei namensbasierten Virtual Hosts werden von einer Instanz von Apache mehrere Domains bedient. Die Einrichtung mehrerer IPs für einen Rechner ist hierbei nicht nötig. Dies ist die einfachste Alternative und sie sollte bevorzugt werden. Mehr zu Gründen, die gegen die Verwendung von namensbasierten Virtual Hosts sprechen können, findet man in der Dokumentation zu Apache.

Diese Konfiguration geschieht direkt über die Konfigurationsdatei `/etc/apache2/httpd.conf`. Um namensbasierte Virtual Hosts zu aktivieren, muss man eine passende Direktive angeben: `NameVirtualHost *`. Hier reicht die Angabe von `*`, damit Apache einfach alle eingehenden Anfragen entgegen nimmt. Dann müssen noch die einzelnen Hosts konfiguriert werden:

```
<VirtualHost *>
    ServerName www.meinefirma.de
    DocumentRoot /srv/www/htdocs/meinefirma.de
    ServerAdmin webmaster@meinefirma.de
    ErrorLog /var/log/httpd/www.meinefirma.de-error_log
    CustomLog /var/log/httpd/www.meinefirma.de-access_log common
</VirtualHost>

<VirtualHost *>
    ServerName www.meineanderefirma.de
    DocumentRoot /srv/www/htdocs/meineanderefirma.de
    ServerAdmin webmaster@meineanderefirma.de
    ErrorLog /var/log/httpd/www.meineanderefirma.de-error_log
    CustomLog /var/log/httpd/www.meineanderefirma.de-access_log common
</VirtualHost>
```

Hier wie im Folgenden sollte für Apache 2 der Pfad zu den Logdateien von `/var/log/httpd` in `/var/log/apache2` geändert werden. Für die Domain, die der Server ursprünglich gehostet hat (`www.meinefirma.de`), muss dabei ebenfalls ein VirtualHost-Eintrag angelegt werden. In diesem

Beispiel wird also die ursprüngliche Domain und zusätzlich eine weitere Domain (`www.meineanderefirma.de`) auf demselben Server gehostet.

In den `VirtualHost`-Direktiven wird wie bei `NameVirtualHost` ebenfalls ein `*` angegeben. Den Zusammenhang zwischen der Anfrage und dem Virtual Host stellt Apache über das `Host`-Feld im HTTP-Header her. Die Anfrage wird an den Virtual Host weitergeleitet, dessen `ServerName` mit dem in diesem Feld angegebenen Hostnamen übereinstimmt.

Bei den Direktiven `ErrorLog` und `CustomLog` ist es nicht entscheidend, dass die Log-Dateien den Domain-Namen enthalten, man kann hier beliebige Namen verwenden.

`ServerAdmin` benennt die E-Mail-Adresse eines Verantwortlichen, an den man sich bei Problemen wenden kann. Treten Fehler auf, dann gibt Apache diese Adresse in Fehlermeldungen an, die er an den Client zurückschickt.

15.10.2 IP-basierte Virtual Hosts

Für diese Alternative muss man auf einem Rechner mehrere IPs einrichten. Eine Instanz von Apache bedient dann mehrere Domains, wobei jede Domain einer IP zugewiesen ist. Das folgende Beispiel zeigt, wie man Apache so einrichtet, dass er außer auf seiner ursprünglichen IP `192.168.1.10` noch zwei weitere Domains auf zusätzlichen IPs (`192.168.1.20` und `192.168.1.21`) hostet. Dieses konkrete Beispiel funktioniert natürlich nur in einem Intranet, da IPs aus dem Bereich von `192.168.0.0` bis `192.168.255.0` im Internet nicht weitergeleitet (geroutet) werden.

IP-Aliasing einrichten

Damit Apache mehrere IPs hosten kann, muss der Rechner, auf dem er läuft, Anfragen für mehrere IPs akzeptieren. Dies bezeichnet man auch als Multi-IP-Hosting. Dazu muss als erstes IP-Aliasing im Kernel aktiviert sein. Dies ist bei SUSE LINUX standardmäßig der Fall.

Ist der Kernel für IP-Aliasing konfiguriert, kann man mit den Befehlen `ifconfig` und `route` weitere IPs auf dem Rechner einrichten. Um diese Kommandos einzugeben, muss man als `root` eingeloggt sein. Im Folgenden wird angenommen, dass der Rechner bereits eine eigene IP-Adresse, zum Beispiel `192.168.1.10` hat, die dem Netzwerkdevice `eth0` zugewiesen ist.

Welche IP der Rechner verwendet, lässt sich durch Eingabe von `ifconfig` feststellen. Weitere IPs fügt man dann zum Beispiel auf folgende Weise hinzu:

```
/sbin/ifconfig eth0:0 192.168.1.20
/sbin/ifconfig eth0:1 192.168.1.21
```

Alle diese IPs sind dann demselben physikalischen Netzwerkdevice (`eth0`) zugewiesen.

Virtual Hosts mit IPs

Ist IP-Aliasing auf dem System eingerichtet oder der Rechner mit mehreren Netzwerkkarten konfiguriert worden, kann man Apache konfigurieren. Für jeden virtuellen Server gibt man einen eigenen `VirtualHost`-Block an:

```
<VirtualHost 192.168.1.20>
    ServerName www.meineanderefirma.de
    DocumentRoot /srv/www/htdocs/meineanderefirma.de
    ServerAdmin webmaster@meineanderefirma.de
    ErrorLog /var/log/httpd/www.meineanderefirma.de-error_log
    CustomLog /var/log/httpd/www.meineanderefirma.de-access_log common
</VirtualHost>

<VirtualHost 192.168.1.21>
    ServerName www.noeheinefirma.de
    DocumentRoot /srv/www/htdocs/noeheinefirma.de
    ServerAdmin webmaster@noeheinefirma.de
    ErrorLog /var/log/httpd/www.noeheinefirma.de-error_log
    CustomLog /var/log/httpd/www.noeheinefirma.de-access_log common
</VirtualHost>
```

Hier werden `VirtualHost`-Direktiven nur für die zusätzlichen Domains angegeben, die ursprüngliche Domain (`www.meinefirma.de`) wird nach wie vor über die entsprechenden Einstellungen (`DocumentRoot` etc.) außerhalb der `VirtualHost`-Blöcke konfiguriert.

15.10.3 Mehrere Instanzen von Apache

Bei den vorhergehenden Methoden für Virtual Hosts können die Administratoren einer Domain die Daten der anderen Domains lesen. Will man die einzelnen Domains voneinander abschotten, kann man mehrere Instanzen von Apache starten, die jeweils eigene Einstellungen für User, Group etc. in der Konfigurationsdatei verwenden.

In der Konfigurationsdatei gibt man mit der `Listen` Direktive an, für welche IP die jeweilige Instanz von Apache zuständig ist. Analog zum vorhergehenden Beispiel lautet diese Direktive dann für die erste Instanz von Apache:

```
Listen 192.168.1.10:80
```

Für die anderen beiden Instanzen jeweils:

```
Listen 192.168.1.20:80
```

```
Listen 192.168.1.21:80
```

15.11 Sicherheit

15.11.1 Das Risiko gering halten

Wenn man auf einem Rechner keinen Webserver benötigt, sollte man Apache im Runlevel-Editor deaktivieren oder erst gar nicht installieren bzw. deinstallieren. Jeder Server, der auf einem Rechner nicht läuft, ist eine Angriffsmöglichkeit weniger. Dies gilt insbesondere für Rechner, die als Firewalls dienen, auf diesen sollten grundsätzlich nach Möglichkeit keine Server laufen.

15.11.2 Zugriffsrechte

DocumentRoot sollte Root gehören

Per Voreinstellung gehört das Verzeichnis `DocumentRoot (/srv/www/htdocs)` und das CGI-Verzeichnis dem Benutzer `root`. Das sollte man auch so belassen. Sind diese Verzeichnisse für jedermann beschreibbar, kann dort jeder Benutzer Dateien ablegen. Diese Dateien werden dann von Apache ausgeführt, und zwar als Benutzer `wwwrun`. Apache sollte keine Schreibrechte auf die Daten und Skripte haben, die er ausliefert. Deshalb sollten diese nicht dem Benutzer `wwwrun`, sondern zum Beispiel `root` gehören.

Möchte man Usern die Möglichkeit geben, Dateien im Dokument-Verzeichnis von Apache unterzubringen, so sollte man, anstatt dieses für alle beschreibbar zu machen, ein für alle beschreibbares Unterverzeichnis einrichten, zum Beispiel `/srv/www/htdocs/wir_ueber_uns`.

Dokumente aus dem eigenen Home-Verzeichnis veröffentlichen

Eine weitere Möglichkeit dafür zu sorgen, dass Anwender eigene Dateien ins Netz stellen können ist, in der Konfigurationsdatei ein Verzeichnis im Home des Users anzugeben, in dem dieser seine Dateien für die Web-Präsentation ablegen kann (zum Beispiel `~/public_html`). Dies ist bei SUSE LINUX per Voreinstellung aktiviert, die Einzelheiten sind im Abschnitt 15.7.2 auf Seite 436 beschrieben.

Auf diese Webseiten kann dann unter Angabe des Users in der URL zugegriffen werden, die URL enthält die Bezeichnung `<username>` als Kürzel für das entsprechende Verzeichnis im Home-Verzeichnis des Anwenders. Ein Beispiel: Die Eingabe der URL `http://localhost/~tux` in einem Browser zeigt die Dateien aus dem Verzeichnis `public_html` im Home-Verzeichnis des Anwenders `tux` an.

15.11.3 Immer auf dem Laufenden bleiben

Wer einen Webserver betreibt sollte, besonders wenn dieser Webserver öffentlich verfügbar ist, immer auf dem neuesten Stand bleiben, was Fehler und die dadurch möglichen Angriffsflächen angeht.

Quellen für die Recherche nach Exploits und Fixes sind im Abschnitt 15.13.3 auf Seite 450 aufgelistet.

15.12 Troubleshooting

Sollten Probleme auftreten, etwa dass Apache eine Seite gar nicht oder nicht korrekt anzeigt, helfen Ihnen folgende Maßnahmen beim Ermitteln der Fehlerquelle.

- Schauen Sie zunächst in der Fehler-Logdatei nach, ob aus den Meldungen darin hervorgeht, was schief läuft: `/var/log/httpd/error_log` bzw. `/var/log/apache2/error_log`.

Lassen Sie idealerweise in einer Konsole die Logfiles anzeigen, um während der Zugriffe auf den Server parallel mitlesen zu können, wie er reagiert. Geben Sie dazu in einer `root`-Konsole folgenden Befehl ein: `tail -f /var/log/apache2/*_log`.

- Schauen Sie in der Bug-Datenbank nach. Diese ist online unter `http://bugs.apache.org/` verfügbar.

- Lesen Sie die Mailinglisten und Newsgroups. Die Mailingliste für Anwender findet man unter <http://httpd.apache.org/userslist.html>. Als Newsgroup empfehlen sich `comp.infosystems.www.servers.unix` und verwandte Gruppen.
- Wenn alle vorhergehenden Möglichkeiten keine Lösung gebracht haben und Sie sich sicher sind, dass Sie einen Bug in Apache gefunden haben, dann können Sie diesen unter <http://www.suse.de/feedback/> an uns direkt melden.

15.13 Weitere Dokumentation

15.13.1 Apache

Apache kommt mit einer ausführlichen Dokumentation. Wie man diese installiert, ist im Abschnitt 15.6 auf Seite 429 beschrieben. Sie steht dann unter <http://localhost/manual> zur Verfügung. Die aktuellste Dokumentation findet man natürlich immer auf der Homepage von Apache: <http://httpd.apache.org>

15.13.2 CGI

Weitere Informationen zu CGI bieten folgende Seiten:

- <http://apache.perl.org/>
- <http://perl.apache.org/>
- <http://www.modperl.com/>
- <http://www.modperlcookbook.org/>
- <http://www.fastcgi.com/>
- <http://www.boutell.com/cgiic/>

15.13.3 Sicherheit

Unter <http://www.suse.de/security/> werden laufend die aktuellen Patches für die SUSE Pakete zur Verfügung gestellt. Diese URL sollten Sie regelmäßig besuchen, dort können Sie auch die SUSE Security Announcements per Mailingliste abonnieren.

Das Apache-Team betreibt eine offene Informationspolitik, was Fehler in Apache angeht. Aktuelle Meldungen über Bugs und dadurch mögliche Angriffsstellen findet man unter http://httpd.apache.org/security_report.html.

Hat man selber ein Sicherheitsproblem entdeckt (bitte erst auf den eben genannten Seiten verifizieren, ob es wirklich neu ist), kann man es per Mail an security@suse.de oder auch security@apache.org melden.

Weitere Quellen für Informationen über Sicherheitsprobleme bei Apache (und anderen Internet-Programmen) sind:

- <http://www.cert.org/>
- <http://www.vnunet.com/>
- <http://www.securityfocus.com/>

Eine deutsche Seite zum Thema Sicherheit ist die Heise Security Liste <http://www.heisec.de/>

15.13.4 Weitere Quellen

Es empfiehlt sich, bei Schwierigkeiten einen Blick in die SUSE Support-Datenbank zu werfen: <http://sdb.suse.de/>

Eine Online-Zeitung rund um Apache gibt es unter der URL: <http://www.apacheweek.com/>

Die Entstehungsgeschichte von Apache wird unter http://httpd.apache.org/ABOUT_APACHE.html beschrieben. Hier erfährt man auch, warum der Server eigentlich Apache heisst.

Datei-Synchronisation

Viele Menschen benutzen heutzutage mehrere Computer. Ein Computer zu Hause, ein oder mehrere Rechner am Arbeitsplatz und eventuell noch einen Laptop oder PDA für unterwegs. Viele Dateien benötigt man auf allen Computern und möchte sie auch bearbeiten. Dennoch sollen alle Daten überall in aktueller Version zur Verfügung stehen.

16.1	Software zur Datensynchronisation	452
16.2	Kriterien für die Programmauswahl	454
16.3	Einführung InterMezzo	458
16.4	Einführung unison	461
16.5	Einführung CVS	463
16.6	Einführung mailsync	467

16.1 Software zur Datensynchronisation

Auf Computern, die ständig miteinander über ein schnelles Netzwerk in Verbindung stehen, ist die Datensynchronisation kein Problem. Man wählt ein Netzwerkdateisystem, wie zum Beispiel NFS, und speichert die Dateien auf einem Server. Alle Rechner greifen dabei über das Netzwerk auf ein und dieselben Daten zu.

Dieser Ansatz ist unmöglich, wenn die Netzverbindung schlecht oder teilweise gar nicht vorhanden ist. Wer mit einem Laptop unterwegs ist, ist darauf angewiesen, von allen benötigten Dateien Kopien auf der lokalen Festplatte zu haben. Wenn Dateien bearbeitet werden stellt sich aber schnell das Problem der Synchronisierung. Wird auf einem Computer eine Datei verändert, muss man tunlichst aufpassen, dass man die Kopie der Datei auf allen anderen Rechnern auch aktualisiert. Dies kann bei seltenen Kopiervorgängen von Hand mit Hilfe von scp oder rsync erledigt werden. Bei vielen Dateien wird das schnell aufwändig und erfordert hohe Aufmerksamkeit vom Benutzer, um Fehler, wie zum Beispiel das Überschreiben einer neuen mit einer alten Datei, zu vermeiden.

Achtung

Datenverlust droht

Man sollte in jedem Fall sich mit dem verwendeten Programm vertraut machen und seine Funktion testen, bevor man seine Daten über ein Synchronisationssystem verwaltet. Für wichtige Dateien ist ein Backup unerlässlich.

Achtung

Um diese zeitraubende und fehlerträchtige Handarbeit der Datensynchronisation zu vermeiden, gibt es Software, die mit verschiedenen Ansätzen diese Arbeit automatisiert. Die folgenden Kurzeinführungen sollen dem Nutzer eine Vorstellung davon liefern, wie diese Programme funktionieren und genutzt werden können. Vor dem tatsächlichen Einsatz empfehlen wir, die Programmdokumentation sorgfältig zu lesen.

16.1.1 InterMezzo

Die Idee von InterMezzo ist es, ein Dateisystem zu konstruieren, das wie NFS die Dateien über Netzwerk austauscht, dabei aber auf jedem Com-

puter lokale Kopien speichert, sodass auch bei Verlust der Netzwerkverbindung die Dateien zur Verfügung stehen. Die lokalen Kopien können bearbeitet werden. In einer speziellen Logdatei werden alle Veränderungen notiert. Bei Wiederherstellung der Verbindung werden diese Veränderungen dann automatisch weitergegeben und die Dateien abgeglichen. Mehr Informationen zu `intersync` finden Sie sofern das Paket installiert ist unter `/usr/share/doc/packages/intersync/doc/InterMezzo-HOWTO.html`.

16.1.2 unison

Bei `unison` handelt es sich nicht um ein Netzwerkdateisystem. Stattdessen werden Dateien ganz normal lokal gespeichert und bearbeitet. Von Hand kann das Programm `unison` aufgerufen werden, um Dateien zu synchronisieren. Beim ersten Abgleich wird auf den beteiligten zwei Computern eine Datenbank angelegt, in der Prüfsummen, Zeitstempel und Berechtigungen der ausgewählten Dateien gespeichert sind.

Beim nächsten Aufruf kann `unison` erkennen, welche Dateien verändert wurden und die Übertragung vom oder zum anderen Rechner vorschlagen. Im besten Fall kann man alle Vorschläge annehmen.

16.1.3 CVS

Meist zur Versionsverwaltung von Quelltexten von Programmen benutzt bietet CVS die Möglichkeit, Kopien der Dateien auf mehreren Computern zu haben. Damit eignet es sich auch für unseren Zweck.

Bei CVS gibt es eine zentrale Datenbank (`repository`) auf dem Server, welche nicht nur die Dateien, sondern auch die Veränderungen an ihnen abspeichert. Veränderungen, die man lokal durchführt, werden in die Datenbank eingchecked (`commit`) und können von anderen Computern wieder abgeholt werden (`update`). Beides muss vom Benutzer initiiert werden.

Dabei ist CVS bei Veränderungen auf mehreren Computern sehr fehler-tolerant: Die Veränderungen werden zusammengeführt und nur wenn in den gleichen Zeilen Veränderungen stattfanden, gibt es einen Konflikt. Die Datenbank bleibt im Konfliktfall in einem konsistenten Zustand und der Konflikt ist nur auf dem Client Computer sichtbar und zu lösen.

16.1.4 mailsync

Im Vergleich zu den bisher erwähnten Synchronisationswerkzeugen dient Mailsync einzig und allein der Aufgabe, E-Mails zwischen Mailboxen zu synchronisieren. Es kann sich sowohl um lokale Mailbox-Dateien als auch um Mailboxen handeln, die auf einem IMAP-Server untergebracht sind.

Dabei wird für jede Nachricht aufgrund der im E-Mail-Header enthaltenen Message-ID einzeln entschieden, ob sie synchronisiert bzw. gelöscht werden muss. Es ist sowohl die Synchronisation zwischen einzelnen Mailboxen, als auch zwischen Hierarchien von Mailboxen möglich.

16.2 Kriterien für die Programmauswahl

16.2.1 Client-Server-Modell versus Gleichberechtigung

Zur Verteilung von Daten sind zwei verschiedene Modelle verbreitet. Einerseits kann man einen zentralen Server verwenden, mit dem alle anderen Computer (sog. Clients) ihre Dateien abgleichen. Der Server muss dann zumindest zeitweise über ein Netzwerk von allen Clients erreichbar sein. Dieses Modell wird von CVS und InterMezzo verwendet. Andererseits können alle Computer gleichberechtigt vernetzt sein und ihre Daten gegenseitig abgleichen. Diesen Ansatz verfolgt unison.

16.2.2 Portabilität

Bei InterMezzo handelt es sich um eine Lösung, die momentan lediglich auf Linux-Systemen funktioniert. In der Vergangenheit war sie sogar auf Architekturen mit 32bit little-endian (ix86) beschränkt. Mit dem Umstieg vom perl-basierten lento auf InterSync ist diese Beschränkung aber mittlerweile aufgehoben. Dennoch ist beim Abgleich zwischen verschiedenen Architekturen Vorsicht angezeigt, da dies ein wenig getestetes Feature ist. CVS und unison sind auch auf vielen anderen Betriebssystemen wie anderen Unices und Windows erhältlich.

16.2.3 Interaktiv vs. Automatisch

Bei InterMezzo erfolgt der Datenabgleich normalerweise automatisch im Hintergrund, sobald die Netzwerkverbindung zum Server hergestellt werden kann. Lediglich wenn Konflikte auftreten, muss eingegriffen werden.

Bei CVS und unison wird der Datenabgleich manuell vom Benutzer angestoßen. Dies erlaubt die genauere Kontrolle über die abzugleichenden Dateien und einen einfacheren Umgang mit Konflikten. Andererseits kann es leicht passieren, dass der Abgleich zu selten durchgeführt wird, wodurch sich die Chancen auf einen Konflikt erhöhen.

16.2.4 Geschwindigkeit

unison und CVS erscheinen aufgrund ihres interaktiven Charakters langsamer als InterMezzo, welches im Hintergrund arbeitet. CVS ist im Allgemeinen etwas schneller als unison.

16.2.5 Konflikte: Auftreten und Lösung

Konflikte treten bei CVS nur selten auf, selbst wenn mehrere Leute an einem großen Programmprojekt arbeiten. Die Dokumente werden hier zeilenweise zusammengeführt. Wenn ein Konflikt auftritt, dann ist davon immer nur ein Client betroffen. In der Regel sind Konflikte mit CVS einfach zu lösen. Bei unison bekommt man Konflikte mitgeteilt und kann die Datei einfach vom Abgleich ausnehmen. Änderungen lassen sich aber nicht so einfach zusammenführen wie bei CVS.

Aufgrund des nicht-interaktiven Charakters von InterMezzo lassen sich Konflikte nicht einfach interaktiv lösen. Wenn Konflikte auftreten, bricht InterSync mit einer Warnung ab. Der Systemadministrator muss in diesem Fall eingreifen und ggf. Dateien von Hand (rsync/scp) übertragen, um wieder einen konsistenten Zustand herzustellen.

16.2.6 Dateiwahl, Dateien hinzufügen

Bei InterMezzo wird ein ganzes Dateisystem synchronisiert. Neu hinzugekommene Dateien innerhalb des Dateisystems erscheinen automatisch auch auf den anderen Computern.

Bei unison wird in der einfachsten Konfiguration ein ganzer Verzeichnisbaum synchronisiert. Dort neu erscheinende Dateien werden auch automatisch in die Synchronisation mit einbezogen.

Bei CVS müssen neue Verzeichnisse und Dateien explizit mittels `cvsv add` hinzugefügt werden. Daraus resultiert eine genauere Kontrolle über die zu synchronisierenden Dateien. Andererseits werden neue Dateien gerne vergessen, vor allem, wenn aufgrund der Anzahl der Dateien die '?' in der Ausgabe von `cvsv update` ignoriert werden.

16.2.7 Geschichte

CVS bietet eine Rekonstruktion alter Dateiversionen als zusätzliches Feature. Bei jeder Veränderung kann man einen kurzen Bearbeitungsvermerk hinzufügen und später die Entwicklung der Dateien aufgrund des Inhalts und der Vermerke gut nachvollziehen. Für Diplomarbeiten und Programmtexte ist dies eine wertvolle Hilfe.

16.2.8 Datenmenge / Plattenbedarf

Auf jedem der beteiligten Computer benötigt man für alle verteilten Daten genügend Platz auf der Festplatte. Bei CVS fällt zusätzlich der Platzbedarf für die Datenbank (dem repository) auf dem Server an. Da dort auch die Geschichte der Dateien gespeichert wird, ist dieser deutlich größer als der reine Platzbedarf. Bei Dateien im Textformat hält sich dies in Grenzen, da nur geänderte Zeilen neu gespeichert werden müssen. Bei binären Dateien wächst hingegen der Platzbedarf bei jeder Änderung um die Größe der Datei.

16.2.9 GUI

unison kommt mit einer grafischen Oberfläche, die anzeigt, welche Abgleiche unison vornehmen möchte. Man kann den Vorschlag annehmen oder einzelne Dateien vom Abgleich ausnehmen. Daneben kann man auch im Textmodus interaktiv die einzelnen Vorgänge bestätigen.

CVS wird von erfahrenen Benutzern normalerweise an der Kommandozeile benutzt. Es gibt jedoch grafische Oberflächen für Linux (`cervisia`, ...) und auch für Windows (`wincvs`). Viele Entwicklungstools (zum Beispiel `kdevelop`) und Texteditoren (zum Beispiel `emacs`) haben eine Unterstützung für CVS. Die Behebung von Konflikten wird mit diesen Frontends oft sehr vereinfacht.

So viel Komfort bietet InterMezzo nicht. Andererseits erfordert es ja normalerweise keine Interaktion und sollte – einmal eingerichtet – einfach nur im Hintergrund seinen Dienst tun.

16.2.10 Anforderungen an den Benutzer

Die Einrichtung von InterMezzo ist relativ schwierig und sollte nur von einem Systemverwalter mit etwas Erfahrung im Linux-Bereich vorgenommen werden. Zur Einrichtung sind `root`-Rechte nötig. `unison` ist recht einfach zu benutzen und bietet sich auch für Anfänger an.

CVS ist etwas schwieriger zu benutzen. Man sollte zu dessen Verwendung das Zusammenspiel zwischen Repository, und lokalen Daten verstanden haben. Veränderungen an den Daten sollten zunächst immer lokal mit dem Repository zusammengeführt werden. Hierzu dient der Befehl `cvs update`. Nachdem dies geschehen ist, müssen die Daten mit dem Befehl `cvs commit` wieder in das Repository zurückgeschickt werden. Wenn man dies verinnerlicht hat, ist CVS auch für Anfänger leicht zu benutzen.

16.2.11 Sicherheit gegen Angriffe

Die Sicherheit bei der Übertragung der Daten gegenüber Abhören oder gar Verändern der Daten sollte idealerweise gewährleistet werden.

Sowohl `unison` als auch CVS lassen sich einfach über `ssh` (Secure Shell) benutzen und sind dann gut gegen obige Angriffe gesichert. Es sollte vermieden werden, CVS oder `unison` über `rsh` (Remote Shell) einzusetzen und auch Zugriffe über den CVS `pserver` Mechanismus sind in ungeschützten Netzwerken nicht empfehlenswert.

Bei InterMezzo erfolgt der Datenabgleich über `http`. Dieses Protokoll kann leicht abgehört oder verfälscht werden. Zur Erhöhung der Sicherheit ist der Einsatz von SSL vorgesehen, derzeit ist dieses Feature aber noch nicht benutzbar.

16.2.12 Sicherheit gegen Datenverlust

CVS wird schon sehr lange von vielen Entwicklern zur Verwaltung ihrer Programmprojekte benutzt und ist ausgesprochen stabil. Durch das Speichern der Entwicklungsgeschichte ist man bei CVS sogar gegen gewisse Benutzerfehler (zum Beispiel irrtümlichen Löschen einer Datei) geschützt. `unison` ist noch relativ neu, aber weist eine hohe Stabilität auf. Es ist empfindlicher gegen Benutzerfehler. Wenn man der Synchronisierung eines Löschvorgangs bei einer Datei einmal zustimmt, ist diese nicht mehr zu retten.

InterMezzo sollte gegenwärtig noch als experimentell gelten. Da die Dateien in einem darunter liegenden Dateisystem gespeichert werden, ist die

Wahrscheinlichkeit eines größeren Datenverlustes relativ gering. Aber der Datenabgleich selbst kann prinzipiell schief gehen und zerstörte Dateien hinterlassen. Auch bei Benutzerfehlern ist die Toleranz gering: Das lokale Löschen einer Datei wird auf allen anderen synchronisierten Computern einfach nachvollzogen. Insofern sind Backups sehr ratsam.

Tabelle 16.1: Features der Datensynchronisationstools -- = sehr schlecht, - = schlecht bzw. nicht vorhanden, o = mittelmäßig, + = gut, ++ = sehr gut, x = vorhanden

	InterMezzo	unison	CVS	mailsync
CS/Gleich	C-S	gleich	C-S	gleich
Portabil.	Linux(i386)	Lin,Un*x,Win	Lin,Un*x,Win	Lin,Un*x
Interaktiv	-	x	x	-
Geschwind.	++	-	o	+
Konflikte	-	o	++	+
Dateiwahl	Dateisystem	Verzeichnis	Auswahl	Mailbox
Geschichte	-	-	x	-
Plattenbed.	o	o	--	+
GUI	-	+	o	-
Schwierigk.	-	+	o	o
Angriffe	-	+(ssh)	+(ssh)	+(SSL)
Datenverlust	o	+	++	+

16.3 Einführung InterMezzo

16.3.1 Architektur

Bei InterMezzo handelt es sich um einen eigenen Dateisystemtyp. Die Dateien werden auf jedem der Rechner lokal auf der Festplatte gespeichert. Dazu wird eines der vorhandenen Dateisysteme von Linux benutzt, vorzugsweise ext3 oder eines der anderen Dateisysteme mit Journaling. Nach entsprechender Vorbereitung der Partition wird das Dateisystem mit dem Typ `intermezzo` gemountet (eingehängt). Der Kernel lädt ein Modul mit

der InterMezzo-Unterstützung und fortan werden Veränderungen, die in diesem Dateisystem durchgeführt werden, in eine Log-Datei geschrieben. Nach diesen Vorarbeiten kann InterSync gestartet werden. Dieser startet einen Webserver, wie zum Beispiel Apache, auf den andere Computer zugreifen können, um Daten auszutauschen. Wenn man einen Client konfiguriert, muss man InterSync den Namen des Servers mitteilen. Dieser wird dann kontaktiert. Zur Erkennung des Dateisystems wird dabei eine frei wählbare Bezeichnung für das Dateisystem übergeben, das so genannte `fileset`.

InterSync ist die Weiterentwicklung vom älteren InterMezzo System, welches zum Datenabgleich einen in Perl geschriebenen Daemon namens `lento` benutzte. In der Dokumentation zu InterSync finden sich gelegentlich noch Referenzen zu diesem älteren System. Es wurde durch InterSync jedoch abgelöst. Das Modul, welches sich in Standardkernels befindet ist leider noch auf dem Niveau von `lento` und funktioniert nicht mit InterSync. Beim SUSE-Kernel ist jedoch ein neueres Modul vorhanden. Bei selbstgebaute Kerneln, sollte das Kernel-Modul mit Hilfe des Pakets `km_inter-sync` gebaut werden.

Zum Einrichten von InterMezzo werden Systemadministratorrechte benötigt. Wie aus dem Vergleich hervorgeht, ist die Einrichtung von InterMezzo vergleichsweise schwierig und sollte daher nur von erfahrenen Systemadministratoren durchgeführt werden. Die unten beschriebene Konfiguration sieht keinerlei Schutzmechanismen vor. Das heißt, dass bössartige Netzbewohner ohne größere Schwierigkeiten Ihre über InterMezzo synchronisierten Daten ausspähen und manipulieren können. Die Einrichtung sollte nur in einer vertrauenswürdigen Umgebung wie zum Beispiel einem privaten, kabelgebundenen Netzwerk hinter einer Firewall stattfinden.

16.3.2 Einrichten eines InterMezzo-Servers

Einer der Computer, vorzugsweise einer mit guter Anbindung ans Netzwerk, bekommt die Rolle des Servers zugewiesen. Über ihn läuft der gesamte Verkehr zum Datenabgleich.

Zum Speichern der Daten muss ein eigenes Dateisystem eingerichtet werden. Falls man keine Partition mehr zur Verfügung stehen hat und nicht LVM benutzt, kann man das Dateisystem am einfachsten über ein so genanntes `loop device` anlegen. Dabei wird eine Datei im lokalen Dateisystem als eigenes Dateisystem behandelt.

Im folgenden Beispiel soll ein InterMezzo/`ext3` Dateisystem im Wurzelverzeichnis mit 256 MB Größe eingerichtet werden. Das `fileset` soll die Bezeichnung `fset0` bekommen.

```
dd if=/dev/zero of=/izo0 bs=1024 count=262144
mkizofs -r fset0 /izo0 # Die Warnung kann ignoriert werden
```

Dieses Dateisystem wird nun in `/var/cache/intermezzo` eingehängt:

```
mount -t intermezzo -o fileset=fset0,loop /izo /var/cache/intermezzo
```

Später wird man dies durch einen Eintrag in der Datei `/etc/fstab` automatisch beim Booten erledigen lassen. Nun sollte InterSync konfiguriert werden. Dazu ist die Anpassung von `/etc/sysconfig/intersync` nötig. Tragen Sie folgende Zeilen in die Datei ein:

```
INTERSYNC_CLIENT_OPTS="--fset=fset0"
INTERSYNC_CACHE=/var/cache/intermezzo}
INTERSYNC_PROXY=""
```

Starten Sie InterSync mit dem Befehl: `/etc/init.d/intersync start`. Um dies in Zukunft beim Systemstart automatisch zu veranlassen, kann der Dienst in die Liste der zu startenden Dienste eingetragen werden:
`insserv intersync`.

16.3.3 Einrichten von InterMezzo-Clients

Die Einrichtung der Clients (ein Server kann viele Clients bedienen) unterscheidet sich kaum von der des Servers. Der einzige Unterschied ist, dass man bei der Konfiguration von `/etc/sysconfig/intersync` bei der Variable `INTERSYNC_CLIENT_OPTS` zusätzlich den Namen des Servers mit angeben muss:

```
INTERSYNC_CLIENT_OPTS="--fset=fset0 --server=sun.example.com"
```

Für `sun.example.com` ist natürlich der Netzwerkname des Servers einzutragen. Es empfiehlt sich übrigens, die Dateisysteme auf allen Computern mit der gleichen Größe anzulegen.

16.3.4 Problembehebung

Sobald ein Client gestartet ist, sollten Veränderungen von Dateien unter dem Verzeichnis `/var/cache/intermezzo/` auch auf dem Server und allen anderen Clients sichtbar werden. Falls dies nicht der Fall ist,

liegt das meistens daran, dass keine Verbindung zum Server zustande kommt oder ein Konfigurationsfehler, wie zum Beispiel das Nichtübereinstimmen der fileset Bezeichnung vorliegt. Zur Diagnose ist es hilfreich, die Logmeldungen im Systemlog `/var/log/messages` zu analysieren. Der gestartete Webserver protokolliert seine Daten im Verzeichnis `/var/intermezzo-X/`. Die Logdatei des Kernels, welche Veränderungen am Dateisystem protokolliert, befindet sich unter `/var/cache/intermezzo/.intermezzo/fset0/kml` und kann mittels `kmlprint` ausgegeben werden.

Bei auftretenden Konflikten wird normalerweise einer der InterSync-Prozesse beendet. Wenn Dateisynchronisation nicht mehr stattfindet, sollte man auch nach entsprechenden Hinweisen in den Logdateien suchen und mit `/etc/init.d/intersync status` überprüfen, ob der Synchronisationsservice noch läuft.

Ansonsten muss hier auf die Dokumentation des Pakets verwiesen werden, zu finden im Verzeichnis `/usr/share/doc/packages/intersync/`, oder auf die Webseite von Intermezzo: <http://www.inter-mezzo.org/>

16.4 Einführung unison

16.4.1 Einsatzgebiete

Unison ist hervorragend für den Abgleich und Transfer ganzer Verzeichnissbäume geeignet. Der Abgleich findet in beide Richtungen statt und lässt sich intuitiv über ein grafisches Frontend steuern (alternativ kann aber auch die Konsolen-Version verwenden). Der Abgleich lässt sich auch automatisieren (das heißt keine Interaktion mit dem Benutzer), wenn man weiß, was man tut.

16.4.2 Voraussetzungen

Unison muss sowohl auf dem Client, als auch auf dem Server installiert sein, wobei mit Server ein zweiter, entfernter Rechner gemeint ist (im Gegensatz zu CVS, siehe Abschnitt 16.1.3 auf Seite 453).

Da wir uns im Folgenden auf die Benutzung von unison mit ssh beschränken, muss ein ssh-Client auf dem Client und ein ssh-Server auf dem Server installiert sein.

16.4.3 Bedienung

Das Grundprinzip bei Unison ist, zwei Verzeichnisse (so genannte "roots") aneinander zu binden. Diese Bindung ist symbolisch zu verstehen, es handelt sich also nicht um eine Online-Verbindung. Angenommen, wir haben folgendes Verzeichnis-Layout:

Client:	/home/tux/dir1
Server:	/home/geeko/dir2

Diese beiden Verzeichnisse sollen synchronisiert werden. Auf dem Client ist der User als tux bekannt, auf dem Server dagegen als geeko. Zunächst sollte ein Test durchgeführt werden, ob die Kommunikation zwischen Client und Server funktioniert:

```
unison -testserver /home/tux/dir1 ssh://geeko@server//homes/geeko/dir2
```

Die häufigsten Probleme, die hierbei auftreten können:

- die auf dem Client und Server eingesetzten Versionen von unison sind nicht kompatibel
- der Server lässt keine SSH-Verbindung zu
- keiner der beiden angegebenen Pfade existiert

Funktioniert soweit alles, lässt man die Option `-testserver` weg. Bei der Erstsynchronisierung kennt unison das Verhältnis der beiden Verzeichnisse noch nicht und macht von daher Vorschläge für die Transferrichtung der einzelnen Dateien und Verzeichnisse. Die Pfeile in der Spalte Action geben die Transferrichtung an. Ein '?' bedeutet, dass unison keinen Vorschlag bzgl. der Transferrichtung machen kann, da beide Versionen in der Zwischenzeit verändert wurden bzw. neu sind.

Mit den Pfeiltasten kann man die Transferrichtung für jeden Eintrag einstellen. Stimmen die Transferrichtungen für alle angezeigten Einträge, dann klickt man auf 'Go'.

Das Verhalten von unison (zum Beispiel ob in eindeutigen Fällen die Synchronisation automatisch durchgeführt werden soll), lässt sich beim Starten per Kommandozeilenparameter steuern. Eine komplette Liste aller Parameter liefert `unison -help`.

Über die Synchronisation wird für jede Bindung im Benutzer-Verzeichnis `~/.unison` Protokoll geführt. In diesem Verzeichnis lassen sich auch Konfigurationssets ablegen, zum Beispiel `~/.unison/example.prefs`:

Beispiel 16.1: Die Datei `/.unison/example.prefs`

```
root=/home/foobar/dir1
root=ssh://fbar@server//homes/fbar/dir2
batch=true
```

Um die Synchronisation anzustoßen, genügt es dann einfach, diese Datei als Kommandozeilenargument anzugeben: `unison example.prefs`

16.4.4 Weiterführende Literatur

Die offizielle Dokumentation zu `unison` ist äußerst umfangreich; in diesem Abschnitt wurde nur eine Kurzeinführung dargestellt. Unter <http://www.cis.upenn.edu/~bcpierce/unison/> bzw. im SUSE-Paket `unison` ist ein komplettes Handbuch verfügbar.

16.5 Einführung CVS

16.5.1 Einsatzgebiete

CVS bietet sich zur Synchronisation an, wenn einzelne Dateien häufig bearbeitet werden und in einem Dateiformat vorliegen wie ASCII-Text oder Programmquelltext. Die Verwendung von CVS für die Synchronisation von Daten in anderen Formaten (zum Beispiel JPEG-Dateien) ist zwar möglich, führt aber schnell zu großen Datenmengen, da jede Variante einer Datei dauerhaft auf dem CVS-Server gespeichert wird. Zudem bleiben in solchen Fällen die meisten Möglichkeiten von CVS ungenutzt.

Die Verwendung von CVS zur Dateisynchronisation ist nur dann möglich, wenn alle Arbeitsplatzrechner auf denselben Server zugreifen können!

Im Gegensatz dazu wäre zum Beispiel mit dem Programm `unison` auch folgendes Szenario möglich:

$A > B > C > S$

A, B, C sind Rechner, die die fraglichen Daten bearbeiten können.

16.5.2 Einrichten eines CVS-Servers

Der Server ist der Ort, wo alle gültigen Dateien liegen, d. h. insbesondere die aktuelle Version jeder Datei. Als Server kann zum Beispiel ein fest installierter Arbeitsplatzrechner dienen. Wünschenswert ist, dass die Daten des CVS-Servers regelmäßig in ein Backup mit einbezogen werden.

Ein sinnvoller Weg, einen CVS-Server aufzusetzen, ist, dem Benutzer Zugang per SSH auf den Server zu gestatten. So kann zum Beispiel ein fest installierter Arbeitsplatzrechner als Server dienen.

Ist auf diesem Server der Benutzer als tux bekannt und sowohl auf dem Server als auch auf dem Client (zum Beispiel Notebook) die CVS-Software installiert, sollte man auf der Client-Seite dafür Sorge tragen, dass folgende Umgebungsvariablen gesetzt sind:

```
CVS_RSH=ssh CVSROOT=tux@server:/serverdir
```

Mit dem Befehl `cvs init` lässt sich dann von der Client-Seite aus der CVS-Server initialisieren (dies muss nur einmal geschehen).

Abschließend muss ein Name für die Synchronisation festgelegt werden. Wählen oder erzeugen Sie auf einem Client ein Verzeichnis, das ausschließlich Dateien enthält, die von CVS verwaltet werden sollen (es kann auch leer sein). Der Name des Verzeichnisses spielt dabei keine Rolle und soll in diesem Beispiel `synchome` sein. Wechseln Sie in dieses Verzeichnis. Um den Synchronisationsnamen auf `synchome` zu setzen, gibt man Folgendes ein:

```
cvs import synchome tux tux_0
```

Hinweis: Viele Kommandos von CVS verlangen einen Kommentar. Zu diesem Zweck ruft CVS einen Editor auf (den in der Umgebungsvariable `$EDITOR` definierten, ansonsten `vi`). Den Aufruf des Editors kann man umgehen, indem man den Kommentar bereits auf der Kommandozeile angibt, wie zum Beispiel in

```
cvs import -m 'dies ist ein Test' synchome tux tux_0
```

16.5.3 Benutzung von CVS

Ab diesem Zeitpunkt kann das Synchronisationsrepository von beliebigen Rechnern ausgecheckt werden:

```
cv$ co synchome
```

Man erhält dadurch ein neues Unterverzeichnis `synchome` auf dem Client. Hat man Änderungen durchgeführt, die man an den Server übermitteln will, so wechselt man in das `synchome`-Verzeichnis (oder auch ein Unterverzeichnis desselben) und gibt folgenden Befehl ein:

```
cv$ commit
```

Dabei werden standardmäßig alle Dateien, die unterhalb des aktuellen Verzeichnisses liegen, und zum lokalen CVS gehören an den Server übermittelt. Will man nur einzelne Dateien oder Verzeichnisse übermitteln, so muss man diese angeben:

```
cv$ commit dateil ... verzeichnisl ...
```

Neue Dateien/Verzeichnisse muss man vor der Übermittlung an den Server als dem CVS zugehörig deklarieren:

```
cv$ add dateil ... verzeichnisl ...
```

und danach übermitteln

```
cv$ commit dateil ... verzeichnisl ...
```

Wechselt man nun den Arbeitsplatz, sollte, falls dies nicht schon in früheren Sessions am gleichen Arbeitsplatz geschehen ist, das Synchronisationsrepository ausgecheckt werden. Der Abgleich mit dem Server wird über das Kommando angestoßen:

```
cv$ update
```

Man kann auch selektiv Dateien/Verzeichnisse updaten:

```
cv$ update dateil ... verzeichnisl ...
```

Will man im voraus die Unterschiede zu den auf dem Server gespeicherten Versionen sehen, so geht dies mit dem Befehl `cv$ diff` oder explizit mit:

```
cv$ diff dateil ... verzeichnisl ...
```

Alternativ kann man sich auch anzeigen lassen, welche Dateien von einem Update betroffen wären: `cvs -nq update`. Bei einem Update werden (u. a.) folgende Status-Symbole verwendet:

- U** Die lokale Version wurde auf den neuesten Stand gebracht. Dies betrifft alle Dateien, die der Server bereitstellt, die aber nicht lokal existierten.
- M** Die lokale Version wurde modifiziert. Soweit sich diese auf dem Server verändert hat, konnten die Änderungen auch lokal eingepflegt werden.
- P** Die lokale Version wurde mit Hilfe eines Patches auf den aktuellen Stand gebracht.
- ?** Diese Datei ist nicht im CVS.

Der Status **M** markiert Dateien, die gerade bearbeitet werden. Um die Änderungen zum Server zurückzusenden, muss man den Befehl `cvs commit` ausführen. Wenn man stattdessen auf die eigenen Änderungen verzichten möchte, um den aktuellen Stand des Servers zu übernehmen, entfernt man die lokale Kopie, und führt erneut ein Update durch. Die fehlende Datei wird dann vom Server geholt.

Wenn von verschiedenen Benutzern die gleiche Datei an derselben Stelle editiert wurde, entsteht eine Situation, in der CVS nicht entscheiden kann, welche Version verwendet werden soll. Dieser Fall wird bei einem Update mit dem Symbol **C** gekennzeichnet. Zur Lösung eines Konfliktes bieten sich verschiedene Vorgehensweisen an. In der entsprechenden Datei werden an den betreffenden Stellen Konfliktmarken eingefügt, die manuell editiert werden können. Für Anfänger ist es empfehlenswert, in diesem Fall auf ein Hilfsprogramm wie `cervisia` zurückzugreifen. Alternativ kann man auch die eigene Datei umbenennen und erneut ein Update ausführen. Sobald man die Änderungen an der aktuellen Datei beendet hat, sollte man diese dem Server mit dem Befehl `cvs commit` übergeben. Dadurch wird die Wahrscheinlichkeit für Konflikte reduziert.

16.5.4 Weiterführende Literatur

Die Möglichkeiten von CVS sind umfangreich und es konnte hier nur ein kleiner Einblick gegeben werden. Weiterführende Dokumentation gibt es unter anderem unter <http://www.cvshome.org/> und <http://www.gnu.org/manual/>.

16.6 Einführung mailsync

16.6.1 Einsatzgebiet

Mailsync bietet sich im Wesentlichen für drei Aufgaben an:

- Synchronisation lokal gespeicherter E-Mails mit E-Mails, die auf einem Server gespeichert sind.
- Migration von Mailboxen in ein anderes Format bzw. auf einen anderen Server.
- Integritätscheck einer Mailbox bzw. der Suche nach Duplikaten.

16.6.2 Konfiguration und Benutzung

Mailsync unterscheidet zwischen der Mailbox an sich (einem so genannten Store) und der Verknüpfung zwischen zwei Mailboxen (einem so genannten Channel). Die Definitionen der Stores und Channels wird in der Datei `~/.mailsync` abgelegt. Im Folgenden sollen einige Beispiele für Stores vorgestellt werden. Eine einfache Definition sieht zum Beispiel so aus:

```
store saved-messages {  
    pat      Mail/saved-messages  
    prefix   Mail/  
}
```

`Mail/` ist ein Unterverzeichnis im Home des Benutzers, welches Ordner mit E-Mails enthält, unter anderem den Ordner `saved-messages`. Ruft man nun mailsync mit dem Befehl `mailsync -m saved-messages` auf, wird ein Index aller Nachrichten in `saved-messages` aufgelistet. Eine weitere Definition kann wie folgt aussehen:

```
store localdir {  
    pat      Mail/*  
    prefix   Mail/  
}
```

Hier bewirkt der Aufruf von `mailsync -m localdir` das Auflisten aller Nachrichten, die in den Ordnern unter `Mail/` gespeichert sind. Der Aufruf `mailsync localdir` listet dagegen die Ordernamen.

Die Spezifikation eines Stores auf einem IMAP-Server sieht zum Beispiel so aus:

```
store imapinbox {
    server {mail.uni-hannover.de/user=gulliver}
    ref    {mail.uni-hannover.de}
    pat    INBOX
}
```

Im obigen Fall wird nur der Hauptordner auf dem IMAP-Server adressiert, ein Store für die Unterordner sieht dagegen wie folgt aus:

```
store imapdir {
    server {mail.uni-hannover.de/user=gulliver}
    ref    {mail.uni-hannover.de}
    pat    INBOX.*
    prefix INBOX.
}
```

Unterstützt der IMAP-Server verschlüsselte Verbindungen, sollte man die Server-Spezifikation wie folgt abändern:

```
server {mail.uni-hannover.de/ssl/user=gulliver}
```

bzw. (falls das Server-Zertifikat nicht bekannt ist) in

```
server {mail.uni-hannover.de/ssl/novalidate-cert/user=gulliver}
```

Nun sollen die Ordner unter Mail/ mit den Unterverzeichnissen auf dem IMAP-Server verbunden werden:

```
channel Ordner localdir imapdir {
    msinfo .mailsync.info
}
```

Dabei wird sich Mailsync in der mit `msinfo` angegebenen Datei merken, welche Nachrichten schon synchronisiert wurden. Ein Aufruf von `mailsync Ordner` bewirkt nun Folgendes:

- Auf beiden Seiten wird das Mailbox-Muster (`pat`) expandiert.
- Von den dabei entstehenden Ordnernamen wird jeweils das Präfix (`prefix`) entfernt.
- Die Ordner werden paarweise synchronisiert (bzw. angelegt, falls noch nicht vorhanden).

Ein Ordner `INBOX.sent-mail` auf dem IMAP-Server wird also mit dem lokalen Ordner `Mail/sent-mail` synchronisiert (obige Definitionen vorausgesetzt). Dabei wird die Synchronisation zwischen den einzelnen Ordnern folgendermaßen durchgeführt:

- Existiert eine Nachricht schon auf beiden Seiten, passiert gar nichts.
- Fehlt die Nachricht auf einer Seite und ist neu (d. h. nicht in der `msinfo`-Datei protokolliert) wird sie dorthin übertragen.
- Existiert die Nachricht nur auf einer Seite und ist alt (d. h. bereits in der `msinfo`-Datei protokolliert), wird sie dort gelöscht (da sie hoffentlich auf der anderen Seite existiert hatte und dort gelöscht wurde).

Um im voraus ein Bild davon zu erhalten, welche Nachrichten bei einer Synchronisation übertragen und welche gelöscht werden, ruft man `Mail-sync` mit einem Channel *und* einem Store gleichzeitig auf: `mailsync Ordner localdir`.

Dadurch erhält man eine Liste aller Nachrichten, die lokal neu sind, als auch eine Liste aller Nachrichten, die bei einer Synchronisation auf der IMAP-Seite gelöscht werden würden!

Spiegelbildlich erhält man mit `mailsync Ordner imapdir` eine Liste aller Nachrichten, die auf der IMAP-Seite neu sind, als auch eine Liste aller Nachrichten, die bei einer Synchronisation lokal gelöscht werden würden.

16.6.3 Mögliche Probleme

Im Fall eines Datenverlustes ist es das sicherste Vorgehen, die zugehörige Channel-Protokolldatei `msinfo` zu löschen. Dadurch gelten alle Nachrichten, die nur auf jeweils einer Seite existieren, als neu und werden beim nächsten Sync übertragen.

Es werden nur solche Nachrichten in die Synchronisation einbezogen, die eine Message-ID tragen. Nachrichten, in denen diese fehlt, werden schlichtweg ignoriert, das heisst weder übertragen noch gelöscht. Das Fehlen einer Message-ID kommt in der Regel durch fehlerhafte Programme im Prozess der Mailzustellung oder -erzeugung zustande.

Auf bestimmten IMAP-Servern wird der Hauptordner mittels `INBOX`, Unterordner mittels eines beliebigen Namen angesprochen (im Gegensatz zu `INBOX` und `INBOX.name`). Dadurch ist es bei solchen IMAP-Server nicht möglich, ein Muster ausschließlich für die Unterordner zu spezifizieren.

Die von Mailsync benutzen Mailbox-Treiber (c-client), setzen nach erfolgreicher Übertragung der Nachrichten auf einen IMAP-Server ein spezielles Status-Flag, wodurch es manchen E-Mail-Programmen, wie zum Beispiel muft, nicht möglich ist, die Nachrichten als neu zu erkennen. Das Setzen dieses speziellen Status-Flags lässt sich in Mailsync mit der Option `-n` unterbinden.

16.6.4 Weiterführende Literatur

Das im Paket mailsync enthaltene README unter `/usr/share/doc/packages/maailsync/` enthält weitere Informationen und Hinweise. Von besonderem Interesse ist in diesem Zusammenhang auch das RFC 2076 "Common Internet Message Headers".

Heterogene Netzwerke

Linux kann nicht nur mit anderen Linux-Rechnern, sondern auch mit Windows- und Macintosh-Rechnern sowie über Novell-Netzwerke kommunizieren. Dieses Kapitel zeigt Ihnen, worauf Sie dabei achten müssen und wie Sie entsprechende heterogene Netzwerke konfigurieren können.

17.1 Samba	472
17.2 Netatalk	481
17.3 NetWare-Emulation mit MARSNWE	488

17.1 Samba

17.1.1 Einführung in Samba

Mit dem Programmpaket Samba kann ein Unix-Rechner zu einem File- und Printserver für DOS-, Windows- und OS/2 Rechner ausgebaut werden. Das Samba-Projekt wird vom Samba Team betreut und wurde ursprünglich von dem Australier Andrew Tridgell entwickelt.

Samba ist inzwischen ein sehr umfassendes Produkt, so dass wir an dieser Stelle lediglich einen Einblick in seine Funktionalität liefern können. Jedoch kommt die Software mit umfassender digitaler Dokumentation. Diese besteht einerseits aus Handbuchseiten — zwecks Umfang rufen Sie bitte `apropos samba` auf der Kommandozeile auf — und andererseits aus Dokumenten und Beispielen, die Sie bei installiertem Samba auf Ihrem System unter `/usr/share/doc/packages/samba` finden. Dort finden Sie im Unterverzeichnis `examples` auch die kommentierte Beispielkonfiguration `smb.conf`. SuSE.

Beginnend mit SUSE LINUX Version 9.1 steht Ihnen das Paket `samba` in der Version 3 zur Verfügung. Einige wichtige Neuerungen dieses Paketes sind:

- Active Directory support.
- Unicode Support wurde stark verbessert.
- Die internen Authentifizierungsmechanismen wurden komplett überarbeitet.
- Verbesserte Unterstützung für das Windows 200x/XP Drucksystem.
- Konfiguration als Mitglieds-Server in Active-Directory-Domänen.
- NT4-Domänenübernahme um von einer NT4 Domäne zu einer Samba Domäne zu migrieren.

Samba benutzt das SMB-Protokoll (Server Message Block), das auf den NetBIOS Diensten aufgesetzt ist. Auf Drängen der Firma IBM gab die Firma Microsoft das Protokoll frei, sodass auch andere Software-Hersteller Anbindungen an ein Microsoft-Domain-Netz finden konnten. Samba setzt das SMB- auf das TCP/IP-Protokoll auf. Entsprechend muss auf allen Clients das Protokoll TCP/IP installiert sein. Wir empfehlen die ausschließliche Verwendung von TCP/IP auf den Clients.

Hinweis

Migration nach Samba3

Wenn Sie von Samba 2.x nach Samba 3 migrieren möchten, so sind einige Besonderheiten zu beachten. Diesem Thema wurde in der Samba-HOWTO-Collection ein eigenes Kapitel gewidmet. Nach der Installation des Paketes `samba-doc` finden Sie das HOWTO unter `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf`.

Hinweis

NetBIOS

NetBIOS ist eine Softwareschnittstelle (API), die zur Rechnerkommunikation entworfen wurde. Dabei wird ein Namensdienst *name service* bereitgestellt, der zur gegenseitigen Identifikation der Rechner dient. Für die Namensvergabe gibt es keine zentrale Instanz, die Rechte vergeben oder überprüfen könnte. Jeder Rechner am Netz kann beliebig Namen für sich reservieren, sofern diese noch nicht vergeben sind. Die NetBIOS-Schnittstelle kann auf unterschiedlichen Netzarchitekturen implementiert werden. Eine Implementation erfolgt relativ „dicht“ an der Netzwerkhardware und nennt sich NetBEUI. NetBEUI wird häufig als NetBIOS bezeichnet. Netzwerkprotokolle, mit denen NetBIOS implementiert wurde, sind IPX (NetBIOS über TCP/IP) von Novell und TCP/IP.

Die NetBIOS-Namen, die auch bei der Implementation von NetBIOS mittels TCP/IP vergeben werden, haben nichts mit den in der Datei `/etc/hosts` oder per DNS vergebenen Namen zu tun. NetBIOS ist ein vollständig eigener Namensraum. Es empfiehlt sich jedoch zwecks vereinfachter Administration, zumindest für die Server NetBIOS-Namen zu vergeben, die ihrem DNS-Hostnamen entsprechen. Für einen Samba-Server ist das die Voreinstellung.

Clients

Alle gängigen Betriebssysteme wie Mac OS X, Windows und OS/2 unterstützen das SMB-Protokoll. Auf den Rechnern muss das TCP/IP Protokoll installiert sein. Für die verschiedenen UNIX Versionen stellt Samba einen Client zur Verfügung. Für Linux gibt es zudem ein Dateisystem-Kernel-Modul für SMB, dass das Einbinden von SMB-Ressourcen auf Linux-Systemebene gestattet.

SMB-Server stellen den Clients Plattenplatz in Form von Freigaben, so genannten „Shares“ zur Verfügung. Dabei umfasst ein Share ein Verzeichnis mit allen Unterverzeichnissen auf dem Server. Es wird unter einem eigenen Namen exportiert und kann von Clients unter diesem Namen angesprochen werden. Dabei kann der Sharename frei vergeben werden. Er muss nicht dem Namen des exportierten Verzeichnisses entsprechen. Ebenso wird einem exportierten Drucker ein Name zugeordnet, unter dem Clients darauf zugreifen können.

17.1.2 Installation und Konfiguration des Servers

Wenn Sie Samba als Server einsetzen möchten, installieren Sie das Paket `samba`. Manuell werden die für Samba erforderlichen Dienste mit `rcnmb start & & rcsmb start` gestartet und mit `rcsmb stop & & rcnmb stop` beendet.

Die zentrale Konfigurationsdatei von Samba ist `/etc/samba/smb.conf`. Die Datei kann man logisch in zwei Bereiche trennen. In der so genannten `[global]`-Section werden zentrale und übergreifende Einstellungen vorgenommen. Im zweiten Teilbereich, den `[share]`-Sections, werden die einzelnen Datei- und Drucker-Freigaben definiert. Mittels dieses Vorgehens können Details der Freigaben unterschiedlich oder in der `[global]`-Sektion übergreifend gesetzt werden. Letzteres trägt zur Übersichtlichkeit der Konfigurationsdatei bei.

global-Section anhand der Beispielkonfiguration

Die folgenden Parameter der `global`-Section sind den Gegebenheiten Ihres Netzwerkes anzupassen, damit Ihr Samba-Server im Windows-Netz von anderen Systemen per SMB erreichbar ist.

workgroup = TUX-NET Der Samba-Server wird mittels dieser Zeile einer Arbeitsgruppe zugeordnet. Zum Betrieb passen Sie TUX-NET an die bei Ihnen vorhandene Arbeitsgruppe an oder konfigurieren Ihren Clients auf den hier gewählten Wert. Ihr Samba-Server erscheint bei dieser Konfiguration mit seinem DNS-Namen in der gewählten Arbeitsgruppe, insoweit der Name noch nicht vergeben ist.

Sollte der Name bereits vergeben sein, kann er mit `netbios name = MEINNAME` abweichend vom DNS-Namen gesetzt werden. Details zu diesem Parameter sind per `man smb.conf` verfügbar.

os level = 2 Anhand dieses Parameters entscheidet Ihr Samba-Server, ob er versucht, LMB *Local Master Browser* für seine Arbeitsgruppe zu werden. Der im Beispiel genutzte Wert ist bewusst niedrig gewählt, damit ein vorhandenes Windows-Netz nicht durch einen falsch konfigurierten Samba-Server gestört wird. Details zu diesem wichtigen Thema finden Sie in den Dateien `BROWSING.txt` und `BROWSING-Config.txt` im Unterverzeichnis `textdocs/` der Paketdokumentation.

Wird nicht bereits ein SMB-Server — zum Beispiel Windows NT, 2000 Server — betrieben und soll der Samba-Server im lokalen Netz die Namen der verfügbaren Systeme vorhalten, so erhöhen Sie den `os level` auf einen höheren Wert (zum Beispiel 65), um die Wahl zum LMB zu gewinnen.

Bei der Änderung dieses Wertes sollten Sie besonders vorsichtig sein, da Sie den Betrieb eines vorhandenen Windows-Netzes stören können. Reden Sie mit Ihrem Administrator, testen Sie Änderungen zuerst in einem isolierten Netz oder zu unkritischen Zeiten.

wins support und wins server Wenn Sie den Samba-Server in ein vorhandenes Windows-Netz integrieren möchten, in dem bereits ein WINS-Server betrieben wird, benötigen Sie den Parameter `wins server`. Dieser Parameter muss auf die IP-Adresse Ihres WINS-servers gesetzt werden.

Wenn Ihre Windows-Systeme in getrennten Sub-Netzen betrieben werden, und sich gegenseitig sehen sollen, benötigen Sie einen WINS-Server. Um den Samba-Server zum WINS-Server zu machen, benötigen Sie die Option `wins support = Yes`. Achten Sie unbedingt darauf, dass Sie diesen Parameter ausschließlich bei einem Samba-Server aktivieren.

In Ihrer `smb.conf` dürfen nie beide Optionen, `wins server` und `wins support`, zusammen aktiviert werden.

Freigaben

In den folgenden Beispielen werden einerseits das CD-ROM-Laufwerk und andererseits die Verzeichnisse der Nutzer, `homes` für SMB-Clients freigegeben.

[cdrom] Um die versehentliche Freigabe einer CD-ROM zu verhindern, sind alle erforderlichen Zeilen dieser Freigabe mittels Kommentarzeichen – hier Semikolons – deaktiviert. Wollen Sie das CD-ROM-

Laufwerk per Samba freigeben, entfernen Sie bitte die Semikolons in der ersten Spalte.

Beispiel 17.1: CD-ROM-Freigabe

```
[cdrom]
;      comment = Linux CD-ROM
;      path = /media/cdrom
;      locking = No
```

[cdrom] **und** comment Der Eintrag [cdrom] ist der den SMB-Clients sichtbare Freigabename. Mittels comment kann den Clients eine aussagekräftigere Bezeichnung der Freigabe mitgeteilt werden.

path = /media/cdrom Mit path wird das Verzeichnis media/cdrom/ exportiert.

Diese Art der Freigabe ist aufgrund einer bewusst restriktiv gewählten Voreinstellung lediglich für die auf dem System vorhandenen Nutzer verfügbar. Soll die Freigabe für jedermann bereitgestellt werden, ermöglicht man dies mit der zusätzlichen Zeile `guest ok = Yes`. Aufgrund der sich daraus ergebenden Lesemöglichkeit für jedermann, sollte man mit dieser Einstellung sehr vorsichtig umgehen und sie allein auf ausgesuchte Freigaben anwenden. Für die Verwendung in der [global]-Section gilt besondere Vorsicht.

[homes] Eine besondere Stellung nimmt die so genannte [homes]-Freigabe ein. Hat der Benutzer auf dem Linux-File-Server einen gültigen Account und ein eigenes Home-Verzeichnis, so kann sich sein Client bei gültiger Nutzerkennung und Passwort mit diesem verbinden.

Beispiel 17.2: Freigabe homes

```
[homes]
      comment = Home Directories
      valid users = %S
      browseable = No
      read only = No
      create mask = 0640
      directory mask = 0750
```


[homes] Insoweit keine ausdrückliche Freigabe mit dem Freigabenamen des sich verbindenden Nutzers existiert, wird aufgrund der [homes]-Freigabe dynamisch eine Freigabe erzeugt. Dabei ist der Freigabename identisch mit dem Nutzernamen.

valid users = %S Das %S wird nach erfolgreichem Verbindungsaufbau durch den konkreten Freigabenamen ersetzt. Da dies bei der [homes]-Freigabe immer mit dem Nutzernamen identisch ist, werden die zulässigen Nutzer auf den Eigentümer des Nutzerverzeichnisses beschränkt. Dies ist eine Möglichkeit, um den Zugriff allein dem Eigentümer zu gestatten.

browseable = No Durch diese Einstellung ist die [homes]-Freigabe nicht in der Liste der Freigaben sichtbar.

read only = No Samba verbietet in der Voreinstellung den Schreibzugriff auf exportierte Freigaben, `read only = Yes`. Soll also ein Verzeichnis als schreibbar freigegeben werden, muss man den Wert `read only = No` setzen. Dies ist gleichbedeutend mit `writable = Yes`.

create mask = 0640 Nicht auf MS Windows NT basierende Systeme kennen das Konzept der Unix-Zugriffsrechte nicht. Daher können sie bei der Erstellung von Dateien auch nicht angeben, mit welchen Zugriffsrechten dies zu geschehen hat. Der Parameter `create mask` legt fest, mit welchen Zugriffsrechten Dateien angelegt werden. Dieses gilt nur für schreibbare Shares. Konkret wird hier dem Eigentümer das Lesen und Schreiben und Mitgliedern der primären Gruppe des Eigentümers das Lesen erlaubt. Bitte beachten Sie, dass `valid users = %S` selbst dann den lesenden Zugriff verhindert, wenn die Gruppe leseberechtigt ist. Entsprechend muss bei gewünschtem Lese- oder Schreibzugriff für die Gruppe die Zeile `valid users = %S` deaktiviert werden.

Security Level

Das SMB-Protokoll kommt aus der DOS-/Windows-Welt und berücksichtigt die Sicherheitsproblematik direkt. Jeder Zugang zu einem Share kann mit einem Passwort geschützt werden. SMB kennt drei verschiedene Möglichkeiten der Berechtigungsprüfung.

Share Level Security (security = share):

Bei der Share Level Security wird einem Share ein Passwort fest zugeordnet. Jeder, der dieses Passwort kennt, hat Zugriff auf das Share.

User Level Security (security = user):

Diese Variante führt das Konzept des Benutzers in SMB ein. Jeder Benutzer muss sich bei einem Server mit einem Passwort anmelden. Nach der Authentifizierung kann der Server dann, abhängig vom Benutzernamen, Zugang zu den einzelnen, exportierten Shares gewähren.

Server Level Security (security = server):

Samba behauptet gegenüber den Clients, im User Level Mode zu arbeiten. Allerdings übergibt es alle Passwortanfragen an einen anderen User Level Mode Server, der die Authentifizierung übernimmt. Diese Einstellung erwartet einen weiteren Parameter (`password server =`).

Die Unterscheidung zwischen Share, User und Server Level Security gilt für den gesamten Server. Es ist nicht möglich, einzelne Shares einer Server-Konfiguration per Share Level Security und andere per User Level Security zu exportieren. Jedoch können Sie auf einem System pro konfigurierter IP-Adresse einen eigenen Samba-Server betreiben.

Weitere Infos zu diesem Thema finden Sie in der Samba-HOWTO-Collection. Für mehrere Server auf einem System beachten Sie bitte die Parameter `interfaces` und `bind interfaces only`.

Hinweis

Für die einfache Administration des Samba-Servers gibt es noch das Programm `swat`. Es stellt ein einfaches Webinterface zur Verfügung, mit dem Sie bequem den Samba-Server konfigurieren können. Rufen Sie in einem Webbrowser `http://localhost:901` auf und loggen Sie sich als Benutzer `root` ein. Bitte beachten Sie, dass `swat` auch in den Dateien `/etc/xinetd.d/samba` und `/etc/services` aktiviert ist. Hierzu müssen Sie in `/etc/xinetd.d/samba` den Parameter `disable` auf `no` ändern. Weitere Informationen zu `swat` finden Sie in der Manualpage von `swat`.

Hinweis

17.1.3 Samba als Anmelde-Server

In Netzwerken, in denen sich überwiegend Windows-Clients befinden, ist es oft wünschenswert, dass sich die Benutzer nur mit gültigem Account und Passwort anmelden dürfen. Dies kann mit Hilfe eines Samba-Servers

realisiert werden. In einem Windows basierten Netzwerk übernimmt ein Windows-NT-Server diese Aufgabe, dieser ist als so genannter Primary Domain Controller (PDC) konfiguriert. Es müssen Einträge in die `[global]`-Section der `smb.conf` vorgenommen werden wie in Beispiel 17.3.

Beispiel 17.3: *Global-Section in smb.conf*

```
[global]
    workgroup = TUX-NET
    domain logons = Yes
    domain master = Yes
```

Werden verschlüsselte Passwörter zur Verifizierung genutzt - dies ist Standard mit gepflegten MS Windows 9x Versionen, MS Windows NT 4.0 ab service pack 3 und allen späteren Produkten -, muss der Samba Server damit umgehen können. Der Eintrag `encrypt passwords = yes` in der `[global]`-Section ermöglicht dies und ist bei Samba ab Version 3 default. Außerdem müssen die Benutzeraccounts bzw. die Passwörter in eine Windows konforme Verschlüsselungsform gebracht werden. Das geschieht mit dem Befehl `smbpasswd -a name`. Da nach dem Windows NT Domänenkonzept auch die Rechner selbst einen Domänen-Account benötigen, wird dieser mit den folgenden Befehlen angelegt:

Beispiel 17.4: *Anlegen eines Maschinenaccounts*

```
useradd rechnername\$
smbpasswd -a -m rechnername
```

Bei dem Befehl `useradd` wurde ein Dollarzeichen hinzugefügt. Der Befehl `smbpasswd` fügt dieses bei der Verwendung des Parameters `-m` selbst hinzu.

In der kommentierten Beispielskonfiguration `/usr/share/doc/packages/samba/examples/smb.conf` SuSE sind Einstellungen vorgesehen, die diese Arbeiten automatisieren.

Beispiel 17.5: Automatisiertes Anlegen eines Maschinenaccounts

```
add machine script = /usr/sbin/useradd -g machines \  
                  -c "NT Machine Account" -d \  
                  /dev/null -s /bin/false %m\%
```

Damit dieses Skript von Samba richtig ausgeführt werden kann, benötigen Sie noch einen Samba Benutzer mit Administrator Rechten. Fügen Sie hierzu die Gruppe ntadmin dem ausgewählten Benutzer hinzu. Danach können Sie alle Benutzer dieser Unix Gruppe zu den „Domain Admins“ mit folgendem Befehl hinzufügen:

```
net groupmap add ntgroup="Domain Admins" unixgroup=ntadmin
```

Mehr Informationen hierzu finden Sie in der Samba-HOWTO-Collection im Kapitel 12: `/usr/share/doc/packages/samba/Samba-HOWTO-Collection.pdf`.

17.1.4 Installation der Clients

Clients können den Samba-Server nur über TCP/IP erreichen. NetBEUI oder NetBIOS über IPX sind mit Samba nicht verwendbar.

Windows 9x/ME

Windows 9x/ME bringt die Unterstützung für TCP/IP bereits mit. Wie bei Windows for Workgroups wird sie jedoch in der Standardinstallation nicht mitinstalliert. Um TCP/IP nachzuinstallieren, wählt man im Netzwerk-Applet der Systemsteuerung 'Hinzufügen...' unter 'Protokolle' TCP/IP von Microsoft. Nach einem Neustart des Windows-Rechners können Sie den Samba-Server durch Doppelklick auf das Desktop-Symbol für die Netzwerkumgebung finden.

Hinweis

Um einen Drucker auf dem Samba-Server zu nutzen, sollte man den allgemeinen oder den Apple PostScript-Druckertreiber von der jeweiligen Windows-Version installieren; am besten verbindet man dann mit der Linux Drucker-Queue, die PostScript als Input Format akzeptiert.

Hinweis

17.1.5 Optimierung

Eine Möglichkeit der Optimierung bietet `socket options`. Die Voreinstellung in der mitgelieferten Beispielkonfiguration orientiert sich an einem lokalen Ethernet-Netzwerk. Weitere Details finden Sie in der Manualpage von `smb.conf` im Abschnitt `socket options` und der Manualpage von `socket(7)`. Weitere Informationen hierzu sind in der Samba-HOWTO-Collection im Kapitel `Samba performance tuning` enthalten.

Die Standardkonfiguration in `/etc/samba/smb.conf` versucht sinnvolle Werte vorzuschlagen und orientiert sich dabei an Voreinstellungen des Samba-Teams. Eine fertige Konfiguration ist jedoch insbesondere hinsichtlich der Netzwerkkonfiguration und des Arbeitsgruppennamens nicht möglich. In der kommentierten Beispielkonfiguration `examples/smb.conf` SuSE finden Sie zahlreiche weiterführenden Hinweise, die bei der Anpassung an lokale Gegebenheiten hilfreich sind.

Hinweis

Das Samba-Team liefert in der Samba-HOWTO-Collection einen Abschnitt zur Fehlersuche. In Part V ist außerdem eine Schritt-für-Schritt-Anleitung zur Überprüfung der Konfiguration enthalten.

Hinweis

17.2 Netatalk

Mit dem `netatalk` können Sie einen leistungsfähigen File- und Druckserver für Mac OS-Clients realisieren. Es ist möglich, von einem Macintosh aus auf Daten des Linux-Rechners zuzugreifen oder auf einem angeschlossenen Drucker zu drucken.

`Netatalk` ist eine Suite von Unix-Programmen, die auf dem im Kernel implementierten DDP (Datagram Delivery Protocol) aufsetzen und die AppleTalk-Protokoll-Familie (ADSP, ATP, ASP, RTMP, NBP, ZIP, AEP und PAP) implementieren.

AppleTalk ist im Prinzip ein Äquivalent zum wesentlich weiter verbreiteten TCP (Transmission Control Protocol). Viele auf TCP/IP aufsetzende Dienste, zum Beispiel zur Auflösung von Hostnamen und Zeitsynchronisation, finden ihre Entsprechung unter AppleTalk. Beispielsweise wird an Stelle von `ping` (ICMP ECHO_REQUEST, Internet Control Message Protocol) der Befehl `aecho` (AEP, AppleTalk Echo Protocol) verwendet.

Folgende drei Daemonen werden normalerweise auf dem Server gestartet:

- Der `atalkd` („AppleTalk-Netzwerk-Manager“), der dem Programm `ip` entspricht;
- `afpd` („AppleTalk Filing Protocol daemon“), der für Macintosh-Clients ein Interface zu Unix-Dateisystemen zur Verfügung stellt;
- `papd` („Printer Access Protocol daemon“), der Drucker im (AppleTalk-) Netz bereitstellt.

Sie können ohne weiteres – und in heterogenen Netzwerkumgebungen ist dies sehr nützlich – Verzeichnisse auf dem Server nicht nur über `Netatalk`, sondern gleichzeitig über Samba (für Windows-Clients, siehe voriges Kapitel) und über NFS (siehe 14.9 auf Seite 408), exportieren. Datensicherung und die Verwaltung der Nutzerrechte können zentral auf dem Linux-Server erfolgen.

Beachten Sie bitte bei der Verwendung von `Netatalk` folgende Beschränkungen:

- Wegen einer Einschränkung der Macintosh-Clients dürfen die Passwörter der Benutzer auf dem Server maximal 8 Zeichen lang sein.
- Auf Unix-Dateien mit Namen länger als 31 Zeichen können Macintosh-Clients nicht zugreifen.
- Dateinamen dürfen keine Doppelpunkte (:) enthalten, weil diese unter Mac OS als Separator in Pfadnamen dienen.

17.2.1 Konfiguration des Fileservers

In der Standardkonfiguration ist `Netatalk` als Fileserver für die auf dem Linux-System eingetragenen Benutzer schon voll funktionsfähig. Um die weitergehenden Features zu nutzen, müssen Sie einige Einstellungen in den Konfigurationsdateien vornehmen. Diese befinden sich im Verzeichnis `/etc/netatalk/`.

Alle Konfigurationsdateien sind reine Textdateien. Text, der hinter einer Raute # steht, wird ignoriert („Kommentare“), leere Zeilen ebenso. Über die Datei `/etc/netatalk/netatalk.conf` werden die verschiedenen Dienste (Drucker, Appletalk Broadcast, Appletalk via TCP/IP, Timeserver) aktiviert:

```
ATALKD_RUN=yes
PAPD_RUN=yes
AFPD_RUN=yes
TIMELORD_RUN=no
```

Netz konfigurieren – atalkd.conf

In `/etc/netatalk/atalkd.conf` legt man fest, über welche Interfaces die Dienste angeboten werden. Meist ist dies `eth0`, und es genügt, wenn hier wie in der Beispieldatei als einziger Wert `eth0` eingetragen ist. Tragen Sie weitere Interfaces ein, wenn Sie zum Beispiel mehrere Netzwerkkarten gleichzeitig verwenden. Wird der Server gestartet, sucht er im Netzwerk nach bereits vorhandenen Zonen und Servern und verändert die entsprechende Zeile, indem er die konfigurierten AppleTalk-Netzwerk-Adressen einträgt. Am Ende der Datei finden Sie dann eine Zeile entsprechend folgender:

```
eth0 -phase 2 -net 0-65534 -addr 65280.57
```

Sollten Sie komplexere Konfigurationen vornehmen wollen, finden Sie in der Konfigurationsdatei Beispiele. Dokumentation über weitere Optionen können Sie außerdem der Manual-Page zum `afpd` entnehmen.

Fileserver definieren – afpd.conf

In der Datei `afpd.conf` wird festgelegt, wie Ihr Fileserver auf Mac-OS-Rechnern in der 'Auswahl' erscheint. Wie die anderen Konfigurationsdateien enthält auch diese ausführliche Kommentare, die die vielfältigen Optionen erklären.

Ändern Sie hier nichts, wird einfach der Default-Server gestartet und in der 'Auswahl' mit dem Hostnamen angezeigt. Sie müssen also hier nicht unbedingt etwas eintragen, allerdings ist es auch möglich, Fileserver mit verschiedenen Namen und Optionen zu definieren, um zum Beispiel einen speziellen „Guest Server“ anzubieten, auf dem man als „Gast“ Dateien ablegen kann:

```
"Guest server" -uamlist uams_guest.so
```

Sie können auch einen Server definieren, der keinen Gastzugang erlaubt, sondern nur für Benutzer zugänglich ist, die auf dem Linux-System existieren:

```
"Font server" -uamlist uams_clrtxt.so,uams_dhx.so
```

Dieses Verhalten wird gesteuert durch die Option `uamlist` gefolgt von einer durch Kommata getrennten Liste der zu verwendenden Authentifizierungsmodule. Default ist, dass alle Verfahren aktiv sind.

Ein AppleShare-Server stellt seine Dienste standardmäßig nicht nur über AppleTalk, sondern auch („encapsulated“) über TCP/IP zur Verfügung. Der Default-Port ist 548. Für zusätzliche AppleShare-Server (auf dem gleichen Rechner) müssen Sie, wenn diese ebenfalls auch über TCP laufen sollen, dedizierte Ports zuweisen. Die Bereitstellung des Dienstes über TCP/IP ermöglicht den Zugriff auf den Server auch über nicht AppleTalk-Netze wie zum Beispiel das Internet.

Die Syntax wäre dann zum Beispiel:

```
"Font server" -uamlist uams_clrtxt.so,uams_dhx.so -port 12000
```

Der AppleShare-Server erscheint hier im Netz mit dem Namen `Font Server`, erlaubt keinen Zugriff für Gäste und ist auf den Port 12000 eingestellt. Damit ist er auch über TCP/IP-Router hinweg erreichbar.

Welche (auf dem Server liegenden) Verzeichnisse der jeweilige AppleShare-Server dann als Netz-*Volumes* bereitstellt, wird in der Datei `AppleVolumes.default` definiert (die weiter unten näher erläutert wird). Mit der `-defaultvol` Option können Sie für einen einzelnen AppleShare-Server auch eine andere Datei festlegen, in der abweichende Vorgaben gemacht werden, zum Beispiel (in einer Zeile):

```
"Guest server" -uamlist uams_guest.so -defaultvol \  
/etc/netatalk/AppleVolumes.guest
```

Weitere Optionen sind in der Datei `afpd.conf` selbst erklärt.

Verzeichnisse und Zugriffsrechte – AppleVolumes.default

Die Verzeichnisse, die exportiert werden sollen, werden in der Datei `AppleVolumes.default` ausgewählt. Die Zugriffsrechte werden dabei durch die unter Unix üblichen Benutzer- und Gruppen-Rechte festgelegt.

Hinweis

Hier hat sich die Syntax teilweise geändert. Bitte berücksichtigen Sie dies, wenn Sie von einer älteren Version updaten; zum Beispiel heißt es statt `access=` jetzt `allow=`: (ein charakteristisches Symptom wäre, wenn Sie auf den Mac-Clients unter AppleTalk statt der Laufwerksbezeichnung deren Optionen angezeigt bekommen.) Da bei einem Update die neuen Dateien mit der Endung `.rpmnew` angelegt werden, kann es sein, dass Ihre alten Einstellungen unter Umständen wegen der geänderten Syntax nicht mehr funktionieren.

Wir empfehlen Ihnen, ein Backup von Ihren Konfigurationsdateien zu machen, aus diesen Ihre alten Einstellungen in die neuen Dateien zu übernehmen und diese dann umzubenennen. So profitieren Sie auch von den aktuellen ausführlichen Kommentaren, die zur Erklärung der diversen Optionen in den Konfigurationsdateien enthalten sind.

Hinweis

Neben `AppleVolumes.default` können zusätzliche Dateien angelegt werden, zum Beispiel `AppleVolumes.guest`, die von bestimmten Servern benutzt werden (indem in der Datei `afpd.conf` die `-defaultvol-` Option benutzt wird – siehe voriger Abschnitt).

Die Syntax ist denkbar einfach:

```
/usr/local/psfonts "PostScript Fonts"
```

bedeutet, dass das in dem Rootverzeichnis liegende Linux-Verzeichnis `/usr/local/psfonts/` als AppleShare-Volume mit dem Namen „PostScript Fonts“ freigegeben wird.

Optionen werden, durch Leerzeichen getrennt, an die Zeile angehängt. Eine sehr nützliche Option ist die Zugriffsbeschränkung:

```
/usr/local/psfonts "PostScript Fonts" allow:User1,@gruppe0
```

was den Zugriff auf das Volume „PostScript Fonts“ auf den Benutzer `User1` und alle Mitglieder der Gruppe `gruppe0` beschränkt. Diese müssen natürlich dem Server bekannt sein. Entsprechend können Sie mit `deny:User2` auch explizit Nutzer ausschließen.

Bitte berücksichtigen Sie, dass diese Einschränkungen für den Zugriff über AppleTalk gelten und nichts mit den Rechten zu tun haben, die der User hat, wenn er sich auf dem Server selber einloggen kann.

Netatalk legt zur Abbildung der Mac-OS-typischen Ressource-Fork von Dateien im Linux-Dateisystem `.AppleDouble`-Verzeichnisse an. Mit der Option `noadouble` können Sie bestimmen, dass diese Verzeichnisse erst dann angelegt werden, wenn sie tatsächlich benötigt werden. Syntax:

```
/usr/local/guests "Guests" options:noadouble
```

Weitere Optionen und Möglichkeiten entnehmen Sie bitte den Erklärungen in der Datei selbst.

Übrigens: In dieser Konfigurationsdatei finden Sie ebenfalls eine kleine unschuldige Tilde (``~'`). Diese Tilde steht für das Homeverzeichnis eines jeden Benutzers auf dem Server. Dadurch kann jedem Benutzer automatisch sein Homeverzeichnis bereitgestellt werden, ohne dass jedes einzelne hier explizit angegeben werden müsste. Die installierte Beispieldatei enthält bereits eine Tilde, weshalb Netatalk standardmäßig die Homeverzeichnisse bereitstellt, wenn Sie an dieser Datei nichts ändern.

Der `afpd` sucht außerdem im Homeverzeichnis eines angemeldeten Benutzers nach einer Datei `AppleVolumes` oder `.AppleVolumes`. Einträge in dieser Datei ergänzen die Einträge in den Serverdateien `AppleVolumes.system` und `AppleVolumes.default`, um weitere individuelle `type/creator`-Zuordnungen zu ermöglichen und auf Dateisysteme zuzugreifen. Diese Einträge sind Ergänzungen und ermöglichen keine Zugriffe, die nicht von Serverseite für diesen Benutzer erlaubt sind.

Die Datei `netatalk.pamd` dient der Authentifizierung über PAM (Pluggable Authentication Modules), was in unserem Rahmen hier ohne Bedeutung ist.

Dateizuordnungen – AppleVolumes.system

In der Datei `AppleVolumes.system` legen Sie fest, welche (Mac-OS-typischen) `Type`- und `Creator`-Zuordnungen zu bestimmten Dateiendungen erfolgen soll. Eine ganze Reihe von Standardwerten sind schon vorgegeben. Wenn eine Datei mit einem generischen weißen Icon angezeigt wird, ist in diesem Fall noch kein Eintrag vorhanden. Sollten Sie Probleme haben, eine Textdatei eines anderen Systems unter Mac OS korrekt öffnen zu können, bzw. das umgekehrte Problem, kontrollieren Sie dort die Einträge.

17.2.2 Konfiguration des Druckservers

Über die Datei `papd.conf` konfigurierbar wird ein Laserwriter-Dienst zur Verfügung gestellt. Der Drucker `lpd` muss lokal schon funktionieren (sie-

he Kapitel 5 auf Seite 99). Wenn Sie mit dem Kommando `lpr datei.txt` lokal drucken können, ist der erste Schritt erfolgreich getan.

Sie müssen in `lpd.conf` nichts eingeben, wenn unter Linux ein lokaler Drucker eingerichtet ist, da ohne weitere Angaben Druckaufträge einfach an den Druck-Daemon `lpd` weitergegeben werden. Der Drucker meldet sich im AppleTalk-Netz als Laserwriter. Sie können aber auch bestimmte Drucker wie folgt eintragen:

```
Drucker_Empfang:pr=lp:pd=/etc/netatalk/kyocera.ppd
```

Dies lässt den Drucker mit dem Namen `Drucker_Empfang` in der Auswahl erscheinen. Die entsprechende Druckerbeschreibungsdatei gibt es gewöhnlich beim Hersteller. Ansonsten nehmen Sie einfach die Datei `Laserwriter` aus dem Ordner `Systemerweiterungen/`; allerdings können Sie dann meist nicht alle Features benutzen.

17.2.3 Starten des Servers

Der Server wird per `Init-Skript` beim Systemstart gestartet oder per Hand mit: `rcatalk start`. Das `Init-Skript` befindet sich in `/etc/init.d/atalc`. Den Start erledigt das Startskript im Hintergrund; es dauert ca. eine Minute, bis die AppleTalk-Interfaces konfiguriert und erreichbar sind. Sie können mit einer Statusabfrage sehen, ob es soweit ist (erkennbar daran, dass dreimal OK ausgegeben wird):

```
rcatalk status
```

```
Checking for service atalk:OKOKOK
```

Gehen Sie nun an einen Mac, der unter Mac OS läuft. Kontrollieren Sie, dass AppleTalk aktiviert ist, wählen Sie 'Filesharing', doppelklicken Sie 'Appleshare'; in dem Fenster sollten Sie nun den Namen Ihres Servers sehen. Doppelklicken Sie ihn und melden sie sich an. Wählen Sie das Laufwerk und – voilà – hier ist Ihr Netzlaufwerk unter Mac OS.

Mit Servern, die nur über TCP und nicht über DDP laufen, können Sie sich verbinden, indem Sie in der 'Auswahl' auf 'Server IP-Adresse' klicken und die entsprechende IP-Adresse, gegebenenfalls gefolgt von einem Doppelpunkt und der Portnummer, eingeben.

17.2.4 Weiterführende Informationen

Um alle Möglichkeiten, die das `netatalk` bietet, voll auszuschöpfen, empfiehlt es sich, in den entsprechenden Manual-Pages zu stöbern. Diese finden Sie mit dem Befehl: `rpm -qd netatalk` Noch ein Hinweis: Die Datei `/etc/netatalk/netatalk.conf` wird in unserer Version von `netatalk` nicht verwendet, Sie können sie einfach ignorieren. Hilfreiche URLs:

- <http://netatalk.sourceforge.net/>
- <http://www.umich.edu/~rsug/netatalk/>
- <http://www.anders.com/projects/netatalk/>

17.3 NetWare-Emulation mit MARSNWE

Der NetWare-Emulator MARSNWE kann einen Novell NetWare 2.2 bzw. 3.11 Server für Datei- und Druckdienste relativ leicht ersetzen und kann dabei auch als IPX-Router verwendet werden. Die Funktionalität neuerer NetWare-Versionen, wie z.B. *NDS NetWare Directory Services* kann er allerdings nicht bieten. Arbeitsstationen, die mit DOS oder Windows laufen und bereits für den Zugriff auf einen NetWare 2.2/3.11/3.12-Server konfiguriert sind, können mit minimalen Änderungen den Linux-Server mit dem NetWare-Emulator MARSNWE als Server nutzen. Die Administration kann man unter Linux erledigen.

17.3.1 NetWare-Emulator MARSNWE starten

Der MARSNWE auf SUSE LINUX kann nach der Installation gestartet werden, da er bereits soweit vorkonfiguriert ist, dass man ihn sofort testen kann. Die erforderliche IPX-Unterstützung seitens des Kernels ist als ladbares Kernelmodul vorhanden und wird vom Startskript bei Bedarf automatisch geladen. Das Aufsetzen des IPX-Interfaces wird vom MARSNWE automatisch durchgeführt. Netznummer und zu verwendendes Protokoll werden dabei der ausführlich kommentierten Konfigurationsdatei `/etc/nwserve.conf` entnommen. Gestartet wird der MARSNWE mit dem Kommando `rcnwe start`. Meldet er dabei am rechten Bildschirmrand in grün `done`, wurde er erfolgreich gestartet.

Ob der NetWare-Emulator läuft, überprüft man mit `rcnwe status`, und beendet wird er mit `rcnwe stop`.

17.3.2 Die Konfigurationsdatei `/etc/nwserv.conf`

Die Konfigurationsoptionen sind zu durchnummerierten Sections zusammengefasst. Jede Konfigurationszeile beginnt dabei immer mit der Nummer der jeweiligen Section. Interessant sind lediglich die Sections 1 bis 22, wobei aber nicht alle Nummern verwendet werden. Im Normalfall kommt man mit folgenden Sections für die Konfiguration aus:

- 1 NetWare Volumes
- 2 Servername
- 4 IPX-Netzwerk
- 13 Benutzernamen
- 21 Drucker

Nach Änderungen an der Konfiguration, muss MARSNWE mit dem Befehl `rcnwe restart` neu gestartet werden.

Die Konfigurations-Optionen im Detail:

Volumes (Section 1):

```
1   SYS           /usr/local/nwe/SYS/      kt      711 600
```

Hier werden die zu exportierenden Volumes definiert. Jede Zeile beginnt mit der Nummer der Section (hier 1), danach folgt der Volume-Name und dann der Pfad des Verzeichnisses auf dem Server. Zusätzlich können noch diverse Optionen angegeben werden, die durch einzelne Buchstaben dargestellt sind, sowie jeweils eine Umask für das Erzeugen von Verzeichnissen und eine für Dateien. Wird keine Umask angegeben, wird der Standardwert aus Section 9 verwendet. Das Volume für SYS ist bereits eingetragen. Um Probleme mit Groß- und Kleinschreibung bei Dateinamen zu vermeiden, empfiehlt sich die Verwendung der Option `k`, denn dann werden alle Dateinamen in Kleinschreibung konvertiert.

Servername (Section 2):

```
2   MARS
```

Diese Angabe ist optional, standardmäßig wird der Hostname verwendet.

Interne Netznummer (Section 3):

```
3      auto
```

Die interne Netznummer wird aus der MAC-Adresse der Netzwerkkarte generiert, wenn hier `auto` angegeben ist. Normalerweise behält man diese Einstellung bei.

IPX-Konfiguration (Section 4):

```
4      0x0      *      AUTO      1
4      0x22     eth0     ethernet_ii  1
```

Hier gibt man die NetWare-Netznummer an und auf welche Netzwerk-Schnittstelle es mit welchem Protokoll gebunden werden soll. Das erste Beispiel setzt alles automatisch auf, während das zweite die Netznummer `0x22` auf die Netzwerkkarte `eth0` mit dem Frametyp `Ethernet-II` bindet. Hat man mehrere Netzwerkkarten und trägt diese alle mit unterschiedlichen Netznummern ein, so wird IPX dazwischen geroutet.

Create Mode (Section 9):

```
9      0751      0640
```

Gibt die Standardrechte an, mit denen Verzeichnisse und Dateien angelegt werden.

GID und UID mit minimalen Rechten (Section 10, 11):

```
10     65534
11     65534
```

Gruppen- und Benutzer-ID für nicht angemeldete Benutzer. Hier `nogroup` und `nobody`.

Supervisor Login (Section 12):

```
12    SUPERVISOR      root
```

Der Supervisor wird auf den Benutzer `root` abgebildet.

Benutzer Logins (Section 13):

```
13    LINUX           linux
```

Die Zuordnung der NetWare-Benutzer zu den Linux-Usern wird hier festgelegt. Optional kann ein festes Passwort mit eingetragen werden.

Automatische Benutzer-Abbildung (Section 15):

```
15    0              top-secret
```

Gibt man hier statt der 0 eine 1 an, werden die Linux-Logins automatisch als NetWare-Logins zur Verfügung gestellt, das Passwort ist in diesem Fall „top-secret“.

Drucker-Queues (Section 21):

```
21    LP             -      lpr -
```

Der erste Parameter `LP` ist der Name des NetWare-Druckers, als zweites kann man den Namen des Spool-Verzeichnisses angeben und als drittes das Druckkommando.

Print-Server (Section 22):

```
22    PS_NWE  LP_PS  1
```

Drucker die über `pserver` aus dem `ncpfs` angesprochen werden, können hier definiert werden.

17.3.3 Zugriff auf NetWare-Server und deren Administration

Das `nnpfs` ist eine Sammlung kleiner Programme, mit denen man einen NetWare 2.2/3.11 Server von Linux aus administrieren, NetWare-Volumes mounten oder Drucker verwalten kann. Will man auf neuere NetWare-Server ab Version 4 zugreifen, muss auf diesen die Bindery-Emulation und IPX aktiviert sein.

Folgende Programme stehen dafür zur Verfügung, deren Funktion man den entsprechenden Manual-Pages entnehmen kann:

<code>nwmmsg</code>	<code>ncopy</code>	<code>nnpmount</code>	<code>nnpumount</code>
<code>nprint</code>	<code>nsend</code>	<code>nwauth</code>	<code>nwbocreate</code>
<code>nwbols</code>	<code>nwboprops</code>	<code>nwborm</code>	<code>nwbppadd</code>
<code>nwbppcreate</code>	<code>nwbpprm</code>	<code>nwbppset</code>	<code>nwbppvalues</code>
<code>nwdir</code>	<code>nwdppvalues</code>	<code>nwfscrtl</code>	<code>nwfinfo</code>
<code>nwfstime</code>	<code>nwgrant</code>	<code>nwpasswd</code>	<code>nwpurge</code>
<code>nwrevoke</code>	<code>nwrights</code>	<code>nwsfind</code>	<code>nwtrustee</code>
<code>nwtrustee2</code>	<code>nwuserlist</code>	<code>nwvolinfo</code>	<code>pqlist</code>
<code>pqrm</code>	<code>pqstat</code>	<code>pserver</code>	<code>slist</code>

Wichtig ist zum Beispiel `nnpmount`, mit dem man Volumes von einem NetWare-Server unter Linux einhängen (`mount`) kann und `nnpumount`, um es wieder auszuhängen (`umount`).

Außerdem enthält das `nnpfs` Tools zur Konfiguration des IPX-Protokolls und IPX-Routing:

```
ipx_cmd
ipx_configure
ipx_interface
ipx_internal_net
ipx_route
```

Mit `ipx_configure` oder `ipx_interface` kann man die IPX-Konfiguration der Netzwerkkarte vornehmen. Hat man MARSNWE laufen, macht dieser das aber bereits automatisch.

17.3.4 IPX-Router mit ipxrip

Ein weiteres Paket, um Linux in einen IPX-Router zu verwandeln ist `ipxrip`. In der Regel wird man es aber nicht benötigen, da man mit MAR-SNWE oder den Tools aus `ncpfs` ebenfalls einen IPX-Router konfigurieren kann.

Internet

Das Internet hat sich als Kommunikationsplattform weltweit durchgesetzt. Linux als Netzwerkbetriebssystem kann vielfältige Aufgaben sowohl als Client als auch als Server in diesem Netz wahrnehmen. In diesem Kapitel sollen einige interessante Themen hierzu beschrieben werden: der Einwahlhelfer `smpppd` (SUSE Meta PPP-Daemon), die manuelle Konfiguration eines ADSL-Zuganges, falls es bei der Einrichtung mit YaST Probleme geben sollte, und die Konfiguration des Proxies Squid.

18.1 Der <code>smpppd</code> als Einwahlhelfer	496
18.2 Konfiguration eines ADSL / T-DSL Anschlusses . .	498
18.3 Proxy-Server: Squid	500

18.1 Der smpppd als Einwahlhelfer

18.1.1 Programmkomponenten zur Einwahl ins Internet

Die meisten Heimanwender besitzen keine feste Anbindung an das Internet, sondern wählen sich bei Bedarf ein. Die Kontrolle über diese Verbindung hat dabei je nach Einwahlart (ISDN oder DSL) der `ipppd` oder der `pppd`. Im Prinzip reicht es, diese Programme korrekt zu starten, um online zu sein.

Sofern man über eine Flatrate verfügt, die bei der Einwahl keine zusätzlichen Kosten verursacht, reicht es tatsächlich aus, wenn man den Daemon entsprechend startet. Oftmals wünscht man sich jedoch, die Einwahl besser kontrollieren zu können, sei es über ein KDE-Applet oder auch über ein Kommandozeileninterface. Hinzu kommt, dass das Internet-Gateway oft nicht der eigentliche Arbeitsrechner ist, so dass man die Einwahl in einem per Netz erreichbaren Rechner steuern möchte.

An dieser Stelle kommt der `smpppd` (SUSE Meta PPP-Daemon) ins Spiel. Er stellt Hilfsprogrammen eine einheitliche Schnittstelle zur Verfügung, die in zwei Richtungen funktioniert. Zum einen programmiert er den jeweils nötigen `pppd` oder `ipppd`, und steuert dessen Einwahlverhalten. Zum anderen stellt er den Benutzerprogrammen verschiedene Provider zur Verfügung, und übermittelt Informationen über den aktuellen Zustand der Verbindung. Da der `smpppd` auch über das Netz steuerbar ist, eignet er sich gut, die Einwahl ins Internet von einer Workstation im privaten Subnetz aus zu steuern.

18.1.2 Die Konfiguration des smpppd

Die Konfiguration der Verbindungen, die der `smpppd` zur Verfügung stellt, wird automatisch durch YaST vorgenommen. Die eigentlichen Einwahlprogramme `kinternet` und `cinternet` werden ebenfalls vorkonfiguriert. Handarbeit ist dann gefragt, wenn Sie weitere Features des `smpppd`, etwa eine remote Bedienung, einrichten möchten.

Die Konfigurationsdatei des `smpppd` liegt unter `/etc/smpppd.conf`. Sie ist so eingestellt, dass standardmäßig keine remote Bedienung möglich ist. Die interessantesten Optionen dieser Konfigurationsdatei sind:

open-inet-socket = `<yes|no>` Wenn eine Steuerung des `smpppd` über das Netzwerk gewünscht ist, muss diese Option auf `yes` gesetzt werden. Der Port, auf dem der `smpppd` dann hört, ist 3185. Wenn

dieser Parameter auf `yes` gesetzt ist, sollten Sie auch die Parameter `bind-address`, `host-range` und `password` sinnvoll setzen.

bind-address = `<ip>` Wenn ein Rechner mehrere IP-Adressen hat, kann damit festgelegt werden, über welche IP-Adresse der `smpppd` Verbindungen akzeptiert.

host-range = `<min ip> <max ip>`

Der Parameter `host-range` kann verwendet werden, um einen Netzbereich zu definieren. Den Rechnern, deren IP-Adressen in diesem Bereich liegen, wird der Zugang zum `smpppd` erlaubt. Anders ausgedrückt, es werden alle Rechner abgewiesen, die nicht in diesem Bereich liegen.

password = `<password>` Mit der Vergabe eines Passworts kann eine Einschränkung der Clients auf berechnete Rechner geschehen. Da dies ein Klartextpassword ist, sollte man die Sicherheit, die es bietet nicht überbewerten. Wenn kein Passwort vergeben wird, dann sind alle Clients berechtigt, auf den `smpppd` zuzugreifen.

Weitere Informationen zum `smpppd` finden Sie in den Manualpages `man smpppd` und `man smpppd.conf`.

18.1.3 kinternet und cinternet im Remote-Einsatz

Die Programme `kinternet` und `cinternet` können sowohl lokal verwendet werden als auch einen entfernten `smpppd` steuern. `cinternet` ist hierbei auf der Kommandozeile die Entsprechung zum grafischen `kinternet`. Wenn Sie diese Utilities zum Einsatz mit einem remote `smpppd` vorbereiten möchten, müssen Sie die Konfigurationsdatei `/etc/smpppd-c.conf` manuell oder mit Hilfe von `kinternet` editieren. Diese Datei kennt nur drei Optionen:

server = `<server>` An dieser Stelle können Sie den Rechner spezifizieren, auf dem der `smpppd` läuft. Wenn dies gleichzeitig das default gateway des Rechners ist, reicht es, `gateway-fallback` auf `yes` zu setzen.

gateway-fallback = `<yes|no>` Wenn weder ein Server spezifiziert wurde noch einer lokal läuft, sprechen Sie einen `smpppd` auf dem default gateway an. Diese Option ist standardmäßig aktiviert.

password = <password> Setzen Sie an dieser Stelle das Passwort ein, das auch für den `smpppd` ausgewählt wurde.

Sofern der `smpppd` läuft, können Sie jetzt versuchen, auf den `smpppd` zuzugreifen. Dazu bietet sich der Befehl `cinternet --verbose --interface-list` an. Sollten Sie an dieser Stelle noch Schwierigkeiten haben, dann lesen Sie bitte die Manualpages `man smpppd-c.conf` und `man cinetwork`.

18.2 Konfiguration eines ADSL / T-DSL Anschlusses

18.2.1 Standardkonfiguration

Momentan werden von SuSE Linux DSL-Zugänge unterstützt, die mit dem Point-to-Point-over-Ethernet-Protokoll (PPPoE) arbeiten. Dieses Protokoll wird von allen großen Anbietern benutzt. Sollten Sie sich nicht sicher sein, welches Protokoll Ihr Provider verwendet, gibt dieser sicherlich gerne Auskunft.

1. Die Pakete `ppp` und `smpppd` müssen installiert werden. Verwenden Sie dazu am besten YaST.
2. Konfigurieren Sie Ihre Netzwerkkarte mit YaST. Verwenden Sie nicht `dhcp`, sondern vergeben Sie eine statische IP Adresse, zum Beispiel `192.168.2.22`.
3. Die Parameter, die Sie mit dem YaST DSL-Modul bearbeiten, werden in der Datei `/etc/sysconfig/network/providers/provider0/` abgespeichert. Zusätzlich gibt es noch Konfigurationsdateien für den `smpppd` (SuSE Meta-PPP-Daemon) und seine Frontends `kinetwork` und `cinternet`. Bitte beachten Sie dazu die Manualpage `man smpppd`.
4. Starten Sie das Netzwerk ggf. mit dem Befehl `rcnetwork start` und danach den `smpppd` mit dem Befehl `rcsmpppd start`.
5. Mit den Befehlen `cinternet --start` und `cinternet --stop` können Sie auf einem System ohne graphischer Oberfläche eine Verbindung herstellen bzw. abbrechen. Auf einer graphischen Benutzeroberfläche können Sie dazu `kinetwork` benutzen. Dieses Programm

wird unter KDE automatisch gestartet, falls Sie DSL mit YaST eingerichtet haben. Klicken Sie auf das Zahnrad-Icon in der Buttonleiste. Wählen Sie 'Kommunikation/Internet' (→) 'Internet Tools' (→) 'kinternet'. Nun erscheint in der Buttonleiste das Steckersymbol. Ein Klick darauf startet die Verbindung und ein zweiter Klick beendet sie wieder.

18.2.2 DSL Verbindung per Dial-on-Demand

Dial-on-Demand bedeutet, dass die Verbindung automatisch aufgebaut wird, sobald ein User auf das Internet zugreift, zum Beispiel indem er eine Webseite mit einem Browser anwählt oder E-Mails verschickt. Nach einer bestimmten Zeit (Idlezeit), in der keine Daten gesendet oder empfangen werden, wird die Verbindung wieder getrennt. Da die Einwahl mit PPPoE, dem Protokoll für ADSL, sehr schnell geht, entsteht mitunter der Eindruck, als hätte man eine Standleitung in das Internet.

Dies ist aber nur sinnvoll, wenn Sie eine Flatrate besitzen. Wird Ihr Zugang zeitabhängig abgerechnet, müssen Sie darauf achten, dass kein periodischer Prozess, zum Beispiel ein cronjob, immer wieder eine Verbindung aufbaut. Das könnte Ihre Kosten in die Höhe treiben.

Obwohl mit einer DSL-Flatrate auch eine permanente Einwahl möglich wäre, sprechen doch einige Punkte für eine Verbindung, die nur kurz und nach Bedarf besteht:

- Die meisten Provider trennen die Verbindung nach einer gewissen Zeit.
- Eine permanente Verbindung kann als Ressourcenverschwendung betrachtet werden (zum Beispiel IP-Adressen).
- Vor allem ist es ein enormes Sicherheitsrisiko, permanent online zu sein, da ein Angreifer das System auf Schwachstellen absuchen kann. Ein System, das nur bei Bedarf im Internet erreichbar ist und immer wieder eine andere IP-Adresse hat, ist viel schwieriger zu attackieren.

Dial-on-Demand können Sie mit YaST aktivieren (siehe auch das Benutzer-Handbuch) oder Sie richten es manuell ein. Setzen Sie in der Datei `/etc/sysconfig/network/providers/provider0/` den Parameter `DEMAND=` auf `"yes"` und definieren Sie eine Idlezeit mit der Variable: `IDLETIME="60"`. Damit wird eine unbenutzte Verbindung nach 60 Sekunden beendet.

Zur Einrichtung eines DSL-Gateways für private Netzwerke empfehlen wir folgenden Artikel in unserer Supportdatenbank: <http://portal.suse.de/sdb/de/2002/07/masq80.html>

18.3 Proxy-Server: Squid

Squid ist ein weit verbreiteter Proxy-Cache für Linux/UNIX-Plattformen. Wir werden beschreiben, wie er zu konfigurieren ist, welche Systemanforderungen bestehen, wie das eigene System konfiguriert sein muss, um transparentes Proxying durchzuführen, wie man Statistiken über den Nutzen des Cache mithilfe von Programmen wie Calamaris und cachemgr erhält oder wie man Web-Inhalte mit squidGuard filtert.

18.3.1 Was ist ein Proxy-Cache?

Squid fungiert als Proxy-Cache. Es verhält sich wie ein Makler, der Anfragen von Clients erhält (in diesem Fall Web-Browser) und an den zuständigen Server-Provider weiterleitet. Wenn die angeforderten Objekte beim Vermittler ankommen, behält er eine Kopie davon in einem Festplatten-Cache.

Vorteilhaft ist das, wenn mehrere Clients dasselbe Objekt anfordern: Sie können nun direkt aus dem Festplatten-Cache bedient werden, also wesentlich schneller als aus dem Internet. Dies spart gleichzeitig Netzwerk Transfervolumen.

Hinweis

Squid bietet ein großes Spektrum an Features, zum Beispiel die Festlegung von Hierarchien für die Proxy-Server zum Verteilen der Systemlast, Aufstellen fester Zugriffsregeln an alle Clients, die auf den Proxy zugreifen wollen, Erteilen oder Verweigern von Zugriffsrechten auf bestimmte Webseiten mit Hilfe anderer Applikationen oder die Ausgabe von Statistiken der meistbesuchten Webseiten, wie zum Beispiel das Surfverhalten der Benutzer.

Hinweis

Squid ist kein generischer Proxy. Normalerweise vermittelt er nur zwischen HTTP-Verbindungen. Außerdem unterstützt er die Protokolle FTP, Gopher, SSL und WAIS, jedoch keine anderen Internet-Protokolle wie

Real Audio, News oder Videokonferenzen. Squid greift auf das UDP-Protokoll nur zur Unterstützung der Kommunikation zwischen verschiedenen Caches zurück. Aus diesem Grund werden auch andere Multimedia-Programme nicht unterstützt.

18.3.2 Informationen zu Proxy-Cache

Squid und Sicherheit

Man kann Squid zusammen mit einer Firewall verwenden, um interne Netzwerke durch den Einsatz eines Proxy-Cache nach außen zu schützen. Die Firewall verweigert mit Ausnahme von Squid allen Clients den Verbindungsaufbau zu externen Diensten. Alle WWW-Verbindungen müssen dann durch den Proxy aufgebaut werden.

Im Falle einer Firewall-Konfiguration mit einem DMZ würde man dort den Proxy einsetzen. In diesem Fall ist es wichtig, dass alle Rechner in der DMZ ihre Protokolldateien an Rechner innerhalb des gesicherten Netzwerks senden.

Ein Möglichkeit der Implementierung eines so genannten „transparenten“ Proxy wird in Abschnitt 18.3.6 auf Seite 512 behandelt.

Mehrere Caches

Man kann mehrere Proxies so konfigurieren, dass Objekte zwischen ihnen ausgetauscht werden können, um die Systemlast zu reduzieren und die Wahrscheinlichkeit zu steigern, ein bereits im lokalen Netzwerk vorhandenes Objekt zu finden. Möglich sind auch Cache-Hierarchien, so dass ein Cache in der Lage ist, Objektanfragen an Caches der gleichen Hierarchie weiterzuleiten oder einen übergeordneten Cache zu veranlassen, die Objekte von einem anderen Cache im lokalen Netzwerk oder direkt aus der Quelle herunterzuladen.

Die Wahl der richtigen Topologie für die Cache-Hierarchie ist sehr wichtig, da Netzwerkverkehr insgesamt nicht erhöht werden soll. In einem großen Netzwerk bietet es sich an, für jedes Subnetz einen Proxy-Server zu konfigurieren und diesen dann mit einem übergeordneten Proxy zu verbinden, der wiederum mit dem Proxy-Cache vom ISP verbunden wird.

Die gesamte Kommunikation wird vom ICP *Internet Cache Protocol* gesteuert, das auf dem UDP-Protokoll aufgesetzt ist. Der Datenaustausch zwischen Caches geschieht mittels HTTP *Hyper Text Transmission Protocol* basierend auf TCP.

Um den besten Server für die gewünschten Objekte zu finden, schickt ein Cache an alle Proxies der gleichen Hierarchie eine ICP-Anfrage. Die Proxies werden mittels ICP-Antworten mit dem Code „HIT“ auf die Anfragen reagieren, falls das Objekt gefunden wurde oder, falls nicht, mit dem Code „MISS“. Im Falle mehrerer HIT-Antworten wird der Proxy-Server einen Server für das Herunterladen bestimmen. Diese Entscheidung wird unter anderem dadurch bestimmt, welcher Cache die schnellste Antwort sendet oder welcher näher ist. Bei einer nicht zufrieden stellenden Antwort wird die Anfrage an den übergeordneten Cache geschickt.

Hinweis

Zur Vermeidung einer mehrfachen Speicherung von Objekten in verschiedenen Caches des Netzwerks werden ebenfalls ICP-Protokolle verwendet, wie zum Beispiel *CARP Cache Array Routing Protocol* oder *HTCP Hyper-Text Cache Protocol*. Je mehr Objekte sich im Netzwerk befinden, desto leichter wird es, das Gesuchte zu finden.

Hinweis

Zwischenspeichern von Internetobjekten

Nicht alle im Netzwerk verfügbaren Objekte sind statisch. Es existieren viele dynamisch generierte CGI-Seiten, Zugriffszähler oder verschlüsselte SSL-Dokumente für eine höhere Sicherheit. Aus diesem Grund werden solche Objekte nicht im Cache gehalten: Bei jedem neuen Zugriff hat sich das Objekt bereits wieder verändert.

Für alle anderen im Cache befindlichen Objekte stellt sich jedoch die Frage, wie lange sie dort bleiben sollen. Für diese Entscheidung werden alle Objekte im Cache verschiedenen Stadien zugeordnet.

Durch Header wie `Last modified` („zuletzt geändert“) oder `Expires` („läuft ab“) und dem entsprechenden Datum informieren sich Web- und Proxy-Server über den Status eines Objekts. Es werden auch andere Header verwendet, die zum Beispiel anzeigen, dass ein Objekt nicht zwischengespeichert werden muss.

Objekte im Cache werden normalerweise aufgrund fehlenden Speichers ersetzt, und zwar durch Algorithmen wie LRU (engl. *Last Recently Used*), die zum Ersetzen von Cache-Objekten entwickelt wurden. Das Prinzip besteht im Wesentlichen darin, zuerst die am seltensten gewünschten Objekte zu ersetzen.

18.3.3 Systemanforderungen

Zuerst sollte die maximale Systemlast bestimmt werden. Es ist wichtig, den Systemspitzen besondere Aufmerksamkeit zu schenken, da diese mehr als viermal so hoch wie der Tagesdurchschnitt sein können. Im Zweifelsfall ist es besser, die Systemanforderungen zu überschätzen, da ein am Limit arbeitender Squid zu einem ernsthaften Qualitätsverlust des Dienstes führen kann.

Geordnet nach Wichtigkeit werden in den folgenden Abschnitten die verschiedenen Systemfaktoren aufgezeigt.

Festplatte

Für das Zwischenspeichern spielt Geschwindigkeit eine hohe Rolle. Man sollte sich also um diesen Faktor besonders kümmern. Bei Festplatten ist dieser Parameter als „Zugriffszeit“ in Millisekunden beschrieben. Als Faustregel gilt: Je niedriger dieser Wert, desto besser. Für eine hohe Geschwindigkeit empfiehlt es sich, schnelle Festplatten zu wählen.

Da Squid zumeist kleinere Datenblöcke von der Festplatte zu liest oder abspeichert, ist die Zugriffszeit einer Festplatte wichtiger als der Durchsatz. Gerade hierbei rentieren sich Festplatten mit hohen Drehzahlen, die eine schnelle Positionierung des Lesekopfes ermöglichen. Schnelle SCSI Festplatten erreichen heute Zugriffszeiten unter 4 Millisekunden.

Eine Möglichkeit, die Geschwindigkeit zu erhöhen, ist gleichzeitiger Einsatz mehrerer Festplatten oder striping Raid Arrays.

Größe des Festplatten-Cache

In einem kleinen Cache ist die Wahrscheinlichkeit eines HIT (das gewünschte Objekt befindet sich bereits dort) sehr gering, da der Cache schnell gefüllt sein wird. In diesem Fall werden die selten gewünschten Objekte durch neue ersetzt. Steht jedoch zum Beispiel 1 GB für den Cache zur Verfügung und die Benutzer benötigen nur 10 MB pro Tag zum Surfen, dann dauert es mehr als hundert Tage, bis der Cache voll ist.

Am leichtesten lässt sich die Größe des Cache durch die maximale Übertragungsrate der Verbindung bestimmen. Mit einer Verbindung von 1 Mbit/s wird die maximale Übertragungsrate bei 125 KB/s liegen. Landet der gesamte Datenverkehr im Cache, kommen innerhalb einer Stunde 450 MB zusammen. Wenn man nun annimmt, dass der gesamte Datenverkehr lediglich während acht Arbeitsstunden erzeugt wird, erreicht man innerhalb eines Tages 3,6 GB. Da die Verbindung normalerweise nicht bis zur Kapazitätsgrenze ausgeschöpft wird, kann man davon

ausgehen, dass die gesamte Datenmenge, die der Cache bearbeitet, bei ungefähr 2 GB liegt. In diesem Beispiel werden demnach 2 GB Speicher für Squid benötigt, um die Daten aller aufgerufenen Seiten *eines* Tages im Cache zu halten.

RAM

Der von Squid benötigte Speicher ist abhängig von der Anzahl der im Cache zugewiesenen Objekte. Squid speichert Cache-Objektverweise und häufig angeforderte Objekte zusätzlich im Hauptspeicher, damit diese Daten schneller abgefragt werden können. Der Hauptspeicher ist sehr viel schneller als eine Festplatte!

Squid hält auch andere Daten im Speicher, zum Beispiel eine Tabelle mit allen vergebenen IP-Adressen, einen genau festgelegten Domainnamen-Cache, die am häufigsten gewünschten Objekte, Puffer, Zugriffskontrolllisten, etc.

Es ist sehr wichtig, dass ausreichend Speicher für den Squid-Prozess zur Verfügung steht. Sollte er auf Festplatte ausgelagert werden müssen, wird sich die Systemleistung drastisch reduzieren. Für die Cache-Speicherverwaltung kann das Tool `cachemgr.cgi` verwendet werden. Es wird im Abschnitt 18.3.7 auf Seite 515 erläutert.

CPU

Das Programm Squid benötigt nicht viel CPU. Nur beim Start und während der Überprüfung des Cache-Inhalts ist die Prozessorlast höher. Der Einsatz eines Multiprozessorrechners steigert die Systemleistung nicht. Zur Effektivitätssteigerung ist es besser, schnellere Festplatten zu verwenden oder mehr Speicher hinzuzufügen.

18.3.4 Squid starten

Der Squid auf SUSE LINUX ist bereits soweit vorkonfiguriert, dass man ihn problemlos sofort nach der Installation starten kann. Als Voraussetzung für einen reibungslosen Start sollte das Netzwerk soweit konfiguriert sein, dass mindestens ein Nameserver und sinnvollerweise auch das Internet erreichbar sind. Probleme kann es bereiten, wenn man eine Wählverbindung mit dynamischer DNS-Konfiguration verwendet. In so einem Fall sollte mindestens der Nameserver fest eingetragen sein, da Squid erst gar nicht startet, wenn er in der `/etc/resolv.conf` keinen DNS Server findet.

Um Squid zu starten, gibt man auf der Kommandozeile (als `root`) den Befehl `rcsquid start` ein. Beim ersten Mal wird zunächst die Verzeichnisstruktur in `/var/squid/cache` angelegt. Dies wird vom Startskript `/etc/init.d/squid` automatisch durchgeführt und kann ein paar Sekunden bis Minuten dauern. Erscheint rechts in grün *done*, wurde Squid erfolgreich gestartet. Auf dem lokalen System kann man die Funktionsfähigkeit von Squid sofort testen, indem man im Browser als Proxy `localhost` und Port `3128` einträgt. Um den Zugriff auf Squid und somit das Internet für alle zu ermöglichen, braucht man in der Konfigurationsdatei `/etc/squid/squid.conf` lediglich den Eintrag `http_access deny all` auf `http_access allow all` zu ändern. Allerdings sollte man dabei bedenken, dass man den Squid damit komplett für jedermann öffnet. Von daher sollte man unbedingt ACLs definieren, die den Zugriff auf den Proxy regeln. Dazu mehr im Abschnitt 18.3.5 auf Seite 509.

Hat man Änderungen an der Konfigurationsdatei `/etc/squid/squid.conf` vorgenommen, muss man Squid dazu bringen, diese neu einzulesen. Das gelingt mit: `rcsquid reload`.

Alternativ kann man Squid auch komplett neu starten: `rcsquid restart`. Wichtig ist noch folgendes Kommando: `rcsquid status`. Damit kann man feststellen, ob der Proxy läuft und mit `rcsquid stop` wird Squid beendet. Letzteres kann eine Weile dauern, da Squid bis zu einer halben Minute (Option `shutdown_lifetime` in `/etc/squid/squid.conf`) wartet, bevor die Verbindungen zu den Clients unterbrochen werden und er dann noch seine Daten auf Platte schreiben muss.

Achtung

Beenden von Squid

Beendet man Squid mit einem `kill` oder `killall`, kann das einen zerstörten Cache zur Folge haben, den man dann löschen muss, um Squid wieder starten zu können.

Achtung

Beendet sich Squid nach kurzer Zeit, obwohl er scheinbar erfolgreich gestartet wurde, kann das an einem fehlerhaften Nameserver-Eintrag oder einer fehlenden `/etc/resolv.conf` liegen. Den Grund für einen gescheiterten Start protokolliert Squid dabei in der Datei `/var/squid/logs/cache.log`. Soll Squid bereits beim Booten automatisch gestartet werden, muss im YaST Runlevel-Editor Squid für bestimmte Runlevel aktiviert werden.

Bei einer Deinstallation von Squid werden weder Cache noch Log-Dateien entfernt. Man muss das Verzeichnis `/var/cache/squid` manuell löschen.

Lokaler DNS-Server

Einen lokalen DNS-Server wie BIND9 aufzusetzen, ist durchaus sinnvoll, auch wenn er keine eigene Domain verwaltet. Er fungiert dann lediglich als „Caching-only DNS“ und kann ohne spezielle Konfiguration DNS-Anfragen über die Root-Nameserver auflösen. Trägt man diesen in der `/etc/resolv.conf` mit der IP-Adresse `127.0.0.1` für `localhost` ein, findet Squid beim Starten immer einen gültigen Nameserver. Es reicht aus, das Paket zu installieren und BIND zu starten. Den Nameserver des Providers sollte man in der Konfigurationsdatei `/etc/named.conf` unter `forwarders` mit seiner IP-Adresse eintragen. Falls man eine Firewall laufen hat, muss man aber darauf achten, dass die DNS-Anfragen auch durchgelassen werden.

18.3.5 Die Konfigurationsdatei `/etc/squid/squid.conf`

Alle Einstellungen zum Squid Proxyserver sind in der Datei `/etc/squid/squid.conf` vorzunehmen. Um Squid erstmalig starten zu können, sind darin keine Änderungen erforderlich, der Zugriff von externen Clients ist jedoch zunächst gesperrt. Für `localhost` ist der Proxy freigegeben und als Port wird standardmäßig 3128 verwendet. Die Optionen sind ausführlich und mit vielen Beispielen in der vorinstallierten `/etc/squid/squid.conf` dokumentiert. Annähernd alle Einträge sind am Zeilenanfang durch ein `#`-Zeichen auskommentiert; am Zeilenende befinden sich die relevanten Spezifikationen. Die angegebenen Werte entsprechen fast immer den voreingestellten Werten, so dass das Entfernen des Kommentarzeichens, ohne den Parameter der Option zu ändern, bis auf wenige Ausnahmen keine Wirkung hat. Besser ist es, das Beispiel stehen zu lassen und die Option mit dem geänderten Parameter in der Zeile darunter neu einzufügen. So kann man die voreingestellten Werte und Änderungen problemlos nachvollziehen.

Hinweis

Update von Version 2.4 auf Version 2.5

Nach einem Update von Squid von Version 2.4 auf Version 2.5 muss der Cache des Squid gelöscht werden, da sich das Layout der Verzeichnisstruktur geändert hat.

Hinweis

Hat man ein Update von einer älteren Squid-Version durchgeführt, ist es unbedingt zu empfehlen, die neue `/etc/squid/squid.conf` zu verwenden.

den und nur die Änderungen von der ursprünglichen Datei zu übernehmen. Versucht man die alte `squid.conf` weiter zu verwenden, läuft man Gefahr, dass die Konfiguration nicht mehr funktioniert, da Optionen immer wieder geändert werden und neue hinzukommen.

Allgemeine Konfigurations-Optionen (Auswahl)

http_port 3128 Das ist der Port, auf dem Squid auf Anfragen der Clients lauscht. Voreingestellt ist 3128, gebräuchlich ist auch 8080. Es ist möglich, hier mehrere Portnummern, durch Leerzeichen getrennt, anzugeben.

cache_peer <hostname> <type> <proxy-port> <icp-port>
Hier kann man einen übergeordneten Proxy als „Parent“ eintragen, zum Beispiel wenn man den des Providers nutzen will oder muss. Als <hostname> trägt man den Namen bzw. die IP-Adresse des zu verwendenden Proxies und als <type> *parent* ein. Für <proxy-port> trägt man die Portnummer ein, die der Betreiber des Parent auch zur Verwendung im Browser angibt, meist 8080. Den <icp-port> kann man auf 7 oder 0 setzen, wenn man den ICP-Port des Parent nicht kennt und die Benutzung dieses mit dem Provider nicht vereinbart wurde. Zusätzlich sollte man dann noch *default* und *no-query* nach den Portnummern angeben, um die Verwendung des ICP-Protokolls ganz zu unterbinden. Squid verhält sich dann gegenüber dem Proxy des Providers wie ein normaler Browser.

cache_mem 8 MB Dieser Eintrag gibt an, wie viel Arbeitsspeicher von Squid für das Cachen maximal verwendet wird. Voreingestellt sind 8 MB.

cache_dir ufs /var/cache/squid 100 16 256
Der Eintrag *cache_dir* gibt das Verzeichnis an, in dem alle Objekte auf Platte abgelegt werden. Die Zahlen dahinter geben den maximal zu verwendenden Plattenplatz in MB und die Anzahl der Verzeichnisse in erster und zweiter Ebene an. Den Parameter *ufs* sollte man unverändert lassen. Voreingestellt sind 100 MB Plattenplatz im Verzeichnis `/var/cache/squid` zu belegen und darin 16 Unterverzeichnisse anzulegen, die jeweils wiederum 256 Verzeichnisse enthalten. Bei Angabe des zu verwendenden Plattenplatzes sollte man genügend Reserven lassen, sinnvoll sind Werte zwischen 50 und maximal 80 Prozent des verfügbaren Platzes. Die beiden letzten Zahlen für die Anzahl der Verzeichnisse sollte man nur mit Vorsicht vergrößern, da zu viele Verzeichnisse auch wieder auf Kosten der Performance

gehen können. Hat man mehrere Platten, auf die der Cache verteilt werden soll, kann man entsprechend viele *cache_dir*-Zeilen eintragen.

cache_access_log /var/log/squid/access.log

Pfadangabe für Log-Dateien.

cache_log /var/log/squid/cache.log

Pfadangabe für Log-Dateien.

cache_store_log /var/log/squid/store.log

Pfadangabe für Log-Dateien. Diese drei Einträge geben den Pfad zur Protokolldatei von Squid an. Normalerweise wird man daran nichts ändern. Wird der Squid stark beansprucht, kann es sinnvoll sein, den Cache und die Log-Dateien auf verschiedene Platten zu legen.

emulate_httppd_log off Ändert man diesen Eintrag auf *on*, erhält man lesbare Log-Dateien. Allerdings kommen manche Auswerteprogramme damit nicht zurecht.

client_netmask 255.255.255.255 Mit diesem Eintrag kann man die protokollierten IP-Adressen in den Log-Dateien maskieren, um die Identität der Clients zu verbergen. Trägt man hier *255 . 255 . 255 . 0* ein, wird die letzte Stelle der IP-Adresse auf Null gesetzt.

ftp_user Squid@ Hiermit kann man das Passwort setzen, welches Squid für den anonymen FTP-Login verwenden soll. Es kann aber sinnvoll sein, hier eine gültige E-Mail-Adresse in der eigenen Domain anzugeben, da einige FTP-Server diese auf Gültigkeit überprüfen.

cache_mgr webmaster Eine E-Mail-Adresse, an die Squid eine Nachricht schickt, wenn er unerwartet abstürzt. Voreingestellt ist *webmaster*.

logfile_rotate 0 Squid ist in der Lage, die gesicherten Log-Dateien zu rotieren, wenn man *squid -k rotate* aufruft. Die Dateien werden dabei, entsprechend der angegebenen Anzahl, durchnummeriert, und nach Erreichen des angegebenen Wertes wird die jeweils älteste Datei wieder überschrieben. Dieser Wert steht standardmäßig auf *0*, weil das Archivieren und Löschen der Log-Dateien bei SUSE LINUX von einem eigenen Cronjob durchgeführt wird, dessen Konfiguration man in der Datei */etc/logrotate/squid* findet.

append_domain <domain> Mit *append_domain* kann man angeben, welche Domain automatisch angehängt wird, wenn keine angegeben wurde. Meist wird man hier die eigene Domain eintragen, dann genügt es, im Browser *www* einzugeben, um auf den eigenen Webserver zu gelangen.

forwarded_for on Setzt man diesen Eintrag auf *off*, entfernt Squid die IP-Adresse bzw. den Systemnamen des Clients aus den HTTP-Anfragen.

negative_ttl 5 minutes; negative_dns_ttl 5 minutes

Normalerweise braucht man diese Werte nicht zu verändern. Hat man aber eine Wählleitung, kann es vorkommen, dass das Internet zeitweilig nicht erreichbar ist. Squid merkt sich dann die erfolglosen Anfragen und weigert sich, diese neu anzufragen, obwohl die Verbindung in das Internet wieder steht. Für diesen Fall sollte man die *minutes* in *seconds* ändern, dann führt auch ein *Reload* im Browser, wenige Sekunden nach der Einwahl, wieder zum Erfolg.

never_direct allow <acl_name> Will man verhindern, dass Squid Anfragen direkt aus dem Internet fordert, kann man hiermit die Verwendung eines anderen Proxies erzwingen. Diesen muss man zuvor unter *cache_peer* eingetragen haben. Gibt man als *<acl_name>* *all* an, erzwingt man, dass sämtliche Anfragen direkt an den *parent* weitergegeben werden. Das kann zum Beispiel nötig sein, wenn man einen Provider verwendet, der die Verwendung seines Proxies zwingend vorschreibt oder die Firewall keinen direkten Zugriff auf das Internet durchlässt.

Optionen zur Zugriffskontrolle

Squid bietet ein ausgeklügeltes System, um den Zugriff auf den Proxy zu steuern. Durch die Verwendung von ACLs ist es einfach und vielseitig konfigurierbar. Dabei handelt es sich um Listen mit Regeln, die der Reihe nach abgearbeitet werden. ACLs müssen zuerst definiert werden, bevor sie verwendet werden können. Einige Standard-ACLs wie *all* und *localhost* sind bereits vorhanden. Das Festlegen einer ACL an sich bewirkt aber noch gar nichts. Erst wenn sie tatsächlich eingesetzt wird, zum Beispiel in Verbindung mit *http_access*, werden die definierten Regeln abgearbeitet.

acl <acl_name> <type> <data> Eine ACL benötigt zur Definition mindestens drei Angaben. Der Name *<acl_name>* kann frei gewählt werden. Für *<type>* kann man aus einer Vielzahl unterschiedlicher Möglichkeiten auswählen, die man im Abschnitt *ACCESS CONTROLS* in der */etc/squid/squid.conf* nachlesen kann. Was für *<data>* anzugeben ist, hängt vom jeweiligen Typ der ACL ab und kann auch aus einer Datei, zum Beispiel mit Rechnernamen, IP-Adressen oder URLs eingelesen werden. Im folgenden einige einfache Beispiele:

```
acl meinesurfer srcdomain .meine-domain.com
acl lehrer src 192.168.1.0/255.255.255.0
```

```
acl studenten src 192.168.7.0-192.168.9.0/255.255.255.0
acl mittags time MTWHF 12:00-15:00
```

http_access allow <acl_name> Mit *http_access* wird festgelegt, wer den Proxy verwenden darf und auf was er im Internet zugreifen darf. Dabei sind ACLs anzugeben, *localhost* und *all* sind weiter oben bereits definiert, die mit *deny* oder *allow* den Zugriff sperren oder freigeben. Man kann hier eine Liste mit vielen *http_access*-Einträgen erstellen, die von oben nach unten abgearbeitet werden; je nachdem, was zuerst zutrifft, wird der Zugriff auf die angeforderte URL freigegeben oder gesperrt. Als letzter Eintrag sollte immer *http_access deny all* stehen. Im folgenden Beispiel hat *localhost*, also der lokale Rechner, freien Zugriff auf alles, während er für alle anderen komplett gesperrt ist:

```
http_access allow localhost
http_access deny all
```

Noch ein Beispiel, in dem die zuvor definierten ACLs verwendet werden: Die Gruppe *lehrer* hat jederzeit Zugriff auf das Internet, während die Gruppe *studenten* nur Montags bis Freitags, und da nur mittags, surfen darf:

```
http_access deny localhost
http_access allow lehrer
http_access allow studenten mittags
http_access deny all
```

Die Liste mit den eigenen *http_access*-Einträgen sollte man der Übersichtlichkeit halber nur an der dafür vorgesehenen Stelle in der */etc/squid/squid.conf* eintragen. Das bedeutet zwischen dem Text

```
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR
# CLIENTS
```

und dem abschließenden

```
http_access deny all
```

redirect_program /usr/bin/squidGuard

Mit dieser Option kann man einen „Redirector“, wie zum Beispiel squidGuard angeben, der in der Lage ist, unerwünschte URLs zu sperren. In Verbindung mit Proxy-Authentifizierung und den passenden ACLs kann man so den Zugriff auf das Internet für verschiedene Benutzergruppen sehr differenziert steuern. squidGuard ist ein eigenes Paket, das separat zu installieren und konfigurieren ist.

authenticate_program /usr/sbin/pam_auth

Sollen sich die Benutzer am Proxy authentifizieren müssen, kann man hier ein entsprechendes Programm wie zum Beispiel pam_auth angeben. Bei der Verwendung von pam_auth öffnet sich für den Anwender beim ersten Zugriff ein Loginfenster, in dem er Benutzername und Passwort eingeben muss. Zusätzlich ist noch eine ACL erforderlich, damit nur Clients mit gültigem Login surfen können:

```
acl password proxy_auth REQUIRED

http_access allow password
http_access deny all
```

Das *REQUIRED* nach *proxy_auth* kann man auch durch eine Liste von erlaubten Benutzernamen oder einen Pfad zu solch einer Liste ersetzen.

ident_lookup_access allow <acl_name>

Hiermit erreicht man, dass auf alle durch die ACL definierten Clients eine Ident-Anfrage ausgeführt wird, um die Identität des jeweiligen Benutzers zu ermitteln. Setzt man für <acl_name> *all* ein, erfolgt dies generell für alle Clients. Auf den Clients muss dazu ein Ident-Daemon laufen, bei Linux kann man dafür das Paket *pidentd* installieren, für Windows gibt es freie Software, die man sich aus dem Internet besorgen kann. Damit nur Clients mit erfolgreichem Ident-Lookup zugelassen werden, ist auch hier wieder eine entsprechende ACL zu definieren:

```
acl identhhosts ident REQUIRED

http_access allow identhhosts
http_access deny all
```

Auch hier kann man das *REQUIRED* wieder durch eine Liste erlaubter Benutzernamen ersetzen. Die Verwendung von *Ident* kann den

Zugriff merklich verlangsamen, da die Ident-Lookups durchaus für jede Anfrage wiederholt werden.

18.3.6 Transparente Proxy-Konfiguration

Normalerweise schickt der Web-Browser an einen bestimmten Port des Proxy-Servers Anfragen und der Proxy stellt die angeforderten Objekte zur Verfügung, ob sie nun im Cache sind oder nicht. Innerhalb eines echten Netzwerks können verschiedene Situationen auftreten:

- Aus Sicherheitsgründen ist es besser, wenn alle Clients zum Surfen im Internet einen Proxy verwenden.
- Es ist notwendig, dass alle Clients einen Proxy verwenden, egal ob sie sich dessen bewusst sind oder nicht.
- In einem Netzwerk wird der Proxy umgezogen, die bestehenden Clients sollen jedoch ihre alte Konfiguration behalten.

In jedem dieser Fälle kann ein transparenter Proxy eingesetzt werden. Das Prinzip ist denkbar einfach: Der Proxy nimmt die Anfragen des Web-Browsers entgegen und bearbeitet sie, sodass der Web-Browser die angeforderten Seiten erhält ohne zu wissen, woher sie kommen. Der gesamte Prozess wird transparent ausgeführt, daher der Name für den Vorgang.

Kernel-Konfiguration

Zuerst sollte sichergestellt sein, dass der Kernel des Proxy-Servers einen transparenten Proxy unterstützt. Andernfalls muss man dem Kernel diese Optionen hinzufügen und ihn neu kompilieren. Genauere Informationen dazu entnehmen Sie bitte dem Kapitel 11 auf Seite 287. Kernel Module verändern sich von Version zu Version. Prüfen Sie den aktuellen Stand unter `/usr/share/doc/howto/en/html/mini/TransparentProxy-3.html` bzw. im Internet: <http://www.tldp.org/HOWTO/mini/TransparentProxy-3.html>.

Konfigurationsoptionen in `/etc/squid/squid.conf`

Folgende Optionen in der Datei `/etc/squid/squid.conf` müssen aktiviert werden, um einen transparenten Proxy aufzusetzen:

- `httpd_accel_host virtual`

- `httpd_accel_port 80` # Port, auf dem sich der tatsächliche HTTP-Server befindet.
- `httpd_accel_with_proxy on`
- `httpd_accel_uses_host_header on`

Firewall-Konfiguration mit SuSEfirewall2

Alle durch die Firewall eingehenden Anfragen müssen mithilfe einer Port-Weiterleitungsregel an den Squid-Port weitergeleitet werden. Dafür eignet sich das SuSE-eigene Tool SuSEfirewall2. Dessen Konfigurationsdatei findet man in der Datei `/etc/sysconfig/SuSEfirewall2`. Die Konfigurationsdatei wiederum setzt sich aus gut dokumentierten Einträgen zusammen. Auch wenn wir nur einen transparenten Proxy einrichten wollen, müssen wir einige Firewall-Optionen konfigurieren. Beispielsweise:

- Gerät zeigt auf Internet: `FW_DEV_EXT="eth1"`
- Gerät zeigt auf Netzwerk: `FW_DEV_INT="eth0"`

Auf Ports und Dienste (siehe `/etc/services`) in der Firewall wird von nicht vertrauenswürdigen Netzwerken also dem Internet zugegriffen. In diesem Beispiel bieten wir lediglich Web-Dienste nach außen hin an:

```
FW_SERVICES_EXT_TCP="www"
```

Auf Ports/Dienste (siehe `/etc/services`) in der Firewall wird vom sicheren Netzwerk, sowohl TCP und UDP, zugegriffen.

```
FW_SERVICES_INT_TCP="domain www 3128"
```

```
FW_SERVICES_INT_UDP="domain"
```

Wir greifen auf Web-Dienste und Squid (dessen Standardport ist 3128) zu. Der oben beschriebene Dienst „Domain“ steht für DNS oder Domain Name Server. Es ist üblich, diesen Dienst zu nutzen. Andernfalls entfernen wir ihn einfach aus obigem Eintrag und setzen folgende Option auf `no`:

```
FW_SERVICE_DNS="yes"
```

Die wichtigste Option ist die Ziffer 15:

Beispiel 18.1: Option 15 der Firewallkonfiguration

```
#
# 15.)
# Welcher Zugriff auf die einzelnen Dienste soll an einen lokalen
# Port auf dem Firewall-Rechner umgeleitet werden?
#
# Damit können alle internen Benutzer gezwungen werden, über den
# Squid-Proxy zu surfen oder es kann eingehender Webverkehr
# transparent an einen sicheren Web-Server umgeleitet werden.
#
# Wahl: keinen Eintrag vornehmen oder folgend erklärte Syntax von
# Umleitungsregeln, getrennt durch Leerzeichen, verwenden.
# Eine Umleitungsregel besteht aus 1) Quelle IP/Netz, 2) Ziel
# IP/Netz, 3) ursprünglicher Zielport und 4) lokaler Port, an den
# der Verkehr umgeleitet werden soll, getrennt durch Kommata, z.B.:
# "10.0.0.0/8,0/0,80,3128 0/0,172.20.1.1,80,8080"
#
```

Im obigen Kommentar wird die einzuhaltende Syntax gezeigt. Zuerst greifen die IP-Adresse und die Netzwerkmaske der „internen Netzwerke“ auf die Proxy-Firewall zu. Dann die IP-Adresse und die Netzwerkmaske, an die Anfragen von den Clients „gesendet“ werden. Im Fall von Web-Browsern wählt man die Netzwerke 0/0. Dies ist eine Wildcard und bedeutet „überallhin“. Danach kommt der „ursprüngliche“ Port, an den diese Anfragen geschickt wurden und schließlich folgt der Port, an den die Anfragen „umgeleitet“ wurden.

Da Squid mehr Protokolle unterstützt als nur HTTP, können auch Anfragen von anderen Ports an den Proxy umgeleitet werden, so zum Beispiel FTP (Port 21), HTTPS oder SSL (Port 443).

Im konkreten Fall werden Web-Dienste (Port 80) auf den Proxy-Port (hier 3128 umgeleitet. Falls mehrere Netzwerke oder Dienste hinzugefügt werden sollen, müssen diese durch ein Leerzeichen im entsprechenden Eintrag getrennt werden.

```
FW_REDIRECT_TCP="192.168.0.0/16,0/0,80,3128
192.168.0.0/16,0/0,21,3128"
```

```
FW_REDIRECT_UDP="192.168.0.0/16,0/0,80,3128
192.168.0.0/16,0/0,21,3128"
```

Zum Starten der Firewall und der neuen Konfiguration muss man einen Eintrag in der Datei `/etc/sysconfig/SuSEfirewall12` editieren. Der Eintrag `START_FW` muss auf "yes" gesetzt werden:

Starten Sie Squid wie in Abschnitt 18.3.4 auf Seite 504 beschrieben. Anhand der Protokolldateien in `/var/log/squid/access.log` kann überprüft werden, ob alles richtig funktioniert. Um zu überprüfen, ob alle Ports korrekt konfiguriert wurden, kann von jedem beliebigen Rechner außerhalb unserer Netzwerke auf dem Rechner ein Portscan ausgeführt werden. Nur der Web-Dienst-Port (80) sollte offen sein. Der Portscan führt über `nmap -O <IP-Adresse>`.

18.3.7 Squid und andere Programme

In diesem Abschnitt wird gezeigt, wie andere Applikationen mit Squid interagieren. `cachemgr.cgi` ermöglicht dem Systemadministrator, den benötigten Speicher für das Zwischenspeichern von Objekten zu überprüfen. `squidGuard` filtert Webseiten, und `calamaris` ist ein Berichtsgenerator für Squid.

cachemgr.cgi

Der Cache-Manager (`cachemgr.cgi`) ist ein CGI-Hilfsprogramm zur Ausgabe von Statistiken über den benötigten Speicher des laufenden Squid-Prozesses. Im Gegensatz zum Protokollieren erleichtert dies die Cache-Verwaltung und die Anzeige von Statistiken.

Einrichten

Zuerst wird ein lauffähiger Web-Server auf dem System benötigt. Als Benutzer `root` gibt man Folgendes ein, um herauszufinden, ob Apache bereits läuft: `rcapache status`.

Erscheint eine Nachricht wie die folgende, läuft Apache auf unserem Rechner:

```
Checking for service httpd: OK
Server uptime: 1 day 18 hours 29 minutes 39 seconds
```

Andernfalls müssen Sie folgenden Befehl eingeben: `rcapache start`. So wird Apache mit den Standardeinstellungen von SUSE LINUX gestartet.

Als letzten Schritt muss man die Datei `cachemgr.cgi` aus dem Verzeichnis `/usr/share/doc/packages/squid/scripts/` in das Verzeichnis `/srv/www/cgi-bin` von Apache kopieren.

Cache-Manager ACLs in `/etc/squid/squid.conf`

Folgende Standardeinstellungen sind für den Cache-Manager erforderlich:

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
```

Folgende Regeln sollten enthalten sein:

```
http_access allow manager localhost
http_access deny manager
```

Die erste ACL ist am wichtigsten, da der Cache-Manager versucht, mit dem Squid über das `cache_object`-Protokoll zu kommunizieren. Die folgenden Regeln setzen voraus, dass der Web-Server und Squid auf demselben Rechner laufen. Die Kommunikation zwischen dem Cache-Manager und Squid entsteht beim Web-Server, nicht beim Browser. Befindet sich der Web-Server also auf einem anderen Rechner, müssen Sie extra eine ACL wie in der Beispieldatei 18.2 hinzufügen.

Beispiel 18.2: Zugriffsregeln

```
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl webserver src 192.168.1.7/255.255.255.255 # IP Webserver
```

Dann werden noch folgende Regeln aus Datei 18.3 benötigt.

Beispiel 18.3: Zugriffsregeln

```
http_access allow manager localhost
http_access allow manager webserver
http_access deny manager
```

Es ist auch möglich, ein Passwort für den Manager zu konfigurieren, wenn auf mehrere Optionen zugegriffen werden soll, wie z. B. Schließen des Cache von Remote oder Anzeigen weiterer Informationen über den Cache. Dann müssen Sie den Eintrag `cachemgr_passwd` und die Optionenliste, die angezeigt werden soll, mit einem Passwort für den Manager konfigurieren. Diese Liste erscheint als Teil der Eintragskommentare in `/etc/squid/squid.conf`.

Immer wenn sich die Konfigurationsdatei geändert hat, muss Squid neu gestartet werden. Dies geschieht am einfachsten mit dem Befehl:

```
rcsquid reload
```


Statistiken anzeigen

Gehen Sie zur entsprechenden Web-Seite, beispielsweise `http://webserver.example.org/cgi-bin/cachemgr.cgi`. Drücken Sie auf 'continue' und lassen Sie sich die verschiedenen Statistiken anzeigen. Weitere Informationen über die einzelnen Einträge, die vom Cache-Manager ausgegeben werden, finden sich in den FAQs zu Squid: `http://www.squid-cache.org/Doc/FAQ/FAQ-9.html`

squidGuard

Dieses Kapitel soll lediglich eine Einführung zur Konfiguration von squidGuard sowie ein paar Ratschläge zu dessen Einsatz geben. Auf eine umfangreiche Erklärung wird an dieser Stelle verzichtet. Tiefer gehende Informationen finden sich auf den Webseiten zu squidGuard: `http://www.squidguard.org`

squidGuard ist ein freier (GPL), flexibler und ultraschneller Filter, ein Umleiter und „PlugIn“ zur Zugriffskontrolle für Squid. Er ermöglicht das Festlegen einer Vielzahl von Zugriffsregeln mit unterschiedlichen Beschränkungen für verschiedene Benutzergruppen für einen Squid-Cache. squidGuard verwendet die Standardschnittstelle von Squid zum Umleiten. squidGuard kann u. a. für Folgendes verwendet werden:

- Beschränkung des Internetzugriffs für einige Benutzer auf bestimmte akzeptierte/bekannte Web-Server und/oder URLs.
- Zugriffsverweigerung für einige Benutzer auf bestimmte Web-Server und/oder URLs.
- Zugriffsverweigerung für einige Benutzer auf URLs, die bestimmte reguläre Ausdrücke oder Wörter enthalten.
- Umleiten gesperrter URLs an eine „intelligente“ CGI-basierte Infoseite.
- Umleiten nicht registrierter Benutzer an ein Registrationsformular.
- Umleiten von Bannern an ein leeres GIF.
- Unterschiedliche Zugriffsregeln abhängig von der Uhrzeit, dem Wochentag, dem Datum etc.
- Unterschiedliche Regeln für die einzelnen Benutzergruppen.

Weder mit squidGuard noch mit Squid ist Folgendes möglich:

- Text innerhalb von Dokumenten filtern, zensieren oder editieren.
- In HTML eingebettete Skriptsprachen wie JavaScript oder VBScript filtern, zensieren oder editieren.

Verwendung von squidGuard

Installieren Sie das squidGuard. Editieren Sie die Konfigurationsdatei `/etc/squidguard.conf`. Es gibt zahlreiche andere Konfigurationsbeispiele unter <http://www.squidguard.org/config/>. Sie können später mit komplizierteren Konfigurationseinstellungen experimentieren.

Der nächste Schritt besteht darin, eine Dummy-Seite „Zugriff verweigert“ oder eine mehr oder weniger intelligente CGI-Seite zu erzeugen, um Squid umzuleiten, falls der Client eine verbotene Webseite anfordert. Der Einsatz von Apache wird auch hier wieder empfohlen.

Nun müssen wir Squid sagen, dass er squidGuard benutzen soll. Dafür verwenden wir folgende Einträge in der Datei `/etc/squid/squid.conf`:

```
redirect_program /usr/bin/squidGuard
```

Eine andere Option namens `redirect_children` konfiguriert die Anzahl der verschiedenen auf dem Rechner laufenden „redirect“, also Umleitungsprozesse (in diesem Fall squidGuard). squidGuard ist schnell genug, um eine Vielzahl von Anfragen (squidGuard ist wirklich schnell: 100.000 Anfragen innerhalb von 10 Sekunden auf einem 500MHz Pentium mit 5900 Domains, 7880 URLs, gesamt 13780) zu bearbeiten. Es wird daher nicht empfohlen, mehr als 4 Prozesse festzusetzen, da die Zuweisung dieser Prozesse unnötig viel Speicher braucht.

```
redirect_children 4
```

Als Letztes lassen Sie den Squid seine neue Konfiguration einlesen: `rcsquid reload`. Nun können Sie Ihre Einstellungen in einem Browser testen.

Erzeugen von Cache-Berichten mit Calamaris

Calamaris ist ein Perl-Skript, das zur Erzeugung von Aktivitätsberichten des Cache im ASCII- oder HTML-Format verwendet wird. Es arbeitet mit Squid-eigenen Zugriffsprotokolldateien. Die Homepage zu Calamaris befindet sich unter: <http://Calamaris.Cord.de/>. Das Programm ist einfach zu verwenden. Melden Sie sich als root an und geben Sie folgenden Befehl ein: `cat access.log.files | calamaris [options] > reportfile`.

Beim Verketteten mehrerer Protokolldateien ist die Beachtung der chronologischen Reihenfolge wichtig, das heisst ältere Dateien kommen zuerst. Die verschiedenen Optionen:

- a wird normalerweise zur Ausgabe aller verfügbaren Berichte verwendet, mit
- w erhält man einen HTML-Bericht und mit
- l eine Nachricht oder ein Logo im Header des Berichts.

Weitere Informationen über die verschiedenen Optionen finden Sie in der Manual Page zu `calamari`: `man calamari`.

Ein weiteres, leistungsstarkes Tool zum Erzeugen von Cache-Berichten ist SARG (Squid Analysis Report Generator). . Weitere Informationen dazu gibt es auf der entsprechenden Internetseite unter: <http://web.onda.com.br/orso/>

18.3.8 Weitere Informationen zu Squid

Besuchen Sie die Homepage von Squid: <http://www.squid-cache.org/>. Hier finden Sie den Squid User Guide und eine sehr umfangreiche Sammlung von FAQs zu Squid. Das Mini-Howto zum transparenten Proxy im howtoen finden Sie unter: `/usr/share/doc/howto/en/mini/TransparentProxy.gz`

Des Weiteren gibt es Mailinglisten für Squid unter: `squid-users@squid-cache.org`. Das Archiv dazu befindet sich unter: <http://www.squid-cache.org/mail-archive/squid-users/>.

Sicherheit im Netzwerk

Masquerading, Firewall und Kerberos bilden die Grundlagen für ein sicheres Netzwerk, welche für einen kontrollierten Datenverkehr sorgen. Die Secure Shell (SSH) gibt dem Benutzer die Möglichkeit, über eine verschlüsselte Verbindung auf entfernten Rechner sich anzumelden. Um all diese umfangreichen Möglichkeiten zu nutzen, werden die grundlegenden Themen zur Netzwerksicherheit besprochen.

19.1 Masquerading und Firewall	522
19.2 SSH – secure shell, die sichere Alternative	529
19.3 Netzwerkauthentifizierung — Kerberos	535
19.4 Installation und Administration von Kerberos	543
19.5 Sicherheit ist Vertrauenssache	561

19.1 Masquerading und Firewall

Wegen seiner herausragenden Netzwerkfähigkeiten wird Linux immer häufiger als Router für Wählleitungen oder auch Standleitungen verwendet. Der Begriff Router bezieht sich hierbei auf einen Rechner, der mehr als ein Netzwerkinterface hat und der Pakete, die nicht für eines seiner eigenen Netzwerkinterfaces bestimmt sind, an seine jeweiligen Kommunikationspartner weiterleitet. Ein Router wird häufig auch Gateway genannt. Die im Linux-Kernel vorhandenen Paketfilter ermöglichen eine präzise Steuerung dafür, welche Pakete des Datenverkehrs nun passieren dürfen und welche nicht.

Das Festlegen der genauen Filterregeln für diesen Paketfilter erfordert in der Regel etwas Erfahrung seitens des Administrators. SUSE Linux hält für den weniger erfahrenen Benutzer ein `SuSEfirewall12` bereit, das das Einstellen dieser Regeln erleichtert.

Die `SuSEfirewall12` ist flexibel konfigurierbar und eignet sich deswegen auch zum Aufbau von komplexeren Paketfilterkonstrukten. Das Paketfilter-Paket erlaubt es, einen Linux-Rechner mittels Masquerading als Router zur Anbindung eines internen Netzwerks mit nur einer einzigen von außen sichtbaren IP-Adresse zu betreiben. Masquerading wird also mit Hilfe von Regeln eines Paketfilters realisiert.

Achtung

Die hier vorgestellten Verfahren gelten als Standard und funktionieren in der Regel. Es gibt jedoch keine Garantie dafür, dass sich nicht doch in diesem Buch oder woanders ein Fehler eingeschlichen hat. Sollten Cracker trotz umfassender korrekter Schutzmaßnahmen Ihrerseits in Ihr System eindringen, dann machen Sie bitte nicht die Buchautoren verantwortlich. Auch wenn Sie nicht direkt eine Antwort erhalten, können Sie sicher sein, dass wir für Kritik und Anregungen dankbar sind und Verbesserungen einbringen werden.

Achtung

19.1.1 Grundlagen des Masquerading

Masquerading ist der Linux-Spezialfall von NAT (*Network Address Translation*), der Übersetzung von Netzwerkadressen. Das Prinzip dahinter ist einfach: Ihr Router hat mehr als ein Netzwerkinterface, typischerweise sind das eine Netz Karte und eine Schnittstelle zum Internet (zum Beispiel ein

ISDN-Interface). Eines dieser Interfaces wird Sie nach außen anbinden, eines oder mehrere andere verbinden Ihren Rechner mit den weiteren Rechnern in Ihrem Netz. In einem Beispiel soll nun per ISDN nach außen gewählt werden, das äußere Netzwerkinterface ist `ipp0`. Sie haben mehrere Rechner im lokalen Netz mit der Netzwerkkarte Ihres Linux-Routers verbunden, die in diesem Beispiel `eth0` heißt. Die Rechner im Netz senden alle Pakete, die nicht für das eigene Netz bestimmt sind, an den Default-Router oder das Default-Gateway.

Hinweis

Achten Sie beim Konfigurieren Ihres Netzwerks immer auf übereinstimmende broadcast-Adressen und Netzwerkmasken!

Hinweis

Wird nun einer der Rechner in Ihrem Netz ein Paket fürs Internet abschicken, dann landet es beim Default-Router. Dieser muss so konfiguriert sein, dass er solche Pakete auch weiterleitet. Aus Sicherheitsgründen wird eine SUSE LINUX Installation dies in der Voreinstellung nicht tun! Ändern Sie die Variable `IP_FORWARD` in der Datei `/etc/sysconfig/sysctl` auf `IP_FORWARD=yes`. Nach einem Reboot oder dem folgenden Kommando ist das Weiterleiten aktiviert: `echo 1 > /proc/sys/net/ipv4/ip_forward`

Der Zielrechner der Verbindung kennt nur Ihren Router, nicht aber den eigentlichen Absender-Rechner in Ihrem inneren Netzwerk, der hinter Ihrem Router versteckt ist. Daher kommt der Begriff Masquerading. Die Zieladresse für Antwortpakete ist wegen der Adressübersetzung wieder unser Router. Dieser muss die Pakete erkennen und die Zieladresse so umschreiben, dass sie zum richtigen Rechner im lokalen Netz gelangen.

Diese Erkennung von Paketen, die zu Verbindungen gehören, die durch Masquerading durch den Router entstanden sind, geschieht mit Hilfe einer Tabelle, die direkt im Kernel Ihres Routers gehalten wird, solange die dazugehörigen Verbindungen aktiv sind. Diese Tabelle kann der Superuser (`root`) sogar mit dem Kommando `iptables` einsehen. Bitte konsultieren Sie die Manpage dieses Kommandos für genauere Anleitung. Für die Identifizierung einzelner Masquerade Verbindungen sind neben Absender- und Zieladresse auch Port-Nummern und die beteiligten Protokolle an sich relevant. Damit ist es möglich, dass Ihr Router für jeden einzelnen Ihrer lokalen Rechner viele Verbindungen gleichzeitig bereitstellen kann.

Da der Weg der Pakete von außen nach innen von der Masquerading-Tabelle abhängt, gibt es keine Möglichkeit, von außen eine Verbindung nach innen zu öffnen. Für diese Verbindung gäbe es keinen Eintrag in der

Tabelle. Eine etablierte Verbindung hat darüber hinaus in der Tabelle einen zugeordneten Status, so dass dieser Tabelleneintrag nicht von einer zweiten Verbindung genutzt werden kann.

In der Folge ergeben sich nun Probleme mit manchen Anwendungen, zum Beispiel ICQ, cucme, IRC (DCC, CTCP), Quake und FTP (im PORT-Mode). Netscape, das Standard-FTP-Programm und viele andere benutzen den PASV-Modus, der im Zusammenhang mit Paketfiltern und Masquerading weit weniger problembehaftet ist.

19.1.2 Grundlagen Firewalling

Firewall ist wohl der am weitesten verbreitete Begriff für einen Mechanismus, der zwei Netze miteinander verbindet, aber für möglichst kontrollierten Datenverkehr sorgt. Es gibt verschiedene Bauarten von Firewalls, die sich hauptsächlich in der logisch-abstrakten Ebene unterscheiden, auf der sie den Datenverkehr untersuchen und regulieren. Die Methode, die wir hier vorstellen, müsste sich eigentlich genauer Paketfilter nennen. Ein Paketfilter regelt den Durchlass anhand von Kriterien wie Protokoll, Port und IP-Adresse. Auf diese Weise können Sie also Pakete abfangen, die aufgrund ihrer Adressierung nicht in Ihr Netz durchdringen sollen. Beispielsweise sollten Sie Pakete abfangen, die den telnet-Dienst Ihrer Rechner auf port 23 zum Ziel haben. Wenn Sie beispielsweise Zugriffe auf Ihren Webserver zulassen wollen, müssen Sie den dazugehörigen Port freischalten. Der Inhalt dieser Pakete, falls sie legitim adressiert sind (also beispielsweise mit Ihrem Webserver als Ziel), wird nicht untersucht. Das Paket könnte insofern einen Angriff auf ein CGI-Programm auf Ihrem Webserver enthalten und wird vom Paketfilter durchgelassen.

Ein wirksames — wenn auch komplexeres — Konstrukt ist die Kombination von mehreren Bauarten, beispielsweise ein Paketfilter mit zusätzlichem Application Gateway/Proxy. Der Paketfilter wehrt Pakete ab, die zum Beispiel an nicht freigeschaltete Ports gerichtet sind. Nur Pakete für ein Application Gateway sollen durchgelassen werden. Dieses Proxy tut nun so, als wäre es der eigentliche Kommunikationspartner des Servers, der mit uns eine Verbindung herstellt. In diesem Sinne kann ein solches Proxy als eine Masquerading-Maschine auf der Ebene des Protokolls der jeweiligen Anwendung angesehen werden. Ein Beispiel für solch ein Proxy ist Squid, ein HTTP Proxy Server, für den Sie Ihren Browser so konfigurieren müssen, dass Anfragen für HTML-Seiten zuerst an den Speicher des Proxy gehen und nur, wenn dort die Seite nicht zu finden ist, vom Proxy in das Internet geschickt werden. Die SUSE proxy suite (das proxy-suite) enthält übrigens einen Proxy-Server für das FTP-Protokoll.

Im Folgenden wollen wir uns auf das Paketfilter-Paket bei SUSE LINUX konzentrieren. Für mehr Informationen und weitere Links zum Thema Firewall lesen Sie bitte das Firewall-HOWTO, enthalten im `howtode`. Es lässt sich mit dem Kommando `less /usr/share/doc/howto/de/DE-Firewall-HOWTO.txt.gz` lesen, wenn das Paket `howtode` installiert ist.

19.1.3 SuSEfirewall2

Die Konfiguration der SuSEfirewall2 erfordert einiges an Wissen und Erfahrung. Unter `/usr/share/doc/packages/SuSEfirewall2` finden Sie Dokumentation zur SuSEfirewall2.

Die Konfiguration lässt sich entweder mit YaST vornehmen (s. Abschnitt 19.1.3 auf Seite 528) oder kann direkt in der Datei `/etc/sysconfig/SuSEfirewall2` erfolgen, die ausführliche englische Kommentare enthält.

Manuelle Konfiguration

Wir werden Ihnen nun Schritt für Schritt eine erfolgreiche Konfiguration vorführen. Es ist bei jedem Punkt angeführt, ob er für Masquerading oder Firewall gilt. In der Konfigurationsdatei ist auch von einer DMZ (Demilitarisierte Zone) die Rede, auf die an dieser Stelle nicht näher eingegangen wird.

Falls Sie wirklich nicht mehr als Masquerading brauchen, füllen Sie nur die mit *Masquerading* bezeichneten Zeilen aus.

- Aktivieren Sie zunächst mit dem YaST Runlevel Editor die SuSEfirewall2 für Ihren Runlevel (wahrscheinlich 3 oder 5). Dadurch werden symbolische Links für die `SuSEfirewall2_*` Skripte in den Verzeichnissen `/etc/init.d/rc?.d/` angelegt.
- `FW_DEV_WORLD` (Firewall, Masquerading): zum Beispiel `eth0`, als Device, das ins Internet führt. Bei ISDN ist es zum Beispiel `ipp0`.
- `FW_DEV_INT` (Firewall, Masquerading): Geben Sie hier das Device an, das ins innere, private Netz zeigt. Falls kein inneres Netz vorhanden ist, einfach leer lassen.
- `FW_ROUTE` (Firewall, Masquerading): Wenn Sie Masquerading brauchen, müssen Sie hier auf jeden Fall `yes` eintragen. Ihre internen

Rechner sind nicht von außen sichtbar, da diese private Netzwerkadressen (zum Beispiel 192.168.x.x) haben, die im Internet gar nicht geroutet werden.

Bei einer Firewall ohne Masquerading wählen Sie hier nur dann *yes*, wenn Sie Zugang zum internen Netz erlauben wollen. Dazu müssen die internen Rechner offiziell zugewiesene IP-Adressen haben. Im Normalfall sollten Sie allerdings den Zugang von außen auf die internen Rechner *nicht* erlauben!

- **FW_MASQUERADE** (Masquerading): Wenn Sie Masquerading brauchen, müssen Sie hier *yes* eintragen. Beachten Sie, dass es sicherer ist, wenn die Rechner des internen Netzes über Proxy-Server auf das Internet zugreifen.
- **FW_MASQ_NETS** (Masquerading): Tragen Sie hier die Rechner oder Netzwerke ein, für die Masquerading vorgenommen werden soll. Trennen Sie die einzelnen Einträge durch Leerzeichen. Zum Beispiel: `FW_MASQ_NETS="192.168.0.0/24 192.168.10.1"`
- **FW_PROTECT_FROM_INTERNAL** (Firewall): Tragen Sie hier *yes* ein, wenn Sie den Firewall-Rechner auch durch Angriffe vom internen Netz schützen wollen. Dann müssen Sie die Services, die für das innere Netz verfügbar sind, explizit freigeben. Siehe auch `FW_SERVICES_INTERNAL_TCP` und `FW_SERVICES_INTERNAL_UDP`.
- **FW_AUTOPROTECT_GLOBAL_SERVICES** (Firewall): Im Normalfall auf *yes* lassen.
- **FW_SERVICES_EXTERNAL_TCP** (Firewall): Tragen Sie hier die Services ein, auf die zugegriffen werden soll; zum Beispiel `www smtp ftp domain 443` – für den Rechner zu Hause, der keine Dienste anbieten soll, tragen Sie meist nichts ein.
- **FW_SERVICES_EXTERNAL_UDP** (Firewall): Wenn Sie nicht gerade einen Nameserver betreiben, auf den von außen zugegriffen werden soll, lassen Sie dieses Feld leer. Ansonsten fügen Sie hier die benötigten Ports ein.
- **FW_SERVICES_INTERNAL_TCP** (Firewall): Hier werden die für das innere Netz zur Verfügung stehenden Dienste deklariert. Die Angaben sind analog zu denen unter `FW_SERVICES_EXTERNAL_TCP`, beziehen sich hier aber auf das *interne* Netz.
- **FW_SERVICES_INTERNAL_UDP** (Firewall): Siehe oben.

- **FW_TRUSTED_NETS (Firewall):** Hier tragen Sie die Rechner ein, denen Sie *wirklich* vertrauen können (Trusted Hosts). Beachten Sie zudem, dass auch diese Rechner vor Eindringlingen geschützt sein müssen. `172.20.0.0/16 172.30.4.2` bedeutet, dass alle Rechner, deren IP-Adresse mit `172.20.x.x` beginnt, sowie der Rechner mit der IP-Adresse `172.30.4.2` durch die Firewall hindurch können.
- **FW_SERVICES_TRUSTED_TCP (Firewall):** Hier legen Sie die TCP-Portadressen fest, die von den Trusted Hosts benutzt werden können. Geben Sie zum Beispiel `1:65535` ein, wenn die vertrauenswürdigen Rechner auf alle Services zugreifen dürfen. Normalerweise sollte es reichen, wenn man hier als Service `ssh` eingibt.
- **FW_SERVICES_TRUSTED_UDP (Firewall):** Wie oben, nur auf UDP bezogen.
- **FW_ALLOW_INCOMING_HIGHPORTS_TCP (Firewall):** Wenn Sie mit normalem (aktivem) FTP arbeiten wollen, so tragen Sie hier `ftp-data` ein.
- **FW_ALLOW_INCOMING_HIGHPORTS_UDP (Firewall):** Tragen Sie hier `dns` ein, damit Sie die in `/etc/resolv.conf` eingetragenen Nameserver verwenden können. Mit `yes` geben Sie alle hohen Portnummern frei.
- **FW_SERVICE_DNS (Firewall):** Falls bei Ihnen ein Nameserver läuft, auf den von außen zugegriffen werden soll, tragen Sie hier `yes` ein; in `FW_TCP_SERVICES_*` muss zugleich der Port 53 freigeschaltet sein.
- **FW_SERVICE_DHCLIENT (Firewall):** Wenn Sie `dhclient` benutzen, um Ihre IP-Adresse zu beziehen, so müssen Sie hier `yes` eintragen.
- **FW_LOG_*:** Stellen Sie hier ein, was Sie mitloggen wollen. Für den laufenden Betrieb reicht `yes` bei `FW_LOG_DENY_CRIT`.
- **FW_STOP_KEEP_ROUTING_STATE (Firewall):** Falls Sie automatisch per `diad` oder über ISDN (dial on demand) ins Internet gehen, so tragen Sie hier `yes` ein.

Damit ist die Konfiguration abgeschlossen. Vergessen Sie nicht, die Firewall zu testen (zum Beispiel `telnet` von außen); Sie sollten dann in `/var/log/messages` in etwa folgende Einträge sehen:

```
Mar 15 13:21:38 linux kernel: SFW2-INext-DROP-DEFLT IN=eth0 OUT=
MAC=00:80:c8:94:c3:e7:00:a0:c9:4d:27:56:08:00 SRC=192.168.10.0
```

```
DST=192.168.10.1 LEN=60 TOS=0x10 PREC=0x00 TTL=64 ID=15330 DF  
PROTO=TCP SPT=48091 DPT=23 WINDOW=5840 RES=0x00 SYN URGP=0 OPT  
(020405B40402080A061AFEB0000000001030300)
```

Konfiguration mit YaST

Die grafisch geführte Konfiguration mit YaST erreichen Sie über das YaST-Kontrollzentrum. Wählen Sie aus der Kategorie 'System und Benutzer' den Unterpunkt 'Firewall'. Die Konfiguration ist in vier Teilabschnitte gegliedert:

Grundeinstellungen Legen Sie die abzusichernden Interfaces fest. Ist ein einzelner Rechner ohne internes Netz dahinter abzusichern, geben Sie nur die nach außen ins Internet gerichtete Schnittstelle an. Ist ein internes Netz hinter Ihrem System geschaltet, muss auch die nach innen gerichtete Schnittstelle angegeben werden. Verlassen Sie diesen Dialog mit 'Weiter'.

Dienste Diese Option ist nur relevant, falls Sie über Ihr System Dienste anbieten wollen, die vom Internet aus verfügbar sein sollen (Web-Server, Mail-Server etc.). Aktivieren Sie die entsprechenden Check-boxen und/oder nehmen Sie über den Button 'Experten ...' die Freischaltung bestimmter Dienste über deren Portnummern (nachzulesen in `/etc/services`) vor. Soll Ihr Rechner nicht als Server betrieben werden, verlassen Sie diesen Dialog ohne jegliche Änderung mit 'Weiter'.

Features Hier selektieren Sie die wichtigsten Features, die Ihre Firewall auszeichnen sollen:

- 'Traceroute erlauben' hilft, das Routing zu Ihrer Firewall hin zu überprüfen.
- 'Daten weiterleiten und Masquerading durchführen' schirmt Rechner aus dem internen Netz gegen das Internet ab — alle Internetdienste werden scheinbar von Ihrer Firewall benutzt, während die internen Rechner unsichtbar bleiben.
- 'Alle laufenden Dienste schützen' bedeutet, dass jeglicher Netzwerkzugriff auf TCP- und UDP-Dienste der Firewall verhindert wird. Ausgenommen hiervon sind die Dienste, die Sie im vorhergehenden Schritt explizit freigeschaltet haben.
- 'Vor internem Netz schützen' Nur die freigegebenen Dienste der Firewall sind für die *internen* Rechner verfügbar. Da hier keine

Freigabe von Diensten möglich ist, sollten Sie diese Option besser deaktivieren, wenn Sie Zugriff aus dem internen Netz wünschen.

Ist die Featurekonfiguration abgeschlossen, verlassen Sie diese Maske mit 'Weiter'.

Protokollierung Hier legen Sie den Umfang der Protokollierung Ihrer Firewall fest. Bevor Sie die 'Optionen zur Fehlersuche' aktivieren, bedenken Sie, dass diese Logfiles große Ausgabemengen erzeugen. Mit der Konfiguration der Protokollierung ist die Konfiguration Ihrer Firewall abgeschlossen. Verlassen Sie den Dialog mit 'Weiter' und bestätigen Sie die nun erscheinende Meldung zur Aktivierung der Firewall.

19.2 SSH – secure shell, die sichere Alternative

Vernetztes Arbeiten erfordert oft auch den Zugriff auf entfernte Systeme. Hierbei muss sich der Benutzer über sein Login und ein Passwort authentifizieren. Unverschlüsselt im Klartext versandt, könnten diese sensiblen Daten jederzeit von Dritten mitgeschnitten und in ihrem Sinne eingesetzt werden, um zum Beispiel den Zugang des Benutzers ohne sein Wissen nutzen. Abgesehen davon, dass die Angreifer so sämtliche privaten Daten des Benutzers einsehen können, können sie den erworbenen Zugang nutzen, um von dort aus andere Systeme anzugreifen, oder Administrator- bzw. Rootrechte auf dem betreffenden System zu erlangen. Früher wurde zur Verbindungsaufnahme zwischen zwei entfernten Rechnern Telnet verwendet, das keinerlei Verschlüsselungs- oder Sicherheitsmechanismen gegen ein Abhören der Verbindungen vorsieht. Ebenso wenig geschützt sind einfache FTP- oder Kopierverbindungen zwischen entfernten Rechnern.

Die SSH-Software liefert den gewünschten Schutz. Die komplette Authentifizierung, in der Regel Benutzername und Passwort, und Kommunikation erfolgen hier verschlüsselt. Zwar ist auch hier weiterhin das Mitschneiden der übertragenen Daten möglich, doch kann der Inhalt mangels fehlendem Schlüssel durch einen Dritten nicht wieder entschlüsselt werden. So wird sichere Kommunikation über unsichere Netze wie das Internet möglich. SUSE LINUX bietet das Paket OpenSSH an.

19.2.1 Das OpenSSH-Paket

Per Default wird unter SUSE LINUX das Paket OpenSSH installiert. Es stehen Ihnen daher die Programme `ssh`, `scp` und `sftp` als Alternative für `telnet`, `rlogin`, `rsh`, `rcp` und `ftp` zur Verfügung.

19.2.2 Das ssh-Programm

Mit `ssh` können Sie Verbindung zu einem entfernten System aufnehmen und dort interaktiv arbeiten. Es ist somit gleichermaßen ein Ersatz für `telnet` und `rlogin`. Aufgrund der Verwandtschaft zu `rlogin` zeigt der zusätzliche symbolische Name `slogin` ebenfalls auf `ssh`. Zum Beispiel kann man sich mit dem Befehl `ssh sun` auf dem Rechner `sun` anmelden. Anschließend wird man nach seinem Passwort auf dem System `sun` gefragt.

Nach erfolgreicher Authentifizierung kann dort auf der Kommandozeile oder interaktiv, zum Beispiel mit YaST, gearbeitet werden. Sollten sich der lokale Benutzername und der auf dem entfernten System unterscheiden, kann ein abweichender Name angegeben werden, zum Beispiel `ssh -l august sun` oder `ssh august@sun`.

Darüber hinaus bietet `ssh` die von `rsh` bekannte Möglichkeit, Kommandos auf einem anderen System auszuführen. Im nachfolgenden Beispiel wird das Kommando `uptime` auf dem Rechner `sun` ausgeführt und ein Verzeichnis mit dem Namen `tmp` angelegt. Die Programmausgabe erfolgt auf dem lokalen Terminal des Rechners `earth`.

```
ssh sonne "uptime; mkdir tmp"
tux@sonne's password:
1:21pm up 2:17, 9 users, load average: 0.15, 0.04, 0.02
```

Anführungszeichen sind hier zum Zusammenfassen der beiden Anweisungen in einem Kommando erforderlich. Nur so wird auch der zweite Befehl auf dem Rechner `sun` ausgeführt.

19.2.3 scp – sicheres Kopieren

Mittels `scp` kopieren Sie Dateien auf einen entfernten Rechner. `scp` ist der sichere, verschlüsselte Ersatz für `rcp`. Zum Beispiel kopiert `scp MeinBrief.tex sun`: die Datei `MeinBrief.tex` vom Rechner `earth` auf den Rechner `sun`. Insoweit sich die beteiligten Nutzernamen

auf earth und sun unterscheiden, geben Sie bei scp die Schreibweise `Nutzername@Rechnername` an. Eine Option `-l` existiert nicht.

Nachdem das Passwort eingegeben wurde, beginnt scp mit der Datenübertragung und zeigt dabei den Fortschritt anhand eines von links nach rechts anwachsenden Balkens aus Sternen an. Zudem wird am rechten Rand die geschätzte Restübertragungszeit *estimated time of arrival* angezeigt. Jegliche Ausgabe kann durch die Option `-q` unterdrückt werden.

scp bietet neben dem Kopieren einzelner Dateien ein rekursives Verfahren zum Übertragen ganzer Verzeichnisse: `scp -r src/ sun:backup/` kopiert den kompletten Inhalt des Verzeichnisses `src/` inklusive aller Unterverzeichnisse auf den Rechner `sun` und dort in das Unterverzeichnis `backup/`. Dieses wird automatisch angelegt wenn es fehlt.

Mittels der Option `-p` kann scp die Zeitstempel der Dateien erhalten. `-C` sorgt für eine komprimierte Übertragung. Dadurch wird einerseits das zu übertragende Datenvolumen minimiert, andererseits aber ein höherer Rechenaufwand erforderlich.

19.2.4 sftp - sicherere Dateiübertragung

Alternativ kann man zur sicheren Datenübertragung sftp verwenden. sftp bietet innerhalb der Sitzung viele der von ftp bekannten Kommandos. Gegenüber scp mag es vor allem beim Übertragen von Daten, deren Dateinamen unbekannt sind, von Vorteil sein.

19.2.5 Der SSH Daemon (sshd) – die Serverseite

Damit ssh und scp, die Clientprogramme des SSH-Paketes, eingesetzt werden können, muss im Hintergrund der SSH-Daemon, ein Server, laufen. Dieser erwartet seine Verbindungen auf TCP/IP Port 22.

Während des ersten Starts generiert der Daemon drei Schlüsselpaare. Die Schlüsselpaare bestehen aus einem privaten und einem öffentlichen *public* Teil. Deshalb bezeichnet man dies als ein public-key basiertes Verfahren. Um die Sicherheit der Kommunikation mittels SSH zu gewährleisten, darf ausschließlich der Systemadministrator die Dateien der privaten Schlüssel einsehen können. Die Dateirechte werden per Voreinstellung entsprechend restriktiv gesetzt. Die privaten Schlüssel werden lediglich lokal vom SSH-Daemon benötigt und dürfen an niemanden weitergegeben werden. Demgegenüber werden die öffentlichen Schlüsselbestandteile (an der Namensendung `.pub` erkennbar) an Kommunikationspartner weitergegeben und sind entsprechend für alle Benutzer lesbar.

Eine Verbindung wird vom SSH-Client eingeleitet. Der wartende SSH-Daemon und der anfragende SSH-Client tauschen Identifikationsdaten aus, um die Protokoll- und Softwareversion abzugleichen und die Verbindung zu einem falschen Port auszuschließen. Da ein Kindprozess des ursprünglichen SSH-Daemons antwortet, sind gleichzeitig viele SSH-Verbindungen möglich.

OpenSSH unterstützt zur Kommunikation zwischen SSH-Server und SSH-Client das SSH-Protokoll in den Versionen 1 und 2. Nach einer Neuinstallation von SUSE LINUX wird automatisch die aktuelle Protokoll-Version 2 eingesetzt. Möchten Sie nach einem Update weiterhin SSH 1 beibehalten, folgen Sie den Anweisungen in `/usr/share/doc/packages/openssh/README.SuSE`. Dort ist ebenfalls beschrieben, wie Sie in wenigen Schritten eine SSH 1-Umgebung in eine funktionierende SSH 2-Umgebung umwandeln.

Bei Verwendung der SSH Protokoll-Version 1 sendet der Server sodann seinen öffentlichen `host key` und einen stündlich vom SSH-Daemon neu generierten `server key`. Mittels beider verschlüsselt *encrypt* der SSH-Client einen von ihm frei gewählten Sitzungsschlüssel *session key* und sendet ihn an den SSH-Server. Er teilt dem Server zudem die gewählte Verschlüsselungsmethode *cipher* mit.

Die SSH Protokoll-Version 2 kommt ohne den `server key` aus. Stattdessen wird ein Algorithmus nach Diffie-Hellman verwendet, um die Schlüssel auszutauschen.

Die zur Entschlüsselung des Sitzungsschlüssels zwingend erforderlichen privaten `host` und `server keys`, können nicht aus den öffentlichen Teilen abgeleitet werden. Somit kann allein der kontaktierte SSH-Daemon mit seinen privaten Schlüsseln den Sitzungsschlüssel entziffern (vgl. `man /usr/share/doc/packages/openssh/RFC.nroff`). Diese einleitende Phase der Verbindung kann man mittels der Fehlersuchoption `-v` des SSH-Clientprogramms gut nachvollziehen. Per Default wird SSH Protokoll-Version 2 verwendet, man kann jedoch mit dem Parameter `-1` auch die SSH Protokoll-Version 1 erzwingen. Indem der Client alle öffentlichen `host keys` nach der ersten Kontaktaufnahme in `~/.ssh/known_hosts` ablegt, können so genannte man-in-the-middle Angriffe unterbunden werden. SSH-Server, die versuchen, Name und IP-Adresse eines anderen vorzutauschen, werden durch einen deutlichen Hinweis enttarnt. Sie fallen entweder durch einen gegenüber `~/.ssh/known_hosts` abweichenden `host`-Schlüssel auf, oder können mangels passendem privaten Gegenstück den vereinbarten Sitzungsschlüssel nicht entschlüsseln.

Es empfiehlt sich, die in `/etc/ssh/` abgelegten privaten und öffentlichen Schlüssel extern und gut geschützt zu archivieren. So können Änderungen

der Schlüssel festgestellt und nach einer Neuinstallation die alten wieder eingespielt werden. Dies erspart den Benutzern die beunruhigende Warnung. Ist es sichergestellt, dass es sich trotz der Warnung um den korrekten SSH-Server handelt, muss der vorhandene Eintrag zu diesem System aus `~/.ssh/known_hosts` entfernt werden.

19.2.6 SSH-Authentifizierungsmechanismen

Jetzt erfolgt die eigentliche Authentifizierung, die in ihrer einfachsten Weise aus der Eingabe eines Passwortes besteht, wie es in den oben aufgezeigten Beispielen erfolgte. Ziel von SSH war die Einführung einer sicheren, aber zugleich einfach zu nutzenden Software. Wie bei den abzulösenden Programmen `rsh` und `rlogin` muss deshalb auch SSH eine im Alltag einfach zu nutzende Authentifizierungsmethode bieten. SSH realisiert dies mittels eines weiteren hier vom Nutzer erzeugten Schlüsselpaares. Dazu liefert das SSH-Paket das Hilfsprogramm `ssh-keygen` mit. Nach der Eingabe von `ssh-keygen -t rsa` oder `ssh-keygen -t dsa` wird das Schlüsselpaar generiert und der Basisdateiname zur Ablage der Schlüssel erfragt:

```
Enter file in which to save the key (/home/tux/.ssh/id_rsa):
```

Bestätigen Sie die Voreinstellung und beantworten Sie die Frage nach einer Passphrase. Auch wenn die Software eine leere Passphrase nahelegt, sollte bei der hier vorgeschlagenen Vorgehensweise ein Text von zehn bis 30 Zeichen Länge gewählt werden. Verwenden Sie möglichst keine kurzen und einfachen Worte oder Sätze. Nach erfolgter Eingabe wird zur Bestätigung eine Wiederholung der Eingabe verlangt. Anschließend wird der Ablageort des privaten und öffentlichen Schlüssels, in unserem Beispiel der Dateien `id_rsa` und `id_rsa.pub`, ausgegeben.

```
Enter same passphrase again:
```

```
Your identification has been saved in /home/tux/.ssh/id_rsa
Your public key has been saved in /home/tux/.ssh/id_rsa.pub.
The key fingerprint is:
79:c1:79:b2:e1:c8:20:c1:89:0f:99:94:a8:4e:da:e8 tux@sun
```

Verwenden Sie `ssh-keygen -p -t rsa` bzw. `ssh-keygen -p -t dsa`, um Ihre alte Passphrase zu ändern. Kopieren Sie den öffentlichen Teil des Schlüssels (in unserem Beispiel `id_rsa.pub`) auf den entfernten Rechner und legen Sie ihn dort als `~/.ssh/authorized_keys` ab. Zur Authentifizierung werden Sie beim nächsten Verbindungsaufbau nach Ihrer Passphrase gefragt. Sollte dies nicht der Fall sein, überprüfen Sie bitte Ort und Inhalt der zuvor erwähnten Dateien.

Auf Dauer ist diese Vorgehensweise mühsamer, als die Eingabe eines Passwortes. Entsprechend liefert das SSH-Paket ein weiteres Hilfsprogramm, den `ssh-agent`, der für die Dauer einer X-session private Schlüssel bereit hält. Dazu wird das gesamte X als Kindprozess des `ssh-agent`s gestartet. Sie erreichen dies am einfachsten, indem Sie am Anfang der Datei `.xsession` die Variable `usessh` auf `yes` setzen und sich über einen Displaymanager, zum Beispiel KDM oder XDM, anmelden. Alternativ können Sie `ssh-agent startx` verwenden.

Nun können Sie wie gewohnt `ssh` oder `scp` nutzen. Insoweit Sie Ihren öffentlichen Schlüssel wie zuvor verteilt haben, sollten Sie jetzt nicht mehr nach dem Passwort gefragt werden. Achten Sie beim Verlassen Ihres Rechners darauf, dass Sie Ihre X-session beenden oder mittels einer passwortgeschützten Bildschirmsperre, zum Beispiel `xlock`, verriegeln.

Alle wichtigen Änderungen die sich mit der Einführung von SSH Protokoll-Version 2 ergeben haben, wurden auch in der Datei `/usr/share/doc/packages/openssh/README.SuSE` noch einmal dokumentiert.

19.2.7 X-, Authentifizierungs- und sonstige Weiterleitung

Über die bisher beschriebenen sicherheitsrelevanten Verbesserungen hinaus erleichtert `ssh` auch die Verwendung von entfernten X-Anwendungen. Insoweit Sie `ssh` mit der Option `-X` aufrufen, wird auf dem entfernten System automatisch die `DISPLAY`-Variable gesetzt und alle X-Ausgaben werden durch die bestehende `ssh`-Verbindung auf den Ausgangsrechner weitergeleitet. Diese bequeme Funktion unterbindet gleichzeitig die bisher bestehenden Abhörmöglichkeiten bei entfernt aufgerufenen und lokal betrachteten X-Anwendungen.

Durch die gesetzte Option `-A` wird der Mechanismus zur Authentifizierung des `ssh-agent` auf den nächsten Rechner mit übernommen. Man kann so von einem Rechner zum anderen gehen, ohne ein Passwort eingeben zu müssen. Allerdings nur, wenn man zuvor seinen öffentlichen Schlüssel auf die beteiligten Zielrechner verteilt und korrekt abgelegt hat.

Beide Mechanismen sind vorsichtshalber in der Voreinstellung deaktiviert, können jedoch in der systemweiten Konfigurationsdatei `/etc/ssh/ssh_config` oder der nutzereigenen `~/.ssh/config` permanent eingeschaltet werden.

Man kann `ssh` auch zur beliebigen Umleitung von TCP/IP-Verbindungen benutzen. Als Beispiel sei hier die Weiterleitung des SMTP- und POP3-Ports aufgeführt:

```
ssh -L 25:sun:25 earth
```

Hier wird jede Verbindung zu earth Port 25, SMTP auf den SMTP-Port von sun über den verschlüsselten Kanal weitergeleitet. Dies ist insbesondere für Nutzer von SMTP-Servern ohne SMTP-AUTH oder POP-before-SMTP-Fähigkeiten von Nutzen. Mail kann so von jedem beliebigen Ort mit Netzanschluss zur Auslieferung durch den heimischen Mailserver übertragen werden. Analog können mit folgendem Befehl alle POP3-Anfragen (Port 110) an earth auf den POP3-Port von sun weitergeleitet werden:

```
ssh -L 110:sun:110 earth
```

Beide Beispiele müssen Sie als Benutzer `root` ausführen, da auf privilegierte, lokale Ports verbunden wird. Bei bestehender SSH-Verbindung wird Mail wie gewohnt als normaler Benutzer versandt und abgeholt. Der SMTP- und POP3-Host muss dabei auf `localhost` konfiguriert werden. Zusätzliche Informationen entnehmen Sie den Manualpages der einzelnen Programme und den Dateien unter `/usr/share/doc/packages/openssh`.

19.3 Netzwerkauthentifizierung — Kerberos

Ein offenes Netzwerk bietet außer den gewöhnlichen Passwortmechanismen — die von Natur aus unsicher sind — keinerlei Möglichkeit, um sicherzustellen, dass ein Arbeitsplatzrechner seine Benutzer eindeutig identifizieren kann. Das bedeutet, dass eine beliebige Person unter der Vorgabe einer anderen Identität dessen E-Mails abholen, auf dessen private Dateien zugreifen oder einen Dienst starten könnte. Ihre Netzwerkkumgebung muss daher die folgenden Anforderungen erfüllen, um wirklich sicher zu sein:

- Lassen Sie alle Benutzer für jeden gewünschten Dienst ihre Identität nachweisen und stellen Sie sicher, dass niemand die Identität eines anderen Benutzers annehmen kann.
- Stellen Sie außerdem sicher, dass jeder Netzwerkservers seine Identität nachweist. Falls Sie dies nicht tun, könnte es einem Angreifer gelingen, sich als der von ihnen angefragte Server auszugeben und vertrauliche Informationen abfangen, die Sie dem Server senden. Dieser

Vorgang wird als „Mutual Authentication“ (gegenseitige Authentifizierung) bezeichnet, weil sich der Client beim Server und der Server beim Client authentifiziert.

Durch stark verschlüsselte Authentifizierung hilft Ihnen Kerberos, die o. g. Anforderungen zu erfüllen. Die folgenden Abschnitte zeigen Ihnen, wie dies erreicht wird. Bitte beachten Sie, dass hier nur die grundlegende Arbeitsweise von Kerberos dargelegt wird. Ausführlichere technische Anweisungen sind in der mit Ihrer Kerberos-Implementierung mitgelieferten Dokumentation enthalten.

Hinweis

Das ursprüngliche Kerberos wurde am MIT entwickelt. Neben MIT Kerberos existieren noch verschiedene andere Implementierungen von Kerberos. SUSE LINUX enthält eine freie Implementierung von Kerberos 5, das so genannte Heimdal Kerberos 5 von KTH. Da sich der folgende Text auf gemeinsame Eigenschaften aller Implementierungen von Kerberos bezieht, bezeichnen wir das Programm als Kerberos, es sei denn, es handelt sich um spezifische Information über Heimdal.

Hinweis

19.3.1 Kerberos-Terminologie

Bevor wir auf die Einzelheiten von Kerberos eingehen, wollen wir einen Blick auf das folgende Glossar werfen, das Ihnen helfen wird, mit der Kerberos-Terminologie zurechtzukommen.

Credential Benutzer oder Clients müssen Credentials (Berechtigungsnachweise) vorweisen können, die sie berechtigen, Dienste anzufordern. Kerberos kennt zwei Arten von Berechtigungsnachweisen — Tickets und Authenticators.

Ticket Ein Ticket ist ein serverbezogener Berechtigungsnachweis, den ein Client benutzt, um sich bei einem Server zu authentifizieren, von dem er einen Dienst anfordert. Es enthält den Namen des Servers, den Namen des Clients, die Internetadresse des Clients, einen Zeitstempel (engl. *timestamp*), eine Lebensdauer und einen zufällig generierten Session Key. Alle diese Daten werden mit dem Schlüssel des Servers verschlüsselt.

Authenticator In Verbindung mit dem Ticket wird ein Authenticator benutzt, um zu beweisen, dass der Client, der ein Ticket vorlegt, tatsächlich derjenige ist, der er zu sein vorgibt. Ein Authenticator wird anhand des Namens des Clients, der IP-Adresse des Arbeitsplatzrechners und der aktuellen Uhrzeit am Arbeitsplatzrechner erstellt — verschlüsselt mit dem Session Key, der nur dem Client und dem Server, von dem er einen Dienst anfordert, bekannt ist. Im Gegensatz zu einem Ticket kann ein Authenticator nur einmal benutzt werden. Ein Client kann selbst einen Authenticator erzeugen.

Principal Ein Kerberos-Principal ist eine unverwechselbare Einheit (ein Benutzer oder ein Dienst), der ein Ticket zugewiesen werden kann. Ein Principal setzt sich aus den folgenden Bestandteilen zusammen:

- **Primary** – Der erste Teil des Principals, der im Falle eines Benutzers mit dem Benutzernamen identisch sein kann.
- **Instance** – Optionale Information, die den Primary beschreibt. Diese Zeichenkette ist durch ein / vom Primary getrennt.
- **Realm** – Der Realm legt Ihren Kerberos-Bereich fest. Normalerweise ist Ihr Realm Ihr Domainname in Großbuchstaben.

Mutual Authentication Kerberos sorgt dafür, dass sich sowohl der Client als auch der Server über die Identität der jeweiligen Gegenpartei sicher sein können. Sie teilen sich einen Session Key, mit dem sie sicher kommunizieren können.

Session Key Session Keys (Sitzungsschlüssel) sind temporäre private Schlüssel, die von Kerberos generiert werden. Sie sind dem Client bekannt und werden zur Verschlüsselung der Kommunikation zwischen dem Client und dem Server benutzt, von dem der Client ein Ticket angefordert und bekommen hat.

Replay Fast alle Nachrichten, die in einem Netzwerk versendet werden, können abgehört, entwendet und erneut versendet werden. Im Zusammenhang mit Kerberos könnte dies äußerst gefährlich sein, falls es einem Angreifer gelingen sollte, Ihre Anforderung für einen Dienst abzufangen, die Ihr Ticket und Ihren Authenticator enthält. Er könnte daraufhin versuchen, sie erneut zu versenden („Replay“) und sich als Sie ausgeben. Allerdings implementiert Kerberos verschiedene Mechanismen, um diesem Problem vorzubeugen.

Server oder Service „Service“ (Dienst) wird benutzt, wenn eine bestimmte Aktion durchgeführt werden soll. Der zugrunde liegende Prozess wird als „Server“ bezeichnet.

19.3.2 Wie funktioniert es?

Kerberos wird oft als „Trusted Third Party“-Authentifizierungsdienst bezeichnet. Das heißt, dass sich alle Clients im Hinblick auf die Identität eines anderen Clients auf die Einschätzung von Kerberos verlassen. Kerberos unterhält eine Datenbank über alle Benutzer und ihre privaten Schlüssel.

Um sicherzustellen, dass Kerberos das in ihn gesetzte Vertrauen auch wirklich verdient, müssen Authentifizierungsserver und Ticket-Granting-Server auf einer dedizierten Maschine laufen. Sorgen Sie dafür, dass nur der Administrator physisch und über das Netzwerk Zugang zu dieser Maschine hat und beschränken Sie die (Netzwerk-)Dienste, die auf diesem Server laufen, auf das absolute Minimum — lassen Sie nicht einmal `sshd` laufen.

Erste Kontaktaufnahme Ihr erster Kontakt mit Kerberos ähnelt dem gewöhnlichen Einloggen an einem normalen Netzwerksystem. Geben Sie Ihren Benutzernamen ein. Diese Information und der Name des Ticket-Granting Services werden dem Authentifizierungsserver (Kerberos) zugesendet. Falls der Authentifizierungsserver von Ihrer Existenz weiß, generiert er einen (zufälligen) Session Key für den weiteren Gebrauch zwischen Ihrem Client und dem Ticket-Granting Server. Nun wird der Authentifizierungsserver ein Ticket für den Ticket-Granting Server erstellen. Das Ticket enthält die folgenden Informationen, die alle mit einem Session Key verschlüsselt sind, den nur der Authentifizierungsserver und der Ticket-Granting Server kennen:

- die Namen des Clients und des Ticket-Granting Servers
- die aktuelle Uhrzeit
- die Lebensdauer, die diesem Ticket zugewiesen wurde
- die IP-Adresse des Clients
- den neu generierten Session Key

Dann wird das Ticket zusammen mit dem Session Key nochmals in verschlüsselter Form dem Client zurückgesendet, jedoch unter Benutzung des privaten Schlüssels des Clients. Dieser private Schlüssel ist nur Kerberos und dem Client bekannt, da er von Ihrem Benutzerpasswort abgeleitet ist. Sobald der Client diese Antwort erhält, werden Sie nach Ihrem Passwort gefragt. Dieses Passwort wird in den Schlüssel konvertiert, welcher das vom Authentifizierungsserver gesendete Paket entschlüsseln kann. Das Paket wird entpackt und das

Passwort und der Schlüssel werden aus dem Arbeitsspeicher des Arbeitsplatzrechners gelöscht. Ihr Arbeitsplatzrechner kann Ihre Identität nachweisen, bis die Lebensdauer des Ticket-Granting Tickets erlischt.

Anforderung eines Dienstes Um von einem beliebigen Server im Netzwerk einen Dienst anfordern zu können, muss die Client-Anwendung dem Server ihre Identität nachweisen. Daher generiert die Anwendung einen so genannten „Authenticator“. Dieser setzt sich aus den folgenden Bestandteilen zusammen:

- dem Principal des Clients
- der IP-Adresse des Clients
- der aktuellen Uhrzeit
- einer Prüfsumme (bestimmt durch den Client)

Alle diese Informationen werden mit dem Session Key, den der Client bereits für diesen speziellen Server empfangen hat, verschlüsselt. Der Authenticator und das Ticket für den Server werden an den Server gesendet. Der Server benutzt seine Kopie des Session Keys, um den Authenticator zu entschlüsseln, der ihm sämtliche benötigte Informationen über den Client liefert, der seinen Dienst anfordert. Diese Informationen können mit denen verglichen werden, die im Ticket enthalten sind. So prüft der Server, ob Ticket und Authenticator vom gleichen Client stammen.

Gäbe es auf der Serverseite keine Sicherheitsmaßnahmen, so wäre diese Stufe das ideale Ziel für Replay-Attacken. Jemand mit schlechten Absichten könnte versuchen, eine vorher aus dem Netz gestohlene Anforderung erneut zu versenden. Um dies zu verhindern, nimmt der Server keine Anforderungen an, die mit einem Zeitstempel und einem Ticket versehen sind, die ihm schon vorher zugesendet worden waren. Außerdem können Anforderungen abgelehnt werden, deren Zeitstempel in Bezug auf den Zeitpunkt, an dem die Anforderung empfangen wurde, zu sehr abweichen (in die Zukunft und in die Vergangenheit).

Gegenseitige Authentifizierung Die Kerberos-Authentifizierung kann in beide Richtungen benutzt werden. Es geht nicht nur darum, ob der Client wirklich derjenige ist, der er zu sein vorgibt; auch der Server sollte in der Lage sein, sich gegenüber dem Client zu authentifizieren, der seinen Dienst anfordert. Daher versendet er selber auch eine Art Authenticator. Er addiert der Prüfsumme, die er im Authenticator des

Clients erhalten hat, eins hinzu und verschlüsselt sie mit dem Session Key, den er mit dem Client teilt. Der Client betrachtet diese Antwort als Nachweis für die Echtheit des Servers, wonach die Zusammenarbeit zwischen dem Client und dem Server beginnen kann.

Ticket-Granting — Kontaktaufnahme mit allen Servern

Tickets sind für den Gebrauch für jeweils einen Server bestimmt. Das bedeutet, dass Sie ein neues Ticket brauchen, sobald Sie einen anderen Dienst anfordern. Kerberos implementiert einen Mechanismus zur Beschaffung von Tickets für einzelne Server. Dieser Dienst wird als „Ticket-Granting Service“ (Dienst zur Ausstellung von Tickets) bezeichnet. Der Ticket-Granting Service ist ein Dienst wie jeder andere und unterliegt daher den gleichen Zugriffsprotokollen, die bereits erwähnt wurden. Jedes Mal, wenn eine Anwendung ein Ticket benötigt, das noch nicht angefordert wurde, nimmt sie mit dem Ticket-Granting Server Kontakt auf. Diese Anforderung setzt sich aus den folgenden Bestandteilen zusammen:

- dem angeforderten Principal
- dem Ticket-Granting Ticket
- dem Authenticator

Ähnlich wie bei jedem anderen Server überprüft der Ticket-Granting Server das Ticket-Granting Ticket sowie den Authenticator. Falls sie als gültig anerkannt werden, erstellt der Ticket-Granting Server einen neuen Session Key zur Benutzung durch den ursprünglichen Client und den neuen Server. Dann wird das Ticket für den neuen Server mit den folgenden Informationen erstellt:

- dem Principal des Clients
- dem Principal des Servers
- der aktuellen Uhrzeit
- der IP-Adresse des Clients
- dem neu generierten Session Key

Dem neuen Ticket wird eine Lebensdauer zugewiesen, die der verbleibenden Lebensdauer des Ticket-Granting Tickets oder dem Standardwert für den Dienst entspricht, je nachdem, was kürzer ist. Dieses Ticket und der Session Key werden dem Client vom Ticket-Granting Service zugesendet. Dieses Mal ist die Antwort jedoch mit

dem Session Key verschlüsselt, der mit dem ursprünglichen Ticket-Granting Ticket empfangen wurde. Wenn ein neuer Dienst angefordert wird, kann der Client nun die Antwort entschlüsseln, ohne das Benutzerpasswort erneut anzufordern. So kann Kerberos für den Client ein Ticket nach dem anderen erlangen, ohne den Benutzer mehr als einmal beim Login zu belästigen.

Kompatibilität mit Windows 2000 Windows 2000 enthält eine Microsoft-Implementierung von Kerberos 5. Da SUSE LINUX die Heimdal-Implementierung von Kerberos 5 benutzt, werden Sie in der Heimdal-Dokumentation bestimmt einige nützliche Informationen und Anleitungen finden; siehe Abschnitt 19.3.4 auf der nächsten Seite.

19.3.3 Auswirkungen von Kerberos für den Benutzer

Im Idealfall kommt ein Benutzer ausschließlich beim Login an seinem Arbeitsplatzrechner mit Kerberos in Kontakt. Beim Einloggen wird ein Ticket-Granting Ticket erlangt. Beim Ausloggen werden die Kerberos-Tickets des Benutzers automatisch vernichtet, wodurch verhindert wird, dass sich ein anderer Benutzer als dieser spezielle Benutzer ausgibt, wenn dieser nicht eingeloggt ist. Die automatische Vernichtung von Tickets führt zu einer schwierigen Situation, wenn die Sitzung des Benutzers länger dauert als die Höchstlebensdauer, die dem Ticket-Granting Ticket zugewiesen wird (10 Stunden ist ein vernünftiger Wert). Der Benutzer kann sich jedoch ein neues Ticket-Granting Ticket besorgen, indem er kinit startet. Er braucht nur sein Passwort erneut einzugeben — Kerberos wird dafür sorgen, dass er zu jedem gewünschten Dienst Zugang hat, ohne nochmals eine Authentifizierung zu verlangen. Diejenigen, die an einer Liste aller Tickets interessiert sind, die durch Kerberos im Hintergrund für sie erworben wurden, können diese mit klist abrufen.

Es folgt eine Auswahl von Anwendungen, die sich die Kerberos-Authentifizierung zunutze machen. Diese Anwendungen befinden sich unter `/usr/lib/heimdal/bin/`. Sie alle bieten die volle Funktionalität ihrer gewöhnlichen UNIX/Linux-Geschwister sowie den zusätzlichen Vorteil einer transparenten Authentifizierung mit Hilfe von Kerberos:

- telnet/telnetd
- rlogin
- rsh, rcp, rshd

- popper/push
- ftp/ftpd
- su
- imapd
- pine

Wie Sie sehen werden, brauchen Sie Ihr Passwort nicht einzugeben, um diese Anwendungen benutzen zu können, da Kerberos Ihre Identität bereits nachgewiesen hat. `ssh` — sofern mit Kerberos-Unterstützung kompiliert — kann sogar alle Tickets, die Sie für einen Arbeitsplatzrechner erworben haben, an einen anderen Arbeitsplatz weiterleiten. Wenn Sie `ssh` benutzen, um sich auf einem anderen Arbeitsplatzrechner einzuloggen, sorgt `ssh` dafür, dass die verschlüsselten Inhalte der Tickets der neuen Situation angepasst werden. Es ist nicht ausreichend, die Tickets einfach von einem Arbeitsplatzrechner auf einen anderen zu kopieren, da das Ticket spezifische Information über den Arbeitsplatzrechner enthält (die IP-Adresse). XDM und KDM bieten ebenfalls Kerberos-Unterstützung. Lesen Sie im *Kerberos V5 UNIX User's Guide* unter <http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-user.html> mehr über die Kerberos-Netzwerkanwendungen.

19.3.4 Weitere Informationen über Kerberos

SUSE LINUX enthält eine freie Implementierung von Kerberos, die als Heimdal bezeichnet wird. Die entsprechende Dokumentation wird zusammen mit dem Paket `heimdal` unter `/usr/share/doc/packages/heimdal/doc/heimdal.info` installiert. Die Dokumentation ist auch auf der Internetseite des Projekts unter <http://www.pdc.kth.se/heimdal/> erhältlich.

Auf der offiziellen Website der Kerberos-Implementierung des MIT finden Sie Links zu anderen relevanten Ressourcen im Zusammenhang mit Kerberos: <http://web.mit.edu/kerberos/www/>

Ein „klassischer“ Dialog, der die Arbeitsweise von Kerberos erläutert. Nicht allzu technisch, aber trotzdem hochinteressant: <http://web.mit.edu/kerberos/www/dialogue.html>

Dieses Papier vermittelt ein umfangreiches Verständnis über die grundlegende Arbeitsweise von Kerberos, ist jedoch nicht übermäßig schwer

zu verstehen. Es bietet außerdem eine Menge Möglichkeiten für weitere Nachforschungen zu Kerberos: <ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS>

Diese Links bieten eine kurze Einführung in Kerberos sowie Antworten auf viele Fragen im Zusammenhang mit der Installation, Konfiguration und Administration von Kerberos: <http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-user.html> <http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-install.html> <http://web.mit.edu/kerberos/www/krb5-1.3/krb5-1.3/doc/krb5-admin.html>

Das offizielle Kerberos-FAQ: <http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html>

Tung, Brian: *Kerberos — A Network Authentication System*. Addison Wesley, 1999. - (ISBN 0-201-37924-4)

19.4 Installation und Administration von Kerberos

Dieser Abschnitt erläutert die Installation des Heimdal Kerberos sowie einige Aspekte der Administration. Es wird vorausgesetzt, dass Sie mit den Grundlagen von Kerberos vertraut sind (siehe auch Abschnitt 19.3 auf Seite 535).

19.4.1 Festlegung der Kerberos-Realms

Die „Domain“ einer Kerberos-Installation wird als Realm bezeichnet und hat einen Namen wie `FOOBAR.COM` oder einfach nur `ACCOUNTING`. Da Kerberos Groß-/Kleinbuchstaben unterscheidet, ist `foobar.com` ein anderer Realm als `FOOBAR.COM`. Die Wahl von Groß-/Kleinbuchstaben ist Ihnen überlassen. Es ist jedoch üblich, für Realm-Namen Großbuchstaben zu benutzen.

Es ist empfehlenswert, Ihren DNS-Domainnamen (oder eine Subdomain wie `ACCOUNTING.FOOBAR.COM`) zu benutzen. Wie Sie später sehen werden, haben Sie es als Administrator viel leichter, wenn Sie Ihre Kerberos-Clients so konfigurieren, dass das KDC und andere Kerberos-Dienste via DNS ansprechbar sind. Um dies zu ermöglichen, ist es sinnvoll, wenn der Realm-Name eine Subdomain Ihres DNS-Domainnamens ist.

Im Gegensatz zum DNS-Namensraum ist Kerberos nicht hierarchisch gegliedert. Sie können nicht einen Realm namens `FOOBAR.COM` aufsetzen, darunter zwei „Subrealms“ namens `DEVELOPMENT` und `ACCOUNTING` erstellen und erwarten, dass die beiden untergeordneten Realms irgendwie Principals von `FOOBAR.COM` übernehmen. Stattdessen hätten Sie drei getrennte Realms, für die Sie „Crossrealm“-Authentifizierung konfigurieren müssten, um Benutzern eines Realms zu ermöglichen, mit Servern oder Benutzern eines anderen Realms zu interagieren. Die Einrichtung der Crossrealm-Authentifizierung wird beispielsweise in [19] beschrieben.

Der Einfachheit halber nehmen wir an, dass Sie für Ihre gesamte Organisation nur einen Realm anlegen. Im restlichen Teil dieses Abschnittes wird der Realm-Name `SAMPLE.COM` für alle Beispiele benutzt.

19.4.2 Einrichtung der KDC-Hardware

Wenn Sie Kerberos benutzen möchten, brauchen Sie zunächst einen Rechner, der als Key Distribution Center (KDC) eingesetzt wird. Auf diesem Rechner befindet sich die gesamte Kerberos-Benutzerdatenbank mit den Passwörtern und allen Informationen.

Das KDC ist der wichtigste Teil Ihrer Sicherheitsinfrastruktur — wenn jemand hier eindringt, sind alle Benutzerkonten und die gesamte Infrastruktur, die durch Kerberos geschützt wird, offengelegt. Ein Angreifer, der Zugang zur Kerberos-Datenbank hat, kann ein beliebiges Principal in der Datenbank verkörpern! Sorgen Sie dafür, dass die Sicherheitsvorkehrungen für diesen Rechner so strikt wie möglich sind:

- Stellen Sie den Server an einem physikalisch sicheren Standort auf, zum Beispiel in einem abgeschlossenen Serverraum, zu dem nur ein begrenzter Personenkreis Zugang hat.
- Lassen Sie außer dem KDC keine anderen Netzwerkanwendungen auf dem Rechner laufen. Dies gilt sowohl für Server- als auch für Clientanwendungen. Das KDC sollte beispielsweise keine Dateisysteme über NFS importieren oder DHCP benutzen, um seine Netzwerkkonfiguration abzurufen.

Ein guter Ansatz wäre, zunächst nur ein Minimalsystem zu installieren und dann die Liste aller installierten Pakete zu überprüfen und eventuelle unnötige Pakete zu löschen. Dies schließt Server wie `inetd`, `portmap` und `cups` sowie alles ein, was mit X11 zu tun hat. Selbst die Installation eines SSH-Servers stellt ein potientiellles Sicherheitsrisiko dar.

Auf diesem Rechner gibt es kein grafisches Login, da auch ein X-Server ein potentielles Sicherheitsrisiko darstellt. Kerberos hat jedoch ein eigenes Administrationsinterface.

- Konfigurieren Sie `/etc/nsswitch.conf` so, dass nur in lokalen Dateien nach Benutzern und Gruppen gesucht wird. Ändern Sie die Zeilen für `passwd` und `group` wie folgt:

```
passwd:      files
group:       files
```

Editieren Sie die Dateien `passwd`, `group`, `shadow` und in `/etc/` und entfernen Sie die Zeilen, die mit einem Pluszeichen anfangen (diese werden für NIS-Anfragen benutzt).

Sie sollten sich auch überlegen, DNS-Anfragen zu deaktivieren, da dies einen Risikofaktor darstellt. Falls in der DNS Resolver Library eine Sicherheitslücke ist, könnte ein Angreifer das KDC überlisten, eine DNS-Anfrage durchzuführen, die diese Lücke ausnutzt. Um DNS-Anfragen zu deaktivieren, löschen Sie einfach `/etc/resolv.conf`.

- Deaktivieren Sie alle Benutzerkonten außer dem von Root, indem Sie `/etc/shadow` editieren und die gehashten Passwörter durch Sternchen oder Ausrufezeichen ersetzen.

19.4.3 Zeitsynchronisation

Um Kerberos erfolgreich einsetzen zu können, müssen alle Systemuhren in Ihrer Organisation in einem bestimmten Bereich synchronisiert werden. Der Grund hierfür ist, dass Kerberos versuchen wird, Sie vor erneut versendeten Credentials (Replay) zu schützen. Es könnte einem Angreifer gelingen, Kerberos-Credentials im Netzwerk zu beobachten und diese zu benutzen, um den Server anzugreifen. Kerberos setzt verschiedene Verteidigungsmechanismen ein, um dies zu verhindern. Einer dieser Mechanismen sieht vor, dass die Tickets mit Zeitstempeln versehen werden. Ein Server, der ein Ticket mit einem nicht aktuellen Zeitstempel erhält, wird das Ticket zurückweisen.

Natürlich erlaubt Kerberos beim Vergleichen von Zeitstempeln einen gewissen Spielraum. Computeruhren können jedoch äußerst ungenau sein — es ist nicht ungewöhnlich, dass PC-Uhren im Laufe einer Woche eine halbe Stunde vor- oder zurückgehen. Sie sollten daher alle Hosts im Netzwerk so konfigurieren, dass ihre Uhren mit einer zentralen Zeitquelle synchronisiert werden.

Sie können dies sehr einfach bewerkstelligen, indem Sie auf einem Rechner einen NTP-Zeitserver installieren und alle Clients ihre Uhren mit diesem Server synchronisieren lassen. Dies kann erreicht werden, indem Sie einen NTP-Daemon im Client-Modus auf allen Rechnern laufen lassen oder `ntpdate` einmal am Tag von allen Clients ausführen lassen (diese Lösung funktioniert wahrscheinlich nur bei einer kleineren Anzahl von Clients).

Das KDC selber muss auch mit der gemeinsamen Zeitquelle synchronisiert werden. Da ein NTP-Daemon auf diesem Rechner ein Sicherheitsrisiko darstellen würde, ist es wahrscheinlich das Beste, `ntpdate` via einen Croneintrag auszuführen.

Eine Beschreibung der Konfiguration von NTP finden Sie im Abschnitt 14.11 auf Seite 419. Weiterführende Information ist in der NTP-Dokumentation auf Ihrem installierten System unter `/usr/share/doc/packages/xntp-doc` erhältlich.

Selbstverständlich können Sie die maximale Abweichung, die Kerberos bei der Überprüfung von *time stamps* toleriert, nach eigenen Vorstellungen anpassen. Dieser Wert (`clock skew`) wird in der Konfigurationsdatei `krb5.conf` verändert, wie unter Abschnitt 19.4.6 auf Seite 552 beschrieben.

19.4.4 Konfiguration der Protokollfunktion

Standardmäßig protokollieren die auf dem KDC-Host laufenden Kerberos-Daemons ihre Information zum `syslog`-Daemon. Falls Sie die Aktivitäten Ihres KDC beobachten möchten, ist es vielleicht nützlich, diese Protokolldateien regelmäßig zu verarbeiten und auf ungewöhnliche Ereignisse oder potentielle Probleme zu untersuchen.

Um dies zu erreichen, kann man auf dem KDC-Host ein Protokollscannerskript laufen lassen oder diese Protokolldaten via `rsync` vom KDC auf einen anderen Host kopieren und die Protokollanalyse dort durchführen. Es wird davon abgeraten, die gesamte Protokollausgabe über die Weiterleitungsfunktion von `syslogd` weiterzuleiten, da die Information in unverschlüsselter Form im Netzwerk übertragen wird.

19.4.5 Installation des KDC

Dieser Abschnitt erläutert die Erstinstallation des KDC, einschließlich der Einrichtung eines administrativen Principals.

Installation der RPMs

Bevor Sie anfangen können, müssen Sie die Kerberos-Software installieren. Installieren Sie die RPMs `heimdal`, `heimdal-lib` und `heimdal-tools` auf dem KDC:

```
rpm -ivh heimdal-*.rpm heimdal-lib-*.rpm heimdal-tools*.rpm
```

Setzen des Master Keys

Der nächste Schritt ist die Initialisierung der Datenbank, in der Kerberos sämtliche Informationen über die Principals speichert. Zuerst muss der Master Key der Datenbank gesetzt werden, der benötigt wird, um die Datenbank vor unbeabsichtigter Offenlegung zu schützen, besonders wenn diese auf ein Band gesichert wird.

Der Master Key wird aus einer Passphrase generiert und in einer Datei gespeichert, die als Stash File bezeichnet wird. Daher brauchen Sie nicht jedes Mal, wenn das KDC neu gestartet wird, das Passwort einzugeben. Wählen Sie eine gute Passphrase, beispielsweise einen Satz aus einem Buch, das Sie an einer zufälligen Stelle aufschlagen.

Wenn Sie die Kerberos-Datenbank auf Band sichern (`/var/heimdal/heimdal.db`), sichern Sie bitte nicht die stash Datei (in `/var/heimdal/m-key`). Ansonsten könnte jeder, der das Band lesen kann, die Datenbank entschlüsseln. Aus diesem Grunde ist es empfehlenswert, eine Kopie der Passphrase in einem Safe oder an einem anderen sicheren Ort aufzubewahren, da Sie diese benötigen, wenn Sie nach einem Absturz Ihre Datenbank von Band wiederherstellen.

Um den Master Key zu setzen, starten Sie die Utility `kstash` ohne zusätzliche Argumente und geben die Passphrase zweimal ein:

```
kstash
```

```
Master key:<enter pass phrase>
```

```
Verifying password - Master key:<enter pass phrase again>
```

Anlegen des Realms

Zuletzt müssen die Einträge für Ihren Realm in der Kerberos-Datenbank erstellt werden. Starten Sie die Utility `kadmin` mit der Option `-l`. Diese Option veranlasst `kadmin`, auf die lokale Datenbank zuzugreifen. Standardmäßig versucht `kadmin`, den Kerberos-Administrationsdienst über das Netzwerk zu erreichen. In diesem Stadium würde dies nicht funktionieren, da dieser Dienst noch nicht läuft.

Nun weisen Sie `kadmin` an, Ihren Realm zu initialisieren. `kadmin` wird eine Reihe von Fragen stellen. Zunächst ist es das Beste, die von `kadmin` angebotenen Standardeinstellungen anzunehmen:

```
kadmin -l

kadmin> init SAMPLE.COM
Realm max ticket life [unlimited]: <press return>
Realm max renewable ticket life [unlimited]: <press return>
```

Um zu prüfen, ob etwas geschehen ist, benutzen Sie den Befehl `list`:

```
kadmin> list *

default@SAMPLE.COM
kadmin/admin@SAMPLE.COM
kadmin/hprop@SAMPLE.COM
kadmin/changepw@SAMPLE.COM
krbtgt/SAMPLE.COM@SAMPLE.COM
changepw/kerberos@SAMPLE.COM
```

Dies zeigt, dass es jetzt in der Datenbank eine Reihe von Principals gibt, die alle für den internen Gebrauch durch Kerberos bestimmt sind.

Erstellung eines Principals

Nun schaffen Sie zwei Kerberos-Principals für sich selbst — ein „normales“ Principal für Ihre tägliche Arbeit und eines für administrative Aufgaben im Zusammenhang mit Kerberos. Verfahren Sie wie folgt, um den Login-Namen `newbie` einzurichten:

```
kadmin -l

kadmin> add newbie
Max ticket life [1 day]: <press return>
Max renewable life [1 week]: <press return>
Principal expiration time [never]: <press return>
Password expiration time [never]: <press return>
Attributes [: <press return>
newbie@SAMPLE.COM's Password: <type password here>
Verifying password: <re-type password here>
```


Sie können die Standardwerte mit `(Enter)` bestätigen. Wählen Sie ein geeignetes Passwort.

Danach erstellen Sie ein anderes Principal namens `newbie/admin` durch Eingabe von `add newbie/admin` am `kadmin`-Prompt. Der Suffix `admin` hinter dem Benutzernamen bezeichnet die „Rolle“ *role*. Später werden Sie diese administrative Rolle benutzen, um die Kerberos-Datenbank zu administrieren.

Ein Benutzer kann mehrere „Rollen“ haben, die unterschiedlichen Zwecken dienen. Ihre Handhabung hat dennoch nichts mit Magie zu tun — sehen Sie sie einfach als völlig unterschiedliche Accounts mit ähnlichen Namen an.

Starten des KDC

Starten Sie die KDC-Daemons. Dies schließt den eigentlichen `kdc` (der Daemon, der für die Benutzerauthentifizierung und Ticketanfragen zuständig ist), `kadmind` (der Server für die Fernadministration) sowie `kpasswd` (zuständig für Passwortänderungsanfragen von Benutzern) ein. Um den Daemon manuell zu starten, geben Sie Folgendes ein:

```
rckdc start
Starting kdc                               done
```

Sorgen Sie dafür, dass das KDC standardmäßig gestartet wird, wenn der Server neu gestartet wird. Dies wird mit Hilfe des Befehls `insserv kdc` bewerkstelligt.

19.4.6 Konfiguration von Kerberos-Clients

Die Konfiguration von Kerberos kann grundsätzlich auf zweierlei Weise erfolgen — über eine statische Konfiguration mit der Datei `/etc/krb5.conf` oder über eine dynamische Konfiguration via DNS. Bei der DNS-Konfiguration versuchen Kerberos-Anwendungen, die KDC-Dienste durch DNS-Einträge zu finden. Bei der statischen Konfiguration müssen Sie die Hostnamen Ihres KDC-Servers in der Datei `krb5.conf` eintragen (und die Datei aktualisieren, wenn das KDC „umzieht“ oder Sie Ihren Realm in irgendeiner anderen Weise neu konfigurieren).

Die DNS-basierte Konfiguration ist gewöhnlich viel flexibler und der Konfigurationsaufwand pro Rechner viel geringer. Dieser Ansatz erfordert jedoch, dass Ihr Realm-Name mit Ihrer DNS-Domain identisch ist oder eine Subdomain hiervon ist.

Außerdem verursacht die Konfiguration von Kerberos via DNS ein kleines Sicherheitsproblem, denn ein Angreifer kann Ihre Infrastruktur durch Ihren DNS erheblich stören (durch Abschuss des Nameservers, Verfälschung von DNS-Einträgen [Spoofing] usw.). Im schlimmsten Fall führt dies jedoch zu einem DoS. Ein ähnliches Szenario kann auch bei der statischen Konfiguration auftreten, es sei denn, Sie geben in `krb5.conf` IP-Adressen anstelle von Hostnamen ein.

Statische Konfiguration

Eine Art der Kerberos-Konfiguration ist der Änderung der Konfigurationsdatei `/etc/krb5.conf`. Die Datei, die standardmäßig im installierten System vorhanden ist, enthält einige Beispieleinträge. Entfernen Sie diese, bevor Sie mit Ihrer eigenen Konfiguration beginnen.

`krb5.conf` besteht aus mehreren Abschnitten. Jeder dieser Abschnitte beginnt mit dem Namen des Abschnitts in eckigen Klammern (`[Beispielname]`).

Für die statische Konfiguration fügen Sie bitte den folgenden Abschnitt in `krb5.conf` ein (wobei `kdc.sample.com` der Hostname des KDCs ist):

```
[libdefaults]
    default_realm = SAMPLE.COM

[realms]
    SAMPLE.COM = {
        kdc = kdc.sample.com
        kpasswd_server = kdc.sample.com
        admin_server = kdc.sample.com
    }
```

Über die Zeile `default_realm` wird die standardmäßige Realm für Kerberos-Applikationen festgelegt.

Falls Sie mehrere Realms haben, fügen Sie einfach dem Abschnitt `[realms]` einen weiteren Ausdruck hinzu.

Fügen Sie dieser Datei auch einen Ausdruck hinzu, der besagt, wie Anwendungen Hostnamen zu Realms zuordnen müssen. Wenn man beispielsweise eine Verbindung zu einem entfernten Host aufbaut, muss die Kerberos-Library wissen, in welchem Realm sich dieser Host befindet. Dies muss im Abschnitt `[domain_realms]` konfiguriert werden:

```
[domain_realm]
    .sample.com = SAMPLE.COM
    www.foobar.com = SAMPLE.COM
```

Dieser Eintrag teilt der Library mit, dass sich alle Hosts in den `sample.com` DNS-Domains in dem Kerberos-Realm `SAMPLE.COM` befinden. Außerdem sollte auch ein externer Host namens `www.foobar.com` als Angehöriger des Realms `SAMPLE.COM` betrachtet werden.

DNS-basierte Konfiguration

Die DNS-basierte Kerberos-Konfiguration macht intensiven Gebrauch von SRV-Einträgen (Siehe (RFC2052) *A DNS RR for specifying the location of services* unter <http://www.ietf.org>). Diese Einträge wurden in früheren Implementierungen des BIND-Nameservers noch nicht unterstützt. Deshalb ist BIND Version 8 (oder spätere Versionen) erforderlich.

Was Kerberos betrifft, ist der Name eines SRV-Eintrags immer wie folgt aufgebaut: `_service._proto.realm`, wobei `realm` der Kerberos-Realm ist. Beachten Sie, dass DNS bei Domainnamen keine Groß-/Kleinbuchstaben unterscheidet, so dass Kerberos-Realms, die Groß-/Kleinschreibung unterscheiden, bei dieser Konfigurationsmethode nicht mehr funktionieren würden. `_service` ist der Name eines Dienstes (verschiedene Namen werden benutzt, wenn beispielsweise eine Verbindung zum KDC oder zum Passwortdienst aufgebaut wird). `_proto` kann entweder `_udp` oder `_tcp` sein, aber nicht alle Dienste unterstützen beide Protokolle.

Der Datenteil der SRV Resource Records besteht aus einem Prioritätswert, einer Gewichtung, einer Portnummer und einem Hostnamen. Die Priorität definiert die Reihenfolge, in welcher Hosts versucht werden sollen (kleinere Werte stellen eine höhere Priorität dar). Die Gewichtung wird benutzt, um ein gewisses Load-Balancing zwischen Servern gleicher Priorität zu unterstützen. Diese Funktion wird kaum gebraucht, so dass Sie diese auf Null setzen können. Bei der Suche nach Diensten sucht Heimdal Kerberos zur Zeit nach den folgenden Namen:

`_kerberos` Definiert die Lokalisierung des KDC-Daemons (der Authentifizierungs- und Ticket-Granting-Server). Typischerweise sehen die Einträge wie folgt aus:

```
_kerberos._udp.SAMPLE.COM.  IN  SRV      0 0 88 kdc.sample.com.
_kerberos._tcp.SAMPLE.COM.  IN  SRV      0 0 88 kdc.sample.com.
```

`_kpasswd` Beschreibt die Lokalisierung des Servers für Passwortänderungen. Typischerweise sehen die Einträge wie folgt aus:

```
_kpasswd._udp.SAMPLE.COM.  IN  SRV      0 0 464 kdc.sample.com.
```

Dakpasswd TCP nicht unterstützt, sollte es keinen `_tcp` Eintrag geben.

`_kerberos-adm` Beschreibt die Lokalisierung des Fernadministrationsservers. Typischerweise sehen die Einträge wie folgt aus:

```
_kerberos-adm._tcp.SAMPLE.COM. IN SRV 0 0 749 kdc.sample.com.
```

Da `kadmind` UDP nicht unterstützt, sollte es keinen `_udp` Eintrag geben.

Wie bei der statischen Konfigurationsdatei gibt es einen Mechanismus, der Clients darüber informiert, dass ein bestimmter Host sich in dem Realm `SAMPLE.COM` befindet, selbst wenn er kein Teil der DNS-Domain `sample.com` ist. Dies kann erreicht werden, indem man `_kerberos.hostname` einen TXT-Eintrag hinzufügt:

```
_kerberos.www.foobar.com. IN TXT "SAMPLE.COM"
```

Anpassung der Uhrabweichung

Über die Variable `clock skew` setzen Sie die Toleranzgrenzen fest, innerhalb derer Tickets akzeptiert werden, deren Zeitstempel nicht exakt mit der Systemuhr des Hostsystems übereinstimmen.

Im Normalfall ist diese Größe mit 300 Sekunden (5 Minuten) angegeben. Ein Ticket kann also einen Zeitstempel tragen, der fünf Minuten in Vergangenheit oder Zukunft von der Systemzeit des Servers abweicht, um noch akzeptiert zu werden.

Verwenden Sie NTP zur Zeitsynchronisation auf allen Hosts, kann dieser Wert auf eine Minute reduziert werden.

Die `clock skew` Variable passen Sie in `/etc/krb5.conf` wie folgt an:

```
[libdefaults]
    clockskew = 120
```

19.4.7 Einrichtung der Fernadministration

Um der Kerberos-Datenbank Principals hinzufügen bzw. löschen zu können, ohne direkten Zugang zur Konsole des KDC zu haben, teilen Sie dem Kerberos-Adminserver mit, welche Principals hierzu berechtigt sind.

Sie können dies erreichen, indem Sie die Datei `/var/heimdal/kadmind.acl` editieren (ACL ist die Abkürzung von Access Control List). Die ACL-Datei ermöglicht eine Spezifizierung der Vorrechte und die Feineinstellung des Kontrollgrads. Nähere Informationen sind auf der Manpage (`man 8 kadmind`) erhältlich.

Erlauben Sie sich nun, mit der Datenbank alles zu tun, was Sie möchten, indem Sie der Datei die folgende Zeile hinzufügen:

```
newbie/admin all
```

Ersetzen Sie den Benutzernamen `newbie` mit Ihrem eigenen Benutzernamen. Starten Sie nun den KDC neu, um Ihre Änderungen aktiv werden zu lassen.

Fernadministration mittels kadmin

Die Fernadministration von Kerberos sollte nun mit Hilfe des Tools `kadmin` möglich sein. Zunächst brauchen Sie ein Ticket für Ihr Admin-Principal. Dieses Ticket wird gebraucht, wenn Sie eine Verbindung zum `kadmin`-Server herstellen:

```
kinit newbie/admin
```

```
newbie/admin@SAMPLE.COM's Password: <enter password>
```

```
/usr/sbin/kadmin
```

```
kadmin> privs
```

```
change-password, list, delete, modify, add, get
```

Mit dem Befehl `privs` können Sie überprüfen, welche Vorrechte Sie haben. Die obige Liste führt sämtliche Vorrechte auf.

Zum Beispiel können Sie das Principal `newbie` modifizieren:

```
kadmin> mod newbie
```

```
Max ticket life [1 day]:2 days
```

```
Max renewable life [1 week]:
```

```
Principal expiration time [never]:2005-01-01
```

```
Password expiration time [never]:
```

```
Attributes []:
```

Dies ändert die maximale Lebensdauer des Tickets auf zwei Tage und setzt das Auslaufdatum auf den 01.01.2005.

Grundlegende kadmin Kommandos

Hier folgt eine kurze Liste der wichtigsten kadmin Kommandos; bitte konsultieren Sie die Manualpage von kadmin für weitergehende Informationen.

add principal Fügt ein neues Principal hinzu.

modify principal Editieren verschiedener Attribute eines Principals, wie z. B. der maximalen Lebensdauer der Tickets und das Auslaufdatum des Kontos/Accounts.

delete principal Entfernt ein Principal aus der Datenbank.

rename principal neuername benennt ein Principal in neuername um.

list *<pattern>* Listet alle Principals auf, die dem angegebenen Pattern (Muster) entsprechen. Patterns funktionieren ähnlich wie die Shell Globbing Patterns: `list newbie*` würde in unserem Beispiel `newbie` und `newbie/admin` auflisten.

get *<principal>* Zeigt Detailinformation über das Principal an.

passwd *<principal>* Ändert das Passwort eines Principals.

Hilfe ist zu jeder Zeit durch Eingabe von `(?)` und `(Enter)` erhältlich, auch an Prompts, die von Befehlen wie `modify` und `add` ausgegeben werden.

Der Befehl `init`, der benutzt wird, wenn der Realm erstmals erstellt wird (sowie bei einigen anderen), ist im Remote-Modus nicht verfügbar. Um einen neuen Realm zu erstellen, gehen Sie an die Konsole des KDCs und benutzen kadmin im lokalen Modus (mit der Befehlszeilenoption `-l`).

19.4.8 Erstellung von Kerberos Host Principals

Jede Maschine innerhalb Ihres Netzwerks muss einem Kerberos Realm zugeordnet sein und ein KDC kontaktieren können. Außerdem müssen Sie für sie auch ein so genanntes „Host Principal“ anlegen.

Bis jetzt wurden nur Benutzer Credentials behandelt. „Kerberisierte“ Dienste müssen sich im Normalfall aber auch gegenüber dem Client-Benutzer authentifizieren. Hierzu werden so genannte „Host Principals“ für alle Hosts innerhalb eines Realms in der Kerberos Datenbank vorliegen.

Die entsprechende Namenskonvention lautet:

`host/<hostname>@<REALM>`, `hostname` ist hier der vollständige gültige Name des betreffenden Hosts.

Host Principals ähneln in vielerlei Hinsicht normalen Benutzer Principals. Allerdings gibt einige kleine Unterschiede. Der Hauptunterschied zwischen Benutzer Principal und Host Principal liegt darin, dass der Schlüssel des ersteren passwortgeschützt ist. Erhält ein Benutzer ein Ticket-Granting Ticket vom KDC, muss er sein Passwort eingeben, damit Kerberos das Ticket entschlüsseln kann. Für einen Systemadministrator wäre es folglich sehr unbequem, müsste er für den SSH Daemon alle acht Stunden neue Tickets anfordern.

Im Fall der Host Principals wird dieses Problem folgendermaßen gelöst: Der zur Entschlüsselung des ursprünglichen Tickets für den Host Principal erforderliche Schlüssel wird einmal vom Administrator vom KDC angefordert. Anschließend wird dieser Schlüssel in einer lokalen Datei namens `keytab` gespeichert. Dienste wie der SSH Daemon lesen diesen Schlüssel aus und verwenden ihn, um bei Bedarf automatisch neue Schlüssel zu erhalten. Die Standard `keytab` Datei liegt unter `/etc/krb5.keytab`.

Um einen Host Principal für `machine.sample.com` anzulegen, geben Sie während Ihrer `kadmin` Sitzung Folgendes ein:

```
kinit newbie/admin
```

```
newbie/admin@SAMPLE.COM's Password: <type password>
```

```
kadmin add -r host/machine.sample.com
Max ticket life [1 day]:
Max renewable life [1 week]:
Principal expiration time [never]:
Password expiration time [never]:
Attributes []:
```

Statt ein Passwort für das neue Principal zu setzen, weist die Option `-r` `kadmin` an, einen zufälligen Schlüssel zu generieren. Dies ist hier möglich, da wir für diesen Principal keine Benutzeraktivität wünschen. Es ist ein reiner Serveraccount für diese Maschine.

Abschließend extrahieren Sie den Schlüssel und speichern ihn in der lokalen Keytab-Datei `/etc/krb5.keytab`. Diese Datei gehört dem Superuser, weshalb Sie Root sein müssen, um den folgenden Befehl ausführen zu können:

```
ktutil get host/machine.sample.com
```

Danach sorgen Sie bitte dafür, dass Sie das Admin-Ticket mit `kdestroy` vernichten, das Sie via `krinit` erhalten haben, wie oben beschrieben.

19.4.9 Aktivierung der PAM-Unterstützung für Kerberos

SUSE LINUX wird mit einem PAM-Modul namens `pam_krb5` ausgeliefert, welches die Anmeldung via Kerberos und die Passwortaktualisierung unterstützt. Dieses Modul kann von Anwendungen wie dem Konsole-Login, `su` und grafischen Anwendungen wie KDM gebraucht werden, in denen der Benutzer ein Passwort eingibt und den Authentifizierungsmechanismus benutzen möchte, um ein erstes Kerberos-Ticket zu erhalten.

Ab dieser Version von SUSE LINUX unterstützt das `pam_unix`-Modul Kerberos-Authentifizierung und Passwortänderungen. Um die Kerberos-Unterstützung in `pam_unix` zu aktivieren, ändern Sie die Datei `/etc/security/pam_unix2.conf` wie folgt:

```
auth:          use_krb5 nullok
account:       use_krb5
password:      use_krb5 nullok
session:       none
```

Wenn diese Datei ausgewertet wird, verwenden alle Dienste Kerberos zur Benutzerauthentifizierung. Falls ein Benutzer keinen Kerberos-Principal besitzt, wird `pam_unix` auf den normalen Passwort-Authentifizierungsmechanismus zurückgreifen. Das Kerberos-Passwort sollte nun auch transparent mit dem `passwd` Kommando aktualisierbar sein.

Feineinstellungen von `pam_krb5` nehmen Sie über Änderungen der Datei `/etc/krb5.conf` vor und indem Sie Standardapplikationen für `pam` hinzufügen. Details zur Vorgehensweise entnehmen Sie bitte der Manualpage (`man 5 pam_krb5`).

Das Modul `pam_krb5` war ursprünglich **nicht** für Netzwerkdienste bestimmt, die Kerberos-Tickets als Teil der Benutzerauthentifizierung annehmen — dies ist eine ganz andere Geschichte und wird in den folgenden Abschnitten behandelt.

19.4.10 Konfiguration von SSH für Kerberos-Authentifizierung

OpenSSH unterstützt Kerberos-Authentifizierung sowohl in der Protokollversion 1 und 2. Version 1 verwendet eine bestimmte Art von Protokollmeldungen zur Übermittlung von Kerberos Tickets. Version 2 verwendet

Kerberos nicht mehr direkt, sondern greift auf „GSSAPI“, die so genannte *General Security Services API*, zurück. Diese Programmierschnittstelle ist nicht Kerberos spezifisch. Sie wurde entwickelt, um die Eigenheiten des zugrunde liegenden Authentifizierungssystems vor der Anwendung zu verbergen, egal ob dies Kerberos, SPKM oder ein anderes derartiges System ist. Allerdings unterstützt die aktuelle GSSAPI Bibliothek von SUSE LINUX zur Zeit nur Kerberos.

Um `sshd` mit der Kerberos-Authentifizierung zu benutzen, editieren Sie `/etc/ssh/sshd_config` und setzen die folgenden Optionen:

```
# These are for protocol version 1
KerberosAuthentication yes
KerberosTgtPassing yes
# These are for version 2
GSSAPIAuthentication yes
GSSAPIKeyExchange yes
```

Im Anschluss daran benutzen Sie den Befehl `rcsshd restart`, um Ihren SSH-Daemon neu zu starten.

Wollen Sie Kerberos Authentifizierung mit der Protokollversion 2 nutzen, müssen Sie auch auf der Clientseite die Unterstützung hierfür aktivieren. Entweder Sie erledigen dies systemweit über die Konfigurationsdatei `/etc/ssh/ssh_config` oder auf Benutzerbasis über `~/.ssh/config`. In beiden Fällen müssen Sie der Konfigurationsdatei die Option `GSSAPIAuthentication yes` hinzufügen.

Nun sollten Sie in der Lage sein, eine Verbindung mit Kerberos-Authentifizierung aufzubauen. Verwenden Sie `klist`, um zu überprüfen, ob Sie ein gültiges Ticket für die Verbindungsaufnahme mit dem SSH-Server besitzen. Um die Verwendung des SSH-Protokolls Version 1 zu erzwingen, übergeben Sie die Option `-1` auf der Kommandozeile.

```
ssh earth.sample.com
```

```
Last login: Fri Aug  9 14:12:50 2002 from zamboni.sample.com
Have a lot of fun...
```

19.4.11 Benutzung von LDAP und Kerberos

Bei Benutzung von Kerberos bietet sich mit LDAP ein Weg zur Verteilung von Benutzerinformationen (Benutzer-ID, Gruppen, Homeverzeichnisse etc.) im lokalen Netz. Natürlich erfordert dies die Verwendung eines starken Verschlüsselungsmechanismus, um Package-Spoofing etc. zu vermeiden.

Für die LDAP-Kommunikation lässt sich natürlich auch Kerberos verwenden.

OpenLDAP implementiert die meisten der verschiedenen Authentifizierungstypen über SASL, das *Simple Authentication Session Layer*. SASL ist im Grunde ein Netzwerkprotokoll zur Authentifizierung. SUSE LINUX verwendet die cyrus-sasl Implementierung und unterstützt mehrere Authentifizierungstypen. Kerberos Authentifizierung wird über GSSAPI (General Security Services API) umgesetzt.

Standardmäßig ist das SASL-Plugin für GSSAPI nicht installiert. Installieren Sie es von Hand nach:

```
rpm -ivh cyrus-sasl-gssapi-*.rpm
```

Um Kerberos Binding zum OpenLDAP-Server zu ermöglichen, legen Sie einen Principal `ldap/earth.sample.com` an und fügen Sie ihn der `keytab` hinzu:

```
kadmin add -r ldap/earth.sample.com  
ktutil get ldap/earth.sample.com
```

An dieser Stelle sollten Sie sich über folgenden Stolperstein klar werden: Der LDAP-Server (`slapd`) läuft standardmäßig unter Benutzer und Gruppe `ldap`, während `keytab` nur vom Benutzer `root` gelesen werden kann. Also müssen Sie entweder die LDAP-Konfiguration dahingehend abändern, dass der Server als Benutzer `root` gestartet wird oder die `keytab` für die Gruppe `ldap` lesbar machen.

Um `slapd` als `root` zu betreiben, editieren Sie die Datei `/etc/sysconfig/openldap` und deaktivieren die beiden Variablen `OPENLDAP_USER` und `OPENLDAP_GROUP` durch Einfügen eines Kommentarzeichens am Beginn der Zeile.

Um eine `keytab` Datei für die Gruppe `ldap` lesbar zu machen, gehen Sie folgendermaßen vor:

```
chgrp ldap /etc/krb5.keytab  
chmod 640 /etc/krb5.keytab
```

Keine dieser beiden Lösungen ist perfekt. Allerdings ist es momentan nicht möglich, OpenLDAP so zu konfigurieren, dass es eine eigene `keytab` verwendet.

Abschließend starten Sie den LDAP Server mit dem Befehl `rcldap restart neu`.

Kerberos-Authentifizierung mit LDAP

Nun sollten Sie in der Lage sein, Anwendungen wie `ldapsearch` automatisch mit Kerberos Authentifizierung auszuführen.

```
ldapsearch -b ou=People,dc=suse,dc=de '(uid=newbie)'
```

```
SASL/GSSAPI authentication started SASL SSF: 56
SASL installing layers
[...]
```

```
# newbie, People, suse.de
dn:uid=newbie,ou=People,dc=suse,dc=de
uid: newbie
cn: Olaf Kirch
[...]
```

Wie Sie hier sehen, gibt `ldapsearch` eine Meldung aus, dass es GSSAPI Authentifizierung gestartet hat. Die folgende Meldung ist zugegebenermaßen etwas kryptisch — sie gibt den „Security Strenth Factor“ (SSF) mit 56 an. (Der Wert 56 an dieser Stelle ist etwas willkürlich. Sehr wahrscheinlich wurde er gewählt, da er die Anzahl der Bits in einem DES Enkryption-Key angibt.) Was Ihnen diese Zeilen im Grunde sagen ist, dass die GSSAPI Authentifizierung erfolgreich war und dass die LDAP-Verbindung per Verschlüsselung geschützt wird.

Vergessen Sie nicht, dass Kerberos Authentifizierung immer ein wechselseitiger Prozess ist. Das bedeutet, nicht nur Sie haben sich gegenüber dem LDAP-Server authentifiziert — er hat sich im Gegenzug auch bei Ihnen authentifiziert. Sie können sich also sicher sein, dass Sie mit dem LDAP-Server kommunizieren, den Sie vorgesehen hatten und nicht mit irgendeinem vorgetäuschten Service, den ein Angreifer aufgesetzt hat.

Für den Fall, dass mehrere verschiedene SASL Mechanismen verwendet werden können, können Sie `ldapsearch` durch die Kommandozeilenoption `-Y GSSAPI` zur Verwendung von GSSAPI zwingen.

Kerberos-Authentifizierung und LDAP-Zugangskontrollen

Im vorigen Abschnitt haben wir uns erfolgreich am LDAP-Server authentifiziert. Im nächsten Schritt solle es jedem Benutzer ermöglicht werden, das Login-Shell Attribut in seinen LDAP Benutzerdaten zu ändern.

Angenommen, Sie verwenden ein Schema, nach dem der LDAP-Eintrag des Benutzers `joe` unter `uid=joe,ou=people,dc=suse,dc=de` liegt, können Sie die folgenden Zugangsregeln in der Datei `/etc/openldap/slapd.conf` aufstellen:

```
# This is required for things to work _at all_
access to dn.base="" by * read
# Let each user change their login shell
access to dn="*,ou=people,dc=suse,dc=de" attrs=loginShell
        by self write
# Every user can read everything
access to *
        by users read
```

Über die zweite Anweisung erlaubt authentifizierten Benutzern schreiben den Zugriff auf das `loginShell` Attribut ihres LDAP Eintrags. Die dritte Anweisung gibt allen authentifizierten Benutzern Lesezugriff auf das gesamte LDAP-Verzeichnis.

Wie findet nun der LDAP-Server heraus, dass `joe@SAMPLE.COM` von Kerberos das Äquivalent zum LDAP DN *distinguished name* `uid=joe,ou=people,dc=suse,dc=de` ist? Diese Zuordnung wird manuell über die `saslExpr` Direktive vorgenommen. Im Beispiel fügen Sie der `slapd.conf` folgende Zeilen hinzu:

```
saslRegexp
    uid=(.*) ,cn=GSSAPI,cn=auth
    uid=$1,ou=people,dc=example,dc=com
```

Um diesen Mechanismus zu verstehen, sollten Sie wissen, dass OpenLDAP einen DN bildet, wenn SASL einen Benutzer authentifiziert. Dieser DN setzt sich aus dem von SASL übergebenen Namen (wie zum Beispiel `joe`) und dem Typ der SASL Authentifizierung (`GSSAPI`) zusammen. In diesem Fall wäre `uid=joe,cn=GSSAPI,cn=auth` das Ergebnis.

Ist ein `saslRegexp` konfiguriert, wird der LDAP-Server den DN aus der SASL-Information mit dem ersten Argument als regulärem Ausdruck überprüfen. Trifft dieser reguläre Ausdruck zu, wird der Name durch das zweite Argument der `saslRegexp` Anweisung ersetzt. Die Platzhalter (`$1`) werden durch den Teilausdruck ersetzt, der über den `(.*)` Ausdruck ermittelt wurde.

Selbstverständlich sind auch kompliziertere Suchmuster möglich. Sollten Sie eine kompliziertere Verzeichnisstruktur verwenden oder der Benutzername in dem von Ihnen verwendeten Schema nicht Teil des DN sein, können Sie sogar Suchausdrücke verwenden, die den SASL DN dem Benutzer-DN zuordnen.

19.5 Sicherheit ist Vertrauenssache

19.5.1 Grundlagen

Eines der grundlegendsten Leistungsmerkmale eines Linux/Unix-Systems ist, dass mehrere Benutzer (multiuser) mehrere Aufgaben zur gleichen Zeit auf demselben Rechner (multi-tasking) ausführen können. Das Betriebssystem ist darüber hinaus netzwerktransparent, sodass Benutzer oftmals gar nicht wissen, ob sich die Daten oder Applikationen, mit denen sie arbeiten, lokal auf dem Rechner befinden oder über das Netzwerk bezogen werden.

Damit mehrere Benutzer auf einem System arbeiten können, müssen ihre jeweiligen Daten auch voneinander getrennt verwaltet werden können. Hier geht es unter anderem auch um Sicherheit und den Schutz der Privatsphäre. Datensicherheit war auch schon relevant, als Computer noch nicht miteinander vernetzt waren. Bei Verlust oder Defekt der Datenträger (im Allgemeinen Festplatten) mussten wichtige Daten weiterhin verfügbar sein, auch wenn solche Defekte womöglich den vorübergehenden Ausfall einer größeren Infrastruktur zur Folge hatten.

Auch wenn sich dieses Kapitel des SUSE-Handbuchs in der Hauptsache mit der Vertraulichkeit der Daten und dem Schutz der Privatsphäre der Benutzer beschäftigt, sei betont, dass ein umfassendes Sicherheitskonzept als integralen Bestandteil immer ein regelmäßiges, funktionierendes und überprüftes Backup beinhaltet. Ohne dieses Backup der Daten wird es nicht nur im Fall eines Hardware-Defekts schwierig sein, weiterhin auf die Daten zuzugreifen, sondern insbesondere auch dann, wenn nur der Verdacht besteht, dass jemand sich unbefugterweise an den Daten zu schaffen gemacht hat.

19.5.2 Lokale Sicherheit und Netzwerksicherheit

Es gibt verschiedene Möglichkeiten, auf Daten zuzugreifen:

- Persönliche Kommunikation mit jemand, der über die gewünschten Informationen verfügt bzw. Zugang zu bestimmten Daten auf einem Computer hat,
- direkt an der Konsole eines Rechners (physikalischer Zugriff),
- über eine serielle Schnittstelle oder
- über ein Netzwerk.

Alle diese Fälle sollten eine Gemeinsamkeit haben: Sie sollten sich als Benutzer authentifizieren müssen, bevor Sie Zugriff auf die Ressourcen oder Daten bekommen. Ein Webserver mag da anders geartet sein, aber Sie wollen sicherlich nicht, dass der Webserver Ihre persönlichen Daten an Surfer preisgibt.

Der erste Fall der oben genannten ist der menschlichste von allen: Etwa bei einer Bank müssen Sie einem Angestellten beweisen, dass Ihnen der Zugriff auf Ihre Konten gestattet ist, indem Sie mit Ihrer Unterschrift, einer PIN oder mit einem Passwort beweisen, dass Sie derjenige sind, für den Sie sich ausgeben. In manchen Fällen kann es gelingen, durch das Erwähnen von Kenntnissen oder durch geschickte Rhetorik das Vertrauen eines Wissensträgers zu erschleichen, so dass dieser weitere Information preisgibt, womöglich ohne dass das Opfer dies bemerkt. Man nennt dies in Hackerkreisen Social Engineering. Gegen diese Art von Angriff hilft nur Aufklärung und ein bewusster Umgang mit Information und Sprache. Einbrüchen auf Rechnersystemen geht oft eine Art Social-Engineering-Angriff, etwa auf das Empfangspersonal, Dienstleister in der Firma oder auch Familienmitglieder, voraus, der erst viel später bemerkt wird.

Jemand, der (unbefugt) Zugriff auf Daten erlangen will, könnte auch die traditionellste Methode benutzen, denn die Hardware selbst ist ein Angriffspunkt. Der Rechner muss gegen Entnahme, Austausch und Sabotage von Teilen und Gesamteinheit (und dem Backup der Daten!) sicher verstaut sein - dazu kann auch eine eventuell vorhandene Netzwerkleitung oder ein Stromkabel gehören. Außerdem muss der Startvorgang abgesichert sein, denn allgemein bekannte Tastenkombinationen können den Rechner zu speziellen Reaktionen veranlassen. Dagegen hilft das Setzen von BIOS- und Bootloaderpasswörtern.

Serielle Schnittstellen mit seriellen Terminals sind heute zwar immer noch gebräuchlich, werden aber kaum noch an neuen Arbeitsplätzen installiert. Ein serielles Terminal stellt eine besondere Art des Zugriffs dar: Es ist keine Netzwerkschnittstelle, da kein Netzwerkprotokoll zur Kommunikation zwischen den Systemeinheiten verwendet wird. Ein simples Kabel (oder eine Infrarotschnittstelle) wird als Übertragungsmedium für einfache Zeichen verwendet. Das Kabel selbst ist dabei der einfachste Angriffspunkt: Man muss nur einen alten Drucker daran anschließen und kann die Kommunikation aufzeichnen. Was mit einem Drucker möglich ist, geht selbstverständlich mit beliebigem Aufwand auch anders.

Da das Öffnen einer Datei auf einem Rechner anderen Zugriffsbeschränkungen unterliegt als das Öffnen einer Netzwerkverbindung zu einem Dienst auf einem Rechner, ist es nötig, zwischen lokaler Sicherheit und Netzwerksicherheit zu unterscheiden. Die Trennlinie ist da markiert, wo

Daten in Pakete verschlüsselt werden müssen, um verschickt zu werden und zur Anwendung zu gelangen.

Lokale Sicherheit

Lokale Sicherheit mit den physikalischen Gegebenheiten, in denen der Rechner aufgestellt ist. Wir gehen davon aus, dass Sie Ihren Rechner so aufgebaut haben, dass das Maß an Sicherheit Ihrem Anspruch und Ihren Anforderungen genügt. In Bezug auf Lokale Sicherheit besteht die Aufgabe darin, die einzelnen Benutzer voneinander zu trennen, sodass kein Benutzer die Rechte oder die Identität eines anderen Benutzers annehmen kann. Dies gilt im Allgemeinen, im Speziellen sind natürlich besonders `root`-Rechte gemeint, da der Benutzer `root` im System Allmacht hat; er kann unter anderem ohne Passwort zu jedem lokalen Benutzer werden und jede lokale Datei lesen.

Passwörter

Ihr Linux-System speichert Passwörter nicht etwa im Klartext ab und vergleicht ein eingegebenes Passwort mit dem, was gespeichert ist. Bei einem Diebstahl der Datei, in der die Passwörter stehen, wären dann ja alle Accounts auf Ihrem System kompromittiert. Stattdessen wird Ihr Passwort verschlüsselt abgelegt und jedes Mal, wenn Sie das Passwort eingegeben haben, wird dieses wieder verschlüsselt und das Ergebnis verglichen mit dem, was als verschlüsseltes Passwort abgespeichert ist. Dies macht natürlich nur dann Sinn, wenn man aus dem verschlüsselten Passwort nicht das Klartextpasswort errechnen kann. Dies erreicht man durch so genannte Falltüralgorithmen, die nur in eine Richtung funktionieren. Ein Angreifer, der das verschlüsselte Passwort in seinen Besitz gebracht hat, kann nicht einfach zurückrechnen und das Passwort sehen, sondern er muss alle möglichen Buchstabenkombinationen für ein Passwort durchprobieren, bis er dasjenige findet, welches verschlüsselt so aussieht wie Ihres. Bei acht Buchstaben pro Passwort gibt es beträchtlich viele Kombinationen.

Mit ein Argument für die Sicherheit dieser Methode in den 70er Jahren war, dass der verwendete Algorithmus recht langsam ist und Zeit im Sekundenbereich für das Verschlüsseln von einem Passwort brauchte. Heutige PCs schaffen ohne weiteres mehrere hunderttausend bis Millionen Verschlüsselungen pro Sekunde. Aus diesem Grund dürfen verschlüsselte Passwörter nicht für jeden Benutzer sichtbar sein (`/etc/shadow` ist für einen normalen Benutzer nicht lesbar), und die Passwörter dürfen nicht leicht zu erraten sein, für den Fall, dass die verschlüsselten Passwörter wegen eines Fehlers eben doch sichtbar werden. Ein Passwort wie Phantasie

umzuschreiben in Ph@nt@s13 hilft nicht viel: Solche Vertauschungsregeln können von Knackprogrammen, die Wörterbücher zum Raten benutzen, sehr leicht aufgelöst werden. Besser sind Kombinationen von Buchstaben, die kein bekanntes Wort bilden und nur für den Benutzer eine persönliche Bedeutung haben, etwa die Anfangsbuchstaben der Wörter eines Satzes, oder nehmen Sie zum Beispiel einen Buchtitel wie Der Name der Rose von Umberto Eco. Daraus gewinnen Sie ein gutes Passwort: DNdRvUE9. Ein Passwort wie Bierjunge oder Jasmin76 würde schon jemand erraten können, der Sie oberflächlich gut kennt.

Der Bootvorgang

Verhindern Sie, dass mit einer Diskette oder einer CD-ROM gebootet werden kann, indem Sie die Laufwerke ausbauen oder indem Sie ein BIOS-Passwort setzen und im BIOS ausschließlich das Booten von Festplatte erlauben.

Linux Systeme starten gewöhnlicherweise mit einem Bootloader, der es erlaubt, zusätzliche Optionen an den zu startenden Kernel weiterzugeben. Solche Optionen sind im hohem Maße sicherheitskritisch, weil der Kernel ja nicht nur mit root-Rechten läuft, sondern die root-Rechte von Anfang an vergibt. Wenn Sie LILO als Bootloader verwenden, können Sie dies durch Vergabe eines weiteren Passwortes in `/boot/grub/menu.lst` verhindern (siehe 7 auf Seite 203).

Zugriffsrechte

Es gilt das Prinzip, immer mit den niedrigst möglichen Privilegien für eine jeweilige Aufgabe zu arbeiten. Es ist definitiv nicht nötig, seine E-Mails als root zu lesen und zu schreiben. Wenn das Mailprogramm (MUA = Mail User Agent), mit dem Sie arbeiten, einen Fehler hat, dann wirkt sich dieser Fehler mit genau den Rechten aus, die Sie zum Zeitpunkt des des Angriffs hatten. Hier geht es also auch um Schadensminimierung.

Die einzelnen Rechte der weit über 200.000 Dateien einer SUSE-Distribution sind sorgfältig vergeben. Der Administrator eines Systems sollte zusätzliche Software oder andere Dateien nur unter größtmöglicher Sorgfalt installieren und besonders gut auf die vergebenen Rechte der Dateien achten. Erfahrene und sicherheitsbewusste Admins verwenden bei dem Kommando `ls` stets die Option `-l` für eine ausführliche Liste der Dateien mitsamt den Zugriffsrechten, so dass sie eventuell falsch gesetzte Dateirechte gleich erkennen können. Ein falsch gesetztes Attribut bedeutet nicht nur, dass Dateien überschrieben oder gelöscht werden können, sondern auch dass die veränderten Dateien von root ausgeführt oder im

Fall von Konfigurationsdateien von Programmen als `root` benutzt werden können. Damit könnte ein Angreifer seine Rechte beträchtlich ausweiten. Man nennt solche Angriffe dann Kuckuckseier, weil das Programm (das Ei) von einem fremden Benutzer (Vogel) ausgeführt (ausgebrütet) wird, ähnlich wie der Kuckuck seine Eier von fremden Vögeln ausbrüten lässt.

SUSE-Systeme verfügen über die Dateien `permissions`, `permissions.easy`, `permissions.secure` und `permissions.paranoid` im Verzeichnis `/etc`. In diesen Dateien werden besondere Rechte wie etwa welt-schreibbare Verzeichnisse oder für Dateien `setuser-ID-bits` festgelegt. (Programme mit gesetztem `setuser-ID-bit` laufen nicht mit der Berechtigung des Eigentümers des Prozesses, der es gestartet hat, sondern mit der Berechtigung des Eigentümers der Datei. Dies ist in der Regel `root`.) Für den Administrator steht die Datei `/etc/permissions.local` zur Verfügung, in der er seine eigenen Änderungen festhalten kann.

Die Auswahl, welche der Dateien für Konfigurationsprogramme von SUSE zur Vergabe der Rechte benutzt werden sollen, können Sie auch komfortabel mit YaST unter dem Menüpunkt 'Sicherheit' treffen. Mehr zu diesem Thema erfahren Sie direkt aus der Datei `/etc/permissions` und der Manpage des Kommandos `chmod` (man `chmod`).

Buffer overflows, format string bugs

Wann immer ein Programm Daten verarbeitet, die in beliebiger Form unter Einfluss eines Benutzers stehen oder standen, ist besondere Vorsicht geboten. Diese Vorsicht gilt in der Hauptsache für den Programmierer der Anwendung: Er muss sicherstellen, dass die Daten durch das Programm richtig interpretiert werden, dass die Daten zu keinem Zeitpunkt in Speicherbereiche geschrieben werden, die eigentlich zu klein sind und dass er die Daten in konsistenter Art und Weise durch sein eigenes Programm und die dafür definierten Schnittstellen weiterreicht.

Ein Buffer Overflow passiert dann, wenn beim Beschreiben eines Pufferspeicherbereichs nicht darauf geachtet wird, wie groß der Puffer eigentlich ist. Es könnte sein, dass die Daten (die vom Benutzer kamen) etwas mehr Platz verlangen, als im Puffer zur Verfügung steht. Durch dieses Überschreiben des Puffers über seine Grenze hinaus ist es unter Umständen möglich, dass ein Programm aufgrund der Daten, die er eigentlich nur verarbeiten soll, Programmsequenzen ausführt, die unter dem Einfluss des Users und nicht des Programmierers stehen. Dies ist ein schwerer Fehler, insbesondere wenn das Programm mit besonderen Rechten abläuft (siehe Abschnitt 19.5.2 auf der vorherigen Seite). Format String Bugs funktionieren etwas anders, verwenden aber wieder user-input, um das Programm von seinem eigentlichen Weg abzubringen.

Diese Programmierfehler werden normalerweise bei Programmen ausbeutet, die mit gehobenen Privilegien

ausgeführt werden, also `setuid`- und `setgid`-Programme. Sie können sich und Ihr System also vor solchen Fehlern schützen, indem Sie die besonderen Ausführungsrechte von den Programmen entfernen. Auch hier gilt wieder das Prinzip der geringstmöglichen Privilegien (siehe Abschnitt 19.5.2 auf Seite 564).

Da Buffer Overflows und Format String Bugs Fehler bei der Behandlung von Benutzerdaten sind, sind sie nicht notwendigerweise nur ausbeutbar, wenn man bereits Zugriff auf ein lokales login hat. Viele der bekannt gewordenen Fehler können über eine Netzwerkverbindung ausgenutzt werden. Deswegen sind Buffer Overflows und Format String Bugs nicht direkt auf den lokalen Rechner oder das Netzwerk klassifizierbar.

Viren

Entgegen andersartiger Verlautbarungen gibt es tatsächlich Viren für Linux. Die bekannten Viren sind von ihren Autoren als Proof-of-Concept geschrieben worden, als Beweis, dass die Technik funktioniert. Allerdings ist noch keiner dieser Viren in freier Wildbahn beobachtet worden.

Viren benötigen zur Ausbreitung einen Wirt, ohne den sie nicht überlebensfähig sind. Dieser Wirt ist ein Programm oder ein wichtiger Speicherbereich für das System, etwa der Master-Boot-Record, und er muss für den Programmcode des Virus beschreibbar sein. Linux hat aufgrund seiner Multi-User Fähigkeiten die Möglichkeit, den Schreibzugriff auf Dateien einzuschränken, also insbesondere Systemdateien. Wenn Sie als `root` arbeiten, erhöhen Sie also die Wahrscheinlichkeit, dass Ihr System von solch einem Virus infiziert wird. Berücksichtigen Sie aber die Regel der geringstmöglichen Privilegien, ist es Schwierigkeiten unter Linux einen Virus zu bekommen. Darüber hinaus sollten Sie nie leichtfertig ein Programm ausführen, das Sie aus dem Internet bezogen haben und dessen genaue Herkunft Sie nicht kennen. SUSE-rpm Pakete sind kryptographisch signiert und tragen mit dieser digitalen Unterschrift das Markenzeichen der Sorgfalt beim Bau der Pakete bei SUSE. Viren sind klassische Symptome dafür, dass auch ein hochsicheres System unsicher wird, wenn der Administrator oder auch der Benutzer ein mangelndes Sicherheitsbewusstsein hat.

Viren sind nicht mit Würmern zu verwechseln, die Phänomene auch der Netzwerksicherheit sind und keinen Wirt brauchen, um sich zu verbreiten.

Netzwerksicherheit

Bei der lokalen Sicherheit war es die Aufgabe, die Benutzer, die an demselben Rechner arbeiten, voneinander zu trennen, insbesondere den Benutzer `root`. Im Gegensatz dazu soll bei der Netzwerksicherheit das ganze System gegen Angriffe über das Netzwerk geschützt werden. Benutzerauthentifizierung beim klassischen Einloggen durch Benutzerkennung und Passwort gehört zur lokalen Sicherheit. Beim Einloggen über eine Netzwerkverbindung muss man differenzieren zwischen beiden Sicherheitsaspekten: bis zur erfolgten Authentifizierung spricht man von Netzwerksicherheit, nach dem Login geht es um lokale Sicherheit.

X-Windows (X11-Authentifizierung)

Wie bereits erwähnt ist Netzwerktransparenz eine grundlegende Eigenschaft eines Unix-Systems. Bei X11, dem Windowing-System von Unix, gilt dies in besonderem Maße! Sie können sich ohne Weiteres auf einem entfernten Rechner einloggen und dort ein Programm starten, welches dann über das Netzwerk auf Ihrem Rechner angezeigt wird.

Wenn nun ein X-Client über das Netzwerk bei unserem X-Server angezeigt werden soll, dann muss der Server die Ressource, die er verwaltet (das Display), gegen unberechtigte Zugriffe schützen. Konkret heißt das hier, dass das Client-Programm Rechte bekommen muss. Bei X-Windows geschieht dies auf zwei verschiedene Arten: Host-basierte und cookie-basierte Zugriffskontrolle. Erstere basiert auf der IP-Adresse des Rechners, auf dem das Client-Programm laufen soll und wird mit dem Programm `xhost` kontrolliert. Das Programm `xhost` trägt eine IP-Adresse eines legitimen Client in eine Mini-Datenbank im X-Server ein. Eine Authentifizierung einzig und allein auf einer IP-Adresse aufzubauen gilt jedoch nicht gerade als sicher. Es könnte noch ein zweiter Benutzer auf dem Rechner mit dem Client-Programm aktiv sein, und dieser hätte dann genau wie jemand, der die IP-Adresse stiehlt, Zugriff auf den X-Server. Deswegen soll hier auch nicht näher auf diese Methoden eingegangen werden. Die Manpage des `xhost`-Kommandos gibt mehr Aufschluss über die Funktionsweise (und enthält ebenfalls die Warnung!).

Bei cookie-basierter Zugriffskontrolle wird eine Zeichenkette, die nur der X-Server und der legitim eingeloggte Benutzer kennen, als einem Passwort ähnliches Ausweismittel verwendet. Dieses cookie (das englische Wort cookie bedeutet Keks und meint hier die chinesischen fortune cookies, die einen Spruch enthalten) wird in der Datei `.xauthority` im Home-Verzeichnis des Benutzers beim login abgespeichert und steht somit jedem X-Windows-client, der ein Fenster beim X-Server zur Anzeige bringen will,

zur Verfügung. Das Programm `xauth` gibt dem Benutzer das Werkzeug, die Datei `.Xauthority` zu untersuchen. Wenn Sie `.Xauthority` aus Ihrem `home`-Verzeichnis löschen oder umbenennen, dann können Sie keine weiteren Fenster von neuen X-Clients mehr öffnen. Näheres über Sicherheitsaspekte von X-Windows erfahren Sie in der manpage von `Xsecurity` (`man Xsecurity`).

`ssh` (secure shell) kann über eine vollständig verschlüsselte Netzverbindung für einen Benutzer transparent (also nicht direkt sichtbar) die Verbindung zu einem X-Server weiterleiten. Man spricht von X11-forwarding. Dabei wird auf der Server-Seite ein X-Server simuliert und bei der Shell auf der remote-Seite die `DISPLAY`-Variable gesetzt.

Achtung

Wenn Sie den Rechner, auf dem Sie sich einloggen, nicht als sicher betrachten, dann sollten Sie auch keine X-Windows-Verbindungen weiterleiten lassen. Mit eingeschaltetem X11-forwarding könnten sich auch Angreifer über Ihre `ssh`-Verbindung mit Ihrem X-Server authentifiziert verbinden und beispielsweise Ihre Tastatur belauschen.

Achtung

Buffer Overflows und Format String Bugs

Nicht direkt klassifizierbar in lokal und remote gilt das im Abschnitt Lokale Sicherheit über Buffer Overflows und Format String Bugs Gesagte äquivalent für Netzwerksicherheit. Wie auch bei den lokalen Varianten dieser Programmierfehler führen Buffer Overflows bei Netzwerkdiensten meistens zu `root`-Rechten. Sollte dies nicht der Fall sein, dann könnte sich der Angreifer zumindest Zugang zu einem unprivilegierten lokalen Account verschaffen, mit dem er dann weitere (lokale) Sicherheitsprobleme ausnützen kann, falls diese vorhanden sind.

Über das Netzwerk ausbeutbare Buffer Overflows und Format String Bugs sind wohl die häufigsten Varianten von remote-Angriffen überhaupt. Auf Sicherheitsmailinglisten werden so genannte *exploits* herumgereicht, d. h. Programme, die die frisch gefundenen Lücken ausnutzen. Auch jemand, der nicht die genauen Details der Lücke kennt, kann damit die Lücke ausnutzen. Im Laufe der Jahre hat sich herausgestellt, dass die freie Verfügbarkeit von exploitcodes generell die Sicherheit von Betriebssystemen erhöht hat, was sicherlich daran liegt, dass Betriebssystemhersteller dazu gezwungen waren, die Probleme in ihrer Software zu beseitigen. Da bei freier Software der Sourcecode für jedermann erhältlich ist (SUSE-Linux liefert alle

verfügbaren Quellen mit), kann jemand, der eine Lücke mitsamt exploitcode findet, auch gleichzeitig noch einen Reparaturvorschlag für das Problem anbieten.

DoS — Denial of Service

Ziel dieser Art von Angriff ist das Blockieren eines Dienstes oder sogar des ganzen Systems. Dies kann auf verschiedenste Arten passieren: Durch Überlastung, durch Beschäftigung mit unsinnigen Paketen oder durch Ausnutzen von Remote Buffer Overflows, die nicht direkt zum Ausführen von Programmen auf der remote-Seite ausbeutbar sind.

Der Zweck eines DoS mag meistens darin begründet sein, dass der Dienst einfach nicht mehr verfügbar ist. Dass ein Dienst fehlt, kann aber weitere Konsequenzen haben. Siehe man in the middle: sniffing, tcp connection hijacking, spoofing und DNS poisoning.

man in the middle: sniffing, tcp connection hijacking, spoofing

Ganz allgemein gilt: Ein Angriff vom Netzwerk, bei der der Angreifer eine Position zwischen zwei Kommunikationspartnern einnimmt, nennt sich man in the middle attack. Sie haben in der Regel eines gemeinsam: Das Opfer merkt nichts davon. Viele Varianten sind denkbar: Der Angreifer nimmt die Verbindung entgegen und stellt, damit das Opfer nichts merkt, selbst eine Verbindung zum Ziel her. Das Opfer hat also, ohne es zu wissen, eine Netzwerkverbindung zum falschen Rechner geöffnet, weil dieser sich als das Ziel ausgibt. Der einfachste man in the middle attack ist ein sniffer. Er belauscht einfach nur die Netzverbindungen, die an ihm vorüber geführt werden (sniffing = engl. schnüffeln). Komplexer wird es, wenn der Angreifer in der Mitte versucht, eine etablierte, bestehende Verbindung zu übernehmen (entführen = engl. hijacking). Dafür muss der Angreifer die Pakete, die an ihm vorbeigeführt werden, eine Weile lang analysiert haben, damit er die richtigen TCP-Sequenznummern der TCP-Verbindung vorhersagen kann. Wenn er dann die Rolle des Ziels der Verbindung übernimmt, merkt das das Opfer, weil auf der Seite des Opfers die Verbindung als ungültig terminiert wird.

Der Angreifer profitiert dabei insbesondere bei Protokollen, die nicht kryptographisch gegen hijacking gesichert sind und bei denen zu Beginn der Verbindung eine Authentifizierung stattfindet. Spoofing nennt sich das Verschicken von Paketen mit modifizierten Absenderdaten, also hauptsächlich der IP Adresse. Die meisten aktiven Angriffsvarianten verlangen das Verschicken von gefälschten Paketen, was unter Unix/Linux übrigens nur der Superuser (root) darf.

Viele der Angriffsmöglichkeiten kommen in Kombination mit einem DoS vor. Gibt es eine Möglichkeit, einen Rechner schlagartig vom Netzwerk zu trennen (wenn auch nur für kurze Zeit), dann wirkt sich das förderlich auf einen aktiven Angriff aus, weil keine Störungen mehr erwartet werden müssen.

DNS poisoning

Der Angreifer versucht, mit gefälschten (gespooften) DNS-Antwortpaketen den cache eines DNS-Servers zu vergiften *poisoning*, so dass dieser die gewünschte Information an ein Opfer weitergibt, das danach fragt. Um einem DNS-Server solche falschen Informationen glaubhaft zuschieben zu können, muss der Angreifer normalerweise einige Pakete des Servers bekommen und analysieren. Weil viele Server ein Vertrauensverhältnis zu anderen Rechnern aufgrund ihrer IP Adresse oder ihres Hostnamens konfiguriert haben, kann ein solcher Angriff trotz eines gehörigen Aufwands recht schnell Früchte tragen. Voraussetzung ist allerdings eine gute Kenntnis der Vertrauensstruktur zwischen diesen Rechnern. Ein zeitlich genau abgestimmter DoS gegen einen DNS-Server, dessen Daten gefälscht werden sollen, ist aus Sicht des Angreifers meistens nicht vermeidbar.

Abhilfe schafft wieder eine kryptographisch verschlüsselte Verbindung, die die Identität des Ziels der Verbindung verifizieren kann.

Würmer

Würmer werden häufig mit Viren gleichgesetzt. Es gibt aber einen markanten Unterschied: Ein Wurm muss keinerlei Wirtsprogramm infizieren, und er ist darauf spezialisiert, sich möglichst schnell im Netzwerk zu verbreiten. Bekannte Würmer wie Ramen, Lion oder Adore nutzen wohlbekannte Sicherheitslücken von Serverprogrammen wie `bind8` oder `lprNG`. Man kann sich relativ einfach gegen Würmer schützen, weil zwischen dem Zeitpunkt des Bekanntwerdens der ausgenutzten Lücken bis zum Auftauchen des Wurms normalerweise einige Tage vergehen, so dass update-Pakete vorhanden sind. Natürlich setzt dies voraus, dass der Administrator die Security-updates auch in seine Systeme einspielt.

19.5.3 Tipps und Tricks: Allgemeine Hinweise

Information: Für einen effizienten Umgang mit dem Bereich Sicherheit ist es nötig, mit den Entwicklungen Schritt zu halten und auf dem Laufenden zu sein, was die neuesten Sicherheitsprobleme angeht. Ein sehr guter Schutz gegen Fehler aller Art ist das schnellstmögliche Einspielen von

update-Paketen, die von einem Security-Announcement angekündigt werden. Die SUSE-security-Announcements werden über eine Mailingliste verbreitet, in die Sie sich, den Links unter <http://www.suse.de/security> folgend, eintragen können. suse-security-announce@suse.de ist die erste Informationsquelle für update-Pakete, die vom Security-Team mit neuen Informationen beliefert wird.

Die Mailingliste suse-security@suse.de ist ein lehrreiches Diskussionsforum für den Bereich Sicherheit. Sie können sich auf der gleichen URL wie für suse-security-announce@suse.de für die Liste anmelden.

Eine der bekanntesten Sicherheitsmailinglisten der Welt ist die Liste bugtraq@securityfocus.com. Die Lektüre dieser Liste bei durchschnittlich 15-20 Postings am Tag kann mit gutem Gewissen empfohlen werden. Mehr Information finden Sie auf <http://www.securityfocus.com>.

Einige Grundregeln, die zu kennen nützlich sein kann, sind nachstehend aufgeführt:

- Vermeiden Sie es, als `root` zu arbeiten, entsprechend dem Prinzip, die geringstnötigen Privilegien für eine Aufgabe zu benutzen. Das verringert die Chancen für ein Kuckucksei oder einen Virus, und überdies für Fehler Ihrerseits.
- Benutzen Sie nach Möglichkeit immer verschlüsselte Verbindungen, um Arbeiten remote auszuführen. `ssh` (secure shell) ist Standard, vermeiden Sie `telnet`, `ftp`, `rsh` und `rlogin`.
- Benutzen Sie keine Authentifizierungsmethoden, die alleine auf der IP-Adresse aufgebaut sind.
- Halten Sie Ihre wichtigsten Pakete für den Netzwerkbereich immer auf dem neuesten Stand und abonnieren Sie die Mailinglisten für Announcements der jeweiligen Software (z. B. Beispiel `bind`, `sendmail`, `ssh`). Dasselbe gilt für Software, die nur lokale Sicherheitsrelevanz hat.
- Optimieren Sie die Zugriffsrechte auf sicherheitskritische Dateien im System, indem Sie die `/etc/permissions`-Datei Ihrer Wahl an Ihre Bedürfnisse anpassen. Ein `setuid`-Programm, welches kein `setuid`-bit mehr hat, mag zwar nicht mehr wirklich seine Aufgabe erledigen können, aber es ist in der Regel kein Sicherheitsproblem mehr. Mit einer ähnlichen Vorgehensweise können Sie auf welt-schreibbare Dateien und Verzeichnisse losgehen.

- Deaktivieren Sie jegliche Netzwerkdienste, die Sie auf Ihrem Server nicht zwingend brauchen. Das macht Ihr System sicherer, und es verhindert, dass Ihre Benutzer sich an einen Dienst gewöhnen, den Sie nie absichtlich freigegeben haben (legacy-Problem). Offene Ports (mit socket-Zustand LISTEN) finden Sie mit dem Programm `netstat`. Als Optionen bietet sich an, `netstat -ap` oder `netstat -anp` zu verwenden. Mit der `-p`-Option können Sie gleich sehen, welcher Prozess mit welchem Namen den Port belegt.

Vergleichen Sie die Ergebnisse, die Sie haben, mit einem vollständigen Portscan Ihres Rechners von außen. Das Programm `nmap` ist dafür hervorragend geeignet. Es klopft jeden einzelnen Port ab und kann anhand der Antwort Ihres Rechners Schlüsse über einen hinter dem Port wartenden Dienst ziehen. Scannen Sie niemals einen Rechner ohne das direkte Einverständnis des Administrators, denn dies könnte als aggressiver Akt aufgefasst werden. Denken Sie daran, dass Sie nicht nur TCP-Ports scannen sollten, sondern auf jeden Fall auch UDP-Ports (Optionen `-sS` und `-sU`).

- Zur zuverlässigen Integritätsprüfung der Dateien in Ihrem System sollten Sie `tripwire` benutzen und die Datenbank verschlüsseln, um sie gegen manipulative Zugriffe zu schützen. Darüber hinaus brauchen Sie auf jeden Fall ein Backup dieser Datenbank außerhalb der Maschine auf einem eigenen Datenträger, der nicht über einen Rechner mit einem Netzwerk verbunden ist.
- Seien Sie vorsichtig beim Installieren von Fremdsoftware. Es gab schon Fälle, wo ein Angreifer tar-Archive einer Sicherheitssoftware mit einem trojanischen Pferd versehen hat. Zum Glück wurde dies schnell bemerkt. Wenn Sie ein binäres Paket installieren, sollten Sie sicher sein, woher das Paket kommt.

SUSE rpm-Pakete werden gpg-signiert ausgeliefert. Der Schlüssel, den wir zum Signieren verwenden, ist

ID:9C800ACA 2000-10-19 SUSE Package Signing Key
<build@suse.de>

Key fingerprint = 79C1 79B2 E1C8 20C1 890F 9994 A84E DAE8 9C80 0ACA

Das Kommando `rpm -checksig paket.rpm` zeigt an, ob die Prüfsumme und die Signatur des (nicht installierten!) Pakets stimmen. Sie finden den Schlüssel auf der ersten CD oder DVD von SUSE LINUX und auf den meisten Keyservern der Welt.

- Überprüfen Sie regelmäßig Ihr Backup der Daten und des Systems. Ohne eine zuverlässige Aussage über die Funktion des Backups ist das Backup unter Umständen wertlos.
- Überwachen Sie Ihre Logfiles. Nach Möglichkeit sollten Sie sich ein kleines Script schreiben, welches Ihre Logfiles nach ungewöhnlichen Einträgen absucht. Diese Aufgabe ist alles andere als trivial, denn nur Sie wissen, was ungewöhnlich ist und was nicht.
- Verwenden Sie `tcp_wrapper`, um den Zugriff auf die einzelnen Dienste Ihres Rechners auf die IP-Adressen einzuschränken, denen der Zugriff auf einen bestimmten Dienst explizit gestattet ist. Nähere Information zu den `tcp_wrappern` finden Sie in der Manualpage von `tcpd(8)` und der Manualpage von `hosts_access`.
- Als zusätzlichen Schutz zu dem `tcpd` (`tcp_wrapper`) könnten Sie die `SuSEfirewall` verwenden.
- Legen Sie Ihr Sicherheitsdenken redundant aus: Eine Meldung, die zweimal eintrifft, ist besser als eine, die Sie nie sehen. Dies gilt genauso für Gespräche mit Kollegen.

19.5.4 Zentrale Meldung von neuen Sicherheitsproblemen

Wenn Sie ein Sicherheitsproblem finden (bitte überprüfen Sie die zur Verfügung stehenden Update-Pakete), dann wenden Sie sich bitte vertrauensvoll an die E-Mail-Adresse `mailto:security@suse.de`. Bitte fügen Sie eine genaue Beschreibung des Problems bei, zusammen mit den Versionsnummern der verwendeten Pakete. Wir werden uns bemühen, Ihnen so schnell wie möglich zu antworten. Eine pgp-Verschlüsselung Ihrer E-Mail ist erwünscht. Unser pgp-Key ist:

ID:3D25D3D9 1999-03-06 SUSE Security Team <security@suse.de>

Key fingerprint = 73 5F 2E 99 DF DB 94 C4 8F 5A A3 AE AF 22 F2 D5

Der Schlüssel liegt auch unter <http://www.suse.de/security> zum Download bereit.

Teil V

Anhang

Dateisysteme unter Linux

Linux unterstützt eine ganze Reihe von Dateisystemen. Dieses Kapitel gibt einen kurzen Überblick über die bekanntesten Dateisysteme unter Linux, wobei wir insbesondere auf deren Designkonzept und Vorzüge sowie deren Einsatzbereiche eingehen werden. Weiterhin werden einige Informationen zum „Large File Support“ unter Linux bereitgestellt.

Glossar

Metadaten Die interne Datenstruktur eines Dateisystems, die eine geordnete Struktur und die Verfügbarkeit der Festplattendaten gewährleistet. Im Grunde genommen sind es die „Daten über die Daten“. Nahezu jedes Dateisystem besitzt seine eigene Metadatenstruktur. Hierin liegt zum Teil auch der Grund für die unterschiedlichen Leistungsmerkmale der verschiedenen Dateisysteme. Es ist von äußerster Wichtigkeit, die Metadaten intakt zu halten, da andernfalls das gesamte Dateisystem zerstört werden könnte.

Inode Inodes enthalten alle möglichen Informationen über eine Datei, die Größe, die Anzahl der Links, Datum, Erstellungszeit, Änderungen, Zugriff sowie Zeiger (engl. *pointer*) auf die Festplattenblöcke, wo die Datei gespeichert ist.

Journal Im Zusammenhang mit einem Dateisystem ist ein Journal eine platteninterne Struktur mit einer Art Protokoll, in das der Dateisystemtreiber die zu ändernden (Meta-)daten des Dateisystems einträgt. „Journaling“ verringert die Wiederherstellungszeit eines Linux-Systems enorm, da der Dateisystemtreiber keine umfassende Suche nach zerstörten Metadaten auf der gesamten Platte starten muss. Stattdessen werden die Journal-Einträge wieder eingespielt.

Die wichtigsten Dateisysteme unter Linux

Anders als noch vor zwei oder drei Jahren ist die Auswahl eines Dateisystems für Linux nicht mehr eine Angelegenheit von Sekunden (Ext2 oder ReiserFS?). Kernel ab der Version 2.4 bieten eine große Auswahl an Dateisystemen. Im Folgenden erhalten Sie einen groben Überblick über die grundlegende Funktionsweise dieser Dateisysteme und deren Vorteile.

Seien Sie sich immer bewusst, dass kein Dateisystem allen Applikationen gleichermaßen gerecht werden kann. Jedes Dateisystem hat seine ihm eigenen Stärken und Schwächen, die berücksichtigt werden müssen. Sogar das hochentwickeltste Dateisystem der Welt wird niemals ein vernünftiges Backupkonzept ersetzen.

Die Fachbegriffe „Datenintegrität“ oder „Datenkonsistenz“ beziehen sich in diesem Kapitel nicht auf die Konsistenz der Speicherdaten eines Benutzers (diejenigen Daten, die Ihre Applikation in ihre Dateien schreibt). Die Konsistenz dieser Daten muss von der Applikation selbst gewährleistet werden.

Hinweis

Einrichtung von Dateisystemen

Soweit nicht explizit hier anders beschrieben, lassen sich alle Arbeiten zur Partitionierung und zum Anlegen und Bearbeiten von Dateisystemen bequem mit YaST erledigen.

Hinweis

Ext2

Die Ursprünge von Ext2 finden sich in der frühen Geschichte von Linux. Sein Vorgänger, das Extended File System, wurde im April 1992 implementiert und in Linux 0.96c integriert. Das Extended File System erfuhr eine Reihe von Änderungen und wurde für Jahre als Ext2 das bekannteste Dateisystem unter Linux. Mit dem Einzug der Journaling File Systeme und deren erstaunlich kurzen Wiederherstellungszeiten verlor Ext2 an Wichtigkeit.

Möglicherweise hilft Ihnen eine kurze Zusammenfassung der Stärken von Ext2 beim Verständnis für dessen Beliebtheit unter den Linux-Benutzern, die es teilweise noch heute als Dateisystem bevorzugen.

Stabilität Als wahrer „old-timer“, erfuhr Ext2 viele Verbesserungen und wurde ausführlich getestet. Daher wohl auch sein Ruf als „rock-solid“. Im Falle eines Systemausfalls, bei dem das Dateisystem nicht sauber ungemountet werden konnte, startet `e2fsck` eine Analyse der Dateisystemdaten. Metadaten werden in einen konsistenten Zustand gebracht und schwebende Dateien oder Datenblöcke werden in ein ausgewiesenes Verzeichnis geschrieben (genannt `lost+found/`). Im Gegensatz zu (den meisten) Journaling File Systemen analysiert `e2fsck` das gesamte Dateisystem und nicht nur die kürzlich veränderten Metadatenbits. Dies dauert bedeutend länger als die Überprüfung der Protokolldaten eines Journaling File Systems. Je nach Größe des Dateisystems kann dies eine halbe Stunde und mehr in Anspruch nehmen. Deshalb werden Sie Ext2 für keinen Server wählen, der hochverfügbar sein muss. Da Ext2 jedoch kein Journal pflegen muss und bedeutend weniger Speicher verbraucht, ist es manchmal schneller als andere Dateisysteme.

Leichtes Upgrade Basierend auf dem starken Fundament Ext2 konnte sich Ext3 zu einem gefeierten Dateisystem der nächsten Generation entwickeln. Seine Zuverlässigkeit und Stabilität wurden geschickt mit den Vorzügen eines Journaling File Systems verbunden.

Ext3

Ext3 wurde von Stephen Tweedie entworfen. Anders als alle anderen „next-generation“ Dateisysteme, folgt Ext3 keinem komplett neuen Designprinzip. Es basiert auf Ext2. Diese beiden Dateisysteme sind sehr eng miteinander verwandt. Ein Ext3-Dateisystem kann leicht auf einem Ext2-Dateisystem aufgebaut werden. Der grundlegendste Unterschied zwischen Ext2 und Ext3 liegt darin, dass Ext3 Journaling unterstützt.

Zusammenfassend lassen sich für Ext3 drei Vorteile herausstellen:

Leichte und höchst zuverlässige Dateisystem-Upgrades von Ext2

Da Ext3 auf dem Ext2-Code basiert und sowohl sein platteneigenes Format als auch sein Metadatenformat teilt, sind Upgrades von Ext2 auf Ext3 sehr unkompliziert. Sie können sogar dann durchgeführt werden, wenn Ihre Ext2-Dateisysteme gemountet sind. Anders als beim Umstieg auf andere Journaling File Systeme, wie zum Beispiel ReiserFS, JFS, oder XFS, der sehr mühsam sein kann, (Sie müssen Sicherungskopien des gesamten Dateisystems erstellen und dieses

anschließend von Grund auf neu erstellen), ist ein Umstieg auf Ext3 eine Angelegenheit von Minuten. Zugleich ist er sehr sicher, da die Wiederherstellung eines gesamten Dateisystems von Grund auf nicht immer fehlerlos vonstatten geht. Betrachtet man die Anzahl der vorhandenen Ext2-Systeme, die auf ein Upgrade auf ein Journaling File System warten, kann man sich leicht die Bedeutung von Ext3 für viele Systemadministratoren ausmalen. Ein Downgrade von Ext3 auf Ext2 ist genauso leicht wie das Upgrade. Führen Sie einfach einen sauberen Unmount des Ext3-Dateisystems durch und mounten Sie es als ein Ext2-Dateisystem.

Zuverlässigkeit und Performance Andere Journaling File Systeme folgen dem „metadata-only“-Journaling-Ansatz. Das heisst, Ihre Metadaten bleiben in einem konsistenten Zustand; dies kann jedoch nicht automatisch für die Dateisystemdaten selbst garantiert werden. Ext3 ist in der Lage, sich sowohl um die Metadaten als auch die Daten selbst zu kümmern. Wie eingehend sich Ext3 um Daten und Metadaten kümmert, ist individuell einstellbar. Den höchsten Grad an Sicherheit (d.h. Datenintegrität) erreicht man durch den Start von Ext3 im `data=journal`-Modus; dies jedoch kann das System verlangsamen, da sowohl Metadaten als auch Daten selbst im Journal erfasst werden. Ein relativ neuer Ansatz besteht in der Verwendung des `data=ordered`-Modus, der sowohl die Daten- als auch die Metadatenintegrität gewährleistet, jedoch das Journaling nur für Metadaten verwendet. Der Dateisystemtreiber sammelt alle Datenblöcke, die zu einem Metadaten-Update gehören. Diese Blöcke werden als „transaction“ gruppiert und werden auf die Platte geschrieben, bevor die Metadaten aktualisiert sind. Somit erreicht man Metadaten- und Datenkonsistenz ohne Leistungsverlust. Eine dritte Verwendungsart ist `data=writeback`. Hierbei können Daten in das Hauptdateisystem geschrieben werden, nachdem ihre Metadaten an das Journal übergeben wurden. Diese Option ist nach Meinung vieler aus Performancegründen die beste Einstellung. Jedoch kann es bei dieser Option passieren, dass alte Daten nach einem Absturz und einer Wiederherstellung in Dateien auftauchen, während die interne Dateisystemintegrität gewahrt wird. Sofern nicht anders angegeben, wird Ext3 mit der Standardeinstellung `data=ordered` gestartet.

Hinweis

Umwandeln eines Ext2-Dateisystems in Ext3

Anlegen des Journals: Rufen Sie `tune2fs -j` als Benutzer `root` auf. `tune2fs` legt das Ext3-Journal mit Standardparametern an. Möchten Sie selbst festlegen, wie groß und auf welchem Device das Journal angelegt werden soll, rufen Sie stattdessen `tune2fs -J` mit den beiden Parametern `size=` und `device=` auf. Mehr zu `tune2fs` entnehmen Sie der Manual-Page.

Festlegung des Dateisystemtyps in `/etc/fstab`

Damit das Ext3-Dateisystem auch als solches erkannt wird, öffnen Sie die Datei `/etc/fstab` und ändern Sie den Dateisystemtyp der betroffenen Partition von `ext2` in `ext3`. Nach dem nächsten Neustart des Systems ist Ihre Änderung wirksam.

Hinweis

ReiserFS

Offiziell stand eine der Hauptfunktionen der Kernel-Version 2.4, ReiserFS seit der SUSE LINUX-Version 6.4 als Kernel-Patch für 2.2.x SuSE-Kernel zur Verfügung. ReiserFS stammt von Hans Reiser und dem Namesys-Entwicklungsteam. ReiserFS hat sich als mächtige Alternative zu Ext2 profiliert. Seine größten Vorteile sind bessere Festplattenspeicherverwaltung, bessere Plattenzugriffsleistung und schnellere Wiederherstellung nach Abstürzen. Einen kleinen Wermutstropfen gibt es dennoch: ReiserFS legt großen Wert auf die Metadaten, jedoch nicht auf die Daten selbst. Die nächsten Generationen von ReiserFS werden Data-Journaling beinhalten (sowohl Metadaten als auch tatsächliche Daten werden in das Journal geschrieben) sowie geordnete Schreibzugriffe (siehe `data=ordered` unter Ext3). Die Stärken von ReiserFS im Detail:

Bessere Festplattenspeicherverwaltung

In ReiserFS werden alle Daten in einer Struktur namens B^* -balanced tree organisiert. Die Baumstruktur trägt zur besseren Festplattenspeicherverwaltung bei, da kleine Dateien direkt in den Blättern des B^* trees gespeichert werden können, statt sie an anderer Stelle zu speichern und einfach den Zeiger auf den tatsächlichen Ort zu verwalten. Zusätzlich dazu wird der Speicher nicht in Einheiten von 1 oder 4 kB zugewiesen, sondern in exakt der benötigten Einheit. Ein weiterer

Vorteil liegt in der dynamischen Vergabe von Inodes. Dies verschafft dem Dateisystem eine größere Flexibilität gegenüber herkömmlichen Dateisystemen, wie zum Beispiel Ext2, wo die Inode-Dichte zum Zeitpunkt der Erstellung des Dateisystems angegeben werden muss.

Bessere Festplattenzugriffsleistung

Bei kleinen Dateien werden Sie häufig bemerken können, dass sowohl die Dateidaten als auch die „stat_data“ (Inode)-Informationen nebeneinander gespeichert wurden. Ein einziger Festplattenzugriff reicht somit, um Sie mit allen benötigten Informationen zu versorgen.

Schnelle Wiederherstellung nach Abstürzen

Durch den Einsatz eines Journals zur Nachverfolgung kürzlicher Metadatenänderungen reduziert sich die Dateisystemüberprüfung sogar für große Dateisysteme auf wenige Sekunden.

JFS

JFS, das „Journaling File System“ wurde von IBM für AIX entwickelt. Die erste Betaversion des JFS-Linux-Ports erreichte die Linux-Gemeinde im Sommer 2000. Version 1.0.0 wurde im Jahre 2001 herausgegeben. JFS ist auf die Bedürfnisse von Server-Umgebungen mit hohem Durchsatz zugeschnitten, da hierbei einzig die Performance zählt. Als volles 64-Bit-Dateisystem unterstützt JFS große Dateien und Partitionen (LFS oder *Large File Support*), was ein weiterer Pluspunkt für den Einsatz in Server-Umgebungen ist.

Ein genauerer Blick auf JFS zeigt, warum dieses Dateisystem möglicherweise eine gute Wahl für Ihren Linux-Server darstellt:

Effizientes Journaling JFS folgt wie ReiserFS einem „metadata only“-Ansatz. Anstelle einer ausführlichen Überprüfung werden lediglich Metadatenänderungen überprüft, die durch kürzliche Dateisystemaktivitäten hervorgerufen wurden. Dies spart enorm viel Zeit bei der Wiederherstellung. Zeitgleiche Aktivitäten, die mehrere Protokolleinträge erfordern, können in einem Gruppen-Commit zusammengefasst werden, wobei der Leistungsverlust des Dateisystems durch mehrfachen Schreibvorgang stark verringert wird.

Effiziente Verzeichnisverwaltung JFS hält an unterschiedlichen Verzeichnisstrukturen fest. Bei kleinen Verzeichnissen erlaubt es die direkte Speicherung des Verzeichnisinhaltes in seinem Inode. Für größere

Verzeichnisse werden B^+ trees verwendet, welche die Verzeichnisverwaltung erheblich erleichtern.

Bessere Speichernutzung durch dynamische Vergabe der Inodes

Unter Ext2 müssen Sie die Inode-Dichte (von Verwaltungsinformationen belegter Speicher) vorab angeben. Dadurch wird die maximale Anzahl von Dateien oder Verzeichnissen Ihres Dateisystems limitiert. JFS erspart Ihnen diese Überlegungen — es weist Inode-Speicher dynamisch zu und stellt ihn bei Nichtbedarf wieder zur Verfügung.

XFS

Ursprünglich als Dateisystem für ihr IRIX-Betriebssystem gedacht, startete SGI die Entwicklung von XFS bereits in den frühen 90ern. Mit XFS sollte ein hochperformantes 64-Bit Journaling File System geschaffen werden, das den extremen Herausforderungen der heutigen Zeit gewachsen ist. XFS ist gut geeignet für den Umgang mit großen Dateien und zeigt gute Leistungen auf High-end-Hardware. Jedoch weist sogar XFS eine Schwäche auf. Wie ReiserFS, legt XFS großen Wert auf Metadatenintegrität und weniger auf Datenintegrität.

Ein kurzer Blick auf die Schlüsselfunktionen von XFS erklärt, warum es sich möglicherweise als starke Konkurrenz zu anderen Journaling File Systemen in der High-end-Datenverarbeitung herausstellen könnte.

Hohe Skalierbarkeit durch den Einsatz von „Allocation groups“

Zum Erstellungszeitpunkt eines XFS-Dateisystems wird das dem Dateisystem zugrundeliegende Block-Device in acht oder mehr lineare Bereiche gleicher Größe unterteilt. Diese werden als „allocation groups“ bezeichnet. Jede Allocation group verwaltet Inodes und freien Speicher selbst. Allocation groups können praktisch als „Dateisysteme im Dateisystem“ betrachtet werden. Da Allocation groups relativ autonom sind, kann der Kernel gleichzeitig mehrere von ihnen adressieren. Hier liegt der Schlüssel zur hohen Skalierbarkeit von XFS. Das Konzept der autonomen Allocation groups kommt natürlicherweise den Anforderungen von Multiprozessorsystemen entgegen.

Hohe Performance durch effiziente Festplattenspeicherverwaltung

Freier Speicher und Inodes werden von B^+ trees innerhalb der Allocation groups verwaltet. Der Einsatz von B^+ trees trägt zu einem Großteil zur Leistung und Skalierbarkeit von XFS bei. Ein wahrhaft

einzigartiges Funktionsmerkmal von XFS ist die „delayed allocation“. XFS verarbeitet die Speicherzuweisung (engl. *allocation*) durch Zerteilung des Prozesses. Eine „schwebende“ Transaktion wird in RAM gespeichert und der entsprechende Speicherplatz reserviert. XFS entscheidet noch nicht, wo genau (d.h. in welchen Dateisystemblöcken) die Daten gespeichert werden. Diese Entscheidung wird bis zum letztmöglichen Moment hinausgezögert. Einige kurzlebige, temporäre Daten werden somit niemals auf Platte gespeichert, da sie zum Zeitpunkt der Entscheidung über ihren Speicherort durch XFS bereits obsolet sind. So XFS erhöht die Leistung und verringert die Dateisystemfragmentation. Da allerdings eine verzögerte Zuordnung weniger Schreibvorgänge als in anderen Dateisystemen zur Folge hat, ist es wahrscheinlich, dass der Datenverlust nach einem Absturz während eines Schreibvorgangs größer ist.

Preallocation zur Vermeidung von Dateisystemfragmentation

Vor dem Schreiben der Daten in das Dateisystem reserviert XFS den benötigten Speicherplatz für eine Datei (engl. *preallocate*). Somit wird die Dateisystemfragmentation erheblich reduziert. Die Leistung wird erhöht, da die Dateiinhalte nicht über das gesamte Dateisystem verteilt werden.

Weitere unterstützte Dateisysteme

Tabelle A.1 enthält weitere von Linux unterstützte Dateisysteme. Sie werden hauptsächlich unterstützt, um die Kompatibilität und den Datenaustausch zwischen unterschiedlichen Medien oder fremden Betriebssystemen sicherzustellen.

Tabelle A.1: Dateisystemarten unter Linux

cramfs	<i>Compressed ROM file system</i> : ein komprimiertes Dateisystem mit Lesezugriff für ROMs.
hpfs	<i>High Performance File System</i> : das OS/2-Standarddateisystem — nur im Lesezugriffsmodus unterstützt.
iso9660	Standarddateisystem auf CD-ROMs.
ncpfs	Dateisystem zum Mounten von Novell-Volumes übers Netzwerk.

nfs	<i>Network File System</i> : Hierbei können Daten auf jedem beliebigen Rechner innerhalb eines Netzwerks gespeichert werden und der Zugriff kann über Netzwerk gewährt werden.
smbfs	<i>Server Message Block</i> : verwendet von Produkten wie zum Beispiel Windows für den Dateizugriff über ein Netzwerk.
sysv	verwendet unter SCO UNIX, Xenix und Coherent (kommerzielle UNIX-Systeme für PCs).
ufs	verwendet von BSD, SunOS und NeXTstep. Nur im Lesezugriffs-Modus unterstützt.
umsdos	<i>UNIX on MSDOS</i> : aufgesetzt auf einem normalen fat-Dateisystem. Erhält UNIX-Funktionalität (Rechte, Links, lange Dateinamen) durch die Erstellung spezieller Dateien.
vfat	<i>Virtual FAT</i> : Erweiterung des fat-Dateisystems (unterstützt lange Dateinamen).
ntfs	<i>Windows NT file system</i> : Lesezugriff.

Large File Support unter Linux

Ursprünglich unterstützte Linux Dateien bis zu einer maximalen Größe von 2 GB. Der zunehmende Einsatz von Linux zur Datenbankverwaltung, zur Verarbeitung von Audio- und Videodaten u.v.a.m. machten es nötig, Kernel und GNU C Library (*glibc*) für die Unterstützung größerer Dateien als 2 GB anzupassen. Es wurden neue Interfaces eingeführt, die von Applikationen genutzt werden können. Heutzutage bieten (fast) alle wichtigen Dateisysteme LFS-Unterstützung, die High-End-Datenverarbeitung erlaubt.

Tabelle A.2 bietet einen Überblick über die derzeitigen Beschränkungen von Linux-Dateien und Dateisystemen für Kernel 2.4.x.

Tabelle A.2: Maximale Größe von Dateisystemen (On-Disk Format)

Dateisystem	Max. Dateigröße	Max. Dateisystemgröße
Ext2 oder Ext3 (1 kB Blockgröße)	2 ³⁴ (16 GB)	2 ⁴¹ (2 TB)

Ext2 oder Ext3 (2 kB Blockgröße)	2^{38} (256 GB)	2^{43} (8 TB)
Ext2 oder Ext3 (4 kB Blockgröße)	2^{41} (2 TB)	2^{44} (16 TB)
Ext2 oder Ext3 (8 kB Blockgröße) (Systeme mit Pages von 8 kB (wie Alpha))	2^{46} (64 TB)	2^{45} (32 TB)
ReiserFS 3.5	2^{32} (4 GB)	2^{44} (16 TB)
ReiserFS 3.6 (unter Linux 2.4)	2^{60} (1 EB)	2^{44} (16 TB)
XFS	2^{63} (8 EB)	2^{63} (8 EB)
JFS (512 Bytes Blockgröße)	2^{63} (8 EB)	2^{49} (512 TB)
JFS (4 kB Blockgröße)	2^{63} (8 EB)	2^{52} (4 PB)
NFSv2 (clientseitig)	2^{31} (2 GB)	2^{63} (8 EB)
NFSv3 (clientseitig)	2^{63} (8 EB)	2^{63} (8 EB)

Hinweis

Linux Kernel Limits

Die Tabelle beschreibt die Limits des on-disk Formats. Die maximale Größe einer Datei und eines Dateisystems, die vom Kernel korrekt verarbeitet werden kann, unterliegt unter Kernel 2.6 folgenden Beschränkungen:

- *Dateigröße:* Dateien können auf 32-bit Systemen nicht größer sein als 2 TB (2^{41} Byte).
- *Dateisystemgröße:* Dateisysteme können bis zu 2^{73} Byte groß sein. Dieses Limit schöpft (noch) keine aktuelle Hardware aus.

Hinweis

Weitere Informationen

Jedes der oben beschriebenen Dateisystemprojekte unterhält seine eigene Homepage, wo Sie Informationen aus Mailinglisten und weitere Dokumentation sowie FAQs erhalten.

- <http://e2fsprogs.sourceforge.net/ext2.html>
- <http://www.zipworld.com.au/~akpm/linux/ext3/>
- <http://www.namesys.com/>
- <http://oss.software.ibm.com/developerworks/opensource/jfs/>
- oss.sgi.com/projects/xfs/

Ein umfassendes mehrteiliges Tutorial zu Linux-Dateisystemen findet sich unter *IBM developerWorks*: <http://www-106.ibm.com/developerworks/library/l-fs.html>

Einen Vergleich der verschiedenen Journaling File Systeme unter Linux befindet sich im Beitrag von Juan I. Santos Florido unter *Linuxgazette*: <http://www.linuxgazette.com/issue55/florido.html>

Eine ausführliche Arbeit zu LFS unter Linux erhält man auf Andreas Jaegers LFS-Seiten: http://www.suse.de/~aj/linux_lfs.html

Access Control Lists unter Linux

Dieses Kapitel gibt einen kurzen Einblick in die Hintergründe und Funktionsweise von POSIX ACLs für Linux-Dateisysteme. Sie erfahren, wie das traditionelle Rechtekonzept für Dateisystemobjekte mit Hilfe von ACLs (*Access Control Lists*) erweitert wird und welche Vorteile dieses Konzept bietet.

Warum ACLs?

Hinweis

POSIX ACLs

Der Ausdruck *POSIX ACL* suggeriert, dass es sich um einen echten Standard aus der POSIX (*Portable Operating System Interface*) Familie handelt. Aus verschiedenen Gründen wurden die betreffenden Standardentwürfe POSIX 1003.1e und POSIX 1003.2c zurückgezogen. ACLs unter vielen UNIX-artigen Betriebssystemen basieren allerdings auf diesen Entwürfen. Die in diesem Kapitel beschriebene Implementierung von Dateisystem ACLs folgt den Inhalten dieser beiden Dokumente, die Sie unter folgender URL einsehen können: <http://wt.xpilot.org/publications/posix.1e/>

Hinweis

Traditionell sind für jedes Dateiojekt unter Linux drei Sets von Berechtigungen definiert. Diese Sets geben die Lese- (r), Schreib- (w) und Ausführungsrechte (x) für die drei Benutzerklassen Eigentümer der Datei (engl.

owner), Gruppe (engl. *group*) und „Rest der Welt“ (engl. *other*) wieder. Zusätzlich können noch die *set user id*, *set group id* und *sticky* Bits gesetzt werden. Mehr zu diesem Thema finden Sie im *Benutzerhandbuch* im Abschnitt *Benutzer und Zugriffsrechte*.

Für die meisten in der Praxis auftretenden Fälle reicht dieses schlanke Konzept völlig aus. Für komplexere Szenarien oder fortgeschrittenere Anwendungen mussten Systemadministratoren zuvor eine Reihe von Tricks anwenden, um die Einschränkungen des traditionellen Rechtekonzepts zu umgehen.

In Situationen, in denen das traditionelle Dateirechte-Konzept nicht ausreicht, helfen ACLs. Sie erlauben es, einzelnen Benutzern oder Gruppen Rechte zuzuweisen, auch wenn diese nicht mit dem Eigentümer oder der Gruppe einer Datei übereinstimmen.

Access Control Lists sind ein Feature des Linux-Kernels und werden zur Zeit von ReiserFS, Ext2, Ext3, JFS und XFS unterstützt. Mit ihrer Hilfe können komplexe Szenarien umgesetzt werden, ohne dass auf Applikationsebene komplexe Rechtemodelle implementiert werden müssten.

Ein prominentes Beispiel für die Vorzüge von Access Control Lists ist der Austausch eines Windows-Servers gegen einen Linux-Server. Manche der angeschlossenen Workstations werden auch nach dem Umstieg weiter unter Windows betrieben werden. Das Linux-System bietet den Windows-Clients via Samba Datei- und Druckserver-Dienste an.

Da Samba Access Control Lists unterstützt, können Benutzerrechte sowohl auf dem Linux-Server als auch über eine grafische Benutzeroberfläche unter Windows (nur Windows NT und höher) eingerichtet werden. Über den *winbindd* ist es sogar möglich, Benutzern Rechte einzuräumen, die nur in der Windows-Domain existieren und über keinen Account auf dem Linux-Server verfügen. Auf der Serverseite können Sie die Access Control Lists mithilfe von *getfacl* und *setfacl* bearbeiten.

Definitionen

Benutzerklassen Das herkömmliche POSIX Rechtekonzept kennt drei *Klassen* von Benutzern für die Rechtevergabe im Dateisystem: Eigentümer (engl. *owner*), Gruppe (engl. *group*) und andere Benutzer oder den „Rest der Welt“ (engl. *other*). Pro Benutzerklasse lassen sich jeweils die drei Berechtigungsbits (engl. *permission bits*) für Lesezugriff (r), für Schreibzugriff (w) und für Ausführbarkeit (x) vergeben.

Eine Einführung in das Benutzerkonzept unter Linux finden Sie im *Benutzerhandbuch* im Abschnitt *Benutzer und Zugriffsrechte*.

Access ACL Die Zugriffsrechte für Benutzer und Gruppen auf beliebige Dateisystemobjekte (Dateien und Verzeichnisse) werden über Access ACLs (dt. *Zugriffs-ACLs*) festgelegt.

Default ACL Default ACLs (dt. *Vorgabe-ACLs*) können nur auf Verzeichnisse angewandt werden und legen fest, welche Rechte ein Dateisystemobjekt von seinem übergeordneten Verzeichnis beim Anlegen erbt.

ACL-Eintrag Jede ACL besteht aus einem Satz von ACL-Einträgen (engl. *ACL entries*). Ein ACL-Eintrag hat einen Typ (siehe Tabelle B.1 auf der nächsten Seite), einen Bezeichner für den Benutzer oder die Gruppe, auf die sich dieser Eintrag bezieht, und Berechtigungen. Der Bezeichner für Gruppe oder Benutzer bleibt für einige Typen von Einträgen leer.

Umgang mit ACLs

Im folgenden Abschnitt lernen Sie den Grundaufbau einer ACL und deren verschiedene Ausprägungen kennen. Der Zusammenhang zwischen ACLs und dem traditionellen Rechtekonzept im Linux-Dateisystem wird anhand mehrerer Grafiken kurz erläutert. An zwei Beispielen lernen Sie, selbst ACLs zu erstellen und deren korrekte Syntax zu beachten. Zuletzt erfahren Sie, nach welchem Muster ACLs vom Betriebssystem ausgewertet werden.

Aufbau von ACL-Einträgen

ACLs werden grundsätzlich in zwei Klassen eingeteilt. Eine *minimale* ACL besteht ausschließlich aus den Einträgen vom Typ *owner* (Besitzer), *owning group* (Besitzergruppe) und *other* (Andere), und entspricht den herkömmlichen Berechtigungsbits für Dateien und Verzeichnisse. Eine *erweiterte* (engl. *extended*) ACL geht über dieses Konzept hinaus. Sie muss einen *mask* (Maske) Eintrag enthalten und darf mehrere Einträge des Typs *named user* (namentlich gekennzeichnete Benutzer) und *named group* (namentlich gekennzeichnete Gruppe) enthalten. Tabelle B.1 auf der nächsten Seite fasst die verschiedenen verfügbaren Typen von ACL-Einträgen zusammen.

Tabelle B.1: Überblick: Typen von ACL-Einträgen

Typ	Textform
owner	user::rwx
named user	user:name:rwx
owning group	group::rwx
named group	group:name:rwx
mask	mask::rwx
other	other::rwx

In den Einträgen *owner* und *other* festgelegte Rechte sind immer wirksam. Vom *mask* Eintrag abgesehen, können alle übrigen Einträge (*named user*, *owning group* und *named group*) entweder wirksam oder maskiert werden. Sind Rechte sowohl in einem der oben genannten Einträge als auch in der Maske vorhanden, werden sie wirksam. Rechte, die nur in der Maske oder nur im eigentlichen Eintrag vorhanden sind, sind nicht wirksam. Das nachfolgende Beispiel verdeutlicht diesen Mechanismus (siehe Tabelle B.2):

Tabelle B.2: Maskierung von Zugriffsrechten

Typ	Textform	Rechte
named user	user:jane:r-x	r-x
mask	mask::rw-	rw-
	Wirksame Berechtigungen:	r--

ACL-Einträge und Berechtigungsbits

Die beiden Abbildungen illustrieren die beiden auftretenden Fälle einer minimalen und einer erweiterten ACL (siehe Abb. B.1 auf der nächsten Seite und B.2 auf der nächsten Seite). Die Abbildungen gliedern sich in drei Blöcke. Links die Typangaben der ACL-Einträge, in der Mitte eine beispielhafte ACL und rechts die entsprechenden Berechtigungsbits, wie sie auch `ls -l` anzeigt.

In beiden Fällen werden die *owner class* Berechtigungen dem *owner* ACL-Eintrag zugeordnet. Die Zuordnung der *other class* Berechtigungen zum

entsprechenden ACL-Eintrag ist ebenfalls konstant. Die Zuordnung der *group class* Berechtigungen variiert:

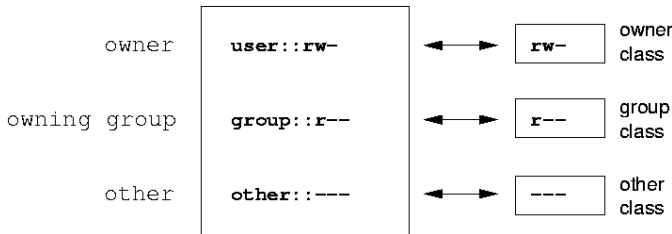


Abbildung B.1: Minimale ACL: ACL-Einträge vs. Berechtigungsbits

- Im Fall einer minimalen ACL — ohne *mask* Eintrag — werden die *group class* Berechtigungen dem *owning group* ACL-Eintrag zugeordnet (siehe Abb. B.1).
- Im Fall einer erweiterten ACL — mit *mask* Eintrag — werden die *group class* Berechtigungen dem *mask* Eintrag zugeordnet (siehe Abb. B.2).

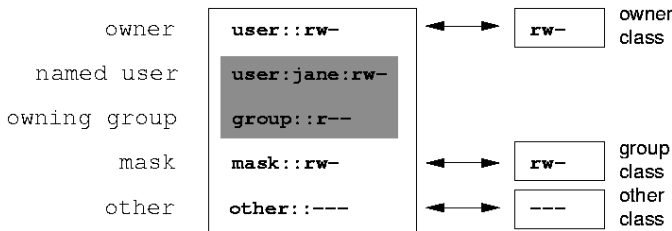


Abbildung B.2: Erweiterte ACL: ACL-Einträge vs. Berechtigungsbits

Durch diese Art der Zuordnung ist die reibungslose Interaktion von Anwendungen mit und ohne ACL-Unterstützung gewährleistet. Die Zugriffsrechte, die mittels der Berechtigungsbits festgelegt wurden, sind die Obergrenze für alle anderen „Feineinstellungen“, die per ACL gemacht werden. Alle Rechte, die hier nicht wiederspiegelt sind, wurden entweder in der ACL nicht gesetzt oder sind nicht effektiv. Werden Berechtigungsbits geändert, spiegelt sich dies in der ACL wider und umgekehrt.

Ein Verzeichnis mit Access ACL

In drei Schritten werden Sie anhand folgendem Beispiel den Umgang mit einer Access ACL lernen:

- Anlegen eines Dateisystemobjekts (hier eines Verzeichnisses)
- Änderungen an der ACL
- Einsatz von Masken

1. Bevor Sie das Verzeichnis anlegen, können Sie mittels des `umask` Befehls festlegen, welche Zugriffsrechte gleich bei der Erstellung maskiert werden sollen:

```
umask 027
```

`umask 027` beschränkt die Rechte der einzelnen Benutzergruppen folgendermaßen: der Besitzer der Datei behält sämtliche Rechte (0), die Besitzergruppe darf nicht schreibend auf die Datei zugreifen (2) und alle anderen Benutzer erhalten keinerlei Zugriff (7). Die Zahlen sind als Bitmaske zu lesen. Details zu `umask` entnehmen Sie der entsprechenden Manualpage (`man umask`).

```
mkdir mydir
```

Das Verzeichnis `mydir/` ist angelegt und hat die durch die `umask` festgelegten Rechte erhalten. Mit

```
ls -dl mydir
```

```
drwxr-x--- ... tux projekt3 ... mydir
```

überprüfen Sie, ob alle Rechte korrekt vergeben wurden.

2. Nachdem Sie sich über den Ausgangszustand der ACL informiert haben, fügen Sie ihr jeweils einen neuen Benutzer- und Gruppen-Eintrag hinzu.

```
getfacl mydir
```

```
# file: mydir
# owner: tux
# group: projekt3
user::rwx
group::r-x
other::---
```

Die Ausgabe von `getfacl` spiegelt exakt die unter Abschnitt B auf Seite 592 beschriebene Zuordnung von Berechtigungsbits und ACL-Einträgen wider. Die ersten drei Zeilen der Ausgabe nennen Namen, Eigentümer und zugehörige Gruppe des Verzeichnisses. Die drei nächsten Zeilen enthalten die drei ACL-Einträge *owner*, *owning group* und *other*. Insgesamt liefert Ihnen der `getfacl` Befehl im Fall dieser einfachsten („minimalen“) ACL keine Information, die Sie mittels `ls` nicht auch erhalten hätten.

Ihr erster Eingriff in die ACL besteht darin, einem zusätzlichen Benutzer *jane* und einer zusätzlichen Gruppen *djungle* Lese-, Schreib- und Ausführrechte zu gewähren.

```
setfacl -m user:jane:rw,group:djungle:rw mydir
```

Die Option `-m` weist `setfacl` an, die bestehende ACL zu modifizieren. Das nachfolgende Argument gibt an, welche ACL-Einträge geändert werden (mehrere werden durch Kommata voneinander getrennt). Abschließend geben Sie den Namen des Verzeichnisses an, für das diese Änderungen gelten sollen.

Die resultierende ACL geben Sie wieder mit `getfacl` aus.

```
# file: mydir
# owner: tux
# group: projekt3
user::rw
user:jane:rw
group::r-x
group:djungle:rw
mask::rw
other::---
```

Zusätzlich zu den von Ihnen initiierten Einträgen für den Benutzer *jane* und die Gruppe *djungle* wurde ein *mask* Eintrag erzeugt. Dieser *mask* Eintrag wird automatisch gesetzt, um alle Einträge in der *group class* auf einen gemeinsamen Nenner zu bringen. Außerdem passt `setfacl` bestehende *mask* Einträge an von Ihnen geänderte Einstellungen automatisch an, so Sie das nicht mit `-n` deaktivieren. *mask* legt die maximal wirksamen Zugriffsrechte für alle Einträge innerhalb der *group class* fest. Dies beinhaltet: *named user*, *named group* und *owning group*. Die *group class* Berechtigungsbits, die ein `ls -dl mydir` ausgeben würde, entsprechen jetzt dem *mask*-Eintrag.

```
ls -dl mydir
```

```
drwxrwx---+ ... tux projekt3 ... mydir
```

Es erscheint in der ersten Spalte der Ausgabe ein +, das auf eine *erweiterte* ACL hinweist.

3. Gemäß der Ausgabe des `ls` Kommandos beinhalten die Rechte für den *mask* Eintrag auch Schreibzugriff. Traditionell würden diese Berechtigungsbits auch darauf hinweisen, dass die *owning group* (hier: projekt3) ebenfalls Schreibzugriff auf das Verzeichnis `mydir/` hätte. Allerdings sind die wirklich wirksamen Zugriffsrechte für die *owning group* als die Schnittmenge aus den für *owning group* und *mask* gesetzten Rechten definiert; also in unserem Beispiel `r-x` (siehe Tabelle B.2 auf Seite 592). Es hat sich auch nach Hinzufügen der ACL-Einträge nichts an den Rechten der *owning group* geändert.

Verändern können Sie den *mask* Eintrag mittels `setfacl` oder `chmod`.

```
chmod g-w mydir
```

```
ls -dl mydir
```

```
drwxr-x---+ ... tux projekt3 ... mydir
```

```
getfacl mydir
```

```
# file: mydir
# owner: tux
# group: projekt3
user::rwx
user:jane:rwx           # effective: r-x
group::r-x
group:djungle:rwx       # effective: r-x
mask::r-x
other::---
```

Nachdem Sie per `chmod` die *group class* Bits um den Schreibzugriff verringert haben, liefert Ihnen schon die Ausgabe des `ls` Kommandos den Hinweis darauf, dass die *mask* Bits über `chmod` entsprechend angepasst wurden. Man erkennt, dass nur der Besitzer Schreibberechtigung im Verzeichnis `mydir/` hat. Noch deutlicher wird dies an der Ausgabe von `getfacl`. `getfacl` fügt für alle Einträge Kommentare hinzu, deren tatsächlich wirksame Berechtigungsbits nicht mit den ursprünglich gesetzten übereinstimmen, weil sie vom *mask* Eintrag

herausgefiltert werden. Sie können den Ausgangszustand mit dem entsprechenden `chmod` Kommando wiederherstellen:

```
chmod g+w mydir
ls -dl mydir

drwxrwx---+ ... tux projekt3 ... mydir

getfacl mydir

# file: mydir
# owner: tux
# group: projekt3
user::rwx
user:jane:rwx
group::r-x
group:djungle:rwx
mask::rwx
other:---
```

Ein Verzeichnis mit Default ACL

Verzeichnisse können mit einer besonderen Art von ACLs versehen werden; einer Default ACL. Diese Default ACL legt fest, welche Zugriffsrechte Unterobjekte dieses Verzeichnisses zum Zeitpunkt ihrer Erstellung erben. Eine Default ACL wirkt sich auf Unterverzeichnisse ebenso wie auf Dateien aus.

Auswirkungen einer Default ACL

Die Zugriffsrechte in einer Default ACL werden an Dateien und Unterverzeichnisse unterschiedlich vererbt:

- Ein Unterverzeichnis erbt die Default ACL des übergeordneten Verzeichnisses sowohl als seine eigene Default ACL als auch als Access ACL.
- Eine Datei erbt die Default ACL als ihre eigene Access ACL.

Alle Systemaufrufe (engl. *system calls*), die Dateisystemobjekte anlegen, verwenden einen `mode` Parameter. Der `mode` Parameter legt die Zugriffsrechte auf das neu anzulegende Dateisystemobjekt fest:

- Hat das übergeordnete Verzeichnis keine Default ACL, ergeben sich die Berechtigungen aus den im `mode`-Parameter angegebenen Berechtigungen, von denen die in der `umask` gesetzten Rechte abgezogen werden.
- Existiert eine Default ACL für das übergeordnete Verzeichnis, werden die Berechtigungsbits entsprechend der Schnittmenge aus dem Wert des `mode` Parameters und den in der Default ACL festgelegten Berechtigungen zusammengesetzt und dem Objekt zugewiesen. Die `umask` wird in diesem Fall nicht beachtet.

Default ACLs in der Praxis

Die drei folgenden Beispiele führen Sie an die wichtigsten Operationen an Verzeichnissen und Default ACLs heran:

- Anlegen einer Default ACL für ein bestehendes Verzeichnis
- Anlegen eines Unterverzeichnisses in einem Verzeichnis mit Default ACL
- Anlegen einer Datei in einem Verzeichnis mit Default ACL

1. Sie fügen dem schon existierenden Verzeichnis `mydir/` eine Default ACL hinzu:

```
setfacl -d -m group:djungle:r-x mydir
```

Die `-d` Option des `setfacl` Kommandos weist `setfacl` an, die folgenden Modifikationen (Option `-m`) auf der Default ACL vorzunehmen.

Sehen Sie sich das Ergebnis dieses Befehls genauer an:

```
getfacl mydir

# file: mydir
# owner: tux
# group: projekt3
user::rwx
user:jane:rwx
group::r-x
group:djungle:rwx
mask::rwx
other:---
```

```
default:user::rwx
default:group::r-x
default:group:djungle:r-x
default:mask::r-x
default:other:---
```

getfacl liefert sowohl die Access ACL als auch die Default ACL zurück. Alle Zeilen, die mit default beginnen, bilden zusammen die Default ACL. Obwohl Sie dem setfacl Befehl lediglich einen Eintrag für die Gruppe djungle in die Default ACL mitgegeben hatten, hat setfacl automatisch alle anderen Einträge aus der Access ACL kopiert, um so eine gültige Default ACL zu bilden. Default ACLs haben keinen direkten Einfluss auf die Zugriffsberechtigungen und wirken sich nur beim Erzeugen von Dateisystemobjekten aus. Beim Vererben wird nur die Default ACL des übergeordneten Verzeichnisses beachtet.

2. Legen Sie im nächsten Beispiel mit mkdir ein Unterverzeichnis in mydir/ an, welches die Default ACL „erben“ wird.

```
mkdir mydir/mysubdir
getfacl mydir/mysubdir

# file: mydir/mysubdir
# owner: tux
# group: projekt3
user::rwx
group::r-x
group:djungle:r-x
mask::r-x
other:---
default:user::rwx
default:group::r-x
default:group:djungle:r-x
default:mask::r-x
default:other:---
```

Wie erwartet, hat das neu angelegte Unterverzeichnis mysubdir/ die Rechte aus der Default ACL des übergeordneten Verzeichnisses. Die Access ACL von mysubdir/ ist ein exaktes Abbild der Default ACL von mydir/, ebenso die Default ACL, die dieses Verzeichnis wiederum an seine Unterobjekte weitervererben wird.

3. Legen Sie im mydir/ Verzeichnis mit touch eine Datei an:

```
touch mydir/myfile
ls -l mydir/myfile

-rw-r-----+ ... tux projekt3 ... mydir/myfile

getfacl mydir/myfile

# file: mydir/myfile
# owner: tux
# group: projekt3
user::rw-
group::r-x      # effective:r--
group:djungle:r-x  # effective:r--
mask::r--
other::---
```

Wichtig an diesem Beispiel: `touch` übergibt `mode` mit dem Wert von `0666`, das bedeutet, dass neue Dateien mit Lese- und Schreibrechten für alle Benutzerklassen angelegt werden, so nicht entweder per `umask` oder Default ACL andere Beschränkungen existieren (siehe Abschnitt B auf Seite 597).

Am konkreten Beispiel heißt dies, dass alle Zugriffsrechte, die nicht im `mode` Wert enthalten sind, aus den entsprechenden ACL-Einträgen entfernt werden: Aus dem ACL-Eintrag der *group class* wurden keine Berechtigungen entfernt, allerdings wurde der *mask* Eintrag dahingehend angepasst, dass nicht per `mode` gesetzte Berechtigungsbits maskiert werden.

Auf diese Weise ist sichergestellt, dass zum Beispiel Compiler reibungslos mit ACLs interagieren können. Sie können Dateien mit beschränkten Zugriffsrechten anlegen und diese anschließend als ausführbar markieren. Über den `mask` Mechanismus ist gewährleistet, dass die richtigen Benutzer und Gruppen schließlich die Datei ausführen können.

Auswertung einer ACL

Nachdem Sie den Umgang mit den wichtigsten Tools zur ACL-Konfiguration bereits verstanden haben, werden Sie im Folgenden kurz an den Auswertungsalgorithmus herangeführt, den jeder Prozess oder jede Anwendung durchlaufen muss, bevor ihm Zugriff auf ein ACL-geschütztes Dateisystemobjekt gewährt werden kann.

Grundsätzlich werden die ACL-Einträge in folgender Reihenfolge untersucht: *owner*, *named user*, *owning group* oder *named group* und *other*. Über den Eintrag, der am besten auf den Prozess passt, wird schließlich der Zugang geregelt.

Komplizierter werden die Verhältnisse, wenn ein Prozess zu mehr als einer Gruppe gehört, also potentiell auch mehrere *group* Einträge passen könnten. Aus den passenden Einträgen mit den erforderlichen Rechten wird ein beliebiger ausgesucht. Für das Endresultat „Zugriff gewährt“ ist es natürlich unerheblich, welcher dieser Einträge den Ausschlag gegeben hat. Enthält keiner der passenden *group* Einträge die korrekten Rechten, gibt wiederum ein beliebiger von ihnen den Ausschlag für das Endresultat "Zugriff verweigert".

Unterstützung in Anwendungen

Wie in den vorangehenden Abschnitten beschrieben, können mit ACLs sehr anspruchsvolle Rechteszenarien umgesetzt werden, die modernen Anwendungen gerecht werden. Das traditionelle Rechtekonzept und ACLs lassen sich geschickt miteinander vereinbaren.

Allerdings fehlt einigen wichtigen Anwendungen noch die Unterstützung für ACLs. Insbesondere auf dem Gebiet der Backup-Anwendungen gibt es mit Ausnahme des *stcr* Archivierers keine Programme, die den vollen Erhalt der ACLs sicherstellen.

Die grundlegenden Dateikommandos (*cp*, *mv*, *ls*, ...) unterstützen ACLs. Viele Editoren und Dateimanager (z.B. Konqueror) beinhalten jedoch keine ACL-Unterstützung. Beim Kopieren von Dateien mit Konqueror gehen zur Zeit noch die ACLs verloren. Wenn eine Datei mit einer Access ACL im Editor bearbeitet wird, hängt es vom Backup-Modus des verwendeten Editors ab, ob die Access ACL nach Abschluss der Bearbeitung weiterhin vorliegt:

- Schreibt der Editor die Änderungen in die Originaldatei, bleibt die Access ACL erhalten.
- Legt der Editor eine neue Datei an, die nach Abschluss der Änderungen in die alte umbenannt wird, gehen die ACLs möglicherweise verloren, es sein denn, der Editor unterstützt ACLs.

Hinweis

Weitere Informationen

Detailinformationen zu ACLs finden Sie unter den folgenden URLs
http://sdb.suse.de/de/sdb/html/81_acl.html <http://acl.bestbits.at/> und auf den Manualpages von `getfacl`,
`acl` und `setfacl`.

Hinweis

Manual-Page von e2fsck

E2FSCK(8)

E2FSCK(8)

NAME

e2fsck - check a Linux second extended file system

SYNOPSIS

```
e2fsck [ -pacnyrdfvstDFSV ] [ -b superblock ] [ -B block
size ] [ -l|-L bad_blocks_file ] [ -C fd ] [ -j external-
journal ] [ -E extended_options ] device
```

DESCRIPTION

e2fsck is used to check a Linux second extended file system (ext2fs). E2fsck also supports ext2 filesystems containing a journal, which are also sometimes known as ext3 filesystems, by first applying the journal to the filesystem before continuing with normal e2fsck processing. After the journal has been applied, a filesystem will normally be marked as clean. Hence, for ext3 filesystems, e2fsck will normally run the journal and exit, unless its superblock indicates that further checking is required.

device is the device file where the filesystem is stored (e.g. /dev/hdc1).

OPTIONS

-a This option does the same thing as the -p option. It is provided for backwards compatibility only; it is suggested that people use -p option whenever possible.

-b superblock

Instead of using the normal superblock, use an alternative superblock specified by superblock. This option is normally used when the primary superblock has been corrupted. The location of the

backup superblock is dependent on the filesystem's blocksize. For filesystems with 1k block sizes, a backup superblock can be found at block 8193; for filesystems with 2k block sizes, at block 16384; and for 4k block sizes, at block 32768.

Additional backup superblocks can be determined by using the mke2fs program using the -n option to print out where the superblocks were created. The -b option to mke2fs, which specifies blocksize of the filesystem must be specified in order for the superblock locations that are printed out to be accurate.

If an alternative superblock is specified and the filesystem is not opened read-only, e2fsck will make sure that the primary superblock is updated appropriately upon completion of the filesystem check.

-B blocksize

Normally, e2fsck will search for the superblock at various different block sizes in an attempt to find the appropriate block size. This search can be fooled in some cases. This option forces e2fsck to only try locating the superblock at a particular blocksize. If the superblock is not found, e2fsck will terminate with a fatal error.

-c

This option causes e2fsck to run the badblocks(8) program to find any blocks which are bad on the filesystem, and then marks them as bad by adding them to the bad block inode. If this option is specified twice, then the bad block scan will be done using a non-destructive read-write test.

-C fd

This option causes e2fsck to write completion information to the specified file descriptor so that the progress of the filesystem check can be monitored. This option is typically used by programs which are running e2fsck. If the file descriptor specified is 0, e2fsck will print a completion bar as it goes about its business. This requires that e2fsck is running on a video console or terminal.

-d

Print debugging output (useless unless you are debugging e2fsck).

-D

Optimize directories in filesystem. This option causes e2fsck to try to optimize all directories, either by reindexing them if the filesystem supports directory indexing, or by sorting and com

pressing directories for smaller directories, or for filesystems using traditional linear directories.

-E extended_options

Set e2fsck extended options. Extended options are comma separated, and may take an argument using the equals ('=') sign. The following options are supported:

ea_ver=extended_attribute_version

Assume the format of the extended attribute blocks in the filesystem is the specified version number. The version number may be 1 or 2. The default extended attribute version format is 2.

-f Force checking even if the file system seems clean.

-F Flush the filesystem device's buffer caches before beginning. Only really useful for doing e2fsck time trials.

-j external-journal

Set the pathname where the external-journal for this filesystem can be found.

-l filename

Add the block numbers listed in the file specified by filename to the list of bad blocks. The format of this file is the same as the one generated by the badblocks(8) program. Note that the block numbers are based on the blocksize of the filesystem. Hence, badblocks(8) must be given the blocksize of the filesystem in order to obtain correct results. As a result, it is much simpler and safer to use the -c option to e2fsck, since it will assure that the correct parameters are passed to the badblocks program.

-L filename

Set the bad blocks list to be the list of blocks specified by filename. (This option is the same as the -l option, except the bad blocks list is cleared before the blocks listed in the file are added to the bad blocks list.)

-n Open the filesystem read-only, and assume an answer of 'no' to all questions. Allows e2fsck to be used non-interactively. (Note: if the -c, -l, or -L options are specified in addition to the -n option, then the filesystem will be opened read-write, to permit the bad-blocks list to be updated. However,

no other changes will be made to the filesystem.)

- p Automatically repair ("preen") the file system without any questions.
- r This option does nothing at all; it is provided only for backwards compatibility.
- s This option will byte-swap the filesystem so that it is using the normalized, standard byte-order (which is i386 or little endian). If the filesystem is already in the standard byte-order, e2fsck will take no action.
- S This option will byte-swap the filesystem, regardless of its current byte-order.
- t Print timing statistics for e2fsck. If this option is used twice, additional timing statistics are printed on a pass by pass basis.
- v Verbose mode.
- V Print version information and exit.
- y Assume an answer of 'yes' to all questions; allows e2fsck to be used non-interactively.

EXIT CODE

The exit code returned by e2fsck is the sum of the following conditions:

- 0 - No errors
- 1 - File system errors corrected
- 2 - File system errors corrected, system should be rebooted
- 4 - File system errors left uncorrected
- 8 - Operational error
- 16 - Usage or syntax error
- 32 - E2fsck canceled by user request
- 128 - Shared library error

SIGNALS

The following signals have the following effect when sent to e2fsck.

SIGUSR1

This signal causes e2fsck to start displaying a completion bar. (See discussion of the -C option.)

SIGUSR2

This signal causes e2fsck to stop displaying a completion bar.

REPORTING BUGS

Almost any piece of software will have bugs. If you manage to find a filesystem which causes e2fsck to crash, or which e2fsck is unable to repair, please report it to the author.

Please include as much information as possible in your bug report. Ideally, include a complete transcript of the e2fsck run, so I can see exactly what error messages are displayed. If you have a writeable filesystem where the transcript can be stored, the script(1) program is a handy way to save the output of e2fsck to a file.

It is also useful to send the output of dumpe2fs(8). If a specific inode or inodes seems to be giving e2fsck trouble, try running the debugfs(8) command and send the output of the stat(1u) command run on the relevant inode(s). If the inode is a directory, the debugfs dump command will allow you to extract the contents of the directory inode, which can sent to me after being first run through uuen code(1).

Always include the full version string which e2fsck displays when it is run, so I know which version you are running.

AUTHOR

This version of e2fsck was written by Theodore Ts'o
<tytso@mit.edu>.

SEE ALSO

mke2fs(8), tune2fs(8), dumpe2fs(8), debugfs(8)

E2fsprogs version 1.34

July 2003

E2FCK(8)

Manual-Page von reiserfsck

REISERFSCK(8)

REISERFSCK(8)

NAME

reiserfsck - check a Linux Reiserfs file system

SYNOPSIS

```
reiserfsck [ -afprVy ] [ --rebuild-sb | --check | --fix-
fixable | --rebuild-tree | --clean-attributes ] [ -j |
--journal device ] [ -z | --adjust-size ] [ -n | --nolog ]
[ -l | --logfile file ] [ -q | --quiet ] [ -y | --yes ] [
-S | --scan-whole-partition ] [ --no-journal-available ]
device
```

DESCRIPTION

Reiserfsck searches for a Reiserfs filesystem on a device, replays any necessary transactions, and either checks or repairs the file system.

device is the special file corresponding to the device or partition (e.g /dev/hdXX for IDE disk partition or /dev/sdXX for SCSI disk partition).

OPTIONS

--rebuild-sb

This option recovers the superblock on a Reiserfs partition. Normally you only need this option if mount reports "read_super_block: can't find a reiserfs file system" and you are sure that a Reiserfs file system is there.

--check

This default action checks file system consistency and reports but does not repair any corruption that it finds. This option may be used on a read-only file system mount.

--fix-fixable

This option recovers certain kinds of corruption that do not require rebuilding the entire file system tree (**--rebuild-tree**). Normally you only need this option if the **--check** option reports "corruption that can be fixed with **--fix-fixable**". This includes: zeroing invalid data-block pointers, correcting **st_size** and **st_blocks** for directories, and deleting invalid directory entries.

--rebuild-tree

This option rebuilds the entire file system tree using leaf nodes found on the device. Normally you only need this option if the **--check** option reports "corruption that can be fixed only during **--rebuild-tree**". You are strongly encouraged to make a backup copy of the whole partition before attempting the **--rebuild-tree** option.

--clean-attributes

This option cleans reserved fields of Stat-Data items.

--journal device , -j device

This option supplies the device name of the current file system journal. This option is required when the journal resides on a separate device from the main data device (although it can be avoided with the expert option **--no-journal-available**).

--adjust-size, -z

This option causes **reiserfsck** to correct file sizes that are larger than the offset of the last discovered byte. This implies that holes at the end of a file will be removed. File sizes that are smaller than the offset of the last discovered byte are corrected by **--fix-fixable**.

- `--logfile file, -l file`
This option causes reiserfsck to report any corruption it finds to the specified log file rather than `stderr`.
- `--nolog, -n`
This option prevents reiserfsck from reporting any kinds of corruption.
- `--quiet, -q`
This option prevents reiserfsck from reporting its rate of progress.
- `--yes, -y`
This option inhibits reiserfsck from asking you for confirmation after telling you what it is going to do, assuming yes. For safety, it does not work with the `--rebuild-tree` option.
- `-a, -p` These options are usually passed by `fsck -A` during the automatic checking of those partitions listed in `/etc/fstab`. These options cause reiserfsck to print some information about the specified file system, check if error flags in the superblock are set and do some light-weight checks. If these checks reveal a corruption or the flag indicating a (possibly fixable) corruption is found set in the superblock, then reiserfsck switches to the fix-fixable mode. If the flag indicating a fatal corruption is found set in the superblock, then reiserfsck finishes with an error.
- `-V` This option prints the reiserfsprogs version and exit.
- `-r, -f` These options are ignored.

EXPERT OPTIONS

DO NOT USE THESE OPTIONS UNLESS YOU KNOW WHAT YOU ARE DOING. WE ARE NOT RESPONSIBLE IF YOU LOSE DATA AS A RESULT OF THESE OPTIONS.

- `--no-journal-available`
This option allows reiserfsck to proceed when the journal device is not available. This option has no

effect when the journal is located on the main data device. NOTE: after this operation you must use reiserfstune to specify a new journal device.

`--scan-whole-partition, -S`

This option causes `--rebuild-tree` to scan the whole partition, not only used space on the partition.

EXAMPLE OF USING

1. You think something may be wrong with a reiserfs partition on `/dev/hda1` or you would just like to perform a periodic disk check.
2. Run `reiserfsck --check --logfile check.log /dev/hda1`. If `reiserfsck --check` exits with status 0 it means no errors were discovered.
3. If `reiserfsck --check` exits with status 1 (and reports about fixable corruptions) it means that you should run `reiserfsck --fix-fixable --logfile fixable.log /dev/hda1`.
4. If `reiserfsck --check` exits with status 2 (and reports about fatal corruptions) it means that you need to run `reiserfsck --rebuild-tree`. If `reiserfsck --check` fails in some way you should also run `reiserfsck --rebuild-tree`, but we also encourage you to submit this as a bug report.
5. Before running `reiserfsck --rebuild-tree`, please make a backup of the whole partition before proceeding. Then run `reiserfsck --rebuild-tree --logfile rebuild.log /dev/hda1`.
6. If the `--rebuild-tree` step fails or does not recover what you expected, please submit this as a bug report. Try to provide as much information as possible and we will try to help solve the problem.

EXIT CODES

`reiserfsck` uses the following exit codes:

- 0 - No errors.
- 1 - File system errors corrected.
- 4 - File system fatal errors left uncorrected, `reiserfsck --rebuild-tree` needs to be launched.
- 6 - File system fixable errors left uncorrected, `reiserfsck --fix-fixable` needs to be launched.
- 8 - Operational error.

16 - Usage or syntax error.

AUTHOR

This version of reiserfsck has been written by Vitaly Fertman <vitaly@namesys.com>.

BUGS

There are likely to be some bugs. Please report bugs to the ReiserFS mail-list <reiserfs-list@namesys.com>.

TODO

Faster recovering, signal handling, i/o error handling, etc.

SEE ALSO

mkreiserfs(8), reiserfstune(8) resize_reiserfs(8), debugreiserfs(8),

Reiserfsprogs-3.6.9

April 2003

REISERFSCK(8)



Deutsche Übersetzung der GNU General Public License

Der folgende Text folgt im Wesentlichen der inoffiziellen Übersetzung von Katja Lachmann und der Überarbeitung von Peter Gerwinski (31. Oktober 1996, 4. Juni 2000).

Diese Übersetzung wird mit der Absicht angeboten, das Verständnis der *GNU General Public License* (GNU-GPL) zu erleichtern. Es handelt sich jedoch nicht um eine offizielle oder im rechtlichen Sinne anerkannte Übersetzung.

Die *Free Software Foundation* (FSF) ist nicht der Herausgeber dieser Übersetzung, und sie hat diese Übersetzung auch nicht als rechtskräftigen Ersatz für die Original-GNU-GPL (siehe <http://www.gnu.org/copyleft/gpl.html>) anerkannt. Da die Übersetzung nicht sorgfältig von Anwälten überprüft wurde, können die Übersetzer nicht garantieren, dass die Übersetzung die rechtlichen Aussagen der GNU-GPL exakt wiedergibt. Wenn Sie sichergehen wollen, dass von Ihnen geplante Aktivitäten im Sinne der GNU-GPL gestattet sind, halten Sie sich bitte an die englischsprachige Originalversion.

Die *Free Software Foundation* möchte Sie darum bitten, diese Übersetzung nicht als offizielle Lizenzbedingungen für von Ihnen geschriebene Programme zu verwenden. Bitte benutzen Sie hierfür stattdessen die von der *Free Software Foundation* herausgegebene englischsprachige Originalversion.

This is a translation of the GNU General Public License into German. This translation is distributed in the hope that it will facilitate understanding, but it is not an official or legally approved translation.

The Free Software Foundation is not the publisher of this translation and has not approved it as a legal substitute for the authentic GNU General Public License. The translation has not been reviewed carefully by lawyers, and therefore the translator cannot be sure that it exactly represents the legal meaning of the GNU General Public License. If you wish to be sure whether your planned activities are permitted by the GNU General Public License, please refer to the authentic English version.

The Free Software Foundation strongly urges you not to use this translation as the official distribution terms for your programs; instead, please use the authentic English version published by the Free Software Foundation.

GNU General Public License

Deutsche Übersetzung der Version 2, Juni 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307, USA

Es ist jedermann gestattet, diese Lizenzurkunde zu vervielfältigen und unveränderte Kopien zu verbreiten; Änderungen sind jedoch nicht erlaubt.

Hinweis

Diese Übersetzung ist kein rechtskräftiger Ersatz für die englischsprachige Originalversion!

Hinweis

Vorwort

Die meisten Softwarelizenzen sind daraufhin entworfen worden, Ihnen die Freiheit zu nehmen, die Software weiterzugeben und zu verändern. Im Gegensatz dazu soll Ihnen die *GNU General Public License*, die Allgemeine Öffentliche GNU-Lizenz, ebendiese Freiheit garantieren. Sie soll sicherstellen, dass die Software für alle Benutzer frei ist. Diese Lizenz gilt für den Großteil der von der *Free Software Foundation* herausgegebenen Software und für alle anderen Programme, deren Autoren ihr Datenwerk dieser Lizenz unterstellt haben. Auch Sie können diese Möglichkeit der Lizenzierung für Ihre Programme anwenden. (Ein anderer Teil der Software der *Free Software Foundation* unterliegt stattdessen der *GNU Library General Public License*, der Allgemeinen Öffentlichen GNU-Lizenz für Bibliotheken. – Mittlerweile

wurde die GNU Library Public License von der GNU Lesser Public License abgelöst.)

Die Bezeichnung „freie“ Software bezieht sich auf Freiheit, nicht auf den Preis. Unsere Lizenzen sollen Ihnen die Freiheit garantieren, Kopien freier Software zu verbreiten (und etwas für diesen Service zu berechnen, wenn Sie möchten), die Möglichkeit, die Software im Quelltext zu erhalten oder den Quelltext auf Wunsch zu bekommen. Die Lizenzen sollen garantieren, dass Sie die Software ändern oder Teile davon in neuen freien Programmen verwenden dürfen – und dass Sie wissen, dass Sie dies alles tun dürfen.

Um Ihre Rechte zu schützen, müssen wir Einschränkungen machen, die es jedem verbieten, Ihnen diese Rechte zu verweigern oder Sie aufzufordern, auf diese Rechte zu verzichten. Aus diesen Einschränkungen folgen bestimmte Verantwortlichkeiten für Sie, wenn Sie Kopien der Software verbreiten oder sie verändern.

Beispielsweise müssen Sie den Empfängern alle Rechte gewähren, die Sie selbst haben, wenn Sie – kostenlos oder gegen Bezahlung – Kopien eines solchen Programms verbreiten. Sie müssen sicherstellen, dass auch die Empfänger den Quelltext erhalten bzw. erhalten können. Und Sie müssen ihnen diese Bedingungen zeigen, damit sie ihre Rechte kennen.

Wir schützen Ihre Rechte in zwei Schritten: (1) Wir stellen die Software unter ein Urheberrecht (Copyright), und (2) wir bieten Ihnen diese Lizenz an, die Ihnen das Recht gibt, die Software zu vervielfältigen, zu verbreiten und/oder zu verändern.

Um die Autoren und uns zu schützen, wollen wir darüberhinaus sicherstellen, dass jeder erfährt, dass für diese freie Software keinerlei Garantie besteht. Wenn die Software von jemand anderem modifiziert und weitergegeben wird, möchten wir, dass die Empfänger wissen, dass sie nicht das Original erhalten haben, damit irgendwelche von anderen verursachte Probleme nicht den Ruf des ursprünglichen Autors schädigen.

Schließlich und endlich ist jedes freie Programm permanent durch Software-Patente bedroht. Wir möchten die Gefahr ausschließen, dass Distributoren eines freien Programms individuell Patente lizenzieren -- mit dem Ergebnis, dass das Programm proprietär würde. Um dies zu verhindern, haben wir klargestellt, dass jedes Patent entweder für freie Benutzung durch jedermann lizenziert werden muss oder überhaupt nicht lizenziert werden darf.

Es folgen die genauen Bedingungen für die Vervielfältigung, Verbreitung und Bearbeitung.

Allgemeine Öffentliche GNU-Lizenz

Bedingungen für die Vervielfältigung, Verbreitung und Bearbeitung

0. Diese Lizenz gilt für jedes Programm und jedes andere Datenwerk, in dem ein entsprechender Vermerk des Copyright-Inhabers darauf hinweist, dass das Datenwerk unter den Bestimmungen dieser *General Public License* verbreitet werden darf. Im Folgenden wird jedes derartige Programm oder Datenwerk als „das Programm“ bezeichnet; die Formulierung „auf dem Programm basierendes Datenwerk“ bezeichnet das Programm sowie jegliche Bearbeitung des Programms im urheberrechtlichen Sinne, also ein Datenwerk, welches das Programm, auch auszugsweise, sei es unverändert oder verändert und/oder in eine andere Sprache übersetzt, enthält. (Im Folgenden wird die Übersetzung ohne Einschränkung als „Bearbeitung“ eingestuft.) Jeder Lizenznehmer wird im Folgenden als „Sie“ angesprochen.

Andere Handlungen als Vervielfältigung, Verbreitung und Bearbeitung werden von dieser Lizenz nicht berührt; sie fallen nicht in ihren Anwendungsbereich. Der Vorgang der Ausführung des Programms wird nicht eingeschränkt, und die Ausgaben des Programms unterliegen dieser Lizenz nur, wenn der Inhalt ein auf dem Programm basierendes Datenwerk darstellt (unabhängig davon, dass die Ausgabe durch die Ausführung des Programmes erfolgte). Ob dies zutrifft, hängt von den Funktionen des Programms ab.

1. Sie dürfen auf beliebigen Medien unveränderte Kopien des Quelltextes des Programms, wie sie ihn erhalten haben, anfertigen und verbreiten. Voraussetzung hierfür ist, dass Sie mit jeder Kopie einen entsprechenden Copyright-Vermerk sowie einen Haftungsausschluss veröffentlichen, alle Vermerke, die sich auf diese Lizenz und das Fehlen einer Garantie beziehen, unverändert lassen und des Weiteren allen anderen Empfängern des Programms zusammen mit dem Programm eine Kopie dieser Lizenz zukommen lassen.

Sie dürfen für den eigentlichen Kopiervorgang eine Gebühr verlangen. Wenn Sie es wünschen, dürfen Sie auch gegen Entgelt eine Garantie für das Programm anbieten.

2. Sie dürfen Ihre Kopie(n) des Programms oder einen Teil davon verändern, wodurch ein auf dem Programm basierendes Datenwerk entsteht; Sie dürfen derartige Bearbeitungen unter den Bestimmungen von Paragraph 1

vervielfältigen und verbreiten, vorausgesetzt, dass zusätzlich alle im Folgenden genannten Bedingungen erfüllt werden:

1. Sie müssen die veränderten Dateien mit einem auffälligen Vermerk versehen, der auf die von Ihnen vorgenommene Modifizierung und das Datum jeder Änderung hinweist.
2. Sie müssen dafür sorgen, dass jede von Ihnen verbreitete oder veröffentlichte Arbeit, die ganz oder teilweise von dem Programm oder Teilen davon abgeleitet ist, Dritten gegenüber als Ganzes unter den Bedingungen dieser Lizenz ohne Lizenzgebühren zur Verfügung gestellt wird.
3. Wenn das veränderte Programm normalerweise bei der Ausführung interaktiv Kommandos einliest, müssen Sie dafür sorgen, dass es, wenn es auf dem üblichsten Wege für solche interaktive Nutzung gestartet wird, eine Meldung ausgibt oder ausdruckt, die einen geeigneten Copyright-Vermerk enthält sowie einen Hinweis, dass es keine Gewährleistung gibt (oder anderenfalls, dass Sie Garantie leisten), und dass die Benutzer das Programm unter diesen Bedingungen weiter verbreiten dürfen. Auch muss der Benutzer darauf hingewiesen werden, wie er eine Kopie dieser Lizenz ansehen kann. (Ausnahme: Wenn das Programm selbst interaktiv arbeitet, aber normalerweise keine derartige Meldung ausgibt, muss Ihr auf dem Programm basierendes Datenwerk auch keine solche Meldung ausgeben.)

Diese Anforderungen gelten für das bearbeitete Datenwerk als Ganzes. Wenn identifizierbare Teile des Datenwerkes nicht von dem Programm abgeleitet sind und vernünftigerweise als unabhängige und eigenständige Datenwerke für sich selbst zu betrachten sind, dann gelten diese Lizenz und ihre Bedingungen nicht für die betroffenen Teile, wenn Sie diese als eigenständige Datenwerke weitergeben. Wenn Sie jedoch dieselben Abschnitte als Teil eines Ganzen weitergeben, das ein auf dem Programm basierendes Datenwerk darstellt, dann muss die Weitergabe des Ganzen nach den Bedingungen dieser Lizenz erfolgen, deren Bedingungen für weitere Lizenznehmer somit auf das gesamte Ganze ausgedehnt werden – und somit auf jeden einzelnen Teil, unabhängig vom jeweiligen Autor.

Somit ist es nicht die Absicht dieses Abschnittes, Rechte für Datenwerke in Anspruch zu nehmen oder Ihnen die Rechte für Datenwerke streitig zu machen, die komplett von Ihnen geschrieben wurden; vielmehr ist es die Absicht, die Rechte zur Kontrolle der Verbreitung von Datenwerken, die auf dem Programm basieren oder unter seiner auszugsweisen Verwendung zusammengestellt worden sind, auszuüben.

Ferner bringt auch das einfache Zusammenlegen eines anderen Datenwerkes, das nicht auf dem Programm basiert, mit dem Programm oder einem auf dem Programm basierenden Datenwerk auf ein- und demselben Speicher- oder Vertriebsmedium dieses andere Datenwerk nicht in den Anwendungsbereich dieser Lizenz.

3. Sie dürfen das Programm (oder ein darauf basierendes Datenwerk gemäß Paragraph 2) als Objectcode oder in ausführbarer Form unter den Bedingungen der Paragraphen 1 und 2 kopieren und weitergeben – vorausgesetzt, dass Sie außerdem eine der folgenden Leistungen erbringen:

1. Liefern Sie das Programm zusammen mit dem vollständigen zugehörigen maschinenlesbaren Quelltext auf einem für den Datenaustausch üblichen Medium aus, wobei die Verteilung unter den Bedingungen der Paragraphen 1 und 2 erfolgen muss. Oder:
2. Liefern Sie das Programm zusammen mit einem mindestens drei Jahre lang gültigen schriftlichen Angebot aus, jedem Dritten eine vollständige maschinenlesbare Kopie des Quelltextes zur Verfügung zu stellen – zu nicht höheren Kosten als denen, die durch den physikalischen Kopiervorgang anfallen – wobei der Quelltext unter den Bedingungen der Paragraphen 1 und 2 auf einem für den Datenaustausch üblichen Medium weitergegeben wird. Oder:
3. Liefern Sie das Programm zusammen mit dem schriftlichen Angebot der Zurverfügungstellung des Quelltextes aus, das Sie selbst erhalten haben. (Diese Alternative ist nur für nicht-kommerzielle Verbreitung zulässig und nur, wenn Sie das Programm als Objectcode oder in ausführbarer Form mit einem entsprechenden Angebot gemäß Absatz 2 erhalten haben.)

Unter dem Quelltext eines Datenwerkes wird diejenige Form des Datenwerkes verstanden, die für Bearbeitungen vorzugsweise verwendet wird. Für ein ausführbares Programm bedeutet „der komplette Quelltext“: Der Quelltext aller im Programm enthaltenen Module einschließlich aller zugehörigen Modulschnittstellen-Definitionsdateien sowie der zur Kompilation und Installation verwendeten Skripte. Als besondere Ausnahme jedoch braucht der verteilte Quelltext nichts von dem zu enthalten, was üblicherweise (entweder als Quelltext oder in binärer Form) zusammen mit den Hauptkomponenten des Betriebssystems (Kernel, Compiler usw.) geliefert wird, unter dem das Programm läuft – es sei denn, diese Komponente selbst gehört zum ausführbaren Programm.

Wenn die Verbreitung eines ausführbaren Programms oder von Objectcode dadurch erfolgt, dass der Kopierzugriff auf eine dafür vorgesehene Stelle

gewährt wird, so gilt die Gewährung eines gleichwertigen Zugriffs auf den Quelltext als Verbreitung des Quelltextes, auch wenn Dritte nicht dazu gezwungen sind, den Quelltext zusammen mit dem Objectcode zu kopieren.

4. Sie dürfen das Programm nicht vervielfältigen, verändern, weiter lizenzieren oder verbreiten, sofern es nicht durch diese Lizenz ausdrücklich gestattet ist. Jeder anderweitige Versuch der Vervielfältigung, Modifizierung, Weiterlizenzierung und Verbreitung ist nichtig und beendet automatisch Ihre Rechte unter dieser Lizenz. Jedoch werden die Lizenzen Dritter, die von Ihnen Kopien oder Rechte unter dieser Lizenz erhalten haben, nicht beendet, solange diese die Lizenz voll anerkennen und befolgen.

5. Sie sind nicht verpflichtet, diese Lizenz anzunehmen, da Sie sie nicht unterzeichnet haben. Jedoch gibt Ihnen nichts anderes die Erlaubnis, das Programm oder von ihm abgeleitete Datenwerke zu verändern oder zu verbreiten. Diese Handlungen sind gesetzlich verboten, wenn Sie diese Lizenz nicht anerkennen. Indem Sie das Programm (oder ein darauf basierendes Datenwerk) verändern oder verbreiten, erklären Sie Ihr Einverständnis mit dieser Lizenz und mit allen ihren Bedingungen bezüglich der Vervielfältigung, Verbreitung und Veränderung des Programms oder eines darauf basierenden Datenwerks.

6. Jedes Mal, wenn Sie das Programm (oder ein auf dem Programm basierendes Datenwerk) weitergeben, erhält der Empfänger automatisch vom ursprünglichen Lizenzgeber die Lizenz, das Programm entsprechend den hier festgelegten Bestimmungen zu vervielfältigen, zu verbreiten und zu verändern. Sie dürfen keine weiteren Einschränkungen der Durchsetzung der hierin zugestandenen Rechte des Empfängers vornehmen. Sie sind nicht dafür verantwortlich, die Einhaltung dieser Lizenz durch Dritte durchzusetzen.

7. Sollten Ihnen infolge eines Gerichtsurteils, des Vorwurfs einer Patentverletzung oder aus einem anderen Grunde (nicht auf Patentfragen begrenzt) Bedingungen (durch Gerichtsbeschluss, Vergleich oder anderweitig) auferlegt werden, die den Bedingungen dieser Lizenz widersprechen, so befreien Sie diese Umstände nicht von den Bestimmungen dieser Lizenz. Wenn es Ihnen nicht möglich ist, das Programm unter gleichzeitiger Beachtung der Bedingungen in dieser Lizenz und Ihrer anderweitigen Verpflichtungen zu verbreiten, dann dürfen Sie als Folge das Programm überhaupt nicht verbreiten. Wenn zum Beispiel ein Patent nicht die gebührenfreie Weiterverbreitung des Programms durch diejenigen erlaubt, die das Programm direkt oder indirekt von Ihnen erhalten haben, dann besteht der einzige Weg, sowohl das Patentrecht als auch diese Lizenz zu befolgen, darin, ganz auf die Verbreitung des Programms zu verzichten.

Sollte sich ein Teil dieses Paragraphen als ungültig oder unter bestimmten Umständen nicht durchsetzbar erweisen, so soll dieser Paragraph seinem Sinne nach angewandt werden; im übrigen soll dieser Paragraph als Ganzes gelten.

Zweck dieses Paragraphen ist nicht, Sie dazu zu bringen, irgendwelche Patente oder andere Eigentumsansprüche zu verletzen oder die Gültigkeit solcher Ansprüche zu bestreiten; dieser Paragraph hat einzig den Zweck, die Integrität des Verbreitungssystems der freien Software zu schützen, das durch die Praxis öffentlicher Lizenzen verwirklicht wird. Viele Leute haben großzügige Beiträge zu dem großen Angebot der mit diesem System verbreiteten Software im Vertrauen auf die konsistente Anwendung dieses Systems geleistet; es liegt am Autor/Geber, zu entscheiden, ob er die Software mittels irgendeines anderen Systems verbreiten will; ein Lizenznehmer hat auf diese Entscheidung keinen Einfluss.

Dieser Paragraph ist dazu gedacht, deutlich klarzustellen, was als Konsequenz aus dem Rest dieser Lizenz betrachtet wird.

8. Wenn die Verbreitung und/oder die Benutzung des Programms in bestimmten Staaten entweder durch Patente oder durch urheberrechtlich geschützte Schnittstellen eingeschränkt ist, kann der Urheberrechtsinhaber, der das Programm unter diese Lizenz gestellt hat, eine explizite geographische Begrenzung der Verbreitung angeben, in der diese Staaten ausgeschlossen werden, so dass die Verbreitung nur innerhalb und zwischen den Staaten erlaubt ist, die nicht ausgeschlossen sind. In einem solchen Fall beinhaltet diese Lizenz die Beschränkung, als wäre sie in diesem Text niedergeschrieben.

9. Die *Free Software Foundation* kann von Zeit zu Zeit überarbeitete und/oder neue Versionen der *General Public License* veröffentlichen. Solche neuen Versionen werden vom Grundprinzip her der gegenwärtigen entsprechen, können aber im Detail abweichen, um neuen Problemen und Anforderungen gerecht zu werden.

Jede Version dieser Lizenz hat eine eindeutige Versionsnummer. Wenn in einem Programm angegeben wird, dass es dieser Lizenz in einer bestimmten Versionsnummer oder „jeder späteren Version“ („*any later version*“) unterliegt, so haben Sie die Wahl, entweder den Bestimmungen der genannten Version zu folgen oder denen jeder beliebigen späteren Version, die von der *Free Software Foundation* veröffentlicht wurde. Wenn das Programm keine Versionsnummer angibt, können Sie eine beliebige Version wählen, die je von der *Free Software Foundation* veröffentlicht wurde.

10. Wenn Sie den Wunsch haben, Teile des Programms in anderen freien Programmen zu verwenden, deren Bedingungen für die Verbreitung anders sind, schreiben Sie an den Autor, um ihn um die Erlaubnis zu bitten.

Für Software, die unter dem Copyright der *Free Software Foundation* steht, schreiben Sie an die *Free Software Foundation*; wir machen zu diesem Zweck gelegentlich Ausnahmen. Unsere Entscheidung wird von den beiden Zielen geleitet werden, zum einen den freien Status aller von unserer freien Software abgeleiteten Datenwerke zu erhalten und zum anderen das gemeinschaftliche Nutzen und Wiederverwenden von Software im allgemeinen zu fördern.

Keine Gewährleistung

11. Da das Programm ohne jegliche Kosten lizenziert wird, besteht keinerlei Gewährleistung für das Programm, soweit dies gesetzlich zulässig ist. Sofern nicht anderweitig schriftlich bestätigt, stellen die Copyright-Inhaber und/oder Dritte das Programm so zur Verfügung, „wie es ist“, ohne irgendeine Gewährleistung, weder ausdrücklich noch implizit, einschließlich – aber nicht begrenzt auf – Marktreife oder Verwendbarkeit für einen bestimmten Zweck. Das volle Risiko bezüglich Qualität und Leistungsfähigkeit des Programms liegt bei Ihnen. Sollte sich das Programm als fehlerhaft herausstellen, liegen die Kosten für notwendigen Service, Reparatur oder Korrektur bei Ihnen.

12. In keinem Fall, außer wenn durch geltendes Recht gefordert oder schriftlich zugesichert, ist irgendein Copyright-Inhaber oder irgendein Dritter, der das Programm wie oben erlaubt modifiziert oder verbreitet hat, Ihnen gegenüber für irgendwelche Schäden haftbar, einschließlich jeglicher allgemeiner oder spezieller Schäden, Schäden durch Seiteneffekte (Nebenwirkungen) oder Folgeschäden, die aus der Benutzung des Programms oder der Unbenutzbarkeit des Programms folgen (einschließlich – aber nicht beschränkt auf – Datenverluste, fehlerhafte Verarbeitung von Daten, Verluste, die von Ihnen oder anderen getragen werden müssen, oder dem Unvermögen des Programms, mit irgendeinem anderen Programm zusammenzuarbeiten), selbst wenn ein Copyright-Inhaber oder Dritter über die Möglichkeit solcher Schäden unterrichtet worden war.

Ende der Bedingungen

Anhang: Wie Sie diese Bedingungen auf Ihre eigenen, neuen Programme anwenden können

Wenn Sie ein neues Programm entwickeln und wollen, dass es vom größtmöglichen Nutzen für die Allgemeinheit ist, dann erreichen Sie das am bes-

ten, indem Sie es zu freier Software machen, die jeder unter diesen Bestimmungen weiterverbreiten und verändern kann.

Um dies zu erreichen, fügen Sie die folgenden Vermerke zu Ihrem Programm hinzu. Am sichersten ist es, sie an den Anfang einer jeden Quelldatei zu stellen, um den Gewährleistungsausschluss möglichst deutlich darzustellen; zumindest aber sollte jede Datei eine Copyright-Zeile besitzen sowie einen kurzen Hinweis darauf, wo die vollständigen Vermerke zu finden sind.

<Program name and short description>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Auf Deutsch:

<Programmnamen und einer kurzen Beschreibung>

Copyright (C) <Jahr> <Name des Autors>

Dieses Programm ist freie Software. Sie können es unter den Bedingungen der GNU General Public License, wie von der Free Software Foundation veröffentlicht, weitergeben und/oder modifizieren, entweder gemäß Version 2 der Lizenz oder (nach Ihrer Option) jeder späteren Version.

Die Veröffentlichung dieses Programms erfolgt in der Hoffnung, dass es Ihnen von Nutzen sein wird, aber OHNE IRGENDNEINE GARANTIE, sogar ohne die implizite Garantie der MARKTREIFE oder der VERWENDBARKEIT FÜR EINEN BESTIMMTEN ZWECK. Details finden

Sie in der GNU General Public License.

Sie sollten eine Kopie der GNU General Public License zusammen mit diesem Programm erhalten haben. Falls nicht, schreiben Sie an die Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307, USA.

Fügen Sie auch einen kurzen Hinweis hinzu, wie Sie elektronisch und per Brief erreichbar sind.

Wenn Ihr Programm interaktiv ist, sorgen Sie dafür, dass es nach dem Start einen kurzen Vermerk ausgibt:

```
Gnomovision version 69, Copyright (C) <year> <name of author>
```

```
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
'show w'. This is free software, and you are welcome to
redistribute it under certain conditions; type 'show c' for
details.
```

Auf Deutsch:

```
Gnomovision Version 69, Copyright (C) <Jahr> <Name des Autors>
```

```
Für Gnomovision besteht KEINERLEI GARANTIE; geben Sie
'show w' für Details ein. Gnomovision ist freie Software, die
Sie unter bestimmten Bedingungen weitergeben
dürfen; geben Sie 'show c' für Details ein.
```

Die hypothetischen Kommandos `show w` und `show c` sollten die entsprechenden Teile der GNU-GPL anzeigen. Natürlich können die von Ihnen verwendeten Kommandos anders heißen als `show w` und `show c`; es könnten auch Mausklicks oder Menüpunkte sein – was immer am besten in Ihr Programm passt.

Soweit vorhanden, sollten Sie auch Ihren Arbeitgeber (wenn Sie als Programmierer arbeiten) oder Ihre Schule einen Copyright-Verzicht für das Programm unterschreiben lassen. Hier ein Beispiel. Die Namen müssen Sie natürlich ändern.

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program 'Gnomovision' (which makes passes at compilers) written
by James Hacker.
```

```
signature of Ty Coon, 1st April 1989 Ty Coon, President of Vice
```

Auf Deutsch:

Die Yoyodyne GmbH erhebt keinen urheberrechtlichen Anspruch auf das von James Hacker geschriebene Programm 'Gnomovision' (einem Schrittmacher für Compiler).

Unterschrift von Ty Coon1. April 1989 Ty Coon, Vizepräsident

Diese *General Public License* gestattet nicht die Einbindung des Programms in proprietäre Programme. Ist Ihr Programm eine Funktionsbibliothek, so kann es sinnvoller sein, das Binden proprietärer Programme mit dieser Bibliothek zu gestatten. Wenn Sie dies tun wollen, sollten Sie die GNU Library General Public License anstelle dieser Lizenz verwenden.

Literaturverzeichnis

- [1] *SuSE Linux (User Guide)*. SuSE, 2. Auflage ©2003 .
- [2] EDWARD C. BAILEY. *Maximum RPM*. ©1997 . ISBN 1-888172-78-9.
- [3] BRYAN COSTALES, ERIC ALLMAN, NEIL RICKERT. *sendmail*. ©1993 . ISBN 1-56592-056-2.
- [4] WERNER ALMESBERGER. *LILO User's guide*.
`file:///usr/share/doc/lilo/user.dvi.`
- [5] OLAF KIRCH. *LINUX Network Administrator's Guide*. ©1995 . ISBN 1-56592-087-2.
- [6] SEBASTIAN HETZE, DIRK HOHNDEL, MARTIN MÜLLER, OLAF KIRCH. *Linux Anwenderhandbuch*. 6. Auflage ©1996 . ISBN 3-929764-05-9.
- [7] SIMON GARFINKEL, GENE SPAFFORD. *Practical UNIX Security*. ©1993 . ISBN 0-937175-72-2.
- [8] CRAIG HUNT. *TCP/IP Network Administration*. ©1995 . ISBN 3-930673-02-9.
- [9] TIM O'REILLY, GRACE TODINO. *Managing UUCP and Usenet*. ©1992 . ISBN 0-937175-93-5.
- [10] MATT WELSH. *Linux Installation and Getting Started*. 2. Auflage ©1994 . ISBN 3-930419-03-3.
- [11] LINDA LAMB. *Learning the vi Editor*. ©1990 . ISBN 0-937175-67-6.
- [12] MATT WELSH, LARS KAUFMAN. *Running Linux*. ©1995 O'Reilly. ISBN 1-56592-100-3.

- [13] JÜRGEN SCHNEIDERER. *Sicherheit Kostenlos – Firewall mit Linux*. ©1998 iX.
- [14] MICHAEL KIENLE. *TIS: Toolkit für anwendungsorientierte Firewall-Systeme*. ©1995 iX.
- [15] ULRICH KUNITZ. *Sicherheit fast kostenlos: Einrichtung eines kostenlosen Firewall-Systems*. ©1995 iX.
- [16] WILLIAM R. CHESWICK, STEVEN M. BELLOVIN. *Firewalls und Sicherheit im Internet*. ©1996 Addison Wesley. ISBN 3-89319-875-x.
- [17] BRENT CHAPMAN, ELISABETH D. ZWICKY. *Einrichten von Internet Firewalls (Sicherheit im Internet gewährleisten)*. ©1996 O'Reilly. ISBN 3-930673312.
- [18] CLIFFORD STOLL. *Kuckucksei. Die Jagd auf die deutschen hacker, die das Pentagon knackten*. ©1998 Fischer-TB. Verlag. ISBN 3-596139848.
- [19] BRIAN TUNG. *Kerberos: A Network Authentication System*. ©1999 Fischer-TB. Verlag. ISBN 0-201-37924-4.
- [20] CHIN FANG, BOB CROSSON, ERIC S. RAYMOND. *The Hitchhiker's Guide to X386/XFree86 Video Timing (or, Tweaking your Monitor for Fun and Profit)*. ©1993.
- [21] SEBASTIAN HETZE, DIRK HOHNDEL, MARTIN MÜLLER, OLAF KIRCH. *Linux Anwenderhandbuch*. 6. Auflage ©1996 LunetIX Softfair. ISBN 3-929764-05-9.

Index

A

ACPI 255
Adressen
- IP 348
- MAC 348
Apache 52, 423–450
- apxs 430
- Beispielumgebung 299
- CGI 438
- Content Negotiation 427
- DocumentRoot 432
- Fehlerbehandlung 427
- Flags 431
- Installation 429–430
- Konfiguration 431–436
- logging 435, 436
- Module 426
 · aktivieren 431
 · laden 432
 · mod_perl 440
 · mod_php4 442
 · mod_python 443
 · mod_ruby 443
- permissions 433
- Sicherheit 447–448
- Squid 515
- SSI 438
- SSI (Server Side Includes) 435
- Standardseite 425
- starten 429
- Threads 428
- Troubleshooting 448
- Virtual Hosts 426, 443–447
- Zugriffsrechte 447
APM 255

- Kernelparameter 50
Apple
- Netatalk 481
Arbeitsspeicher 303
ASCII
- Kodierung 199
ATA-RAID-Controller *siehe* Hardware,
 Promise-Controller
autofs 51

B

bash
- /etc/profile 299
Befehl
- chown 56
- head 56
- nice 56
- sort 56
- tail 56
Benutzer
- Namensänderung 51
- Probleme beim Anlegen 369
Bildschirm
- Auflösung 86
- SuSE-Bildschirm deaktivieren 16
BIND *siehe* DNS
BIOS
- Virus Protection 15
Bluetooth 247
- hciconfig 250
- hcitool 249
- opd 251
- pand 250
- sdptool 250
Bootdiskette 22, 50, 205
- Erzeugen mit dd 21

- Erzeugen mit rawrite	20
Booten	325, 603, 609
- Ablauf	204
- Bootmanager	205
- GRUB	207–218
- initial ramdisk	305–310
- Konzept	205, 325
- LILO	217
- Methoden	15
- Rechner bleibt hängen ..	<i>siehe</i> BIOS, Virus Protection
- von CD2	23
- von Disketten	19
Bootloader	
- GRUB	203, 207
- LILO	217
Bootmanager	203
- GRUB	205
- Windows NT	205
Bootsektor	204

C

CD-ROM-Laufwerk	
- Unterstützung durch Linux	23
CD-ROM-Laufwerke	
- ATAPI	23
chown	56
CID-keyed Fonts	94
CJK	321
Compose <i>siehe</i> Tastaturbelegung, Compose	
Concurrent Version System	<i>siehe</i> CVS
Core-Dateien	302
cpuspeed	268
Crash	603, 609
Cron	
- regelmäßige Wartungsdienste ...	54
cron	300
CVS	453, 463

D

Daemonen	
- lpd	173
Dateien	
- Drucken	140, 142, 175, 178
- finden	51
- synchronisieren	451–470
· CVS	453
· InterMezzo	452
· mailsync	454
· unison	453
Dateisystem	577–587
- Access Controll Lists	589–602
- Beschränkungen	585

- e2fsck	603
- Ext2	578–579
- Ext3	579–581
- FHS	298
- JFS	582–583
- LFS	585–586
- Rechte	301
- ReiserFS	581–582
- reiserfsck	609
- Termini	577
- TeX	298
- XFS	583–584
Deinstallation	
- GRUB	218
- LILO	218
- Linux	218
- Squid	505
DENIC	376
depmod	292
DHCP	
- Server-Konfiguration	414
- statische Adressvergabe	417
Diskette	
- Booten von	205
- formatieren	21
DNS	351, 375
- Forwarding	376
- Logging	380
- Mail Exchanger	352
- NIC	352
- Optionen	378
- Problemanalyse	376
- Squid und	506
- Starten	376
- top level domain	352
- umgekehrte Adress-Auflösung ..	384
- Zonen	380
- Zonendateien	381
Domain	364
Domain Name System	<i>siehe</i> DNS
Druck-System	<i>siehe</i> Spool-System
Drucken	99, 167
- a2ps	194
- Ablauf	100–102
- Aufträge	
· Bearbeitung	119
- aus Anwendungsprogrammen ..	115,
139	
- Bannerseiten	114
- Bearbeitung	119
- CUPS	110, 116–123
- Fehlersuche	122
· OpenOffice.org	121

- cups-lpd	145
- CUPS-Netzwerk-Server	145
- CUPS-Server	145
- Dateien	140, 142, 175, 178
- Druckaufträge	
· Löschen	143
· löschen	141, 176, 179
· Status	140, 176, 178
- Druckerfilter	
· anpassen	182–183
· Beispiel	183
· Fehlersuche	189–190
· konfigurieren	182
· lpdfilter	180–190
- Druckersprache	100
· ASCII	100
· ESC	100
· PCL	100
· PostScript	100
- Druckerwarteschlange	100
- duplex	184
- Fehlersuche	
· CUPS	122
· Netzwerk	159
- foomatic-filters	54
- GDI-Drucker	107–109
· Konfiguration	188
· unterstützt	108
- Ghostscript	190
· Treiber	106–107
- Ghostscript-Treiber	105
- Grundlagen	100–104
- IPP	116
- IPP-Server	145
- Kommandozeile	175
- Kommandozeile, von der	139
- Konfiguration	109
· CUPS	117–118
· Lprng und lpdfilter	173
· Schnittstellen	168–173
· YaST	110
- lpc	176–177
- LPD-Server	145
- lpq	178
- lpr	175, 178
- LPRng	54, 111
· Befehle	175
- lprsetup	173
- Netzwerk	144
· Fehlersuche	159
- Netzwerkdrucker	118
- PPD	118
- Print-Server	144

- Printserver-Box	144
- Protokolle	147
- Spooler	
· lpd	173–174
- Störungsbehebung	179
- Treiber	106–109
- unterstützte Drucker	105
- Voraussetzungen	105
- Warteschlange	109, 113
· kontrollieren	176–179
· Optionen	141
· Tools	175–179
- Warteschlangen	
· color	109
· Druckaufträge löschen ..	176, 179
· entfernt	178–179
· im Netz	142–143
· raw	122, 180
· Status	140, 142, 176, 178
· verwalten	140–144

E

e2fsck	603
Eingabemethode	
- CJK	321
Einwahl	
- smpppd	496
Emacs	304
Erstinstallation	
- Bootdiskette erstellen	
· DOS	19
· Linux, UNIX	21
- Booten von CD2	23
- Booten von Diskette	22
- künftige Boot-Methode	15
- linuxrc	10
- Startbildschirm	8

F

fdisk	218
FHS (File System Hierarchy Standard) ..	298
Firewall	522
- Squid	513
- SuSEfirewall2	522
Font-Systeme	89
- CID-keyed Fonts	94
- X11 Core-Fonts	93
- Xft	89
Fonts	89
- CID-keyed	94
- X11 Core	93
- Xft	89
free	303

FTP-Server	52, 298
- Beispielumgebung	298
Funkverbindung	
- Bluetooth	247

G

GDT RAID5-Controller	<i>siehe</i> ICP Vortex
Ghostscript	190–194
- Treiber	105
GNU Emacs	<i>siehe</i> Emacs
GPL	615
Grafik	
- 3D	94–97
· Diagnose	96
· Installationssupport	97
· SaX2	95
· Support	94
· Testen	96
· Treiber	94
· Troubleshooting	96
- Device-Identifier	86
- Farbtiefe	86
- id	95
Grafischer Hintergrund	<i>siehe</i> SUSE-Bildschirm, deaktivieren
GRUB	203, 207
- /etc/grub.conf	215
- Bootmenü	208
- Bootpasswort	216
- deinstallieren	218
- Gerätenamen	210
- GRUB-Shell	215
- Partitionsnamen	210
- Troubleshooting	217
Gruppen	
- Namensänderung	51
gs	<i>siehe</i> Ghostscript

H

harden_suse	52
Hardware	
- CD-ROM-Laufwerke	
· ATAPI	23
- Laptop	223
- Notebook	223
- Promise-Controller	45
- SCSI-Geräte	
· Konfiguration ändern	25
hciconfig	250
hcitool	249
head	56
Hilfe	
- Info	302

- Manual-Pages	302
- Texinfo	302
- Tkinfo	302
- XInfo	302

Hintergrund

- grafischer ...	<i>siehe</i> SUSE-Bildschirm, deaktivieren
------------------	---

Hotplug	373
HTTP-Server	<i>siehe</i> Apache

I

I18N	322
ICP Vortex-Controller	
- Installation schlägt fehl	15
IDE-Festplatte	
- ATA-RAID-Controller	<i>siehe</i> Hardware, Promise-Controller
inetd	53
init	326
- Skripte	330
- Skripte hinzufügen	332
initial ramdisk (initrd)	305
insmod	292
Installation	
- GRUB	207
- Kernel	295
- Pakete	59
- textbasiert, mit YaST	8
- via FTP	17
- via Netzwerk	17
- via NFS	17
- via PCMCIA	233
Installationssupport	
- 3D-Grafikkarten	97
InterMezzo	452, 458
Internet	
- Proxy	<i>siehe</i> Squid
- smpd	496
- Webserver	<i>siehe</i> Apache

IP-Adressen	348
- IPv6	353, 373
- Namensauflösung	351, 375
- Netzmasken	349
- Netzwerkklassen	349
- privater Adressbereich	351
IrDA	244

J

jade	<i>siehe</i> SGML, openjade
jade_dsl	54

K

Kerberos	535
----------------	-----

- Authenticator	537	- LDAP	393
- Clientkonfiguration	549–552	- LVM	30
- Credential	536	- MARSNWE	489–493
- Host-Principals	554	- Netatalk	481–487
- Installation und Administration	543–560	- Netzwerk	370–375
- KDC	546–549	- NFS	409–413
- Konfiguration von SSH	556	- NIS	404–408
- LDAP und Kerberos	557–560	- Routing	374
- Master Key	547	- Runlevel	327
- Mutual Authentication	537	- Samba	474–481
- PAM-Unterstützung	556	- Soft-RAID	38
- Principal	537, 548	- Squid	506
- Protokollfunktion	546	- SSH	529
- Realm	543, 547	- SuSEfirewall2	525–529
- Replay	537	- Systemeinstellungen	335
- Session Key	537	Konfigurationsdateien	363
- Ticket	536	- /lptions	122, 142
- Zeitsynchronisation	545	- /boot/grub/menu.lst	208
Kernel	287	- /etc/HOSTNAME	369
- Daemon	293	- /etc/conf.modules	<i>siehe</i>
- installieren	295	- /etc/modprobe.conf	
- Kompilierung	287	- /etc/dhcpd.conf	414
- Konfiguration	289	- /etc/exports	411, 413
- Module	291	- /etc/foomatic/filter.conf	54
· übersetzen	294	- /etc/grub.conf	215
· depmod	292	- /etc/gshadow	57
· insmod	292	- /etc/host.conf	366, 367
· modinfo	293	- /etc/hosts	365
· modprobe	292, 293	- /etc/init.d/boot	50
· modprobe.conf	55	- /etc/inittab	326
· Netzwerkkarten	371	- /etc/logfiles	51
· parport	168	- /etc/modprobe.conf	293
· rmmmod	292	- /etc/modules.conf	<i>siehe</i>
- Module Loader	293	- /etc/modprobe.conf	
- Neuheiten der Version 2.6	55	- /etc/named.conf	377
Kernel too big	294	- /etc/networks	366
Kernelparameter		- /etc/nscd.conf	369
- APM	50	- /etc/nsswitch.conf	367
Kmod	<i>siehe</i> Kernel Module Loader	- /etc/nwsvr.conf	489
Kodierung		- /etc/openldap/slapd.conf	393
- UTF-8	56	- /etc/profile	299
Konfiguration		- /etc/resolv.conf	304, 364
- Apache	431–436	- /etc/squid/squid.conf	506, 512, 516
- Bootloader		- /etc/squidguard.conf	518
· GRUB	207	- /etc/sysconfig/network/ifroute-* ..	
- DHCP	414–419	374	
- DNS	375	- /etc/sysconfig/network/routes	374
- Drucken	109–115	- /etc/xinetd.d/cups-lpd	164
- IPv6	373	- /etc/xml/catalog	55
- Kerberos	543–560	- /etc/xml/suse-catalog.xml	55
- Kernel	287–296	- apache2	431
- Laptops	223, 226–235	- cups	
		· lptions	142

- cupsd.conf 118
- httpd.conf 431, 432
- lpd.conf 174
- lpd.perms 174
- lpdfilter 180, 182
- mime.convs 119
- modprobe.conf 55
- modules.conf 168
- printcap 146, 174, 180
- stcany.upp 193
- Konsole
 - virtuell 321
- L**
- L10N 322
- LAN 370
- Laptop 223
- LDAP 387–404
 - Access Control Information 397
 - Daten ändern 401
 - Daten durchsuchen 402
 - Daten hinzufügen 399
 - Daten löschen 403
 - Kerberos und LDAP 557–560
 - ldapadd 399
 - ldapdelete 403
 - ldapmodify 401
 - ldapsearch 402
 - Serverkonfiguration 393
 - Verzeichnisbaum 390
- LDAP (Lightweight Directory Access Protocol) 387
- LFS (Large File Support) 585
- LILO 217
 - deinstallieren 218
- Linux
 - deinstallieren 218
 - Update 43
- linuxrc 310
- linuxthreads 55
- Lizenz *siehe* GPL
- Local Area Network *siehe* LAN
- Locale
 - UTF-8 56
- locate 51
- Logdateien
 - apache2 436, 448
 - httpd 434, 436, 448
- Logfiles *siehe* Protokoll-Dateien
- lprsetup 173
- LSB (Linux Standard Base) 298
 - Pakete installieren 58
- lsmod 293

LVM *siehe* YaST, LVM

M

Mac OS 481

mailsync 454, 467

Manpages *siehe* Hilfe, Manual-Pages

Masquerading 522

Master Boot Record *siehe* MBR

Maus

- pine 51

MBR 204

mkinitrd 309

Modeline 88

modinfo 293

modprobe 292

Modul

- hwinfo 291
- Laden 312
- Parameter 312
- Umgang 292

Multi_key *siehe* Tastaturbelegung, Compose

N

Name Service Cache Daemon 369

Nameserver 364, 375

- BIND 375

Netatalk 481

NetBIOS 473

- Namensdienst 473

Network File System *siehe* NFS

Network Information Service *siehe* NIS

Netzwerk

- Broadcastadresse 351
- DNS 351
- Drucken 118
- Drucken im 144
- IP-Adressen 348
- Kerberos 535
- Konfiguration
 - IPv6 373
- Konfigurationsdateien 363
- Localhost 351
- manuelle Konfiguration 362
- Netzmasken 349
- Netzwerkbasisadresse 350
- Routing 348, 349, 374
- Test 371

NFS 408

- Client 408
- exportieren 410, 411
- importieren 409
- mount 409

- mountd 411
- Server 408
- nfsd 411
- NGPT 55
- nice 56
- NIS 404–408
 - autofs 51
 - Client 407
 - Master 404–407
 - Slave 404–407
- Notebook 223
 - ACPI 255
 - APM 255
 - IrDA 244
 - PCMCIA 373
 - Powermanagement 255
 - SCPM 235
- Notfallsystem 316
- NPTL 55, 56
- NSS (Name Service Switch) 367
- nVidia 53

O

- opd 251
- OpenGL 94–97
 - Testen 96
 - Treiber 94
- OpenLDAP *siehe* LDAP
- OpenOffice.org
 - Drucken
 - Cups 121
- OpenSSH *siehe* SSH

P

- Pakete
 - bauen 54
 - build 68
 - deinstallieren 59
 - installieren 59
 - kompilieren 59, 66
 - LSB 58
 - Paket-Manager 58
 - Paketformat 58
- Paketfilter *siehe* SuSEfirewall2
- pand 250
- Partitionieren
 - Experte 25
 - fdisk 218
 - Partitionstabelle 204
 - Swap 26
- Partitionierer *siehe* YaST, Partitionierer
- PCMCIA 224, 373
 - Cardmanager 225

- Fehlerbehebung 228
- Hilfsprogramme 234
- Installation via 233
- IrDA 244
- ISDN 227
- Konfiguration 226
- Modem 227
- Netzwerkkarten 227
- SCSI 228
- PGP 59
- pine 51
- portmap 411
- Portscan 515
- PostgreSQL
 - Update 45
- PostScript
 - Umformatierung 195–199
- Powermanagement 255, 268–275
 - ACPI 270
 - APM 270
 - cpufrequency 268
 - cpuspeed 268
 - Ladezustand 271
 - Powersave 268
 - YaST 275
- Powersave 268
 - Konfiguration 269
- Programme
 - kompilieren 66
- Programmieren
 - Core-Dateien 302
- Promise-Controller *siehe* Hardware, Promise-Controller
- Protokoll-Dateien 300
- Protokolle
 - ICMP 345
 - IGMP 345
 - IPP 116
 - TCP/IP 344
 - UDP 345
- Proxy *siehe* Squid
- Prozessoren
 - AMD64 283

Q

- Quellen
 - kompilieren 66

R

- RAID-Controller
 - ATA *siehe* Hardware, Promise-Controller

Rechner bleibt hängen *siehe* BIOS, Virus Protection
 Rechte *siehe* Dateisystem, Rechte
 reiserfsck 609
 Remote Login 51
 Rettungssystem 316
 - benutzen 318
 - Rettungsdiskette 316
 - starten 317
 Reverse lookup *siehe* DNS
 rmmod 292
 Routing 348, 374
 - Netzmasken 349
 - routes 374
 - statisch 374
 RPC-Mount-Daemon 411
 RPC-NFS-Daemon 411
 RPC-Portmapper 409, 411
 RPM 58
 - Patches 62
 - rpmnew 59
 - rpmmorig 59
 - rpmsave 59
 - Version 4 54
 rpmbuild 54, 58
 Runlevel 327
 - Runlevel-Editor 334
 - wechseln 328

S

Samba 472–481
 - Freigaben Shares 475
 - Security Level 477
 - Serverkonfiguration 474
 Schnittstelle
 - IrDA 172
 - parallel 168–170
 - seriell 173
 - USB 170–172
 SCPM 235
 - Einrichten 237
 - Profile verwalten 238
 SCSI-Geräte
 - Konfiguration ändern 25
 SCSI-Gerätedateien
 - Namen zuweisen 25
 sdptool 250
 SGML
 - Dateisystem nach FHS 58
 - openjade 54
 Sicherheit 561
 - Firewall 522
 - Squid 501

 - SSH 529–535
 Skript
 - init.d
 · network 370
 · nfsserver 370
 · portmap 370
 · postfix 370
 · squid 504
 · xinetd 370
 · ypbind 370
 · ypserv 370
 - lpdfilter
 · guess 180
 - modify_resolvconf 365
 SMB *siehe* Samba
 smpppd 496
 Soft-RAID *siehe* YaST, Soft-RAID
 sort 56
 Speicher 303
 Spool-System 99
 Squid 500
 - Access Controls 516
 - Apache 515
 - Cache-Größe 503
 - cachemgr.cgi 515
 - Caches 501
 - Calamaris 518
 - CPU 504
 - DNS 506
 - Eigenschaften 500
 - Festplatte 503
 - Firewall 513
 - Konfiguration 506
 - Logdatei 505
 - Objekte speichern 502
 - Proxy-Cache 500
 - RAM 504
 - Rechte 509
 - SARG 519
 - Sicherheit 501
 - squidGuard 517
 - Starten 504
 - Statistik 515
 - transparenter Proxy 512
 - Verzeichnisse 504
 - Zugriffskontrolle 509
 SSH 529–535
 - Authentifizierung 533
 - scp 530
 - sftp 531
 - ssh-agent 533
 - sshd 531
 Startup-Skripten *siehe* Skript, init.d

SuSE Linux	297
- Besonderheiten	297
- Installation	310
- Tastaturbelegung	321
SuSEconfig	335
SuSEfirewall2	522
Swap-Partition	26
sx	54
sysconfig	50, 335
System is too big	294
System-Update	43
Systemdienste konfigurieren <i>siehe</i> sysconfig	
Systeminformationen	311

T

tail	56
Tastaturbelegung	321
- Compose	321
TCP/IP	344
- Dienste	344
- ICMP	345
- IGMP	345
- Pakete	345, 347
- Schichtenmodell	345
- TCP	344
- UDP	345
Thread-Paket	
- NPTL	56
TrueType	<i>siehe</i> X11, TrueType-Font

U

UDP	<i>siehe</i> TCP
ugidd	411
ulimit	302
Umgebungsvariable	
- CUPS_SERVER	116
umgekehrte Adress-Auflösung	
- reverse lookup	384
unison	453, 461
Update	43
- /etc/skel	49
- profile	49
USB-Stick	
- Booten von	205
UTF-8	
- Kodierung	56

V

Vernetzung	343
Virtuelle Konsolen	321
virtueller Bildschirm	86
Virus Protection	<i>siehe</i> BIOS, Virus Protection

Virus-Warnung	15
---------------------	----

W

Webserver	<i>siehe</i> Apache
Weiterführende Hinweise	167
whois	353
Windows	472
- NT Bootmanager	205
- SMB	472

X

X	<i>siehe</i> X11
X Window System	<i>siehe</i> X11
X11	81
- CID-keyed Fonts	94
- Font	88
- Font-Systeme	89
- Optimierung	82
- Treiber	87
- TrueType-Font	88
- X11 Core-Fonts	93
- Xft	89
- xft	88
- Zeichensatz	88
X11 Core-Fonts	93
XF86Config	
- Clocks	86
- Depth	85
- Device	84-86
- Files	83
- InputDevice	83
- Modeline	83
- modeline	86
- Modes	84, 86, 87
- Monitor	83, 85, 87
- Screen	84
- ServerFlags	83
- ServerLayout	84
- Subsection	
· Display	85
- Virtual	86
XFree86	82
Xft	89
xinetd	53
XML	
- Dateisystem nach FHS	58
- Katalog	55
- openjade	54

Y

YaST	50
- 3D	95
- Drucken	110

- LVM (Logical Volume Manager) .	31	- Partitionierer	30
- ncurses	73	- Powermanagement	275
- Netzwerkkonfiguration	371	- Runlevel-Editor	334
- NFS-Client	409	- Soft-RAID	38
- NFS-Server	410	- Sysconfig-Editor	337
- NIS-Client	407	- Tastaturbelegung	73
- NIS-Server	404	- Textmodus	73–79
- Online-Update über die Konsole .	77	YP	<i>siehe</i> NIS