



debian

Debian-Referenz

Osamu Aoki

Copyright © 2013-2018 Osamu Aoki

Diese Debian-Referenz (version 2.77) (2021-01-10 06:32:51 UTC) wurde geschrieben, um für die Zeit nach der Installation einen groben Überblick über das Debian-System in Form eines Benutzerhandbuchs zu bieten. Sie behandelt viele Aspekte der Systemadministration mittels Shell-Befehlsbeispielen für einfache Benutzer.

MITWIRKENDE

	<i>TITEL :</i> Debian-Referenz		
<i>AKTION</i>	<i>NAME</i>	<i>DATUM</i>	<i>UNTERSCHRIFT</i>
VERFASST DURCH	Osamu Aoki	10. Januar 2021	

VERSIONSGESCHICHTE

NUMMER	DATUM	BESCHREIBUNG	NAME

Inhaltsverzeichnis

1	GNU/Linux-Lehrstunde	1
1.1	Grundlagen für die Konsole	1
1.1.1	Die Shell-Eingabeaufforderung (Shell-Prompt)	1
1.1.2	Die Shell-Eingabeaufforderung unter X	2
1.1.3	Das root-Benutzerkonto	2
1.1.4	Die root-shell-Eingabeaufforderung	3
1.1.5	Systemadministrations-Werkzeuge mit grafischer Oberfläche (GUI)	3
1.1.6	Virtuelle Konsolen	3
1.1.7	Wie Sie die Eingabeaufforderung wieder verlassen	4
1.1.8	Wie Sie das System herunterfahren	4
1.1.9	Eine Konsole wiederherstellen	4
1.1.10	Zusätzliche Paketempfehlungen für Neulinge	4
1.1.11	Ein zusätzliches Benutzerkonto	5
1.1.12	sudo-Konfiguration	5
1.1.13	Zeit zum Spielen	6
1.2	Unix-ähnliches Dateisystem	7
1.2.1	Unix-Dateigrundlagen	7
1.2.2	Dateisystem-Internas	8
1.2.3	Dateisystem-Berechtigungen	9
1.2.4	Steuerung der Berechtigungen für neu erzeugte Dateien: umask	11
1.2.5	Berechtigungen für Gruppen von Benutzern (group)	12
1.2.6	Zeitstempel	13
1.2.7	Links	14
1.2.8	Benannte Pipes (FIFOs)	15
1.2.9	Sockets	15
1.2.10	Geräte-dateien	15
1.2.11	Spezielle Geräte-dateien	16
1.2.12	procfs und sysfs	16
1.2.13	tmpfs	17
1.3	Midnight Commander (MC)	17

1.3.1	Anpassen des MC	18
1.3.2	Starten von MC	18
1.3.3	Dateimanager in MC	18
1.3.4	Befehlszeilentricks in MC	18
1.3.5	Der interne Texteditor in MC	19
1.3.6	Der interne Dateibetrachter in MC	19
1.3.7	Autostart-Funktionalitäten von MC	20
1.3.8	Virtuelles FTP-Dateisystem von MC	20
1.4	Die grundlegende Unix-ähnliche Arbeitsumgebung	20
1.4.1	Die Login-Shell	20
1.4.2	Anpassen der bash	21
1.4.3	Spezielle Tastendrucke	21
1.4.4	Mausoperationen im Unix-Stil	22
1.4.5	Der Pager	22
1.4.6	Der Texteditor	22
1.4.7	Einen Standard-Texteditor einstellen	23
1.4.8	Anpassen von vim	23
1.4.9	Aufzeichnen der Shell-Aktivitäten	24
1.4.10	Grundlegende Unix-Befehle	24
1.5	Der einfache Shell-Befehl	26
1.5.1	Befehlsausführung und Umgebungsvariablen	26
1.5.2	Die "\$LANG"-Variable	26
1.5.3	Die "\$PATH"-Variable	28
1.5.4	Die "\$HOME"-Variable	28
1.5.5	Befehlszeilen-Optionen	29
1.5.6	Shell-Glob	29
1.5.7	Rückgabewert eines Befehls	30
1.5.8	Typische Befehlssequenzen und Shell-Weiterleitungen	30
1.5.9	Befehls-Alias	32
1.6	Unix-ähnliche Textverarbeitung	32
1.6.1	Unix-Textverarbeitungswerkzeuge	32
1.6.2	Reguläre Ausdrücke	33
1.6.3	Ersetzungsausdrücke	35
1.6.4	Globale Ersetzungen mit regulären Ausdrücken	35
1.6.5	Extrahieren von Daten aus einer Textdatei-Tabelle	36
1.6.6	Skript-Schnipsel für die Befehlsweiterleitung	38

2	Debian-Paketmanagement	39
2.1	Grundvoraussetzungen für das Debian-Paketmanagement	39
2.1.1	Paketkonfiguration	39
2.1.2	Grundsätzliche Vorsichtsmaßnahmen	40
2.1.3	Leben mit den ewigen Aktualisierungen	41
2.1.4	Grundlagen über das Debian-Archiv	42
2.1.5	Debian ist zu 100% freie Software	45
2.1.6	Paketabhängigkeiten	46
2.1.7	Die Ereignisabfolge für das Paketmanagement	48
2.1.8	Erste Hilfe bei Paketmanagement-Problemen	49
2.2	Grundlegende Paketmanagement-Operationen	49
2.2.1	apt contra apt-get / apt-cache contra aptitude	49
2.2.2	Grundlegende Paketmanagement-Operationen auf der Befehlszeile	50
2.2.3	Interaktive Nutzung von aptitude	52
2.2.4	Tastaturkürzel von aptitude	53
2.2.5	Paketansichten in aptitude	53
2.2.6	Optionen für Suchmethoden mit aptitude	55
2.2.7	Aptitudes Regex-Formel	55
2.2.8	Abhängigkeitsauflösung bei aptitude	57
2.2.9	Protokollierung der Paketaktivitäten	57
2.3	Beispiele für aptitude-Operationen	57
2.3.1	Auflisten von Paketen mit Regex-Suche auf den Paketnamen	57
2.3.2	Durchsuchen mit der Regex-Suche	57
2.3.3	Entfernte Pakete endgültig löschen	57
2.3.4	Automatisch/manuell-Installationsstatus bereinigen	58
2.3.5	Systemweite Hochrüstung	59
2.4	Erweiterte Paketmanagement-Operationen	60
2.4.1	Erweiterte Paketmanagement-Operationen auf der Befehlszeile	60
2.4.2	Verifizierung von installierten Paketdateien	62
2.4.3	Absicherungen für den Fall von Paketproblemen	62
2.4.4	Durchsuchen der Paket-Metadaten	62
2.5	Internas des Debian-Paketmanagements	62
2.5.1	Archiv-Metadaten	62
2.5.2	„Release“-Datei im Wurzelverzeichnis und Authentizität	63
2.5.3	„Release“-Dateien im Archivverzeichnis	64
2.5.4	Empfangen der Metadaten für ein Paket	65
2.5.5	Der Paketstatus für APT	65
2.5.6	Der Paketstatus von aptitude	65
2.5.7	Lokale Kopien der empfangenen Pakete	65

2.5.8	Debian-Paketdateinamen	66
2.5.9	Der dpkg-Befehl	66
2.5.10	Der update-alternatives-Befehl	67
2.5.11	Der dpkg-statoverride-Befehl	68
2.5.12	Der dpkg-divert-Befehl	68
2.6	Wiederherstellung eines beschädigten Systems	68
2.6.1	Inkompatibilität mit alter Benutzerkonfiguration	68
2.6.2	Mehrere Pakete mit überlappenden Dateien	69
2.6.3	Behebung von Problemen aufgrund von beschädigtem Paketskript	69
2.6.4	Systemrettung mit dem dpkg-Befehl	69
2.6.5	Paketauswahldaten wiederherstellen	70
2.7	Tipps für das Paketmanagement	71
2.7.1	Wie Sie Debian-Pakete auswählen	71
2.7.2	Pakete aus gemischten Paketquellen	71
2.7.3	Installationskandidat-Version beeinflussen	73
2.7.4	Aktualisierungen und Backports	74
2.7.5	Über "Recommends" installierte Pakete blockieren	75
2.7.6	Nutzen von Testing mit einigen Paketen aus Unstable	75
2.7.7	Nutzen von Unstable mit einigen Paketen aus Experimental	76
2.7.8	Automatisches Herunterladen und Aktualisieren von Paketen	77
2.7.9	Die Download-Bandbreite für APT einschränken	77
2.7.10	Downgrade im Notfall	77
2.7.11	Wer hat das Paket hochgeladen?	78
2.7.12	Das equivs-Paket	78
2.7.13	Ein Paket auf das Stable-System portieren	78
2.7.14	Proxy-Server für APT	79
2.7.15	Kleines öffentliches Paketarchiv	80
2.7.16	Aufzeichnen und Kopieren der Systemkonfiguration	82
2.7.17	Konvertieren oder Installieren eines Binärpakets mit alien	82
2.7.18	Extrahieren eines Pakets ohne dpkg	82
2.7.19	Weitere Lektüre zum Paketmanagement	83
3	Die Systeminitialisierung	84
3.1	Ein Überblick über den Bootstrap-Prozess	84
3.1.1	Stufe 1: das BIOS	85
3.1.2	Stufe 2: der Bootloader	85
3.1.3	Stufe 3: das Mini-Debian-System	87
3.1.4	Stufe 4: das normale Debian-System	88
3.2	Systemd-Init	88

3.2.1	Der Rechnername	91
3.2.2	Das Dateisystem	91
3.2.3	Initialisierung der Netzwerkschnittstellen	91
3.2.4	Die Kernel-Meldungen	91
3.2.5	Die Systemmeldungen	91
3.2.6	Systemmanagement unter systemd	92
3.2.7	Anpassen von systemd	92
3.3	Das udev-System	94
3.3.1	Die Kernel-Modul-Initialisierung	95
4	Authentifizierung	96
4.1	Normale Unix-Authentifizierung	96
4.2	Verwalten von Konten- und Passwortinformationen	98
4.3	Ein gutes Passwort	98
4.4	Verschlüsselte Passwörter erstellen	99
4.5	PAM und NSS	99
4.5.1	Konfigurationsdateien, auf die PAM und NSS zugreifen	100
4.5.2	Modernes zentralisiertes Systemmanagement	100
4.5.3	”Warum unterstützt GNU su nicht die wheel-Gruppe”	101
4.5.4	Schärfere Passwortregeln	102
4.6	Andere Möglichkeiten zur Zugriffskontrolle	102
4.6.1	sudo	102
4.6.2	PolicyKit	102
4.6.3	SELinux	102
4.6.4	Den Zugriff auf einige Server-Dienste einschränken	103
4.7	Sicherheit der Authentifizierung	103
4.7.1	Sicheres Passwort im Internet	103
4.7.2	Secure Shell (sichere Shell)	104
4.7.3	Zusätzliche Sicherheitsmaßnahmen für das Internet	104
4.7.4	Sichern des root-Passworts	104
5	Netzwerkconfiguration	106
5.1	Die elementare Netzwerkinfrastruktur	106
5.1.1	Die Auflösung des Rechnernamens	108
5.1.2	Der Netzwerkschnittstellenname	109
5.1.3	Der Netzwerkadressbereich für das LAN	109
5.1.4	Unterstützung für Netzwerkgeräte	110
5.2	Moderne Netzwerkconfiguration für Arbeitsplatzsysteme	110
5.2.1	Grafische Netzwerkconfigurations-Werkzeuge	110

5.3	Moderne Netzwerkkonfiguration ohne grafische Oberfläche	111
5.4	Die Netzwerkverbindung und -konfiguration nach altbewährtem System	112
5.5	Die Netzwerkverbindungs-Methoden (nach altbewährtem System)	112
5.5.1	Die DHCP-Verbindung über Ethernet	114
5.5.2	Statische IP-Verbindung über Ethernet	114
5.5.3	Die PPP-Verbindung mit pppconfig	114
5.5.4	Die alternative PPP-Verbindung mit wvdialconf	115
5.5.5	Die PPPoE-Verbindung mit pppoeconf	115
5.6	Die grundlegende Netzwerkkonfiguration mit ifupdown (nach altbewährtem System)	116
5.6.1	Die Befehlssyntax in vereinfachter Form	116
5.6.2	Der grundlegende Aufbau von "/etc/network/interfaces"	117
5.6.3	Die loopback-Netzwerkschnittstelle	117
5.6.4	Die über DHCP bediente Netzwerkschnittstelle	117
5.6.5	Die Netzwerkschnittstelle mit statischer IP-Adresse	118
5.6.6	Die Grundlagen von kabellosen LAN-Schnittstellen	118
5.6.7	Die Wireless-LAN-Schnittstelle mit WPA/WPA2	119
5.6.8	Die Wireless-LAN-Schnittstelle mit WEP	119
5.6.9	Die PPP-Verbindung	120
5.6.10	Die alternative PPP-Verbindung	120
5.6.11	Die PPPoE-Verbindung	120
5.6.12	Der Netzwerkkonfigurations-Status von ifupdown	120
5.6.13	Die grundlegende Netzwerk-Neukonfiguration	121
5.6.14	Das ifupdown-extra-Paket	121
5.7	Die erweiterte Netzwerkkonfiguration mit ifupdown (nach altbewährtem System)	121
5.7.1	Das ifplugd-Paket	122
5.7.2	Das ifmetric-Paket	122
5.7.3	Die virtuelle Schnittstelle	123
5.7.4	Die erweiterte Befehlssyntax	123
5.7.5	Der mapping-Eintrag	124
5.7.6	Die manuell umschaltbare Netzwerkkonfiguration	124
5.7.7	Skriptverarbeitung mit dem ifupdown-System	126
5.7.8	Mapping mit guessnet	127
5.8	Die Netzwerkkonfiguration auf unterster Ebene	127
5.8.1	iproute2-Befehle	127
5.8.2	Sichere Basis-Netzwerkoperationen	127
5.9	Netzwerkoptimierung	129
5.9.1	Die optimale MTU finden	129
5.9.2	Setzen der MTU	130
5.9.3	WAN-TCP-Optimierung	131
5.10	Die Netfilter-Infrastruktur	131

6	Netzwerkapplikationen	132
6.1	Webbrowser	132
6.1.1	Browser-Konfiguration	133
6.2	Das Mail-System	133
6.2.1	Grundlagen des E-Mail-Systems	134
6.2.2	Grundlagen für moderne Mail-Dienste	134
6.2.3	Installationskonzept für Mail-Programme an Arbeitsplatz-Systemen	135
6.3	Mail Transfer Agent (MTA)	136
6.3.1	Die Konfiguration von <code>exim4</code>	137
6.3.2	Die Konfiguration von Postfix mit SASL	138
6.3.3	Die Mail-Adress-Konfiguration	139
6.3.4	Grundlegende MTA-Operationen	140
6.4	Mail User Agent (MUA)	140
6.4.1	Grundlegender MUA -- Mutt	140
6.4.2	Erweiterter MUA -- Mutt + <code>msmtp</code>	142
6.5	Programme zum Abrufen und Weiterleiten fern gespeicherter Mails	144
6.5.1	<code>getmail</code> -Konfiguration	144
6.5.2	<code>fetchmail</code> -Konfiguration	146
6.6	Mail Delivery Agents (MDA) mit Filterfunktion	146
6.6.1	<code>maildrop</code> -Konfiguration	146
6.6.2	<code>procmail</code> -Konfiguration	148
6.6.3	<code>mbox</code> -Inhalte erneut zustellen	148
6.7	POP3-/IMAP4-Server	149
6.8	Der Print-Server und Hilfsprogramme	149
6.9	Der Server für Fernzugriff (SSH) und Hilfsprogramme	150
6.9.1	Grundlagen von SSH	150
6.9.2	Portweiterleitung für SMTP-/POP3-Tunnelung	152
6.9.3	Verbindungen ohne Passwörter für die ferne Seite	153
6.9.4	Der Umgang mit fremden SSH-Clients	153
6.9.5	Einrichten von <code>ssh-agent</code>	154
6.9.6	Wie Sie das ferne System über SSH herunterfahren	154
6.9.7	Fehlersuche bei SSH	154
6.10	Weitere Netzerkanwendungs-Server	155
6.11	Weitere Netzerkanwendungs-Clients	155
6.12	Diagnose von System-Daemons	155

7	Das X-Window-System	157
7.1	Wichtige Pakete	157
7.2	Eine Arbeitsplatz-Umgebung einrichten	157
7.2.1	Debian-Menü	158
7.2.2	Freedesktop.org-Menü	158
7.2.3	Das Debian-Menü erzeugt aus dem Freedesktop.org-Menü	158
7.3	Der Zusammenhang zwischen Server und Client	158
7.4	Der X-Server	159
7.4.1	Die (Neu-)Konfiguration des X-Servers	159
7.4.2	Die Verbindungsmethoden zum X-Server	159
7.5	Starten des X-Window-Systems	160
7.5.1	Starten einer X-Sitzung mit gdm3	161
7.5.2	Anpassen der X-Sitzung (klassische Methode)	161
7.5.3	Anpassen der X-Sitzung (neue Methode)	161
7.5.4	Einen X-Client von fern via SSH verbinden	162
7.5.5	Ein sicheres X-Terminal über das Internet	162
7.6	Schriften in X-Window	162
7.6.1	Basis-Schriftarten	163
7.6.2	Zusätzliche Schriftarten	165
7.6.3	CJK-Schriftarten	165
7.7	X-Anwendungen	166
7.7.1	X-Büroanwendungen	166
7.7.2	X-Werkzeuge	166
7.8	X-Kleinigkeiten	167
7.8.1	Zwischenablage	167
7.8.2	Tastaturbelegungen und Zuweisungen von Zeigegerätetasten in X	167
7.8.3	Klassische X-Clients	167
7.8.4	Der X-Terminal-Emulator - xterm	168
7.8.5	X-Client-Programme als root laufen lassen	168
8	I18N und L10N	169
8.1	Die Tastatureingabe	169
8.1.1	Unterstützung für die Eingabemethode mit IBus	170
8.1.2	Ein Beispiel für Japanisch	170
8.1.3	Die Eingabemethode deaktivieren	171
8.2	Die Bildschirmausgabe	171
8.3	Unbekannte Zeichenbreite bei ostasiatischen Zeichen	171
8.4	Das Gebietsschema (Locale)	171
8.4.1	Grundlagen der Kodierung	172

8.4.2	Argumentation für UTF-8-Gebietsschemata	172
8.4.3	Die Neukonfiguration des Gebietsschemas (Locale)	173
8.4.4	Der Wert der Umgebungsvariable "\$LANG"	173
8.4.5	Spezifisches Gebietsschema nur für X-Window	174
8.4.6	Dateinamenkodierung	174
8.4.7	Lokalisierte Meldungen und übersetzte Dokumentation	175
8.4.8	Auswirkungen des Gebietsschemas	175
9	Systemtipps	176
9.1	Das Programm screen	176
9.1.1	Anwendungsszenario für screen	176
9.1.2	Tastaturkürzel für den screen-Befehl	177
9.2	Datenaufzeichnung und -darstellung	177
9.2.1	Der log-Daemon	177
9.2.2	Analyseprogramme für Logdateien	177
9.2.3	Shell-Aktivitäten sauber aufzeichnen	178
9.2.4	Angepasste Anzeige von Textdaten	179
9.2.5	Angepasste Anzeige von Datum und Zeit	179
9.2.6	Farbige Shell-Ausgabe	179
9.2.7	Farbige Befehle	180
9.2.8	Aufzeichnen von Editor-Aktivitäten für komplexe Wiederholungen	180
9.2.9	Die Bildschirmanzeige einer X-Anwendung aufzeichnen	181
9.2.10	Aufzeichnen von Änderungen in Konfigurationsdateien	181
9.3	Überwachen, Steuern und Starten von Programmaktivitäten	181
9.3.1	Zeitmessung für einen Prozess	181
9.3.2	Die Ablaufpriorität	183
9.3.3	Der Befehl ps	183
9.3.4	Der Befehl top	183
9.3.5	Dateien auflisten, die von einem Prozess geöffnet wurden	183
9.3.6	Programmaktivitäten verfolgen	184
9.3.7	Identifikation von Prozessen, die Dateien oder Sockets verwenden	184
9.3.8	Einen Befehl mit festem Intervall wiederholt ausführen	184
9.3.9	Einen Befehl wiederholt mit einer Schleife über verschiedene Dateien ausführen	184
9.3.10	Ein Programm aus der grafischen Oberfläche heraus starten	185
9.3.11	Anpassen des zu startenden Programms	186
9.3.12	Einen Prozess beenden (kill)	187
9.3.13	Einmalige Aufgaben planen	187
9.3.14	Regelmäßige Aufgaben planen	187
9.3.15	Die Alt-SysRq-Tastenkombination	188

9.4	Tipps zur Systempflege	188
9.4.1	Wer ist/war im System aktiv?	188
9.4.2	Allen eine Warnung schicken	189
9.4.3	Hardware-Identifikation	189
9.4.4	Hardware-Konfiguration	189
9.4.5	System- und Hardware-Zeit	190
9.4.6	Die Terminal-Konfiguration	191
9.4.7	Die Audio-Infrastruktur	191
9.4.8	Deaktivieren des Bildschirmschoners	192
9.4.9	Ausschalten von Pieptönen	192
9.4.10	Arbeitsspeichernutzung	192
9.4.11	Systemsicherheits- und Integritätsüberprüfung	192
9.5	Tipps zur Speicherung von Daten	193
9.5.1	Verwendung des Plattenplatzes	193
9.5.2	Konfiguration der Plattenpartitionen	194
9.5.3	Zugriff auf Partitionen über die UUID-Kennung	195
9.5.4	LVM2	195
9.5.5	Konfiguration von Dateisystemen	195
9.5.6	Dateisystemerzeugung und Integritätsüberprüfung	196
9.5.7	Optimierung von Dateisystemen über mount-Optionen	197
9.5.8	Optimierung von Dateisystemen über den Superblock	198
9.5.9	Optimierung der Festplatte	198
9.5.10	Optimierung von Solid State Disks	198
9.5.11	SMART verwenden, um Festplattenausfälle vorherzusehen	199
9.5.12	Angaben eines Verzeichnisses für temporäre Dateien über \$TMPDIR	199
9.5.13	Vergrößerung des nutzbaren Speicherplatzes mittels LVM	199
9.5.14	Vergrößerung des nutzbaren Speicherplatzes über das Einbinden anderer Partitionen	200
9.5.15	Vergrößerung des nutzbaren Speicherplatzes, indem ein anderes Verzeichnis mit "mount --bind" eingebunden wird	200
9.5.16	Vergrößerung des nutzbaren Speicherplatzes, indem ein anderes Verzeichnis mit "Overlay-mounting" eingebunden wird	200
9.5.17	Vergrößerung des nutzbaren Speicherplatzes über einen symbolischen Link	200
9.6	Das Festplatten-Abbild	201
9.6.1	Erzeugung der Festplatten-Abbild-Datei	201
9.6.2	Direkt auf eine Festplatte schreiben	201
9.6.3	Einbinden der Festplatten-Abbild-Datei	202
9.6.4	Eine Festplatten-Abbild-Datei bereinigen	203
9.6.5	Eine leere Abbild-Datei erstellen	203
9.6.6	Erstellen einer ISO9660-Abbild-Datei	204

9.6.7	Direkt auf die CD/DVD-R/RW schreiben	204
9.6.8	Einbinden einer ISO9660-Abbild-Datei	205
9.7	Binärdaten	205
9.7.1	Betrachten und Bearbeiten von Binärdaten	205
9.7.2	Manipulieren von Dateien ohne Einbinden der Festplatte	206
9.7.3	Datenredundanz	206
9.7.4	Datenwiederherstellung und forensische Analyse	206
9.7.5	Aufsplitten einer großen in mehrere kleine Dateien	206
9.7.6	Leeren von Dateiinhalten	207
9.7.7	Dummy-Dateien	207
9.7.8	Eine vollständige Festplatte löschen	207
9.7.9	Einen ungenutzten Bereich einer Festplatte löschen	208
9.7.10	Wiederherstellen von gelöschten, aber noch geöffneten Dateien	208
9.7.11	Alle harten Links suchen	209
9.7.12	Unsichtbarer Verbrauch von Festplattenplatz	209
9.8	Tipps zur Datenverschlüsselung	209
9.8.1	Verschlüsselung von Wechseldatenträgern mit dm-crypt/LUKS	210
9.8.2	Verschlüsselte Swap-Partition mit dm-crypt	211
9.8.3	Einbinden verschlüsselter Laufwerke mit dm-crypt/LUKS	211
9.8.4	Automatische Verschlüsselung von Dateien mit eCryptfs	211
9.8.5	Automatisches Einbinden von eCryptfs	212
9.9	Der Kernel	212
9.9.1	Linux-Kernel 2.6/3.x	213
9.9.2	Kernel-Parameter	213
9.9.3	Kernel-Header	213
9.9.4	Kompilieren des Kernels und dazugehöriger Module	213
9.9.5	Kompilieren des Kernel-Quellcodes: Empfehlung des Debian-Kernel-Teams	214
9.9.6	Hardware-Treiber und Firmware	215
9.10	Virtualisierte Systeme	215
9.10.1	Virtualisierungswerkzeuge	216
9.10.2	Arbeitsablauf bei Virtualisierung	216
9.10.3	Einbinden des virtuellen Festplatten-Images	217
9.10.4	Chroot-System	218
9.10.5	System mit mehrfachen Arbeitsplatzumgebungen	219

10 Datenmanagement	220
10.1 Austauschen, kopieren und archivieren von Dateien	220
10.1.1 Archivierungs- und Kompressionswerkzeuge	221
10.1.2 Kopier- und Synchronisationswerkzeuge	221
10.1.3 Aufrufe für Archivierungsoperationen	223
10.1.4 Aufrufe für Kopieroperationen	223
10.1.5 Aufrufe für die Auswahl von Dateien	224
10.1.6 Archivierungsmedien	225
10.1.7 Wechseldatenträger	226
10.1.8 Dateisystemauswahl für den Datenaustausch	227
10.1.9 Datenaustausch über das Netzwerk	228
10.2 Datensicherung und -wiederherstellung	229
10.2.1 Programmsammlungen für Datensicherungsaufgaben	230
10.2.2 Ein Beispielskript für die Systemsicherung	232
10.2.3 Ein Kopierskript für die Datensicherung	233
10.3 Datensicherheits-Infrastruktur	234
10.3.1 Schlüsselverwaltung für GnuPG	235
10.3.2 Verwendung von GnuPG mit Dateien	235
10.3.3 Verwendung von GnuPG mit Mutt	237
10.3.4 Verwendung von GnuPG mit Vim	237
10.3.5 Die MD5-Prüfsumme	237
10.4 Werkzeuge zur Quellcode-Zusammenführung	237
10.4.1 Unterschiede für Quelldateien extrahieren	237
10.4.2 Aktualisierungen für Quelldateien zusammenführen	237
10.4.3 Aktualisierung via 3-Wege-Zusammenführung	239
10.5 Versionskontrollsysteme	239
10.5.1 Gegenüberstellung von VCS-Befehlen	240
10.6 Git	241
10.6.1 Konfiguration eines Git-Clients	241
10.6.2 Weitere Referenzen zu Git	241
10.6.3 Git-Befehle	242
10.6.4 Git mit einem Subversion-Depot	243
10.6.5 Git zur Aufzeichnung der Historie von Konfigurationsdateien	243
10.7 CVS	244
10.7.1 Konfiguration eines CVS-Depots	244
10.7.2 Lokaler Zugriff auf CVS	245
10.7.3 Fernzugriff auf CVS mit pserver	245
10.7.4 Fernzugriff auf CVS mit ssh	245
10.7.5 Eine neue Quelle in CVS importieren	245

10.7.6	Dateiberechtigungen im CVS-Depot	246
10.7.7	Arbeitsablauf bei CVS	246
10.7.8	Aktuellste Dateien von CVS	248
10.7.9	Administrierung von CVS	248
10.7.10	Ausführungs-Bit für CVS-Checkout	249
10.8	Subversion	249
10.8.1	Konfiguration eines Subversion-Depots	249
10.8.2	Zugriff auf Subversion über einen Apache2-Server	249
10.8.3	Lokaler Zugriff auf Subversion durch die Gruppe	250
10.8.4	Fernzugriff auf Subversion über SSH	250
10.8.5	Subversion-Verzeichnisstruktur	250
10.8.6	Eine neue Quelle in Subversion importieren	250
10.8.7	Arbeitsablauf bei Subversion	251
11	Datenkonvertierung	254
11.1	Werkzeuge für Textkonvertierung	254
11.1.1	Konvertieren einer Textdatei mit iconv	254
11.1.2	Prüfen mit iconv, ob eine Datei UTF-8-kodiert ist	256
11.1.3	Dateinamen konvertieren mit iconv	256
11.1.4	EOL-Konvertierung	256
11.1.5	TAB-Konvertierung	257
11.1.6	Editoren mit automatischer Konvertierung	257
11.1.7	Extrahieren von reinem Text	258
11.1.8	Hervorheben und Formatieren von reinen Textdaten	259
11.2	XML-Daten	259
11.2.1	Grundlegende Hinweise für XML	260
11.2.2	XML-Verarbeitung	261
11.2.3	Extrahierung von XML-Daten	261
11.3	Textsatz	262
11.3.1	roff-Textsatz	262
11.3.2	TeX/LaTeX	263
11.3.3	Schöner Ausdruck einer Handbuchseite	264
11.3.4	Erstellen einer Handbuchseite	264
11.4	Druckfähige Daten	264
11.4.1	Ghostscript	264
11.4.2	Zwei PS- oder PDF-Dateien zusammenführen	265
11.4.3	Werkzeuge für druckfähige Daten	265
11.4.4	Drucken mit CUPS	266
11.5	Konvertierung von Mail-Daten	266
11.5.1	Grundlagen zu Mail-Daten	266
11.6	Werkzeuge für Grafikdaten	267
11.7	Verschiedene Datenkonvertierungen	267

12 Programmierung	270
12.1 Das Shellskript	271
12.1.1 POSIX-Shell-Kompatibilität	271
12.1.2 Shellparameter	272
12.1.3 Bedingte Ausdrücke in der Shell	273
12.1.4 Shellschleifen	274
12.1.5 Befehlsabfolge auf der Shell	274
12.1.6 Hilfsprogramme für Shellskripte	275
12.1.7 Shellskript-Dialog	276
12.1.8 Shellskript-Beispiel mit zenity	276
12.2 Make	277
12.3 C	278
12.3.1 Ein einfaches C-Programm (gcc)	278
12.4 Fehlersuche (Debugging)	278
12.4.1 Grundlegende Ausführung von gdb	279
12.4.2 Fehlersuche in einem Debian-Paket (Debugging)	279
12.4.3 Gewinnen von Backtrace-Informationen	280
12.4.4 Erweiterte gdb-Befehle	280
12.4.5 Fehleranalyse bei X-Fehlern	281
12.4.6 Überprüfen der Abhängigkeiten von Bibliotheken	281
12.4.7 Werkzeuge zur Erkennung von Speicherlecks	281
12.4.8 Werkzeuge zur statischen Code-Analyse	281
12.4.9 Disassemblieren von Binärdateien	281
12.5 Flex - ein besseres Lex	282
12.6 Bison - ein besseres Yacc	282
12.7 Autoconf	283
12.7.1 Kompilieren und Installieren eines Programms	283
12.7.2 Deinstallation eines Programms	283
12.8 Verrücktheiten bei kurzen Perl-Skripten	283
12.9 Web	284
12.10 Die Quellcode-Übersetzung	284
12.11 Erstellen von Debian-Paketen	285
A Anhang	286
A.1 Das Debian-Labyrinth	286
A.2 Copyright-Vergangenheit	286
A.3 Dokumentenformat	287

Tabellenverzeichnis

1.1	Liste interessanter Textmodus-Programm-Pakete	5
1.2	Liste informativer Dokumentationspakete	5
1.3	Auflistung der Verwendung wichtiger Verzeichnisse	8
1.4	Bedeutungen des ersten Zeichens der "ls -l"-Ausgabe	10
1.5	Der numerische Modus für Dateiberechtigungen in chmod(1)-Befehlen	11
1.6	Beispiele für umask-Werte	11
1.7	Liste erwähnenswerter systemweiter Gruppen	12
1.8	Liste erwähnenswerter, vom System angebotener Gruppen zur Ausführung besonderer Befehle	12
1.9	Liste der Arten von Zeitstempeln	13
1.10	Liste spezieller Gerädateien	17
1.11	Die Tastaturbefehle von MC	19
1.12	Die Reaktion auf die Enter-Taste in MC	20
1.13	Liste von Shell-Programmen	20
1.14	Liste der Tastaturbefehle für bash	22
1.15	Liste der Mausoperationen im Unix-Stil	22
1.16	Liste grundlegender Unix-Befehle	25
1.17	Die drei Teile des locale-Wertes	27
1.18	Liste mit Empfehlungen zum Gebietsschema	27
1.19	Liste der Werte von "\$HOME"	28
1.20	Shell-Glob-Suchmuster	29
1.21	Befehls-Beendigungs-Codes	30
1.22	Abfolgen von Shell-Befehlen	31
1.23	Vordefinierte Datei-Deskriptoren	31
1.24	Metazeichen für BRE und ERE	34
1.25	Der Ersetzungsausdruck	35
1.26	Liste von Skript-Schnipseln für die Befehlsweiterleitung	38
2.1	Liste von Debians Paketmanagement-Werkzeugen	40
2.2	Liste von Debian-Archiv-Seiten	43
2.3	Liste der Debian-Archiv-Bereiche	43

2.4	Zusammenhang zwischen Suite und Codename	44
2.5	Liste wichtiger Websites zur Lösung von Paketproblemen	49
2.6	Grundlegende Paketmanagement-Operationen mit apt(8), aptitude(8) und apt-get(8) / apt-cache(8)	51
2.7	Erwähnenswerte Befehlsoptionen für aptitude(8)	52
2.8	Liste der Tastaturkürzel für aptitude	53
2.9	Liste der Ansichten für aptitude	54
2.10	Die Kategorisierung von Standard-Paketansichten	54
2.11	Liste von aptitudes Regex-Formeln	56
2.12	Protokolldateien für Paketaktivitäten	57
2.13	Liste erweiterter Paketmanagement-Operationen	61
2.14	Inhalt der Metadaten des Debian-Archivs	63
2.15	Namensstruktur von Debian-Paketen	66
2.16	In den einzelnen Komponenten von Debian-Paketnamen zu verwendende Zeichen	66
2.17	Erwähnenswerte Dateien, die durch dpkg erzeugt werden	67
2.18	Liste erwähnenswerter Pin-Prioritäts-Werte für apt-pinning	74
2.19	Liste von Proxy-Hilfsprogrammen speziell für das Debian-Archiv	79
3.1	Liste der Bootloader	86
3.2	Bedeutung der GRUB-Parameter	87
3.3	Liste von Boot-Hilfsprogrammen für das Debian-System	89
3.4	Liste der Schwellwerte zur Filterung von Kernel-Fehler-Meldungen	91
3.5	Liste typischer systemd-Management-Befehle	93
4.1	Wichtige Konfigurationsdateien für pam_unix(8)	96
4.2	Inhalt des zweiten Eintrags in "/etc/passwd"	97
4.3	Liste von Befehlen zur Verwaltung von Konteninformationen	98
4.4	Liste der Werkzeuge zur Passwörterzeugung	99
4.5	Liste von PAM- und NSS-Systemen	100
4.6	Liste von Konfigurationsdateien, auf die PAM und NSS zugreifen	101
4.7	Liste von unsicheren und sicheren Diensten und Ports	104
4.8	Liste von Werkzeugen, die zusätzliche Sicherheitsmaßnahmen ermöglichen	104
5.1	Liste von Werkzeugen zur Netzwerkkonfiguration	107
5.2	Liste der Netzwerkadressbereiche	109
5.3	Liste von Netzwerkverbindungsmethoden und Verbindungspfaden	113
5.4	Liste der Netzwerkverbindungskonfigurationen	113
5.5	Liste der Abkürzungen bei Netzwerkverbindungen	113
5.6	Liste der Konfigurationsdateien für eine PPP-Verbindung mit pppconfig	114
5.7	Liste der Konfigurationsdateien für eine PPP-Verbindung mit wvdialconf	115
5.8	Liste der Konfigurationsdateien für eine PPPoE-Verbindung mit pppoeconf	116

5.9	Liste der Befehle zur grundlegenden Netzwerkkonfiguration mit ifupdown	116
5.10	Liste der Einträge in <code>"/etc/network/interfaces"</code>	117
5.11	Liste der Abkürzungen für WLAN	119
5.12	Liste verschiedener Bezeichnungsweisen für Netzwerkschnittstellen	123
5.13	Liste erweiterter Befehle zur Netzwerkkonfiguration mit ifupdown	124
5.14	Liste der Umgebungsvariablen, die dem ifupdown-System übergeben werden	126
5.15	Gegenüberstellung von <code>net-tools</code> - und <code>iproute2</code> -Befehlen	128
5.16	Liste von Basis-Netzwerkbefehlen	128
5.17	Liste von Werkzeugen zur Netzwerkoptimierung	129
5.18	Wesentliche Grundregeln für den optimalen MTU-Wert	130
5.19	Liste von Firewall-Werkzeugen	131
6.1	Liste der Webbrowser	132
6.2	Liste von Browser-Plugin-Paketen	133
6.3	Liste von grundlegenden Paketen für Mail Transfer Agents	136
6.4	Liste von Mail-Transport-Agent-Paketen im Debian-Archiv	136
6.5	Liste von wichtigen Postfix-Handbuchseiten	138
6.6	Liste von Konfigurationsdateien für Mail-Adressen	139
6.7	Liste grundlegender MTA-Operationen	141
6.8	Liste der Mail User Agents (MUA)	141
6.9	Liste von Programmen zum Abrufen und Weiterleiten von fern gespeicherten Mails	144
6.10	Liste von MDAs mit Filterfunktion	146
6.11	Liste von POP3-/IMAP4-Servern	149
6.12	Liste von Print-Servern und Hilfsprogrammen	149
6.13	Liste von Servern für Fernzugriff und Hilfsprogrammen	150
6.14	Liste von SSH-Authentifizierungsprotokollen und -methoden	151
6.15	Liste von SSH-Konfigurationsdateien	151
6.16	Liste von Beispielen zum Start einer SSH-Verbindung auf einem Client	152
6.17	Liste freier SSH-Clients für andere Plattformen	154
6.18	Liste von weiteren Netzerkennungs-Servern	155
6.19	Liste von Netzerkennungs-Clients	156
6.20	Liste populärer RFCs	156
7.1	Liste von (Meta-)Paketen für X-Window	158
7.2	Liste der Server-/Client-Terminologie	159
7.3	Liste der Verbindungsmethoden zum X-Server	160
7.4	Tabelle der Pakete, die die X-Window-Schriftensysteme unterstützen	163
7.5	Tabelle von sich entsprechenden PostScript-Typ1-Schriftarten	164
7.6	Tabelle von sich entsprechenden TrueType-Schriftarten	164

7.7	Tabelle von in CJK-Schriftnamen verwendeten Schlüsselwörtern zur Erkennung von Schrifttypen	165
7.8	Liste grundlegender X-Büroanwendungen	166
7.9	Liste grundlegender X-Werkzeuge	167
7.10	Liste von Basis-Programmen für die X-Auswahl	167
8.1	Liste der unterstützten Eingabemethoden mit IBus	170
9.1	Liste von Programmen, die Netzwerkverbindungen mit Unterbrechungen unterstützen:	176
9.2	Liste von Tastaturkürzeln für screen	177
9.3	Liste von System-Log-Analysatoren	178
9.4	Anzeigebeispiele von Datum und Zeit für den Befehl "ls -l" unter wheezy	179
9.5	Liste von Werkzeugen zur Bildbearbeitung	181
9.6	Liste von Paketen zur Aufzeichnung der Konfigurations-Historie via VCS	181
9.7	Liste von Werkzeugen zur Überwachung und Steuerung von Programmaktivitäten	182
9.8	Liste der nice-Werte für die Ablaufpriorität	183
9.9	Liste von ps-Befehlen	183
9.10	Liste von häufig verwendeten Signalen für den kill-Befehl	187
9.11	Liste der SAK-Befehlstasten	188
9.12	Liste von Werkzeugen zur Hardware-Identifikation	189
9.13	Liste von Werkzeugen zur Hardware-Konfiguration	190
9.14	Liste von Audio-Paketen	191
9.15	Liste von Befehlen zur Deaktivierung des Bildschirmschoners	192
9.16	Liste der gemeldeten Arbeitsspeichergrößen	193
9.17	Liste von Werkzeugen für eine Systemsicherheits- und Integritätsüberprüfung	193
9.18	Liste von Paketen zum Partitions-Management	194
9.19	Liste von Paketen für das Dateisystem-Management	196
9.20	Liste von Paketen zum Betrachten und Bearbeiten von Binärdaten	205
9.21	Liste von Paketen zur Manipulation von Dateien ohne Einbinden der Festplatte	206
9.22	Liste von Werkzeugen, um Redundanz für Dateien hinzuzufügen	206
9.23	Liste von Paketen für Datenwiederherstellung und forensische Analysen	207
9.24	Liste von Werkzeugen zur Datenverschlüsselung	210
9.25	Liste von Schlüsselpaketen für die Neukompilierung des Kernels auf einem Debian-System	214
9.26	Liste von Virtualisierungswerkzeugen	216
10.1	Liste von Archivierungs- und Kompressionswerkzeugen	222
10.2	Liste von Kopier- und Synchronisationswerkzeugen	222
10.3	Liste von Dateisystemen für Wechseldatenträger mit typischen Anwendungsszenarien	228
10.4	Liste von Netzwerkdiensten mit typischen Anwendungsszenarien	229
10.5	Liste von Datensicherungsprogrammen	231
10.6	Liste von Werkzeugen für die Datensicherheits-Infrastruktur	234

10.7	Liste von GNU Privacy Guard-Befehlen für die Schlüsselverwaltung	235
10.8	Liste der Bedeutungen des Vertrauenscodes	235
10.9	Liste von GNU Privacy Guard-Befehlen mit Dateien	236
10.10	Liste von Werkzeugen zur Quellcode-Zusammenführung	238
10.11	Liste von Werkzeugen für die Versionskontrolle	239
10.12	Gegenüberstellung nativer VCS-Befehle	240
10.13	Liste von zu Git gehörigen Paketen und Befehlen	242
10.14	Erwähnenswerte Optionen für CVS-Befehle (als erste Argumente bei cvs(1) zu verwenden):	248
10.15	Erwähnenswerte Optionen für Subversion-Befehle (als erste Argumente bei svn(1)) zu verwenden):	253
11.1	Liste von Textkonvertierungs-Werkzeugen	254
11.2	Liste von Werten für die Zeichenkodierung und deren Verwendung	255
11.3	Liste der EOL-Codes für verschiedene Plattformen	257
11.4	Liste der Befehle zur TAB-Konvertierung aus den Paketen <code>bsdmainutils</code> und <code>coreutils</code>	257
11.5	Liste von Werkzeugen zum Extrahieren von reinen Textdaten	258
11.6	Liste von Werkzeugen für Hervorhebung/Formatierung von Textdaten	259
11.7	Liste von vordefinierten Entitäten für XML	260
11.8	Liste von XML-Werkzeugen	261
11.9	Liste von DSSSL-Werkzeugen	261
11.10	Liste von Werkzeugen zur Extrahierung von XML-Daten	262
11.11	Liste von XML-Druck-Werkzeugen	262
11.12	Liste von Textsatz-Werkzeugen	262
11.13	Liste von Paketen, die bei der Erstellung einer Handbuchseite helfen	264
11.14	Liste von Ghostscript-PostScript-Interpretern	264
11.15	Liste von Werkzeugen für druckfähige Daten	265
11.16	Liste von Paketen zur Konvertierung von Mail-Daten	266
11.17	Liste von Werkzeugen für Grafikdaten	268
11.18	Liste verschiedener Werkzeuge zur Datenkonvertierung	269
12.1	Liste von Paketen für die Programmierung	270
12.2	Liste typischer Bashisms	272
12.3	Liste von Shellparametern	272
12.4	Liste von Parameterauswertungen	272
12.5	Liste von Shellparameterersetzungen	273
12.6	Liste von Dateivergleichsoperatoren in bedingten Ausdrücken	273
12.7	Liste von String-Vergleichsoperatoren im bedingten Ausdruck	274
12.8	Liste der Pakete, die kleine Hilfsprogramme für Shellskripte enthalten	275
12.9	Liste von Programmen für die Benutzerschnittstelle	276
12.10	Liste von automatischen make-Variablen	277

12.11 Liste von make-Variablenexpandierungen	277
12.12 Liste erweiterter gdb-Befehle	280
12.13 Liste von Werkzeugen zur Erkennung von Speicherlecks	281
12.14 Liste von Werkzeugen für die statische Code-Analyse	282
12.15 Liste Yacc-kompatibler LALR-Parser-Generatoren	282
12.16 Liste von Programmen zur Übersetzung von Quellcode	285

Zusammenfassung

Dieses Buch ist frei; Sie dürfen es unter den Bedingungen der GNU General Public License in jeder Version, die verträglich mit den Debian Richtlinien für Freie Software (DFSG) ist, weiterverteilen und/oder verändern.

Vorwort

Diese [Debian-Referenz \(Version 2.77\)](#) (2021-01-10 06:32:51 UTC) soll für die Zeit nach der Installation einen groben Überblick über das Debian-System in Form eines Benutzerhandbuchs bieten.

Es spricht diejenigen Leser an, die bereit sind, Shellskripte zu lernen, aber nicht bereit sind, alle C-Quellen zu lesen, um herauszufinden, wie das [GNU/Linux](#)-System genau funktioniert.

Anweisungen zur Installation eines Debian-Systems finden Sie unter:

- [Debian GNU/Linux Installationsanleitung für das aktuelle Stable-System](#)
- [Debian GNU/Linux Installationsanleitung für das Testing-System](#)

Haftungsausschluss

Jegliche Gewährleistung wird ausgeschlossen. Alle Handelsmarken sind Eigentum ihrer jeweiligen Markeninhaber.

Das Debian-System selbst ist ein bewegliches Ziel. Das macht es schwer, diese Dokumentation aktuell und korrekt zu halten. Obgleich die aktuelle Unstable-Version des Debian-Systems beim Schreiben dieser Referenz als Basis benutzt wurde, mag einiges vom Inhalt bereits veraltet sein, wenn Sie dies lesen.

Bitte behandeln Sie dieses Dokument als zweitrangige Referenz. Dieses Dokument ersetzt nicht die maßgeblichen Handbücher. Autor und Helfer/Übersetzer übernehmen keine Verantwortung für die Folgen von Fehlern, Auslassungen oder Zweideutigkeiten in diesem Dokument.

Was ist Debian

Das [Debian-Projekt](#) ist eine Vereinigung von Einzelpersonen, die es sich zur gemeinsamen Aufgabe gemacht haben, ein freies Betriebssystem zu erstellen. Seine Distribution zeichnet sich durch folgende Aspekte aus:

- Verpflichtung zur Freiheit von Software: [Debian-Gesellschaftsvertrag und Debian-Richtlinien für Freie Software \(DFSG\)](#);
- internet-basierte, verteilte, unbezahlte und freiwillige Leistung: <https://www.debian.org>;
- eine große Anzahl vorkompilierter, hochqualitativer Software-Pakete;
- Fokus auf Stabilität und Sicherheit mit einfachem Zugang zu Sicherheitsaktualisierungen;
- Fokus auf leichte Aktualisierung auf die neuesten Software-Pakete über die `unstable`- und `testing`-Archive;
- eine große Zahl unterstützter Hardware-Architekturen.

Freie Software in Debian stammt von [GNU](#), [Linux](#), [BSD](#), [X](#), [ISC](#), [Apache](#), [Ghostscript](#), [Common Unix Printing System](#), [Samba](#), [GNOME](#), [KDE](#), [Mozilla](#), [LibreOffice](#), [Vim](#), [TeX](#), [LaTeX](#), [DocBook](#), [Perl](#), [Python](#), [Tcl](#), [Java](#), [Ruby](#), [PHP](#), [Berkeley DB](#), [MariaDB](#), [PostgreSQL](#), [SQLite](#), [Exim](#), [Postfix](#), [Mutt](#), [FreeBSD](#), [OpenBSD](#), [Plan 9](#) und vielen weiteren unabhängigen Freie-Software-Projekten. Debian integriert diese Vielzahl freier Software in ein Ökosystem.

Über dieses Dokument

Leitregeln

Folgende Leitregeln wurden beim Erstellen dieses Dokuments verfolgt:

- liefere einen Überblick und überspringe Sonderfälle (**das Gesamtbild**);
- halte es kurz und einfach (Keep It Short and Simple, **KISS**);
- erfinde das Rad nicht neu (benutze Verweise auf **bestehende Referenzen**);
- Fokus auf nicht-grafische Werkzeuge und Konsolen (benutze **Shell-Beispiele**);
- Sei objektiv (benutze [popcon](#) usw.).

Tipp

Ich habe versucht, hierarchische Aspekte und die unteren Ebenen des Systems zu erläutern.

Vorraussetzungen



Warnung

Sie können davon ausgehen, auch außerhalb dieser Dokumentation leicht selbst Antworten zu finden. Dieses Dokument liefert Ihnen lediglich effiziente Ausgangssituationen.

Verwenden Sie folgende primäre Informationsquellen, um selbst nach Lösungen suchen:

- [Das Debian Administrationshandbuch](#)
- die Debian-Website unter <https://www.debian.org> für grundsätzliche Informationen;
- die Dokumentation im Verzeichnis `/usr/share/doc/<paketname>`;
- die **Handbuchseiten (manpages)** im Unix-Stil: `dpkg -L <paketname> |grep '/man/man.*/'`;
- die **Infoseiten** im GNU-Stil: `dpkg -L <paketname> |grep '/info/'`;
- die Fehlerdatenbank: <http://bugs.debian.org/<paketname>>;
- das Debian Wiki unter <https://wiki.debian.org/> für sich schnell ändernde und spezielle Themen;
- die HOWTOs vom Linux Documentation Project (TLDP) unter <http://tldp.org/>;
- die Single UNIX Specification von Open Group's Homepage des UNIX-Systems unter <http://www.unix.org/>;
- die freie Enzyklopädie von Wikipedia unter <https://www.wikipedia.org/>.

Anmerkung

Für detaillierte Dokumentation zu einem speziellen Programm/Paket müssen Sie möglicherweise das entsprechende Dokumentationspaket mit `- doc` als Anhang hinter dem Namen installieren.

Konventionen

Dieses Dokument bietet Informationen im folgenden vereinfachten Präsentationsstil mit bash(1) Shell-Befehl-Beispielen:

```
# <Befehl als root>
$ <Befehl als normaler Benutzer>
```

Diese Shell-Prompts unterscheiden sich abhängig vom verwendeten Benutzer und sind deckungsgleich mit dem Setzen von Umgebungsvariablen wie "PS1='\\$'" und "PS2=' '". Diese Werte wurden zwecks Lesbarkeit in diesem Dokument ausgewählt und sind nicht zwingend typisch für ein aktuell installiertes System.

Anmerkung

Lesen Sie dazu die Bedeutung der Umgebungsvariablen "\$PS1" und "\$PS2" in bash(1).

Eine vom Systemadministrator geforderte **Aktion** ist in der Imperativ-Form (Befehlsform) geschrieben, z.B. "Enter-Taste drücken nach jedem Eingeben eines Befehls in der Shell".

Spalten wie **Beschreibung** und ähnliche in einer Tabelle können eine **Nominalphrase** im Sinne der [Richtlinien für die Paket-Kurzbeschreibungen](#) enthalten, wobei führende Artikel wie "ein" oder "der" weggelassen werden. Sie können alternativ auch eine Infinitivphrase als **Nominalphrase** ohne das "zu" am Anfang enthalten, folgend den Richtlinien für Befehls-Kurzbeschreibungen in Handbuchseiten. Dies mag einigen Leuten lustig erscheinen, ist aber meine beabsichtigte Stilwahl, um diese Dokumentation so einfach wie möglich zu halten. Diese **Nominalphrasen** beginnen nicht mit einem Großbuchstaben und enden nicht mit einem Punkt, gemäß den Richtlinien für Kurzbeschreibungen.

Anmerkung

Korrekte Nomen inklusive Befehlsnamen bleiben von der Groß-/Kleinschreibung her immer gleich, unabhängig von Ihrer Position im Satz.

Ein **Befehlsschnipsel**, der in einem Textabschnitt zitiert wird, ist in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen dargestellt, z.B. "aptitude safe-upgrade".

Ein **Ausschnitt von Daten** aus einer Konfigurationsdatei, der in einem Textabschnitt zitiert wird, ist in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen dargestellt, z.B. "deb-src".

Ein **Befehl** wird mit seinem Namen in Schreibmaschinen-Schriftart und optional gefolgt von der Nummer seines Handbuchseiten-Abschnitts in Klammern dargestellt, z.B. bash(1). Geben Sie Folgendes ein, um weitere Informationen zu erhalten:

```
$ man 1 bash
```

Eine **Handbuchseite (manpage)** wird mit ihrem Namen in Schreibmaschinen-Schriftart gefolgt von der Nummer ihres Handbuchseiten-Abschnitts in Klammern dargestellt, z.B. sources.list(5). Geben Sie Folgendes ein, um weitere Informationen zu erhalten:

```
$ man 5 sources.list
```

Eine **Infoseite** wird mit dem Befehlsschnipsel in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen dargestellt, z.B. "info make". Geben Sie Folgendes ein, um weitere Informationen zu erhalten:

```
$ info make
```

Ein **Dateiname** wird in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen dargestellt, z.B. "/etc/passwd". Geben Sie Folgendes ein, um z.B. den Inhalt von Konfigurationsdateien anzuzeigen:

```
$ sensible-pager "/etc/passwd"
```

Ein **Verzeichnisname** wird in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen dargestellt, z.B. "/etc/apt/". Geben Sie Folgendes ein, um den Verzeichnisinhalt anzuzeigen:

```
$ mc "/etc/apt/"
```

Ein **Paketname** wird mit seinem Namen in Schreibmaschinen-Schriftart dargestellt, z.B. `vim`. Geben Sie Folgendes ein, um weitere Informationen zu dem Paket zu erhalten:

```
$ dpkg -L vim
$ apt-cache show vim
$ aptitude show vim
```

Bei **Dokumentation** wird der Ablageort entweder über den Dateinamen in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen angezeigt, z.B. `"/usr/share/doc/base-passwd/users-and-groups.txt.gz"` und `"/usr/share/doc/` oder über seine **URL**, z.B. <https://www.debian.org>. Geben Sie Folgendes ein, um die Dokumentation zu lesen:

```
$ zcat "/usr/share/doc/base-passwd/users-and-groups.txt.gz" | sensible-pager
$ sensible-browser "/usr/share/doc/base-passwd/users-and-groups.html"
$ sensible-browser "https://www.debian.org"
```

Eine **Umgebungsvariable** wird mit ihrem Namen und einem führenden "\$" in Schreibmaschinen-Schriftart zwischen doppelten Anführungszeichen dargestellt, z.B. `"$VAR"`. Geben Sie Folgendes ein, um ihren derzeitigen Wert zu erhalten:

```
$ echo "$VAR"
```

Der Popularitäts-Wettbewerb (popcon)

Die Daten des [Debian Popularitäts-Wettbewerbs](#) (Debian Popularity Contest / popcon) stellen ein objektives Messinstrument für die Beliebtheit eines jeden Pakets dar. Sie wurden am 2021-01-10 06:32:17 UTC heruntergeladen und enthalten die gesamte Menge von 197967 Meldungen über 178828 Binärpakete und 26 Architekturen.

Anmerkung

Bitte beachten Sie, dass das amd64-Archiv in `unstable` derzeit nur 62716 Pakete enthält. Die popcon-Daten enthalten auch Berichte von vielen alten Systeminstallationen.

Die popcon-Zahl mit einem vorangestellten "V:" für "votes" wird berechnet über $1000 * (\text{Anzahl der popcon-Berichte, dass das Paket auf dem PC kürzlich ausgeführt wurde}) / (\text{Gesamtanzahl der popcon-Berichte})$.

Die popcon-Zahl mit einem vorangestellten "I:" für "installs" wird berechnet über $1000 * (\text{Anzahl der popcon-Berichte, dass das Paket auf dem PC installiert ist}) / (\text{Gesamtanzahl der popcon-Berichte})$.

Anmerkung

Die popcon-Daten sollten nicht als absolutes Maß für die Wichtigkeit des Pakets angesehen werden. Es gibt viele Faktoren, die die Statistik beeinflussen können. Zum Beispiel könnten einige Systeme, die an dem Popularitäts-Wettbewerb teilnehmen, eingebundene Verzeichnisse wie `/bin` mit der Option `"noatime"` haben, um die System-Performance zu verbessern, und so wäre "votes" für solche Systeme letztlich deaktiviert.

Die Paketgröße

Die Paketgrößen-Daten werden auch als objektives Maß für jedes Paket angeboten. Sie basieren auf der Angabe `"Installed-Size:"`, die vom Befehl `"apt-cache show"` oder `"aptitude show"` angezeigt werden (derzeit von der amd64-Architektur für die `Unstable`-Veröffentlichung). Die Größe wird in KiB ([Kibibyte](#) = Einheit für 1024 Byte) angegeben.

Anmerkung

Ein Paket mit einer sehr kleinen numerischen Paketgröße kann darauf hindeuten, dass das Paket in der `Unstable`-Veröffentlichung ein Dummy-Paket ist, das andere Pakete mit signifikantem Inhalt über die Abhängigkeiten installiert. Dummy-Pakete ermöglichen sanfte Versionsübergänge oder das Aufsplitten von Paketen in mehrere kleinere Einzelpakete.

Anmerkung

Eine Paketgröße gefolgt von "(*)" zeigt an, dass das Paket in der `Unstable`-Veröffentlichung fehlt und stattdessen die Paketgröße aus der `Experimental`-Veröffentlichung angezeigt wird.

Fehlerbericht gegen dieses Dokument

Falls Sie irgendwelche Fehler in diesem Dokument finden, melden Sie diese gegen das Paket `debian-reference`; Sie können dazu `reportbug(1)` verwenden. Fügen Sie bitte Korrekturvorschläge für den Fehler bei, als `"diff -u"` gegen den reinen Text oder die Quellen.

Gedächtnisstützen für neue Benutzer

Hier einige Gedächtnisstützen für neue Benutzer:

- Sichern Sie Ihre Daten
- Halten Sie Passwörter und Schlüssel verschlossen
- [KISS - halte es kurz und einfach \(keep it simple stupid\)](#)
 - Machen Sie Ihr System nicht unnötig kompliziert
- Lesen Sie die Protokolldateien Ihres System
 - Der **ERSTE** Fehler ist der, der zählt
- [RTFM - lesen Sie das tolle Handbuch \(read the fine manual\)](#)
- Suchen Sie im Internet, bevor Sie Fragen stellen
- Arbeiten Sie nicht als root, wenn es nicht erforderlich ist
- Spielen Sie nicht mit dem Paketmanagementsystem herum
- Geben Sie keine Befehle ein, die Sie nicht verstehen
- Ändern Sie keine Dateizugriffsrechte
- Verlassen Sie die root-Shell nicht, bevor Sie Ihre Änderungen **GETESTET** haben
- Halten Sie stets ein alternatives Boot-Medium bereit (USB-Stick, CD-ROM, ...)

Einige Zitate für neue Benutzer

Hier einige interessante Zitate von den Debian-Mailinglisten, die helfen können, neue Benutzer aufzuklären.

- "Dies ist Unix. Es gibt Ihnen genug Seil, damit Sie sich selbst erhängen können." --- Miquel van Smoorenburg <miquels at cistron.nl>
- "Unix IST benutzerfreundlich... Es ist nur sehr wählerisch mit der Entscheidung, wer seine Freunde sind." --- Tollef Fog Heen <tollef at add.no>

Auf Wikipedia gibt es den Artikel "[Unix-Philosophie](#)", der interessante Zitate enthält.

Kapitel 1

GNU/Linux-Lehrstunde

Ich denke, ein Computersystem zu erlernen ist wie das Erlernen einer Fremdsprache. Obwohl Anleitungen und Dokumentation hilfreich sind, müssen Sie es selbst einüben. Um Ihnen einen sanften Start zu verschaffen, werde ich hier einige grundsätzliche Dinge ausführen.

Das kraftvolle Design von [Debian GNU/Linux](#) stammt von dem [Unix](#)-Betriebssystem, was einem [Multiuser](#)- und [Multitasking](#)-Betriebssystem ist. Sie müssen lernen, die Vorteile aus der Kraft dieser Funktionalitäten und den Ähnlichkeiten zwischen Unix und GNU/Linux zu ziehen.

Scheuen Sie sich nicht, Unix-orientierte Texte zu lesen und stützen Sie sich nicht ausschließlich auf GNU/Linux-orientierte Texte, da Sie dann viele nützliche Informationen verpassen würden.

Anmerkung

Falls Sie bereits für eine Weile irgendein [Unix-ähnliches](#) System mit Befehlszeilenwerkzeugen genutzt haben, wissen Sie möglicherweise bereits alles, was ich hier beschreibe. Bitte nutzen Sie dies zum Realitäts-Check und zur Auffrischung.

1.1 Grundlagen für die Konsole

1.1.1 Die Shell-Eingabeaufforderung (Shell-Prompt)

Nach dem Starten des Systems sehen Sie einen textbasierten Anmeldebildschirm (außer Sie haben das [X-Window-System](#) mit einem Displaymanager wie `gdm3` installiert). Angenommen Ihr Rechnername ist `foo`, dann sieht Ihr Anmeldebildschirm wie folgt aus:

```
foo login:
```

Haben Sie eine Umgebung mit [grafischer Benutzerschnittstelle \(GUI\)](#) wie [GNOME](#) oder [KDE](#) installiert, können Sie trotzdem einen solchen textbasierten Anmeldebildschirm bekommen, indem Sie Strg-Alt-F1 drücken; mit Strg-Alt-F7 kehren Sie wieder zur GUI-Umgebung zurück (mehr dazu unter Abschnitt [1.1.6](#)).

In dem Anmeldebildschirm geben Sie Ihren Benutzernamen ein, z.B. `penguin` und drücken die Enter-Taste, dann Ihr Passwort und nochmals Enter.

Anmerkung

Gemäß der Unix-Tradition muss bei Benutzernamen und Passwort auf Groß- und Kleinschreibung geachtet werden. Im Benutzernamen werden für gewöhnlich nur Kleinbuchstaben verwendet. Das erste Benutzerkonto wird normalerweise während der Installation angelegt. Weitere Benutzerkonten können durch `root` mit dem Befehl `adduser(8)` erstellt werden.

Das System startet mit einer Grußnachricht, die in `/etc/motd` (Message Of The Day, Meldung des Tages) gespeichert ist, und zeigt einen Befehls-Prompt an.

```
Debian GNU/Linux jessie/sid foo tty1
foo login: penguin
Password:
Last login: Mon Sep 23 19:36:44 JST 2013 on tty3
Linux snoopy 3.11-1-amd64 #1 SMP Debian 3.11.6-2 (2013-11-01) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
foo:~$
```

Jetzt sind Sie in der [Shell](#). Die Shell interpretiert die Befehle, die Sie eingeben.

1.1.2 Die Shell-Eingabeaufforderung unter X

Falls Sie das [X-Window-System](#) mit einem Displaymanager wie [GNOMEs](#) `gdm3` installiert haben (indem Sie während der Installation die Programmgruppe "Arbeitsplatzumgebung" (Desktop environment) ausgewählt haben), bekommen Sie nach dem Systemstart einen grafischen Anmeldebildschirm. Geben Sie zur Anmeldung Benutzername und Passwort des unprivilegierten Benutzerkontos ein. Verwenden Sie dabei die TAB-Taste, um zwischen Benutzername und Passwort zu wechseln, oder benutzen Sie dazu die Maus.

In der X-Oberfläche bekommen Sie eine Shell-Eingabeaufforderung, indem Sie einen [X-Terminal-Emulator](#) wie `gnome-terminal(1)`, `rxvt(1)` oder `xterm(1)` starten. In der GNOME-Arbeitsplatzumgebung erreichen Sie dies über "Anwendungen" → "Systemwerkzeuge" → "Terminal".

Lesen Sie auch weiter unten den Abschnitt [1.1.6](#).

Bei einigen anderen Arbeitsplatz-Umgebungen (wie `fluxbox`) gibt es möglicherweise keine offensichtliche Möglichkeit, ein Menü zu öffnen. Versuchen Sie in diesem Fall einfach einen Rechtsklick auf den Desktop-Hintergrund und hoffen, dass ein Menü erscheint.

1.1.3 Das root-Benutzerkonto

Das root-Benutzerkonto wird auch [Superuser](#) oder privilegierter Benutzer genannt. Mit diesem Benutzerkonto können Sie folgende Administrationsaufgaben erledigen:

- lesen, schreiben und löschen aller Dateien auf dem System unabhängig von deren Zugriffsrechten;
- setzen von Eigentümer und Zugriffsrechten jeglicher Dateien auf dem System;
- setzen des Passworts von jedem unprivilegierten Benutzer auf dem System;
- anmelden an jedem Benutzerkonto ohne dessen Passwort.

Diese uneingeschränkten Rechte des root-Benutzerkontos erfordern von Ihnen, dass Sie sich besonnen und verantwortungsvoll verhalten, wenn Sie es benutzen.



Warnung

Teilen Sie das root-Passwort niemals anderen mit.

Anmerkung

Bestimmte Dateizugriffsrechte einer Datei (inklusive Hardware-Geräte wie CD-ROM usw., die auf einem Debian-System nichts anderes als eine Datei sind) können dazu führen, dass Sie für nicht-root-Benutzer unbenutzbar sind. Obwohl die Verwendung des root-Benutzerkontos ein schneller Weg ist, solche Situationen zu testen, sollte die Lösung das korrekte Einstellen der Datei-Zugriffsrechte und/oder das Hinzufügen des Benutzers in entsprechende Gruppen sein (siehe dazu Abschnitt [1.2.3](#)).

1.1.4 Die root-shell-Eingabeaufforderung

Hier ein paar grundlegende Methoden, um mittels root-Passwort eine root-shell-Eingabeaufforderung zu bekommen:

- Geben Sie `root` im textbasierten Anmeldebildschirm ein.
- Innerhalb der GNOME-Umgebung klicken Sie auf "Anwendungen" → "Zubehör" → "Root Terminal".
- Tippen Sie in jeglicher Shell "`su - l`" ein.
 - Hierdurch werden NICHT die Umgebungseinstellungen des angemeldeten Benutzers beibehalten.
- Tippen Sie in jeglicher Benutzer-Shell-Eingabeaufforderung "`su`" ein.
 - Hierdurch werden teilweise die Umgebungseinstellungen des angemeldeten Benutzers beibehalten.

1.1.5 Systemadministrations-Werkzeuge mit grafischer Oberfläche (GUI)

Wenn sich über Ihr Arbeitsplatz-Menü keine GUI-Systemadministrations-Werkzeuge mit den benötigten Rechten starten lassen, können Sie diese von einer root-shell-Eingabeaufforderung aus starten, wie z.B. von `gnome-terminal(1)`, `rxvt(1)` oder `xterm(1)`. Lesen Sie dazu Abschnitt [1.1.4](#) und Abschnitt [7.8.5](#).

**Warnung**

Starten Sie niemals das X-Display/den Sitzungsmanager als root, indem Sie `root` im Anmeldebildschirm des Displaymanagers (wie z.B. `gdm3(1)`) eingeben.

**Warnung**

Lassen Sie niemals vertrauensunwürdige GUI-Programme mit Fernanbindung in einem X-Window laufen, wenn dort kritische Daten angezeigt werden, da diese Programme Ihren grafischen Bildschirminhalt abgreifen können.

1.1.6 Virtuelle Konsolen

In einem Standard-Debian-System sind sechs [VT100-ähnliche](#) textbasierte Konsolen verfügbar, zwischen denen Sie hin und her schalten können, und in denen die Befehls-Shell direkt auf dem Linux-Host ausgeführt werden kann. Solange Sie sich nicht in einer GUI-Umgebung befinden, können Sie mit der linken **Alt-Taste** und gleichzeitig einer der Funktionstasten **F1** — **F6** zwischen den virtuellen Konsolen umschalten. Jede textbasierte Konsole erlaubt eine eigenständige Anmeldung mit einem Benutzerkonto und bietet eine Mehrbenutzer-Umgebung. Diese Mehrbenutzer-Umgebung ist eine tolle Unix-Funktionalität und macht sehr leicht süchtig.

Wenn Sie sich in einer X-Window-Umgebung befinden, bekommen Sie mit **Strg-Alt-F1** (d.h. die linke **Strg-Taste**, die linke **Alt-Taste** und **F1** gleichzeitig) Zugang zur textbasierten Konsole 1. Sie können zum X-Window-System zurückkehren, das normalerweise auf der virtuellen Konsole 7 läuft, indem Sie **Alt-F7** drücken.

Alternativ können Sie auch von der Befehlszeile aus zu einer anderen virtuellen Konsole wechseln, z.B. zur Konsole 1:

```
# chvt 1
```

1.1.7 Wie Sie die Eingabeaufforderung wieder verlassen

Drücken Sie in der Eingabeaufforderung **Strg-D** (also die linke **Strg**-Taste und die **d**-Taste gleichzeitig), um die Shell-Aktivitäten zu beenden. Wenn Sie sich auf der textbasierten Konsole befinden, kehren Sie hiermit zum Anmeldebildschirm zurück. Obwohl bei diesen Steuerzeichen von "control D" mit großem D gesprochen wird, müssen Sie hier nicht die Umschalt-Taste drücken. Für **Strg-D** wird auch die Kurzform **^D** benutzt. Alternativ können Sie auch "exit" eingeben.

Wenn Sie sich in einem X-Terminal-Emulator⁽¹⁾ befinden, können Sie hiermit das **X-Terminal-Emulator**-Fenster schließen.

1.1.8 Wie Sie das System herunterfahren

Wie jedes andere moderne Betriebssystem, bei dem Dateioperationen das **Zwischenspeichern (Caching)** beinhalten, um die Performance zu erhöhen, erfordert auch das Debian-System eine Prozedur zum sauberen Herunterfahren, bevor gefahrlos abgeschaltet werden kann. Dies dient dazu, die Integrität der Daten aufrecht zu erhalten, indem alle Änderungen auf die Platte geschrieben werden. Falls Energiekontrolle per Software verfügbar ist, wird im Rahmen des Herunterfahrens die Spannung automatisch abgeschaltet. (Andernfalls müssen Sie den Ein-/Ausschaltknopf mehrere Sekunden gedrückt halten, nachdem das Herunterfahren abgeschlossen ist).

Sie können das System im normalen Mehrbenutzermodus auf der Befehlszeile herunterfahren mittels:

```
# shutdown -h now
```

Sie können das System im Einzelbenutzermodus auf der Befehlszeile herunterfahren mittels:

```
# poweroff -i -f
```

Alternativ können Sie **Strg-Alt-Entf** drücken (die linke **Strg**-Taste, die linke **Alt**-Taste und **Entf** gleichzeitig), um das System herunterzufahren, sofern **/etc/inittab** folgendes enthält: **"ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -h now"**. Details dazu finden Sie unter [inittab\(5\)](#).

Lesen Sie auch Abschnitt [6.9.6](#).

1.1.9 Eine Konsole wiederherstellen

Wenn der Bildschirm verrückt spielt, nachdem Sie lustige Dinge wie **"cat <irgendeine-binärdatei>"** gemacht haben, geben Sie **"reset"** an der Eingabeaufforderung ein. Sie können den Befehl möglicherweise nicht sehen, während Sie ihn eingeben. Mit **"clear"** können Sie bei Bedarf auch den Bildschirm leeren.

1.1.10 Zusätzliche Paketempfehlungen für Neulinge

Obwohl selbst die Minimalinstallation eines Debian-Systems ohne jegliche Arbeitsplatz-Umgebung die grundlegenden Unix-Funktionalitäten bietet, ist es eine gute Idee, mittels **apt-get(8)** einige Befehlszeilen- und curses-basierte Zeichen-Terminal-Pakete zusätzlich zu installieren, wie z.B. **mc** und **vim**; Anfänger können dazu für einen ersten Versuch folgendes verwenden:

```
# apt-get update
...
# apt-get install mc vim sudo
...
```

Falls Sie diese Pakete bereits installiert haben, werden keine neuen Pakete installiert.

Es könnte eine gute Idee sein, einiges an informativer Dokumentation zu lesen.

Sie sollten vielleicht einige dieser Pakete installieren, indem Sie folgendes eingeben:

```
# apt-get install paketname
```

Paket	Popcon	Größe	Beschreibung
mc	V:59, I:236	1482	Ein Textmodus-Dateimanager mit Vollbildschirm-Ansicht
sudo	V:563, I:806	4555	Ein Programm, das Benutzern eingeschränkte root-Privilegien einräumt
vim	V:106, I:398	3231	Unix Texteditor Vi IMproved, ein Texteditor für Programmierer (Standardversion)
vim-tiny	V:62, I:970	1553	Unix Texteditor Vi IMproved, ein Texteditor für Programmierer (Kompaktversion)
emacs-nox	V:4, I:17	18364	Emacs vom GNU-Projekt, der Lisp-basierte erweiterbare Texteditor
w3m	V:31, I:284	2289	Textmodus-WWW-Browser
gpm	V:11, I:17	530	Unix-ähnliches Kopieren-und-Einfügen auf der Textkonsole (Daemon)

Tabelle 1.1: Liste interessanter Textmodus-Programm-Pakete

Paket	Popcon	Größe	Beschreibung
doc-debian	I:854	166	Dokumentation des Debian-Projekts und andere Dokumente
debian-policy	I:36	4306	Debian Policy-Handbuch und zugehörige Dokumente
developers-reference	I:6	1917	Richtlinien und Informationen für Debian-Entwickler
maint-guide	I:4	987	Leitfaden für neue Debian-Betreuer
debian-history	I:1	4285	Vergangenheit des Debian-Projekts
debian-faq	I:849	817	Debian FAQ

Tabelle 1.2: Liste informativer Dokumentationspakete

1.1.11 Ein zusätzliches Benutzerkonto

Falls Sie Ihr Haupt-Benutzerkonto nicht für die folgenden Trainingsaktivitäten nutzen möchten, können Sie mit folgendem Befehl ein Trainings-Benutzerkonto erstellen, hier z.B. `fish`:

```
# adduser fish
```

Beantworten Sie alle Fragen.

Dadurch wird ein neues Konto namens `fish` erstellt. Nach Ihren Übungen können Sie dieses Benutzerkonto und das dazugehörige Heimatverzeichnis entfernen, indem Sie folgendes verwenden:

```
# deluser --remove-home fish
```

1.1.12 sudo-Konfiguration

Für ein typisches Einzelbenutzer-Arbeitsplatzsystem, wie z.B. einen Debian-Desktop auf einem Laptop, ist es gängig, wie folgt eine einfache Konfiguration für `sudo(8)` einzurichten, so dass der nicht-privilegierten Benutzer, z.B. `penguin`, lediglich mit seinem Benutzer-Passwort (aber ohne das root-Passwort) administrative Rechte bekommen kann:

```
# echo "penguin ALL=(ALL) ALL" >> /etc/sudoers
```

Alternativ dazu ist es auch verbreitet, dem nicht-privilegierten Benutzer, hier `penguin`, wie folgt administrative Rechte zu gewähren, ohne dass dieser irgendein Passwort eingeben muss:

```
# echo "penguin ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers
```

Dieser Trick sollte nur für einen Einzelbenutzer-Arbeitsplatz angewandt werden, den Sie selbst administrieren und auf dem Sie der einzige Benutzer sind.

**Warnung**

Richten Sie keine Benutzerkonten regulärer Benutzer auf Mehrbenutzersystemen derartig ein; dies wäre sehr schlecht für die Systemsicherheit.

**Achtung**

Das Passwort und Benutzerkonto von penguin im obigen Beispiel erfordert ebenso viel Schutz wie das root-Passwort und das root-Benutzerkonto.

**Achtung**

Administrative Rechte in diesem Zusammenhang gehören zu jemandem, der berechtigt ist, Systemadministrationsaufgaben auf dem Arbeitsplatzsystem durchzuführen. Gewähren Sie niemals einem Manager in der Verwaltungsabteilung Ihrer Firma oder Ihrem Chef derartige Privilegien, außer diese sind dazu autorisiert und befähigt.

Anmerkung

Um Leuten Zugriffsrechte auf limitierte Geräte und Dateien zu verschaffen, sollten Sie die Verwendung von **group** in Betracht ziehen, um eingeschränkte Rechte einzurichten, statt die root-Privilegien via sudo(8) dafür zu nutzen.

Anmerkung

Mit einer durchdachten und vorsichtigen Konfiguration kann sudo(8) auf einem System, das Sie sich mit mehreren Leuten teilen, limitierte administrative Rechte einräumen, ohne dass Sie anderen das root-Passwort mitteilen müssen. Dies kann Ihnen auf Systemen helfen, wo Sie sich die Verantwortung mit mehreren Administratoren teilen, so dass Sie sagen können, wer was gemacht hat. Auf der anderen Seite möchten Sie vielleicht nicht, dass irgendjemand sonst solche Privilegien hat.

1.1.13 Zeit zum Spielen

Jetzt sind Sie soweit, dass Sie mit dem Debian-System ohne Risiko herumspielen können, solange Sie das nicht-privilegierte Benutzerkonto verwenden.

Das liegt daran, dass sogar nach einer Standardinstallation Ihr Debian-System mit korrekten Dateiberechtigungen konfiguriert ist, die verhindern, dass nicht-privilegierte Benutzer das System beschädigen. Natürlich könnte es immer noch einige Lücken geben, die ausgenutzt werden können, aber diejenigen, die sich darüber Sorgen machen, sollten nicht dieses Kapitel lesen, sondern die [Anleitung zum Absichern von Debian \(Securing Debian Manual\)](#).

Wir lernen das Debian-System mit folgenden Aspekten eines [Unix-ähnlichen](#) Systems kennen:

- [Abschnitt 1.2](#)
 - [Abschnitt 1.3](#)
 - [Abschnitt 1.4](#)
 - [Abschnitt 1.5](#)
 - [Abschnitt 1.6](#)
-

1.2 Unix-ähnliches Dateisystem

In GNU/Linux und anderen [Unix-ähnlichen](#) Betriebssystemen sind [Dateien](#) in [Verzeichnis](#)-Strukturen organisiert. Alle Dateien und Verzeichnisse sind in einem großen Verzeichnisbaum unterhalb von `/` eingeordnet. Dies wird Baum genannt, weil das Dateisystem, wenn Sie es aufmalen, wie ein Baum aussieht, nur dass er auf dem Kopf steht.

Diese Dateien und Verzeichnisse können auf mehrere Geräte verteilt sein. `mount(8)` dient dazu, Dateisysteme, die auf anderen Geräten erkannt werden, in den großen Verzeichnisbaum einzubinden. Umgekehrt dient `umount(8)` dazu, es wieder daraus zu entfernen. Auf aktuellen Linux-Kernel kann `mount(8)` mit bestimmten Optionen Teile des Verzeichnisbaums zusätzlich an anderer Stelle anknüpfen oder die zusätzlichen Eigenschaften `shared`, `private`, `slave` oder `unbindable` verwenden. Unterstützte `mount`-Optionen für verschiedene Dateisystemtypen finden Sie in `/usr/share/doc/linux-doc-*/Documentation/filesystems/`.

Verzeichnisse auf Unix-Systemen werden bei einigen anderen Betriebssystemen auch **Ordner** genannt. Bitte beachten Sie ebenfalls, dass es auf einem Unix-System kein Konzept zur Bezeichnung von **Laufwerken** gibt, das vergleichbar mit z.B. `"A:"` wäre. Es gibt ein Dateisystem, und dort ist alles enthalten. Dies ist ein gewaltiger Unterschied verglichen mit Windows.

1.2.1 Unix-Dateigrundlagen

Hier einige Grundlagen zu Dateien unter Unix:

- Bei Dateinamen ist **Groß-/Kleinschreibung immer relevant**. Das heißt, `"MEINEDATEI"` und `"MeineDatei"` sind zwei unterschiedliche Dateien.
- Das **Wurzel-Verzeichnis (root)** bedeutet Wurzel des Dateisystems und wird einfach als `/` dargestellt. Verwechseln Sie dies nicht mit dem Heimatverzeichnis des root-Benutzers `/root`.
- Jedes Verzeichnis hat einen Namen, der jegliche Buchstaben oder Symbole **außer** `/` enthalten kann. Das Wurzelverzeichnis `root` ist eine Ausnahme; dessen Name ist `/` (ausgesprochen "slash" oder "das root-Verzeichnis") und es kann nicht umbenannt werden.
- Jede Datei oder jedes Verzeichnis ist über einen **voll qualifizierten Dateinamen**, **absoluten Dateinamen** oder **Pfad** gekennzeichnet, der die Abfolge der Verzeichnisse enthält, die zu dessen Erreichen passiert werden müssen. Die drei oben angegebenen Ausdrücke sind alle gleichbedeutend.
- Alle **voll qualifizierten Dateinamen** beginnen mit dem `/`-Verzeichnis, und zwischen zwei Verzeichnissen oder zwischen Verzeichnis und Datei steht ein weiteres `/`. Das erste `/` ist das Wurzelverzeichnis, und die anderen `/` sind jeweils Trenner zwischen den einzelnen Unterverzeichnissen, bis der letzte Eintrag erreicht ist, was der Name der eigentlichen Datei ist. Die Wörter, die hier verwendet werden, können verwirrend sein. Nehmen Sie den folgenden **voll qualifizierten Dateinamen** als Beispiel: `/usr/share/keytables/de.map.gz`. Allerdings spricht man von `de.map.gz` alleine als Dateiname.
- Das Wurzelverzeichnis enthält eine Anzahl von Unterverzeichnisse, wie `/etc/` und `/usr/`. Diese haben wiederum nochmals weitere Unterverzeichnisse, z. B. `/etc/init.d/` und `/usr/local/`. Das Ganze komplett betrachtet wird als **Verzeichnisbaum** bezeichnet. Sie können den absoluten Dateinamen als Route vom Fuss des Baums (`/`) zu dessen Ende einer Verzweigung (einer Datei) betrachten. Manchmal hören Sie auch, wie Leute vom Verzeichnisbaum reden, als wäre es ein **Familien-Stammbaum**, der alle direkten Nachkommen einer einzigen Figur (dem Wurzelverzeichnis `/`) einschließt: demzufolge haben die Unterverzeichnisse **Eltern**, und ein Pfad zeigt die vollständige Herkunft einer Datei. Es gibt auch relative Pfade, die irgendwo beginnen (nicht beim Wurzelverzeichnis). Sie sollten dabei in Erinnerung behalten, dass `./` immer das jeweils übergeordnete Verzeichnis (Eltern-Verzeichnis) bezeichnet. Diese Bezeichnungsweise gilt auch für andere Strukturen, die dem Verzeichnisbaum ähnlich sind, wie hierarchische Datenstrukturen.
- Es gibt keine spezielle Verzeichnispfad-Namenskomponente, die einem physikalischen Gerät, wie einer Festplatte, entspricht. Dies unterscheidet sich von [RT-11](#), [CP/M](#), [OpenVMS](#), [MS-DOS](#), [AmigaOS](#) und [Microsoft Windows](#), wo der Pfad einen Gerätenamen wie `"C:\` enthält. Allerdings existieren Verzeichniseinträge, die sich als Teil des normalen Dateisystems auf physikalische Geräte beziehen. Siehe dazu Abschnitt [1.2.2](#).

Anmerkung

Obwohl Sie nahezu alle Buchstaben oder Symbole in einem Dateinamen verwenden **können**, ist es in der Praxis eine schlechte Idee, dies zu tun. Es ist besser, alle Zeichen zu vermeiden, die auf der Befehlszeile oft eine spezielle Bedeutung haben, darunter Leerzeichen, Tabulatoren, Newlines, und andere spezielle Zeichen wie { } () [] ' ` " \ / > < | ; ! # & ^ * % @ \$. Wenn Sie innerhalb des Namens einzelne Wörter trennen möchten, sind Punkte, Bindestriche und Unterstriche hierfür eine gute Wahl. Sie können auch die Wörter jeweils am Anfang großschreiben, "SowieHier". Erfahrene Benutzer tendieren dazu, Leerzeichen in Dateinamen zu vermeiden.

Anmerkung

Das Wort "root" kann entweder "root-Benutzer" oder "root-Verzeichnis" (Wurzelverzeichnis) bedeuten. Am Zusammenhang bei der Verwendung sollten Sie erkennen können, was gemeint ist.

Anmerkung

Das Wort **Pfad** wird nicht nur wie oben für **voll qualifizierte Dateinamen** verwendet, sondern auch für den **Befehls-Suchpfad**. Die jeweilige Bedeutung erschließt sich für gewöhnlich aus dem Zusammenhang.

Detaillierte bewährte Methoden für die Dateihierarchie sind im Filesystem Hierarchy Standard beschrieben (in `"/usr/share/doc/debian-policy/chrootfiles"` und unter `hier(7)`). Sie sollten folgendes als Ausgangspunkt nutzen:

Verzeichnis	Verwendung des Verzeichnisses
/	das Wurzelverzeichnis (root-Verzeichnis)
/etc/	systemweite Konfigurationsdateien
/var/log/	System-Protokolldateien
/home/	die Heimatverzeichnisse aller nicht-privilegierten Benutzer

Tabelle 1.3: Auflistung der Verwendung wichtiger Verzeichnisse

1.2.2 Dateisystem-Internas

Gemäß der **Unix-Tradition** stellt das Debian GNU/Linux-System ein **Dateisystem** bereit, unterhalb dessen alle physikalischen Daten auf Festplatten und anderen Speichermedien liegen; alles für das Zusammenspiel mit Hardware-Geräten wie Konsolenbildschirmen und von fern zugreifenden seriellen Konsolen benötigte wird in vereinheitlichter Art und Weise unterhalb von `"/dev/"` abgebildet.

Alle Dateien, Verzeichnisse, benannten Pipes (named pipes; ein Verfahren, mit dem zwei Programme Daten austauschen können) oder physikalischen Geräte auf einem Debian GNU/Linux-System haben eine Datenstruktur, **Inode** genannt, die die mit ihr verbundenen Attribute beschreibt, wie der Benutzer, der deren Eigentümer (owner) ist, die Gruppe, der sie angehören, der Zeitpunkt des letzten Zugriffs usw. Die Idee, nahezu alles im Dateisystem darzustellen, war eine Unix-Innovation, und moderne Linux-Kernel haben diese Idee sogar noch weiterentwickelt. Jetzt kann man sogar Informationen über die Prozesse, die auf dem Computer laufen, im Dateisystem finden.

Die abstrakte und vereinheitlichte Abbildung von physikalischen Datensätzen und internen Prozessen ist sehr leistungsfähig, da sie es uns erlaubt, für die gleiche Funktion auf vielen total unterschiedlichen Geräten denselben Befehl zu verwenden. Es ist sogar möglich, die Art und Weise, wie der Kernel arbeitet, zu verändern, indem man Daten in spezielle Dateien schreibt, die mit laufenden Prozessen verbunden sind.

Tipp

Wenn Sie wissen möchten, wie der Verzeichisbaum und die Gerätedateien verknüpft sind, führen Sie `mount(8)` ohne Argumente aus.

1.2.3 Dateisystem-Berechtigungen

[Dateisystem-Berechtigungen](#) auf [Unix-ähnlichen](#) Systemen werden für drei Kategorien beteiligter Benutzer festgelegt:

- der **Benutzer** (user), der Eigentümer der Datei ist (**u**);
- andere Benutzer der **Gruppe** (group), zu der die Datei gehört (**g**);
- alle **anderen** (other) Benutzer (**o**), auch "world" (Welt) oder "everyone" (jeder) genannt.

Bei Dateien gibt es Berechtigung für die folgenden Aktionen:

- Die **Lese**-Berechtigung (read, **r**) erlaubt dem Rechteinhaber, den Inhalt der Datei zu betrachten.
- Die **Schreib**-Berechtigung (write, **w**) erlaubt dem Rechteinhaber, die Datei zu verändern.
- Die **Ausführungs**-Berechtigung (execute, **x**) erlaubt dem Rechteinhaber, die Datei als Befehl auszuführen.

Bei Verzeichnissen gibt es Berechtigung für die folgenden Aktionen:

- Die **Lese**-Berechtigung (read, **r**) erlaubt dem Rechteinhaber, den Inhalt des Verzeichnisses aufzulisten.
- Die **Schreib**-Berechtigung (write, **w**) erlaubt dem Rechteinhaber, Dateien zu dem Verzeichnis hinzuzufügen oder Dateien zu löschen.
- Die **Ausführungs**-Berechtigung (execute, **x**) erlaubt dem Rechteinhaber, auf Dateien in dem Verzeichnis zuzugreifen.

Hierbei bedeutet die **Ausführungs**-Berechtigung für ein Verzeichnis nicht nur, dass der Inhalt der Dateien in diesem Verzeichnis betrachtet werden kann, sondern auch deren Attribute, wie die Dateigröße und der Zeitpunkt der letzten Änderung.

`ls(1)` wird verwendet, um Informationen über die Zugriffsrechte (und mehr) von Dateien und Verzeichnissen anzuzeigen. Wenn es mit der Option `"-l"` aufgerufen wird, werden die folgenden Informationen in der angegebenen Reihenfolge angezeigt:

- **Typ der Datei** (erstes Zeichen);
- **Berechtigungen** der Datei (neun Zeichen, bestehend aus drei Zeichen jeweils für Benutzer, Gruppe und andere, in dieser Reihenfolge);
- **Anzahl der harten Links** auf die Datei;
- Name des **Benutzers**, der Eigentümer der Datei ist;
- Name der **Gruppe**, zu der die Datei gehört;
- **Größe** der Datei in Zeichen (Byte);
- **Datum und Zeit** der Datei (mtime);
- **Name** der Datei.

`chown(1)` wird von dem root-Benutzer verwendet, um den Eigentümer einer Datei zu ändern. `chgrp(1)` wird vom Eigentümer einer Datei oder von root benutzt, um die Gruppe zu ändern, zu der die Datei gehört. `chmod(1)` wird vom Eigentümer oder dem root-Benutzer verwendet, um die Zugriffsberechtigungen für Datei und Verzeichnis zu ändern. Die grundsätzliche Syntax, um die Datei `foo` zu bearbeiten, ist wie folgt:

```
# chown <neuereigentümer> foo
# chgrp <neuegruppe> foo
# chmod [ugoa][+ -=][rwxXst][, ...] foo
```

Sie können zum Beispiel mit folgenden Befehlen einen Verzeichnisbaum so manipulieren, dass `foo` sein Eigentümer wird und `bar` die Gruppe:

Zeichen	Bedeutung
-	normale Datei
d	Verzeichnis
l	symbolischer Link
c	Geräte-Node für zeichenorientierte Geräte
b	Geräte-Node für blockorientierte Geräte
p	benannte Pipe (named pipe)
s	Socket

Tabelle 1.4: Bedeutungen des ersten Zeichens der "ls -l"-Ausgabe

```
# cd /some/location/
# chown -R foo:bar .
# chmod -R ug+rwX,o=rX .
```

Es gibt noch drei weitere, spezielle Zugriffs-Bits:

- das **Setze Benutzer-ID**-Bit (**s**, oder **S** statt dem **x** des Benutzers);
- das **Setze Gruppen-ID**-Bit (**s**, oder **S** statt dem **x** der Gruppe);
- das **sticky (klebrig)**-Bit (**t**, oder **T** statt dem **x** der "anderen").

Hierbei enthält die Ausgabe von "ls -l" für diese Bits die jeweiligen **Großbuchstaben**, wenn die Ausführungs-Bits (x), die bei dieser Ausgabe versteckt sind, **nicht gesetzt** sind.

Das Setzen des **Setze Benutzer-ID**-Bits einer ausführbaren Datei erlaubt dem Benutzer, die ausführbare Datei unter der Benutzer-ID der Datei (zum Beispiel **root**) auszuführen. Ähnlich dazu erlaubt das Setzen des Bits **Setze Gruppen-ID** einer ausführbaren Datei dem Benutzer, die ausführbare Datei unter der Gruppen-ID der Datei (zum Beispiel **root**) auszuführen. Da diese Einstellungen Sicherheitsrisiken verursachen können, erfordert deren Aktivierung besondere Vorsicht.

Das Setzen des **Setze Gruppen-ID**-Bits eines Verzeichnisses aktiviert das **BSD-ähnliche** Dateierstellungs-Schema, bei dem alle in dem Verzeichnis erzeugten Dateien der gleichen **Gruppe** angehören wie das Verzeichnis selbst.

Das Setzen des **sticky (klebrig)**-Bits eines Verzeichnisses verhindert, dass eine Datei in dem Verzeichnis von einem Benutzer gelöscht wird, der nicht Eigentümer der Datei ist. Um den Inhalt einer Datei in einem für alle schreibbaren Verzeichnis wie **/tmp** oder in durch die Gruppe schreibbaren Verzeichnissen sicherzustellen, muss nicht nur die **Schreib**-Berechtigung für die Datei zurückgenommen werden, sondern auch das **sticky (klebrig)**-Bit für das Verzeichnis. Ansonsten könnte jeder Benutzer, der Schreibberechtigung in dem Verzeichnis hat, die Datei löschen und eine neue mit dem gleichen Namen (aber eventuell anderem Inhalt) erstellen.

Hier einige interessante Beispiele von Dateiberechtigungen:

```
$ ls -l /etc/passwd /etc/shadow /dev/ppp /usr/sbin/exim4
crw-----T 1 root root    108, 0 Oct 16 20:57 /dev/ppp
-rw-r--r-- 1 root root    2761 Aug 30 10:38 /etc/passwd
-rw-r----- 1 root shadow 1695 Aug 30 10:38 /etc/shadow
-rwsr-xr-x 1 root root   973824 Sep 23 20:04 /usr/sbin/exim4
$ ls -ld /tmp /var/tmp /usr/local /var/mail /usr/src
drwxrwxrwt 14 root root   20480 Oct 16 21:25 /tmp
drwxrwsr-x 10 root staff   4096 Sep 29 22:50 /usr/local
drwxr-xr-x 10 root root    4096 Oct 11 00:28 /usr/src
drwxrwsr-x  2 root mail    4096 Oct 15 21:40 /var/mail
drwxrwxrwt  3 root root    4096 Oct 16 21:20 /var/tmp
```

Es gibt eine alternative numerische Möglichkeit, um Dateiberechtigungen mit `chmod(1)` darzustellen. Dieser numerische Modus verwendet 3 oder 4 einstellige Oktalzahlen (Basis=8).

Ziffer	Bedeutung
1. Ziffer (optional)	Summe der Bits Setze Benutzer-ID (=4), Setze Gruppen-ID (=2) und sticky (klebrig) (=1)
2. Ziffer	Summer der Rechte Lesen (= 4), Schreiben (= 2) und Ausführen (= 1) für den Benutzer
3. Ziffer	identisch für die Gruppe
4. Ziffer	identisch für alle anderen

Tabelle 1.5: Der numerische Modus für Dateiberechtigungen in chmod(1)-Befehlen

Dies klingt kompliziert, aber es ist letztendlich ganz einfach. Wenn Sie die ersten Spalten (2-10) der "ls -l"-Ausgabe anschauen und sie in binärer Darstellung (Basis = 2) lesen ("-" ist "0" und "rwx" sind "1"), werden Sie die letzten 3 Ziffern des numerischen Modus' im Oktalformat (Basis = 8) erkennen.

Probieren Sie zum Beispiel folgendes:

```
$ touch foo bar
$ chmod u=rw,go=r foo
$ chmod 644 bar
$ ls -l foo bar
-rw-r--r-- 1 penguin penguin 0 Oct 16 21:39 bar
-rw-r--r-- 1 penguin penguin 0 Oct 16 21:35 foo
```

Tipp

Wenn Sie mit einem Shellskript auf Informationen von "ls -l" zugreifen müssen, sollten Sie geeignete Befehle wie test(1), stat(1) und readlink(1) verwenden. Auch in die Shell integrierte Dinge wie "[" oder "test" können genutzt werden.

1.2.4 Steuerung der Berechtigungen für neu erzeugte Dateien: umask

Welche Berechtigungen einer neu erzeugten Datei oder einem neu erzeugten Verzeichnis zugewiesen werden, wird durch den in die Shell integrierten Befehl umask festgelegt/eingeschränkt. Lesen Sie dazu dash(1), bash(1) und builtins(7).

(Dateiberechtigungen) = (angeforderte Dateiberechtigungen) & ~(umask-Wert)

umask	erzeugte Dateirechte	erzeugte Verzeichnisrechte	Verwendung
0022	-rw-r--r--	-rwxr-xr-x	schreibbar nur durch den Benutzer
0002	-rw-rw-r--	-rwxrwxr-x	schreibbar durch die Gruppe

Tabelle 1.6: Beispiele für umask-Werte

Das Debian-System verwendet standardmäßig das UPG-Schema (user private group, eineprivate Gruppe für jeden Benutzer). Eine UPG wird erstellt, immer wenn ein Benutzer dem System hinzugefügt wird. Eine UPG hat den gleichen Namen wie der Benutzer, für den sie erstellt wurde, und dieser Benutzer ist das alleinige Mitglied in dieser Gruppe. Durch das UPG-Schema ist es sicher, die umask auf 0002 zu setzen, da jeder Benutzer seine eigene private Gruppe hat. (Bei einigen Unix-Varianten ist es gängig, alle normalen Benutzer so einzurichten, dass sie einer einzigen **users**-Gruppe angehören und es ist in solchen Fällen eine gute Idee, umask aus Sicherheitsgründen auf 0022 zu setzen.)

Tipp

Sie aktivieren das UPG-Schema, indem Sie "umask 002" zur Datei ~/.bashrc hinzufügen.

1.2.5 Berechtigungen für Gruppen von Benutzern (group)

Um zu erreichen, dass Gruppenberechtigungen für bestimmte Benutzer übernommen werden, muss der Benutzer ein Mitglied dieser Gruppe werden (mittels `sudo vigr` für `/etc/group` und `sudo vigr -s` für `/etc/gshadow`). Sie müssen sich abmelden und neu anmelden, damit die neue Gruppenkonfiguration übernommen wird (oder führen Sie `exec newgrp` aus).

Anmerkung

Alternativ können Sie Benutzer während des Authentifizierungsprozesses dynamisch zu Gruppen hinzufügen, indem Sie die Zeile `auth optional pam_group.so` zu `/etc/pam.d/common-auth` hinzufügen und `/etc/security/group.conf` einrichten. Lesen Sie dazu Kapitel 4.)

Hardware-Geräte sind nur eine andere Art von Datei auf dem Debian-System. Wenn Sie Probleme haben, von einem Benutzerkonto aus auf Geräte wie CD-ROM-Laufwerke oder USB-Memory-Sticks zuzugreifen, sollten Sie den Benutzer zu einem Mitglied der entsprechenden Gruppe machen.

Einige erwähnenswerte systemweite Gruppen erlauben ihren Mitglieder, ohne `root`-Privilegien auf die jeweiligen Dateien und Geräte zuzugreifen:

Gruppe	Beschreibung der Dateien und Geräte, auf die zugegriffen werden kann
<code>dialout</code>	voller und direkter Zugriff auf serielle Ports (<code>/dev/ttyS[0-3]</code>)
<code>dip</code>	eingeschränkter Zugriff auf serielle Ports für Dialup IP -(Einwahl-)Verbindungen zu vertrauenswürdigen Gegenstellen
<code>cdrom</code>	CD-ROM- und DVD+/-RW-Laufwerke
<code>audio</code>	Audiogeräte (z.B. Soundkarten)
<code>video</code>	Videogeräte (z.B. Grafikkarten)
<code>scanner</code>	Scanner
<code>adm</code>	Systemüberwachungs-Protokolle
<code>staff</code>	einige Verzeichnisse für untergeordnete administrative Tätigkeiten: <code>/usr/local</code> , <code>/home</code>

Tabelle 1.7: Liste erwähnenswerter systemweiter Gruppen

Tipp

Sie müssen der Gruppe `dialout` angehören, um ein Modem umzukonfigurieren, sich irgendwo einzuwählen usw. Wenn jedoch `root` vordefinierte Konfigurationsdateien für vertrauenswürdige Gegenstellen in `/etc/ppp/peers/` anlegt, müssen Sie lediglich Mitglied der Gruppe `dip` sein, um mit den Befehlen `pppd(8)`, `pon(1)` und `poff(1)` **Dialup IP**-(Einwahl-)Verbindungen herzustellen.

Einige erwähnenswerte, vom System angebotene Gruppen erlauben Ihren Mitgliedern, besondere Befehle ohne `root`-Rechte auszuführen:

Gruppe	Befehle, die ausgeführt werden können
<code>sudo</code>	<code>sudo</code> ohne sein Passwort ausführen
<code>lpadmin</code>	Befehle ausführen, um Drucker zur Druckerdatenbank hinzuzufügen, zu modifizieren und zu entfernen

Tabelle 1.8: Liste erwähnenswerter, vom System angebotener Gruppen zur Ausführung besonderer Befehle

Eine vollständige Liste der vom System angebotenen Benutzer und Gruppen finden Sie in der letzten Version des Dokuments "Users and Groups" in `/usr/share/doc/base-passwd/users-and-groups.html` aus dem Paket `base-passwd`. Lesen Sie `passwd(5)`, `group(5)`, `shadow(5)`, `newgrp(1)`, `vipw(8)`, `vigr(8)` und `pam_group(8)` für Informationen über Befehle zur Verwaltung des Benutzer- und Gruppensystems.

1.2.6 Zeitstempel

Es gibt drei Arten von Zeitstempeln für eine GNU/Linux-Datei:

Art	Bedeutung (historische Unix-Definition)
mtime	Zeitpunkt der letzten Änderung an der Datei (<code>ls -l</code>)
ctime	Zeitpunkt der letzten Statusänderung der Datei (<code>ls -lc</code>)
atime	Zeitpunkt des letzten Zugriffs auf die Datei (<code>ls -lu</code>)

Tabelle 1.9: Liste der Arten von Zeitstempeln

Anmerkung

ctime ist nicht der Zeitpunkt der Dateierzeugung.

Anmerkung

Der wirkliche Wert von **atime** auf einem GNU/Linux-System könnte sich von dem mit der historischen Unix-Definition konformen Variante unterscheiden.

- Das Überschreiben einer Datei ändert die **mtime**-, **ctime**- und **atime**-Attribute der Datei.
- Das Ändern von Eigentümer oder Zugriffsrechten ändert die **ctime**- und **atime**-Attribute der Datei.
- Das Lesen einer Datei ändert das **atime**-Attribut der Datei auf einem historischen Unix-System.
- Das Lesen einer Datei ändert das **atime**-Attribut der Datei auf einem GNU/Linux-System, wenn das Dateisystem mit dem Argument `"strictatime"` eingebunden wurde.
- Das erstmalige Lesen einer Datei oder das Lesen nach einem Tag ändert das **atime**-Attribut der Datei auf einem GNU/Linux-System, wenn das Dateisystem mit dem Argument `"relatime"` eingebunden wurde (dies ist das Standardverhalten seit Linux 2.6.30).
- Das Lesen einer Datei ändert nicht das **atime**-Attribut der Datei auf einem GNU/Linux-System, wenn das Dateisystem mit dem Argument `"noatime"` eingebunden wurde.

Anmerkung

Die mount-Argument `"noatime"` und `"relatime"` wurden eingeführt, um in normalen Anwendungssituationen die Leistungsfähigkeit der Systeme beim Lesen vom Dateisystem zu verbessern. Der einfache Vorgang zum Lesen einer Datei beinhaltet bei aktivierter `"strictatime"`-Option den zeitaufwendigen Schreibvorgang zum Aktualisieren des **atime**-Attributs. Jedoch wird das **atime**-Attribut selten benutzt, außer bei der `mbox(5)`-Datei. Lesen Sie dazu `mount(8)`.

Verwenden Sie den Befehl `touch(1)`, um den Zeitstempel vorhandener Dateien zu ändern.

Bei Zeitstempeln unterscheidet sich die Ausgabe des `ls`-Befehls, wenn ein nicht-englisches Gebietsschema (Locale) wie `"de_DE.UTF-8"` eingestellt ist, von der Ausgabe bei der alten Locale `"C"`.

```
$ LANG=de_DE.UTF-8 ls -l foo
-rw-rw-r-- 1 penguin penguin 0 16. Okt 21:35 foo
$ LANG=C ls -l foo
-rw-rw-r-- 1 penguin penguin 0 Oct 16 21:35 foo
```

Tipp

Lesen Sie Abschnitt [9.2.5](#), um die Ausgabe von `"ls -l"` anzupassen.

1.2.7 Links

Es gibt zwei Methoden, um die Datei "foo" mit einem anderen Dateinamen, z.B. "bar" zu verbinden:

- [Harter Link](#)
 - doppelter Name für eine existierende Datei
 - "ln foo bar"
- [symbolischer Link \(auch Symlink genannt\)](#)
 - eine spezielle Datei, die über deren Namen auf eine andere Datei verweist
 - "ln -s foo bar"

Schauen Sie sich folgendes Beispiel bezüglich Änderungen bei der Linkanzahl und den feinen Unterschieden im Resultat des rm-Befehls an:

```
$ umask 002
$ echo "Original-Text" > foo
$ ls -li foo
1449840 -rw-rw-r-- 1 penguin penguin 17 Oct 16 21:42 foo
$ ln foo bar      # harter Link
$ ln -s foo baz   # Symlink
$ ls -li foo bar baz
1449840 -rw-rw-r-- 2 penguin penguin 17 Oct 16 21:42 bar
1450180 lrwxrwxrwx 1 penguin penguin  3 Oct 16 21:47 baz -> foo
1449840 -rw-rw-r-- 2 penguin penguin 17 Oct 16 21:42 foo
$ rm foo
$ echo "Neuer Text" > foo
$ ls -li foo bar baz
1449840 -rw-rw-r-- 1 penguin penguin 17 Oct 16 21:42 bar
1450180 lrwxrwxrwx 1 penguin penguin  3 Oct 16 21:47 baz -> foo
1450183 -rw-rw-r-- 1 penguin penguin 12 Oct 16 21:48 foo
$ cat bar
Original-Text
$ cat baz
Neuer Text
```

Der harte Link kann innerhalb des gleichen Dateisystems erstellt werden und teilt sich die gleiche Inode-Nummer mit der Originaldatei, welche Sie mittels der Option "-i" des ls(1)-Befehls herausfinden können.

Der Symlink hat immer nominal die Dateiberechtigungen "rwxrwxrwx", wie in dem obigen Beispiel zu sehen, wobei die letztlich wirksamen Zugriffsrechte durch die Berechtigungen der Datei, auf die verwiesen wird, bestimmt werden.



Achtung

Es ist grundsätzlich eine gute Idee, keine komplizierten symbolischen oder harten Link zu erstellen, außer Sie haben einen sehr guten Grund dafür. Es könnte Ihnen sonst Albträume bescheren, wenn die logische Kombination von symbolischen Links zu Schleifen im Dateisystem führt.

Anmerkung

Im allgemeinen sollten Sie symbolische Links gegenüber harten Links bevorzugen, außer Sie haben einen guten Grund, einen harten Link zu verwenden.

Das Verzeichnis "." verweist auf das Verzeichnis, in dem es erscheint, daher steht die Anzahl der Links in jedem neuen Verzeichnis auf 2. ".." verweist auf das jeweilige Elternverzeichnis, daher erhöht sich die Anzahl der Links für dieses Verzeichnis mit jedem Hinzufügen eines neuen Unterverzeichnisses.

Wenn Sie gerade von Windows zu Linux wechseln, wird Ihnen bald klar werden, wie gut das Verfahren zum Verknüpfen von Dateinamen bei Unix aufgebaut ist, verglichen mit dem Windows-Equivalent der "Shortcuts". Weil es im Dateisystem implementiert ist, bemerken Anwendungen keinen Unterschied zwischen der verknüpften Datei und dem Original. Im Fall eines harten Links gibt es auch tatsächlich keinen Unterschied.

1.2.8 Benannte Pipes (FIFOs)

Eine [benannte Pipe](#) (named pipe) ist eine Datei, die sich wie ein Rohr verhält: wenn Sie etwas in die Datei hineingeben, kommt es am anderen Ende wieder heraus. Daher wird sie auch FIFO genannt, vom englischen First-In-First-Out: das erste, was Sie in das Rohr hineingeben, kommt als erstes am anderen Ende wieder heraus.

Wenn Sie in eine benannte Pipe schreiben, wird der Prozess, der in die Pipe schreibt, nicht eher beendet, als bis die zu schreibenden Informationen von der Pipe gelesen wurden. Wenn Sie aus einer benannten Pipe lesen, wartet der lesende Prozess, bis es nichts mehr zu lesen gibt, bevor er beendet wird. Die Größe der Pipe ist immer Null --- sie speichert keine Informationen, sie verbindet lediglich zwei Prozesse, wie dies die Shell mittels "|" tut. Da diese Pipe allerdings einen Namen hat, müssen die beiden Prozesse nicht auf der gleichen Befehlszeile und nicht einmal unter dem gleichen Benutzerkonto laufen. Pipes waren eine sehr einflußreiche Innovation von Unix.

Probieren Sie zum Beispiel folgendes:

```
$ cd; mkfifo mypipe
$ echo "hallo" >mypipe & # in den Hintergrund schieben
[1] 8022
$ ls -l mypipe
prw-rw-r-- 1 penguin penguin 0 Oct 16 21:49 mypipe
$ cat mypipe
hallo
[1]+  Done                  echo "hallo" >mypipe
$ ls mypipe
mypipe
$ rm mypipe
```

1.2.9 Sockets

Sockets werden von allen Internet-Übertragungen, Datenbanken und dem Betriebssystem selbst umfassend genutzt. Sie sind den benannten Pipes (FIFO) ähnlich und erlauben Prozessen, Informationen sogar zwischen verschiedenen Computern auszutauschen. Für die Nutzung von Sockets müssen diese Prozesse nicht zur gleichen Zeit laufen und auch nicht Kindprozesse desselben Urprozesses sein. Ein Socket ist der Endpunkt für die [Interprozesskommunikation \(IPC\)](#). Der Austausch von Informationen kann über das Netzwerk zwischen verschiedenen Rechnern stattfinden. Die beiden meist verwendeten sind der [Internet-Socket](#) und der [Unix-Domain-Socket](#).

Tipp

"netstat -an" bietet eine sehr nützliche Übersicht aller auf einem entsprechenden System geöffneten Sockets.

1.2.10 Gerätedateien

[Gerätedateien](#) verweisen auf physikalische oder virtuelle Geräte auf Ihrem System, wie Ihre Festplatte, Grafikkarte, Bildschirm oder Tastatur. Ein Beispiel für ein virtuelles Gerät ist die Konsole, repräsentiert durch `/dev/console`.

Es gibt zwei Arten von Gerätedateien:

- **Zeichenorientierte Geräte**

- Zugriff immer nur auf ein Zeichen gleichzeitig
-

- 1 Zeichen = 1 Byte
- z.B. Tastatur, serieller Port, ...
- **Blockorientierte Geräte**
 - Zugriff in größeren Einheiten, Blocks genannt
 - 1 Block > 1 Byte
 - z.B. Festplatten, ...

Sie können Gerätedateien lesen und schreiben, allerdings enthalten diese möglicherweise Binärdaten, die für Menschen nur unverständliches Kauderwelsch darstellen. Daten direkt in diese Dateien zu schreiben, ist manchmal zur Fehlersuche bei Verbindungsproblemen nützlich. Zum Beispiel können Sie eine Textdatei zur Drucker-Gerätedatei `"/dev/lp0"` schicken oder Modembefehle an den entsprechenden seriellen Port `"/dev/ttyS0"`. Aber wenn hierbei nicht mit Vorsicht gearbeitet wird, kann dies eine große Katastrophe verursachen. Seien Sie also auf der Hut.

Anmerkung

Für den normalen Zugriff auf den Drucker verwenden Sie `lp(1)`.

Die Geräte-Node-Nummer wird angezeigt, indem Sie `ls(1)` wie folgt ausführen:

```
$ ls -l /dev/sda /dev/sr0 /dev/ttyS0 /dev/zero
brw-rw---T 1 root disk      8,  0 Oct 16 20:57 /dev/sda
brw-rw---T+ 1 root cdrom    11,  0 Oct 16 21:53 /dev/sr0
crw-rw---T 1 root dialout   4, 64 Oct 16 20:57 /dev/ttyS0
crw-rw-rw- 1 root root      1,  5 Oct 16 20:57 /dev/zero
```

- `"/dev/sda"` hat die major-Gerätenummer 8 und die minor-Gerätenummer 0. Es ist lesbar/schreibbar für Benutzer, die der Gruppe `disk` angehören.
- `"/dev/sr0"` hat die major-Gerätenummer 11 und die minor-Gerätenummer 0. Es ist lesbar/schreibbar für Benutzer, die der Gruppe `cdrom` angehören.
- `"/dev/ttyS0"` hat die major-Gerätenummer 4 und die minor-Gerätenummer 64. Es ist lesbar/schreibbar für Benutzer, die der Gruppe `dialout` angehören.
- `"/dev/zero"` hat die major-Gerätenummer 1 und die minor-Gerätenummer 5. Es ist lesbar/schreibbar für jeden.

Auf modernen Linux-Systemen wird das Dateisystem unterhalb von `"/dev/"` automatisch durch den `udev(7)`-Mechanismus bestückt.

1.2.11 Spezielle Gerätedateien

Es gibt einige spezielle Gerätedateien:

Diese werden oft in Verbindung mit der Shell-Umleitung genutzt (lesen Sie Abschnitt [1.5.8](#)).

1.2.12 `procfs` und `sysfs`

`procfs` und `sysfs`, eingebunden unter `"/proc"` und `"/sys"`, sind Pseudo-Dateisysteme und bringen interne Datenstrukturen des Kernel in den Userspace. Mit anderen Worten: diese Einträge sind virtuell, sie fungieren als komfortables Fenster zu den Operationen des Betriebssystems.

Das Verzeichnis `"/proc"` enthält (neben anderen Dingen) ein Unterverzeichnis für jeden Prozess, der auf dem System läuft, welches nach der Prozess-ID (PID) benannt ist. Systemwerkzeuge, die auf Prozessinformationen zugreifen, wie `ps(1)`, bekommen ihre Informationen aus dieser Verzeichnisstruktur.

Geräte-datei	Aktion	Beschreibung der Antwort
/dev/null	lesen	gibt das "end-of-file"-Zeichen (EOF) zurück
/dev/null	schreiben	gibt nichts zurück (ein bodenloses Loch für Datenmüll)
/dev/zero	lesen	gibt das "\0"-Zeichen (NUL) zurück (das ist nicht dasselbe wie die Zahl Null in ASCII)
/dev/random	lesen	gibt zufällige Zeichen von einem realen Zufallszahlengenerator zurück, der echte Entropie liefert (langsam)
/dev/urandom	lesen	gibt zufällige Zeichen von einem kryptografisch sicheren Pseudo-Zufallszahlengenerator zurück
/dev/full	schreiben	gibt den "disk-full"-Fehler (ENOSPC, Platte voll) zurück

Tabelle 1.10: Liste spezieller Geräte-dateien

Die Verzeichnisse unterhalb von `/proc/sys/` enthalten Schnittstellen, um bestimmte Kernel-Parameter zur Laufzeit zu ändern. (Sie können dies möglicherweise auch mit dem spezialisierten Befehl `sysctl(8)` oder seiner Preload-/Konfigurationsdatei `/etc/sysctl.conf` erreichen.)

Die Leute geraten regelmäßig in Panik, wenn Sie eine bestimmte Datei bemerken - `/proc/kcore` - die wirklich riesig ist. Dies ist (mehr oder weniger) eine Kopie des Inhalts vom Arbeitsspeicher Ihres Rechners. Sie wird für Fehlersuche im Kernel verwendet. Es ist eine virtuelle Datei, die auf den Arbeitsspeicher verweist, sorgen Sie sich daher nicht über ihre Größe.

Das Verzeichnis unterhalb von `/sys` enthält vom Kernel exportierte Datenstrukturen, deren Attribute sowie die Anbindungen zwischen ihnen. Es enthält auch Schnittstellen, um bestimmte Kernel-Parameter zur Laufzeit zu ändern.

Lesen Sie `proc.txt(.gz)`, `sysfs.txt(.gz)` sowie weitere dazugehörige Dokumente aus der Linux-Kernel-Dokumentation (`/usr/share/doc/linux-doc-*/Documentation/filesystems/*`) aus dem Paket `linux-doc-*`.

1.2.13 tmpfs

[tmpfs](#) ist ein temporäres Dateisystem, das alle Dateien im [virtuellen Speicher](#) hält. Die Daten des tmpfs im [page cache](#) des Speichers können - falls nötig - in den [Swap-Bereich](#) auf der Festplatte ausgelagert werden.

Das Verzeichnis `/run` wird im frühen Stadium des Boot-Prozesses als tmpfs eingebunden. Dadurch ist es möglich, in dieses Verzeichnis zu schreiben, auch wenn `/` als read-only (nur lesen) eingebunden ist. Dies ist der neue Ablageort für die Speicherung kurzlebiger Statusdateien und ersetzt verschiedene Orte, die im [Filesystem Hierarchy Standard](#) Version 2.3 beschrieben sind:

- `/var/run` → `/run`
- `/var/lock` → `/run/lock`
- `/dev/shm` → `/run/shm`

Lesen Sie `tmpfs.txt(.gz)` in der Linux-Kernel-Dokumentation (`/usr/share/doc/linux-doc-2.6.*/Documentation`) aus dem Paket `linux-doc-2.6.*`.

1.3 Midnight Commander (MC)

[Midnight Commander \(MC\)](#) ist ein GNU "Schweizer Messer" für die Linux-Konsole und andere Terminal-Umgebungen. Er bringt Neulingen eine menügeführte Konsole, die viel einfacher zu erlernen ist wie die Standard-Unix-Befehle.

Sie müssen möglicherweise wie folgt das Midnight Commander-Paket mit Namen `mc` installieren:

```
$ sudo apt-get install mc
```

Verwenden Sie den `mc(1)`-Befehl, um das Debian-System zu erforschen. Dies ist der beste Weg, um zu lernen. Entdecken Sie einige interessante Orte lediglich durch Verwendung der Pfeiltasten und der Enter-Taste.

- `/etc` und dessen Unterverzeichnisse;
- `/var/log` und dessen Unterverzeichnisse;
- `/usr/share/doc` und dessen Unterverzeichnisse;
- `/sbin` und `/bin`.

1.3.1 Anpassen des MC

Um beim Schließen des MC das Arbeitsverzeichnis zu speichern und beim nächsten Start mit `cd` dorthin zu wechseln, empfehle ich, `~/ .bashrc` zu modifizieren, um ein Skript dort zu integrieren, das vom `mc`-Paket bereitgestellt wird:

```
. /usr/lib/mc/mc.sh
```

Den Grund finden Sie unter `mc(1)` (unter der Option `-P`). (Falls Sie nicht verstehen, worum genau es hier geht, können Sie dies später erledigen.)

1.3.2 Starten von MC

MC kann wie folgt gestartet werden:

```
$ mc
```

MC kümmert sich über sein Menü um alle Dateioperationen und erfordert dabei nur minimale Anstrengungen des Benutzers. Drücken Sie einfach `F1`, um die Hilfe-Seiten zu bekommen. Sie können mit dem MC herumspielen, indem Sie einfach die Pfeiltasten und Funktionstasten drücken.

Anmerkung

In einigen Konsolen, wie `gnome-terminal(1)`, könnten Tastendrucke der Funktionstasten vom Konsolenprogramm "gestohlen" werden. Sie können diese Funktionalität bei `gnome-terminal` über "Bearbeiten" → "Tastenkombinationen" deaktivieren.

Wenn Sie Probleme mit der Zeichenkodierung feststellen und Zeichenmüll angezeigt wird, kann das Hinzufügen von `-a` zur MC-Befehlszeile helfen, Probleme zu vermeiden.

Falls dies Ihre Anzeige Probleme mit MC nicht löst, lesen Sie Abschnitt [9.4.6](#).

1.3.3 Dateimanager in MC

Die Standardanzeige sind zwei Verzeichnisfenster, welche Dateilisten enthalten. Ein anderer nützlicher Modus ist, das rechte Fenster auf "Info" einzustellen, um dort Informationen zu Dateizugriffsrechten und anderem zu erhalten. Wenn der `gpm(8)`-Daemon läuft, kann man auch auf textbasierten Linux-Konsolen eine Maus verwenden. (Stellen Sie sicher, dass Sie die Umschalt-Taste drücken, um im MC das normale Ausschneiden-und-Einfügen-Verhalten zu bekommen.)

1.3.4 Befehlszeilentricks in MC

- `cd`-Befehl wechselt das Verzeichnis, das im gewählten Fenster angezeigt wird.
 - `Strg-Enter` oder `Alt-Enter` kopiert einen Dateinamen auf die Befehlszeile. Nutzen Sie dies mit `cp(1)` und `mv(1)` zusammen mit dem Editieren auf der Befehlszeile.
 - `Alt-Tab` zeigt Auswahlmöglichkeiten für die Auto-Vervollständigung von Dateinamen.
 - Die Start-Verzeichnisse für beide Fenster können MC als Argumente angegeben werden, zum Beispiel `mc /etc /root`.
 - `Esc + n`-Taste → `Fn` (d.h. `Esc + 1` → `F1`, usw.; `Esc + 0` → `F10`).
 - Das Drücken von `ESC` vor einer anderen Buchstabentaste hat den gleichen Effekt wie das Drücken von `Alt` und der anderen Taste zusammen, d. h. drücken Sie `ESC + c` für `Alt-C`. `ESC` wird Meta-Taste genannt und manchmal als "M-" dargestellt.
-

Taste	Tastaturbefehl
F1	Hilfe-Menü
F3	interner Dateibetrachter
F4	interner Texteditor
F9	Aufklapp-Menü aktivieren
F10	Midnight Commander beenden
Tab	zwischen den beiden Fenstern wechseln
Einfg oder Strg-T	Datei für eine mehrere Dateien betreffende Operation wie Kopieren markieren
Entf	Datei löschen (seien Sie vorsichtig --- stellen Sie MC auf "sicheres Löschen")
Pfeiltasten	selbsterklärend

Tabelle 1.11: Die Tastaturbefehle von MC

1.3.5 Der interne Texteditor in MC

Der interne Editor hat ein interessantes Ausschneiden-und-Einfügen-Schema: das Drücken von F3 setzt den Startpunkt einer zu markierenden Auswahl, ein weiteres F3 markiert das Ende der Auswahl und hebt die Auswahl hervor. Dann können Sie Ihren Cursor verschieben. Wenn Sie F6 drücken, wird der ausgewählte Bereich an die Cursor-Position verschoben. Wenn Sie F5 drücken, wird der ausgewählte Bereich kopiert und an der Cursor-Position eingefügt. F2 sichert die Datei. F10 beendet das Ganze. Die meisten Pfeiltasten funktionieren intuitiv.

Dieser Editor kann direkt mit einer Datei gestartet werden, indem einer der folgenden Befehle benutzt wird:

```
$ mc -e name_der_zu_editierenden_Datei
```

```
$ mcedit name_der_zu_editierenden_Datei
```

Dies ist kein Multi-Fenster-Editor, aber man kann mehrere Linux-Konsolen verwenden, um den gleichen Effekt zu erreichen. Um Inhalte zwischen verschiedenen Fenstern hin und her zu kopieren, verwenden Sie die Alt-F<n>-Tasten, um zwischen den virtuellen Konsolen zu wechseln, und "Datei → Datei einfügen" oder "Datei → Kopie in Datei", um Teile einer Datei in eine andere Datei zu kopieren.

Dieser interne Editor kann auf Wunsch durch jeden anderen externen Editor ersetzt werden.

Außerdem verwenden viele Programme die Umgebungsvariablen "\$EDITOR" oder "\$VISUAL", um festzulegen, welcher Editor genutzt wird. Wenn Ihnen vim(1) oder nano(1) nicht behagen, möchten Sie diese vielleicht auf "mcedit" setzen, indem Sie folgende Zeilen zu "~/.bashrc" hinzufügen:

```
export EDITOR=mcedit
export VISUAL=mcedit
```

Ich empfehle, diese wenn möglich auf "vim" zu setzen.

Wenn Sie vim(1) nicht mögen, können Sie für die meisten Systemwartungsaufgaben auch mcedit(1) benutzen.

1.3.6 Der interne Dateibetrachter in MC

MC ist ein sehr toller Dateibetrachter. Er ist ein großartiges Werkzeug, um nach Wörtern in Dokumenten zu suchen. Ich verwende ihn immer für Dateien im /usr/share/doc-Verzeichnis. Dies ist der schnellste Weg, um Massen von Linux-Informationen zu durchsuchen. Dieser Betrachter kann direkt gestartet werden, indem einer der folgenden Befehle benutzt wird:

```
$ mc -v pfad/zur/zu_betrachtenden_datei
```

```
$ mcview pfad/zur/zu_betrachtenden_datei
```

Dateityp	Reaktion auf die Enter-Taste
ausführbare Datei	Befehl ausführen
Handbuchseiten-Datei	Inhalt an Betrachter-Software weiterleiten
html-Datei	Inhalt an Webbrowser weiterleiten
"*.tar.gz"- und "*.deb"-Datei	den Inhalt anzeigen, als wäre es ein Unterverzeichnis

Tabelle 1.12: Die Reaktion auf die Enter-Taste in MC

1.3.7 Autostart-Funktionalitäten von MC

Drücken Sie Enter auf einer Datei und das zugehörige Programm wird den Inhalt der Datei verarbeiten (lesen Sie Abschnitt 9.3.11). Dies ist eine sehr praktische MC-Funktionalität.

Damit diese Betrachter- und virtuellen Dateifunktionalitäten korrekt funktionieren können, sollte bei Dateien, deren Inhalt betrachtet werden kann, nicht die Ausführungs-Berechtigung gesetzt sein. Ändern Sie deren Status ansonsten mit `chmod(1)` oder über das MC-Dateimenü.

1.3.8 Virtuelles FTP-Dateisystem von MC

MC kann verwendet werden, um via FTP über das Internet auf Dateien zuzugreifen. Rufen Sie über F9 das Menü auf, drücken Sie dann "p", um das virtuelle FTP-Dateisystem zu aktivieren. Geben Sie eine URL in der Form `"benutzername:passwort@rechner"` ein, wodurch ein in der Ferne liegendes Verzeichnis abgerufen wird, das wie ein lokales erscheint.

Versuchen Sie `"[deb.debian.org/debian/]"` als URL und durchsuchen Sie das Debian-Archiv.

1.4 Die grundlegende Unix-ähnliche Arbeitsumgebung

Obwohl MC es Ihnen erlaubt, fast alles zu tun, ist es sehr wichtig für Sie, zu lernen, wie die Befehlszeilenwerkzeuge am Shell-Prompt aufgerufen werden, und mit der Unix-ähnlichen Arbeitsumgebung vertraut zu werden.

1.4.1 Die Login-Shell

Sie können Ihre Login-Shell mit `chsh(1)` auswählen.

Paket	Popcon	Größe	POSIX-Shell	Beschreibung
bash	V:791, I:999	6469	Ja	Bash : die GNU Bourne Again SHell (De-Facto-Standard)
tcsh	V:9, I:29	1316	Nein	TENEX C Shell : eine erweiterte Version der Berkeley csh
dash	V:907, I:992	221	Ja	Debian Almquist Shell , gut für Shell-Scripting
zsh	V:37, I:73	2442	Ja	Z Shell : die Standard-Shell mit vielen Erweiterungen
mksh	V:4, I:12	1469	Ja	Eine Version der Korn-Shell
csh	V:2, I:8	343	Nein	OpenBSD C-Shell , eine Version der Berkeley csh
sash	V:0, I:6	1054	Ja	Stand-alone Shell mit integrierten Befehlen (nicht gedacht für Standard <code>"/bin/sh"</code>)
ksh	V:3, I:16	3284	Ja	die echte AT&T-Version der Korn Shell
rc	V:0, I:2	169	Nein	Implementierung der AT&T Plan 9 rc-Shell
posh	V:0, I:0	190	Ja	Policy-compliant (grundsatz-konforme) Ordinary SHell (pdksh-Abkömmling)

Tabelle 1.13: Liste von Shell-Programmen

Tipp

Obwohl sich POSIX-konforme Shells die grundsätzliche Syntax teilen, können sie sich beim Verhalten für Dinge wie Shell-Variablen oder Glob-Ersetzungen unterscheiden. Bitte konsultieren Sie die jeweilige Dokumentation bezüglich weiterer Details.

In dieser Lehrstunde ist mit interaktiver Shell immer `bash` gemeint.

1.4.2 Anpassen der bash

Sie können das Verhalten von `bash(1)` mittels `~/ .bashrc` anpassen.

Probieren Sie zum Beispiel folgendes:

```
# Aktivieren der autom. bash-Vervollständigung
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

# Verzeichniswechsel über das Schließen von MC hinaus
. /usr/lib/mc/mc.sh

# CDPATH auf einen passenden Wert setzen
CDPATH=.:usr/share/doc:~/Desktop:~
export CDPATH

PATH="${PATH+$PATH:}/usr/sbin:/sbin"
# PATH setzen, so dass ein privates bin des Benutzers (falls vorhanden) enthalten ist
if [ -d ~/bin ] ; then
  PATH="$PATH:~/bin"
fi
export PATH

EDITOR=vim
export EDITOR
```

Tipp

Es gibt weitere Tipps zur Anpassung der bash, zum Beispiel Abschnitt [9.2.7](#) in Kapitel [9](#).

Tipp

Das `bash-completion`-Paket enthält eine konfigurierbare Auto-Vervollständigung für bash.

1.4.3 Spezielle Tastendrücke

In einer [Unix-ähnlichen](#) Umgebung gibt es ein paar Tastendrücke, die spezielle Bedeutungen haben. Bitte beachten Sie, dass auf einer normalen textbasierten Linux-Konsole nur die linke `Strg`- und `Alt`-Taste wie erwartet funktionieren. Hier ein paar erwähnenswerte Tastendrücke, die Sie sich merken sollten:

Tipp

Die Terminal-Funktionalität von `Strg-S` kann mittels `stty(1)` deaktiviert werden.

Taste	Beschreibung des Tastaturbefehls
Strg-U	alle Zeichen vor dem Cursor löschen
Strg-H	ein Zeichen vor dem Cursor löschen
Strg-D	Eingabe beenden (Shell beenden, wenn Sie eine benutzen)
Strg-C	einen laufenden Prozess beenden
Strg-Z	Programm vorübergehend stoppen, indem es in den Hintergrund geschoben wird
Strg-S	Ausgabe auf den Bildschirm anhalten
Strg-Q	Ausgabe auf den Bildschirm wieder starten
Strg-Alt-Entf	Neu starten/anhalten des Systems, siehe dazu inittab(5)
Linke-Alt-Taste (optional Windows-Taste)	Meta-Taste für Emacs und ähnliche Benutzerschnittstellen
Pfeiltaste aufwärts	Die zuletzt ausgeführten Befehlen der <code>bash</code> anzeigen
Strg-R	Inkrementelle Suche in den zuletzt ausgeführten Befehlen der <code>bash</code>
Tab	Auto-Vervollständigung des Dateinamens auf der Befehlszeile in der <code>bash</code>
Strg-V Tab	Eingabe von Tab ohne Auto-Vervollständigung auf der Befehlszeile in der <code>bash</code>

Tabelle 1.14: Liste der Tastaturbefehle für `bash`

1.4.4 Mausoperationen im Unix-Stil

Die Mausoperationen im Unix-Stil basieren auf dem 3-Knopf-Maus-System.

Aktion	Reaktion
Links-Klick und Ziehen der Maus	Auswählen und in die Zwischenablage kopieren
Links-Klick	Start der Auswahl festlegen
Rechts-Klick	Ende der Auswahl festlegen und Auswahl in die Zwischenablage kopieren
Mittel-Klick	Inhalt der Zwischenablage an der Cursor-Position einfügen

Tabelle 1.15: Liste der Mausoperationen im Unix-Stil

Das Mittelrad auf einer modernen Radmaus wird als mittlerer Mausknopf betrachtet und kann für den Mittel-Klick verwendet werden. Das gleichzeitige Drücken des rechten und linken Maus-Knopfes wirkt in einem 2-Knopf-Maus-System wie ein Mittel-Klick. Um eine Maus in textbasierten Linux-Konsolen verwenden zu können, müssen Sie `gpm(8)` als Daemon laufen haben.

1.4.5 Der Pager

Der `less(1)`-Befehl ist ein erweiterter Pager (Dateiinhalte-Browser). Er liest die Datei, die per Befehlsargument oder Standard-eingabe angegeben wird. Drücken Sie "h", wenn Sie beim Anzeigen von Dateiinhalten mit `less` Hilfe benötigen. Er hat viel mehr Fähigkeiten als `more(1)` und kann sogar noch weiter aufgebohrt werden, indem "`eval $(lesspipe)`" oder "`eval $(lessfile)`" im Shell-Startskript ausgeführt wird. Weiteres dazu finden Sie in `/usr/share/doc/less/LESSOPEN`. Die Option "-R" erlaubt die Ausgabe von Rohformat-Zeichen und aktiviert ANSI-Color-Escape-Sequenzen. Lesen Sie dazu `less(1)`.

1.4.6 Der Texteditor

Sie sollten lernen, eine der Varianten von [Vim](#) oder [Emacs](#) zu nutzen, die auf Unix-ähnlichen Systemen beliebt sind.

Ich denke, sich an Vim-Befehle zu gewöhnen, ist das richtige, da der Vi-Editor in der Linux-/Unix-Welt immer vorhanden ist. (Eigentlich sind der originale `vi` oder auch der neue `nvi` Programme, die Sie überall finden. Ich habe stattdessen für Neulinge Vim ausgewählt, da es über F1 eine Hilfe anbietet, den anderen sehr ähnlich und gleichzeitig noch leistungsfähiger ist.)

Wenn Sie stattdessen entweder [Emacs](#) oder [XEmacs](#) als Ihren Editor auswählen, ist dies in der Tat eine andere gute Wahl, speziell zum Programmieren. Emacs enthält ebenso eine Fülle an weiteren Funktionen, dazu gehört die Arbeit als Newsreader, Verzeichnis-Editor, Mail-Programm usw. Wenn Sie ihn zum Programmieren oder Editieren von Shellskripten verwenden, erkennt er intelligent das Format, an dem Sie arbeiten, und versucht Sie dabei zu unterstützen. Einige Leute behaupten, dass das einzige Programm, das Sie unter Linux benötigen, Emacs sei. Jetzt zehn Minuten Emacs lernen kann später Stunden an Zeit sparen. Es wird dringend empfohlen, beim Erlernen von Emacs das GNU Emacs-Handbuch als Referenz zur Hand zu haben.

All diese Programme enthalten gewöhnlich Unterrichtseinheiten für Sie, damit Sie sie erlernen können. Starten Sie Vim, indem Sie `"vim"` eingeben, und drücken Sie die F1-Taste. Sie sollten zumindest die ersten 35 Zeilen lesen. Absolvieren Sie dann den Online-Trainingskurs, indem Sie mit dem Cursor auf `"| tutor |"` gehen, und drücken Sie `Strg-]`.

Anmerkung

Gute Editoren wie Vim und Emacs können Texte mit UTF-8 und anderen exotischen Kodierungen korrekt darstellen. Es ist eine gute Idee, die X-Umgebung in der UTF-8-Locale zu verwenden und benötigte Programme und Schriften dafür zu installieren. Editoren haben Optionen, um die Dateikodierung unabhängig von der X-Umgebung zu setzen. Bitte ziehen Sie deren Dokumentation bezüglich Multibyte-Text zu Rate.

1.4.7 Einen Standard-Texteditor einstellen

Debian enthält eine Reihe verschiedener Editoren. Wir empfehlen wie oben bereits erwähnt, das `vim`-Paket zu installieren.

Debian bietet über den Befehl `"/usr/bin/editor"` einen einheitlichen Zugriff auf den Standard-Editor des Systems, so dass andere Programme (z. B. `reportbug(1)`) ihn aufrufen können. Sie können ihn wie folgt ändern:

```
$ sudo update-alternatives --config editor
```

Ich empfehle Neulingen, die Auswahl `"/usr/bin/vim.basic"` gegenüber `"/usr/bin/vim.tiny"` zu bevorzugen, da er Syntaxhervorhebung beherrscht.

Tipp

Viele Programme nutzen die Umgebungsvariablen `"$EDITOR"` oder `"$VISUAL"`, um zu entscheiden, welcher Editor verwendet wird (lesen Sie dazu Abschnitt [1.3.5](#) und Abschnitt [9.3.11](#)). Zwecks Konsistenz auf Debian-Systemen sollten Sie dies auf `"/usr/bin/editor"` setzen. (Früher war `"$EDITOR"` auf `"ed"` und `"$VISUAL"` auf `"vi"` gesetzt.)

1.4.8 Anpassen von vim

Sie können das Verhalten von `vim(1)` mittels `"~/.vimrc"` anpassen.

Probieren Sie zum Beispiel folgendes:

```
" -----
" Lokale Konfiguration
"
set nocompatible
set nopaste
set pastetoggle=<f2>
syn on
if $USER == "root"
    set nomodeline
    set noswapfile
else
    set modeline
    set swapfile
endif
" Füllzeile, um zu vermeiden, dass die obige Zeile als modeline erkannt wird
```

```
" Füllzeile  
" Füllzeile
```

1.4.9 Aufzeichnen der Shell-Aktivitäten

Die Ausgabe eines Shell-Befehls könnte aus Ihrem Bildschirm herauslaufen und für immer verloren sein. Es ist nützlich, Shell-Aktivitäten in einer Datei zu protokollieren, um sie später kontrollieren zu können. Diese Art der Aufzeichnung ist besonders wichtig für jegliche Systemadministrationsaufgaben.

Die grundsätzliche Methode, um die Aktivitäten auf der Shell aufzuzeichnen ist, sie via `script(1)` laufen zu lassen.

Probieren Sie zum Beispiel folgendes:

```
$ script  
Script started, file is typescript (zu deutsch: Script gestartet, die Datei heißt ↵  
    typescript)
```

Führen Sie alle gewünschten Shell-Befehle via `script` aus.

Drücken Sie `Strg-D`, um `script` zu beenden.

```
$ vim typescript
```

Lesen Sie Abschnitt [9.2.3](#).

1.4.10 Grundlegende Unix-Befehle

Wir wollen einige grundlegende Unix-Befehle lernen. Ich verwende hier "Unix" in seiner allgemeinen Bedeutung. Alle Unix-Clone-Betriebssysteme bieten normalerweise entsprechende gleichbedeutende Befehle. Das Debian-System ist hier keine Ausnahme. Sorgen Sie sich nicht, wenn einige Befehle jetzt bei Ihnen nicht wie gewünscht funktionieren. Falls `alias` in der Shell verwendet wird, sind die zugehörigen Befehlsausgaben unterschiedlich. Diese Beispiele sind nicht dazu gedacht, in dieser Reihenfolge ausgeführt zu werden.

Probieren Sie die folgenden Befehle von einem nicht-privilegierten Benutzerkonto:

Anmerkung

Unix hat die Tradition, Dateinamen zu verstecken, die mit einem "." beginnen. Dies sind traditionell Dateien, die Konfigurationsinformationen und Benutzereinstellungen enthalten.

Anmerkung

Bezüglich des `cd`-Befehls schauen Sie unter `builtins(7)`.

Anmerkung

Der Standard-Pager auf einem reinen Debian-System ist `more(1)`, der nicht rückwärts scrollen kann. Indem Sie das `less`-Paket mittels `apt-get install less` installieren, wird `less(1)` zum Standard-Pager und Sie können mit den Pfeiltasten rückwärts scrollen.

Anmerkung

Die Zeichen "[" und "]" in dem regulären Ausdruck des obigen Befehls `ps aux | grep -e "[e]xim4*"` vermeiden, dass `grep` sich selbst findet. Das `"4*"` in dem regulären Ausdruck steht für 0 oder mehr Wiederholungen des Zeichens "4", wodurch `grep` sowohl `"exim"` wie auch `"exim4"` findet. Obwohl `"**"` in dem Dateinamen und dem regulären Ausdruck verwendet wird, ist ihre Bedeutung unterschiedlich. Lernen Sie die regulären Ausdrücke von `grep(1)`.

Befehl	Beschreibung
pwd	Name des derzeitigen/Arbeits-Verzeichnisses
whoami	derzeitigen Benutzernamen anzeigen
id	derzeitige Benutzeridentität anzeigen (Name, uid (Benutzer-ID), gid (Gruppen-ID) und zugehörige Gruppen)
file <foo>	den Dateityp der Datei "<foo>" anzeigen
type -p <befehlsname>	den Speicherort des Befehls <befehlsname> anzeigen
which <befehlsname>	"
type <befehlsname>	Informationen zum Befehl <befehlsname> anzeigen
apropos <schlüsselwort>	Befehle mit Bezug zu <schlüsselwort> finden
man -k <schlüsselwort>	"
whatis <befehlsname>	eine einzeilige Kurzbeschreibung des Befehls <befehlsname> anzeigen
man -a <befehlsname>	Beschreibung zum Befehl <befehlsname> anzeigen (Unix-Stil)
info <befehlsname>	eher lange Beschreibung zum Befehl "<befehlsname>" anzeigen (GNU-Stil)
ls	Inhalte eines Verzeichnisses auflisten (keine versteckten Dateien und Verzeichnisse)
ls -a	Inhalte eines Verzeichnisses auflisten (alle Dateien und Verzeichnisse)
ls -A	Inhalte eines Verzeichnisses auflisten (nahezu alle Dateien und Verzeichnisse, "." und ".." werden übersprungen)
ls -la	alle Inhalte eines Verzeichnisses mit detaillierten Informationen auflisten
ls -lai	alle Inhalte eines Verzeichnisses mit Inode-Nummer und detaillierten Informationen auflisten
ls -d	alle Verzeichnisse im derzeitigen Verzeichnis auflisten
tree	Verzeichnisbaum-Inhalte anzeigen
lsuf <foo>	"geöffnet"-Status der Datei <foo> anzeigen
lsuf -p <pid>	Dateien auflisten, die von dem Prozess mit der ID <pid> geöffnet wurden
mkdir <foo>	ein neues Verzeichnis <foo> im derzeitigen Verzeichnis erstellen
rmdir <foo>	das Verzeichnis <foo> im derzeitigen Verzeichnis löschen
cd <foo>	in das Verzeichnis <foo> im derzeitigen Verzeichnis oder im in der Variable "\$CDPATH" enthaltenen Verzeichnis wechseln
cd /	in das root-Verzeichnis wechseln
cd	in das Heimatverzeichnis des derzeitigen Benutzers wechseln
cd /<foo>	in das Verzeichnis mit dem absoluten Pfad "/"<foo>" wechseln
cd ..	in das übergeordnete Verzeichnis (Eltern-Verzeichnis) wechseln
cd ~<foo>	in das Heimatverzeichnis des Benutzers <foo> wechseln
cd -	in das vorherige Verzeichnis wechseln
</etc/motd pager	Inhalt von "</etc/motd" mit dem Standard-Pager anzeigen
touch <leeredatei>	eine leere Datei mit Namen <leeredatei> erzeugen
cp <foo> <bar>	die vorhandene Datei <foo> in eine neue Datei <bar> kopieren
rm <unnützedatei>	die Datei <unnützedatei> entfernen
mv <foo> <bar>	die vorhandene Datei <foo> umbenennen zum neuen Namen <bar> (<bar> darf nicht existieren)
mv <foo> <bar>	die vorhandene Datei <foo> an einen neuen Ort <bar>/<foo> verschieben (das Verzeichnis <bar> muss existieren)
mv <foo> <bar>/<baz>	die vorhandene Datei <foo> an einen neuen Ort verschieben und gleichzeitig umbenennen nach <bar>/<baz> (das Verzeichnis <bar> muss existieren, aber der Dateiname <bar>/<baz> darf nicht existieren)
chmod 600 <foo>	die vorhandene Datei <foo> als nicht lesbar und nicht schreibbar für andere Benutzer einstellen (und nicht ausführbar für alle)
chmod 644 <foo>	die vorhandene Datei <foo> als lesbar, aber nicht schreibbar für andere Benutzer einstellen (und nicht ausführbar für alle)
chmod 755 <foo>	die vorhandene Datei <foo> als lesbar, aber nicht schreibbar für andere Benutzer einstellen (und als ausführbar für alle)
find . -name <Suchmuster>	passende Dateinamen finden durch Verwendung von Shell-<Suchmuster> (langsamer)
locate -d . <Suchmuster>	passende Dateinamen finden durch Verwendung von Shell-<Suchmuster> (schneller, nutzt eine regelmäßig erstellte Datenbank)
grep -r <Suchmuster> .	das <Suchmuster> in allen Dateien finden, die im derzeitigen Verzeichnis oder in den untergeordneten Verzeichnissen existieren

Bitte durchlaufen Sie zur Übung verschiedene Verzeichnisse und erforschen Sie das System. Wenn Sie Fragen zu einem der Konsolenbefehle haben, seien Sie sicher, dass Sie die Handbuchseite gelesen haben.

Probieren Sie zum Beispiel folgendes:

```
$ man man
$ man bash
$ man builtins
$ man grep
$ man ls
```

An den Stil der Handbuchseiten kann man sich eventuell etwas schwer gewöhnen, da sie eher knapp gehalten sind, speziell die älteren, traditionellen. Aber wenn Sie sich einmal daran gewöhnt haben, werden Sie die knappe Form zu schätzen wissen.

Bitte beachten Sie, dass viele der Unix-ähnlichen Befehle von GNU und BSD eine kurze Hilfe anzeigen, wenn Sie auf eine der folgenden Arten aufgerufen werden (oder in einigen Fällen ohne jegliche Argumente):

```
$ <befehlsname> --help
$ <befehlsname> -h
```

1.5 Der einfache Shell-Befehl

Jetzt haben Sie ein Gefühl, wie das Debian-System genutzt wird. Lassen Sie uns tiefer in den Mechanismus der Befehlsausführung einsteigen. Eine genaue Beschreibung finden Sie unter `bash(1)`.

Ein einfacher Befehl ist eine Abfolge mehrerer Komponenten.

1. Zuweisung von Variablen (optional)
2. Befehlsname
3. Argumente (optional)
4. Weiterleitungen (optional: `>` , `>>` , `<` , `<<` usw.)
5. Steuer-Operator (optional: `&&` , `|` , `<newline>` , `;` , `&` , `(,)`)

1.5.1 Befehlsausführung und Umgebungsvariablen

Die Werte einiger [Umgebungsvariablen](#) verändern das Verhalten einiger Unix-Befehle.

Die Standardwerte von Umgebungsvariablen werden ursprünglich vom PAM-System gesetzt und dann werden einige von ihnen von manchen Anwendungsprogrammen zurückgesetzt.

- Display-Manager wie z.B. `gdm3` setzen Umgebungsvariablen zurück.
- Die Shell setzt durch Ihre Startmechanismen in `"~/ .bash_profile"` und `"~/ .bashrc"` Umgebungsvariablen zurück.

1.5.2 Die "\$LANG"-Variable

Der vollständige Wert für das Gebietsschema (Locale), der an die "\$LANG"-Variable übergeben wird, besteht aus drei Teilen: `"xx_YY.ZZZZ"`.

Einschlägige Beschreibungen zu Sprach- und Länder-Codes finden Sie unter `"info gettext"`.

Auf modernen Debian-Systemen sollten Sie die Zeichenkodierung immer auf **UTF-8** setzen, außer Sie möchten aus einem guten Grund speziell eine der historischen Kodierungen verwenden und haben dazu ausreichende Hintergrundkenntnisse.

Tolle Details zur Konfiguration des Gebietsschemas finden Sie in Abschnitt [8.4](#).

locale-Wert	Bedeutung
xx	Sprach-Code laut ISO 639 (in Kleinbuchstaben) wie z.B. "de"
YY	Länder-Code laut ISO 3166 (in Großbuchstaben) wie z.B. "DE"
ZZZZ	Zeichenkodierung (am besten immer auf "UTF-8" setzen)

Tabelle 1.17: Die drei Teile des locale-Wertes

Anmerkung

Auf Debian-Systemen sollten Sie sicherstellen, dass das Paket `locales-all` installiert ist, wenn Sie alle Gebiets-schemata nutzen möchten.

Anmerkung

"LANG=de_DE" ist weder "LANG=C" noch "LANG=de_DE.UTF-8". Es ist "LANG=de_DE.ISO-8859-1" (siehe Abschnitt [8.4.1](#)).

empfohlenes Gebietsschema	Sprache (Gebiet)
en_US.UTF-8	Englisch (USA)
en_GB.UTF-8	Englisch (Großbritannien)
fr_FR.UTF-8	Französisch (Frankreich)
de_DE.UTF-8	Deutsch (Deutschland)
it_IT.UTF-8	Italienisch (Italien)
es_ES.UTF-8	Spanisch (Spanien)
ca_ES.UTF-8	Katalanisch (Spanien)
sv_SE.UTF-8	Schwedisch (Schweden)
pt_BR.UTF-8	Portugiesisch (Brasilien)
ru_RU.UTF-8	Russisch (Russland)
zh_CN.UTF-8	Chinesisch (Volksrepublik China)
zh_TW.UTF-8	Chinesisch (Taiwan)
ja_JP.UTF-8	Japanisch (Japan)
ko_KR.UTF-8	Koreanisch (Republik Korea)
vi_VN.UTF-8	Vietnamesisch (Vietnam)

Tabelle 1.18: Liste mit Empfehlungen zum Gebietsschema

Eine typische Befehlsausführung verwendet eine Shell-Sequenz wie die folgende:

```
$ date
Sun Jun  3 10:27:39 JST 2007
$ LANG=fr_FR.UTF-8 date
dimanche 3 juin 2007, 10:27:33 (UTC+0900)
```

Hier wird das Programm `date(1)` mit verschiedenen Werten für die Umgebungsvariable "`$LANG`" ausgeführt:

- Beim ersten Befehl ist "`$LANG`" auf den System-[locale](#)-Standardwert "`en_US.UTF-8`" gesetzt.
- Beim zweiten Befehl ist "`$LANG`" auf den französischen UTF-8-[locale](#)-Wert "`fr_FR.UTF-8`" gesetzt.

Die meisten Befehlsausführungen haben keine voreingestellten Definitionen für Umgebungsvariablen. Bei dem obigen Beispiel können Sie alternativ auch folgendes ausführen:

```
$ LANG=fr_FR.UTF-8
$ date
dimanche 3 juin 2007, 10:27:33 (UTC+0900)
```

Wie Sie hier sehen können, wird die Ausgabe des Befehls von der Umgebungsvariable beeinflusst, um eine französische Ausgabe zu erzeugen. Wenn Sie möchten, dass die Umgebungsvariable von Unterprozessen übernommen wird (z.B. beim Aufruf eines Shellskriptes), müssen Sie sie stattdessen wie folgt mit **export** exportieren:

```
$ export LANG
```

Anmerkung

Wenn Sie ein typisches Konsolen-Terminal verwenden, ist die "\$LANG"-Variable gewöhnlich so eingestellt, dass Sie durch die Desktop-Umgebung **exportiert** wird. Daher ist das obige nicht wirklich ein gutes Beispiel, um die Auswirkungen von `export` auszuprobieren.

Tipp

Wenn Sie einen Fehlerbericht einreichen und als Umgebung nicht Englisch verwenden, ist es eine gute Idee, den Befehl mit "LANG=en_US.UTF-8" laufen zu lassen und zu überprüfen.

Informationen zu "\$LANG" und anderen zugehörigen Umgebungsvariablen finden Sie in `locale(5)` und `locale(7)`.

Anmerkung

Ich empfehle Ihnen, die Systemumgebung direkt über die "\$LANG"-Variable zu konfigurieren und die Finger von den "\$LC_*"-Variablen zu lassen, außer es ist absolut nötig.

1.5.3 Die "\$PATH"-Variable

Wenn Sie einen Befehl in die Shell eingeben, sucht die Shell in den durch die Umgebungsvariable "\$PATH" definierten Verzeichnissen nach diesem Befehl. Der Wert der Umgebungsvariablen "\$PATH" wird auch als Shell-Suchpfad bezeichnet.

In der Standard-Debian-Installation enthält die Umgebungsvariable "\$PATH" bei normalen Benutzerkonten nicht die Verzeichnisse "/sbin" und "/usr/sbin". Zum Beispiel muss der `ifconfig`-Befehl mit dem vollständigen Pfad aufgerufen werden, also `/sbin/ifconfig`. Ähnlich dazu ist der `ip`-Befehl in `/bin` abgelegt.

Sie können die Umgebungsvariable "\$PATH" der Bash-Shell über die Dateien `~/ .bash_profile` oder `~/ .bashrc` verändern.

1.5.4 Die "\$HOME"-Variable

Viele Befehle speichern Konfigurationsdateien im Heimatverzeichnis des jeweiligen Benutzers. Über diese lässt sich das Verhalten der Befehle anpassen. Der Ort des Heimatverzeichnisses ist in der Umgebungsvariable "\$HOME" gespeichert.

Wert von "\$HOME"	Situation bei Programmausführung
/	Programm ausgeführt vom init-Prozess (Daemon)
/root	Programm ausgeführt von der root-Shell
/home/<normaler_benutzer>	Programm ausgeführt von der Shell des normalen Benutzers
/home/<normaler_benutzer>	Programm ausgeführt vom GUI-Desktop-Menü des normalen Benutzers
/home/<normaler_benutzer>	Programm ausgeführt von root mittels <code>"sudo programm"</code>
/root	Programm ausgeführt von root mittels <code>"sudo -H programm"</code>

Tabelle 1.19: Liste der Werte von "\$HOME"

Tipp

Die Shell expandiert `~/` zum Heimatverzeichnis des aktuellen Benutzers, also `"$HOME/"`, und `~foo/` zum Heimatverzeichnis des Benutzers `foo`, also `"/home/foo/"`.

1.5.5 Befehlszeilen-Optionen

Einige Befehle akzeptieren Argumente. Argumente, die mit `"-"` oder `"- "` beginnen, werden Optionen genannt und steuern das Verhalten des Befehls.

```
$ date
Mon Oct 27 23:02:09 CET 2003
$ date -R
Mon, 27 Oct 2003 23:02:40 +0100
```

Hier ändert das Befehlszeilen-Argument `"-R"` das Verhalten von `date(1)` zwecks Ausgabe einer [RFC2822](#)-konformen Datumsangabe.

1.5.6 Shell-Glob

Oft möchten Sie einen Befehl auf eine Gruppe von Dateien anwenden, ohne all die Dateinamen einzutippen. Die Dateinamen-Suchmuster mittels **Shell-Glob** (manchmal auch **Wildcards** genannt) erfüllen diese Bedürfnisse.

Shell-Glob-Suchmuster	Beschreibung der Regel
<code>*</code>	Dateiname (oder Segment), der nicht mit <code>"."</code> beginnt
<code>.*</code>	Dateiname (oder Segment), der mit <code>"."</code> beginnt
<code>?</code>	genau ein Zeichen
<code>[...]</code>	genau eins der in Klammern eingeschlossenen Zeichen
<code>[a-z]</code>	genau ein Zeichen von denen zwischen <code>"a"</code> und <code>"z"</code>
<code>[^...]</code>	genau ein Zeichen von denen, die nicht in Klammern eingeschlossen sind (außer <code>"^"</code>)

Tabelle 1.20: Shell-Glob-Suchmuster

Probieren Sie zum Beispiel folgendes:

```
$ mkdir junk; cd junk; touch 1.txt 2.txt 3.c 4.h .5.txt ..6.txt
$ echo *.txt
1.txt 2.txt
$ echo *
1.txt 2.txt 3.c 4.h
$ echo *. [hc]
3.c 4.h
$ echo .*
. . . 5.txt ..6.txt
$ echo .*[^.]*
.5.txt ..6.txt
$ echo [^1-3]*
4.h
$ cd ..; rm -rf junk
```

Weitere Details finden Sie in `glob(7)`.

Anmerkung

Anders als die normale Dateinamenerweiterung der Shell findet das Shell-Suchmuster `"*"`, das in `find(1)` mit `-name` abgefragt wird, den führenden `"."` des Dateinamens (neue [POSIX](#)-Funktionalität).

Anmerkung

`bash` kann dahingehend angepasst werden, sein `glob`-Verhalten mittels der integrierten `shopt`-Optionen wie `"dotglob"`, `"noglob"`, `"nocaseglob"`, `"nullglob"`, `"extglob"` usw. zu verändern. Lesen Sie dazu `bash(1)`.

1.5.7 Rückgabewert eines Befehls

Jeder Befehl gibt seinen Beendigungsstatus (Exit-Status, Variable: "\$?") als Rückgabewert zurück.

Befehls-Beendigungsstatus	numerischer Rückgabewert	logischer Rückgabewert
Erfolg	Null, 0	WAHR
Fehler	nicht-Null, -1	FALSCH

Tabelle 1.21: Befehls-Beendigungs-Codes

Probieren Sie zum Beispiel folgendes:

```
$ [ 1 = 1 ] ; echo $?
0
$ [ 1 = 2 ] ; echo $?
1
```

Anmerkung

Bitte beachten Sie, dass im logischen Zusammenhang für die Shell **Erfolg** als logisches **WAHR** betrachtet wird, was 0 (Null) als Wert bedeutet. Dies ist nicht ganz intuitiv und daher soll hier daran erinnert werden.

1.5.8 Typische Befehlssequenzen und Shell-Weiterleitungen

Versuchen Sie, sich folgende Shell-Befehlsfolgen zu merken, die in einer Zeile eingegeben werden:

Das Debian-System ist ein Multi-Tasking-System. Hintergrundprozesse erlauben es Benutzern, mehrere Programme in einer einzigen Shell laufen zu lassen. Zur Verwaltung von Hintergrundprozessen werden die Shell-Builtins `jobs`, `fg`, `bg` und `kill` benutzt. Bitte lesen Sie die entsprechenden Abschnitte von `bash(1)` bezüglich "SIGNALS", "JOB CONTROL" und `builtins(1)`.

Probieren Sie zum Beispiel folgendes:

```
$ </etc/motd pager
```

```
$ pager </etc/motd
```

```
$ pager /etc/motd
```

```
$ cat /etc/motd | pager
```

Obwohl alle 4 Beispiele von Shell-Weiterleitungen dasselbe anzeigen, führt das letzte Beispiel einen separaten `cat`-Befehl aus und verschwendet ohne Grund zusätzliche Ressourcen.

Die Shell erlaubt Ihnen, Dateien mittels dem `exec`-Builtin mit einem frei wählbaren Datei-Deskriptor zu öffnen.

```
$ echo Hallo >foo
$ exec 3<foo 4>bar # Dateien öffnen
$ cat <&3 >&4 # stdin an 3 weiterleiten und stdout an 4
$ exec 3<&- 4>&- # Dateien schließen
$ cat bar
Hallo
```

Die Datei-Deskriptoren 0-2 sind fest vordefiniert.

Befehlsabfolge	Beschreibung
befehl &	Ausführung von befehl im Hintergrund in einer Unter-Shell
befehl1 befehl2	Weiterleitung (pipe) der Standardausgabe von befehl1 an die Standardeingabe von befehl2 (gleichzeitige Ausführung)
befehl1 2>&1 befehl2	Weiterleitung (pipe) der Standardausgabe und Standardfehler von befehl1 an die Standardeingabe von befehl2 (gleichzeitige Ausführung)
befehl1 ; befehl2	Aufeinander folgende Ausführung von befehl1 und befehl2
befehl1 && befehl2	Ausführung von befehl1; falls erfolgreich, anschließende Ausführung von befehl2 (Erfolg zurückgeben, wenn beide (befehl1 und befehl2) erfolgreich sind)
befehl1 befehl2	Ausführung von befehl1; falls nicht erfolgreich, anschließende Ausführung von befehl2 (Erfolg zurückgeben, wenn befehl1 oder befehl2 erfolgreich ist)
befehl > foo	Weiterleitung der Standardausgabe von befehl in die Datei foo (Datei überschreiben)
befehl 2> foo	Weiterleitung des Standardfehlers von befehl in die Datei foo (Datei überschreiben)
befehl >> foo	Weiterleitung der Standardausgabe von befehl in die Datei foo (an Datei anhängen)
befehl 2>> foo	Weiterleitung des Standardfehlers von befehl in die Datei foo (an Datei anhängen)
befehl > foo 2>&1	Weiterleitung der Standardausgabe und Standardfehler von befehl in die Datei foo
befehl < foo	Weiterleitung der Standardeingabe von befehl in die Datei foo
befehl << begrenzung	Weiterleitung der Standardeingabe von befehl in die folgenden Zeilen, bis "begrenzung" erreicht ist
befehl <<- delimiter	Weiterleitung der Standardeingabe von befehl in die folgenden Zeilen, bis "begrenzung" erreicht ist (die führenden Tab-Zeichen werden von den Eingabezeilen entfernt)

Tabelle 1.22: Abfolgen von Shell-Befehlen

Gerät	Beschreibung	Datei-Deskriptor
stdin	Standardeingabe	0
stdout	Standardausgabe	1
stderr	Standardfehler	2

Tabelle 1.23: Vordefinierte Datei-Deskriptoren

1.5.9 Befehls-Alias

Sie können einen Alias für häufig verwendete Befehle definieren.

Probieren Sie zum Beispiel folgendes:

```
$ alias la='ls -la'
```

Jetzt funktioniert "la" als Kurzform für "ls -la", was alle Dateien in einem langen Listenformat auflistet.

Sie können alle vorhandenen Alias-Definitionen mit `alias` anzeigen (lesen Sie dazu in `bash(1)` den Abschnitt "SHELL BUILTIN COMMANDS").

```
$ alias
...
alias la='ls -la'
```

Sie können den exakten Pfad oder die Identität eines Befehls mit `type` herausfinden (lesen Sie dazu in `bash(1)` den Abschnitt "SHELL BUILTIN COMMANDS").

Probieren Sie zum Beispiel folgendes:

```
$ type ls
ls is hashed (/bin/ls)
$ type la
la is aliased to ls -la
$ type echo
echo is a shell builtin
$ type file
file is /usr/bin/file
```

Hier wurde kürzlich nach "ls" gesucht, nach "file" hingegen nicht, daher ist "ls" "hashed", d.h. die Shell hat einen internen Eintrag für einen schnellen Zugriff auf den Speicherort des "ls"-Befehls.

Tipp

Lesen Sie Abschnitt [9.2.7](#).

1.6 Unix-ähnliche Textverarbeitung

In Unix-ähnlichen Arbeitsumgebungen können Textverarbeitungsvorgänge durchgeführt werden, indem der Text durch Abfolgen mehrerer Standardwerkzeuge geleitet wird. Dies war eine weitere bedeutende Unix-Innovation.

1.6.1 Unix-Textverarbeitungswerkzeuge

Es gibt ein paar Standard-Textverarbeitungswerkzeuge, die auf Unix-ähnlichen Systemen sehr oft verwendet werden.

- Ohne Verwendung regulärer Ausdrücke:
 - `cat(1)` verkettet Dateien und gibt den vollständigen Inhalt aus.
 - `tac(1)` verkettet Dateien und gibt den Inhalt in umgekehrter Reihenfolge aus.
 - `cut(1)` wählt Teile von Zeilen und ausgegebenen Inhalten aus.
 - `head(1)` gibt die ersten Zeilen von Dateien aus.
 - `tail(1)` gibt die letzten Zeilen von Dateien aus.
 - `sort(1)` sortiert Zeilen von Textdateien.
-

- `uniq(1)` entfernt doppelte Zeilen aus einer sortierten Datei.
- `tr(1)` übersetzt oder löscht Zeichen.
- `diff(1)` vergleicht Dateien Zeile für Zeile.
- Mit Verwendung grundlegender regulärer Ausdrücke (basic regular expressions / **BRE**):
 - `grep(1)` findet Text über Suchmuster.
 - `ed(1)` ist ein primitiver Zeilen-Editor.
 - `sed(1)` ist ein Stream-Editor.
 - `vim(1)` ist ein Bildschirm-gestützter Editor.
 - `emacs(1)` ist ein Bildschirm-gestützter Editor (leicht erweiterte **BRE**)
- Mit Verwendung erweiterter regulärer Ausdrücke (extended regular expressions / **ERE**):
 - `egrep(1)` findet Text über Suchmuster.
 - `awk(1)` führt einfache Textverarbeitung durch.
 - `tcl(3tcl)` kann jede erdenkliche Art von Textverarbeitung durchführen. Lesen Sie dazu `re_syntax(3)`. Oft zusammen mit `tk(3tk)` verwendet.
 - `perl(1)` kann jede erdenkliche Art von Textverarbeitung durchführen. Lesen Sie dazu `perlre(1)`.
 - `pcregrep(1)` aus dem `pcregrep`-Paket findet Text über [Perl-kompatible reguläre Ausdrücke \(PCRE\)](#).
 - `python(1)` mit dem `re`-Modul kann jede erdenkliche Art von Textverarbeitung durchführen. Lesen Sie dazu `"/usr/share/doc/p`

Wenn Sie sich nicht sicher sind, was genau diese Befehle tun, verwenden Sie bitte `"man befehl"`, um es selbst herauszufinden.

Anmerkung

Sortierreihenfolge und Bereichsausdruck sind abhängig vom Gebietsschema (Locale). Wenn Sie das traditionelle Verhalten eines Befehls erreichen möchten, verwenden Sie statt der **UTF-8**-Locales die **C**-Locale, indem Sie dem Befehl `"LANG=C"` voranstellen (lesen Sie dazu Abschnitt [1.5.2](#) und Abschnitt [8.4](#)).

Anmerkung

Die regulären [Perl](#)-Erweiterungen (`perlre(1)`), [Perl-kompatiblen regulären Ausdrücke \(PCRE\)](#) sowie regulären [Python](#)-Erweiterungen (bereitgestellt durch das `re`-Modul) enthalten viele gängige Erweiterungen zu den normalen **erweiterten regulären Ausdrücken (ERE)**.

1.6.2 Reguläre Ausdrücke

[Reguläre Ausdrücke](#) werden in vielen Textverarbeitungswerkzeugen verwendet. Sie sind den Shell Globbs ähnlich, aber viel komplexer und leistungsfähiger.

Ein regulärer Ausdruck beschreibt ein Muster und besteht aus Textzeichen und **Metazeichen**.

Ein **Metazeichen** ist einfach ein Zeichen mit einer speziellen Bedeutung. Es gibt zwei Hauptgruppen, **BRE** und **ERE**, abhängig von den Textwerkzeugen, wie unten beschrieben.

Die regulären Ausdrücke von **emacs** gehören grundsätzlich zu den **BRE**, sind aber erweitert worden, damit `"+"` und `"?"` wie bei den **ERE** als **Metazeichen** behandelt werden. Es ist daher nicht nötig, sie bei den regulären Ausdrücken von **emacs** mit `"\"` zu schützen.

`grep(1)` kann verwendet werden, um eine Textsuche mittels regulärer Ausdrücke durchzuführen.

Probieren Sie zum Beispiel folgendes:

BRE	ERE	Beschreibung des regulären Ausdrucks
\ . [] ^ \$ *	\ . [] ^ \$ *	allgemeine Metazeichen
\+ \? \(\) \{ \} \		nur bei BRE: durch "\ " geschützte Metazeichen
	+ ? () { }	nur bei ERE: nicht durch "\ " geschützte Metazeichen
c	c	findet Nicht-Metazeichen "c"
\c	\c	findet den Buchstaben "c", auch wenn "c" selbst ein Metazeichen ist
.	.	findet jegliches Zeichen inklusive Newline
^	^	Position am Anfang einer Zeichenkette
\$	\$	Position am Ende einer Zeichenkette
\<	\<	Position am Anfang eines Wortes
\>	\>	Position am Ende eines Wortes
[abc...]	[abc...]	findet irgendein Zeichen von "abc..."
[^abc...]	[^abc...]	findet irgendein Zeichen außer "abc..."
r*	r*	findet keinen oder mehrere reguläre Ausdrücke festgelegt durch "r"
r\+	r+	findet einen oder mehrere reguläre Ausdrücke festgelegt durch "r"
r\?	r?	findet keinen oder einen regulären Ausdruck festgelegt durch "r"
r1 r2	r1 r2	findet einen der regulären Ausdrücke festgelegt durch "r1" oder "r2"
\(r1 r2\)	(r1 r2)	findet einen der regulären Ausdrücke festgelegt durch "r1" oder "r2" und behandelt es als einen in Klammern gesetzten regulären Ausdruck

Tabelle 1.24: Metazeichen für BRE und ERE


```
$ egrep 'GNU.*LICENSE|Yoyodyne' /usr/share/common-licenses/GPL
GNU GENERAL PUBLIC LICENSE
GNU GENERAL PUBLIC LICENSE
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
```

Tipp

Lesen Sie Abschnitt [9.2.7](#).

1.6.3 Ersetzungsausdrücke

Bei den Ersetzungsausdrücken haben einige Zeichen spezielle Bedeutungen.

Ersetzungsausdruck	Beschreibung des Textes, der den Ersetzungsausdruck ersetzen soll
&	was der reguläre Ausdruck gefunden hat (benutzen Sie \& in emacs)
\n	was der n-te in Klammern gesetzte reguläre Ausdruck gefunden hat ("n" ist ein Zahl)

Tabelle 1.25: Der Ersetzungsausdruck

Bei Ersetzungsausdrücken für Perl wird "\$&" statt "&" verwendet und "\$n" statt "\n".

Probieren Sie zum Beispiel folgendes:

```
$ echo zzz1abc2efg3hij4 | \
sed -e 's/(1[a-z]*)[0-9]*\(.*)$/=&/'
zzz=1abc2efg3hij4=
$ echo zzz1abc2efg3hij4 | \
sed -e 's/(1[a-z]*)[0-9]*\(.*)$/\2===\1/'
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
perl -pe 's/(1[a-z]*)[0-9]*(.*)$/$2===\$1/'
zzzefg3hij4===1abc
$ echo zzz1abc2efg3hij4 | \
perl -pe 's/(1[a-z]*)[0-9]*(.*)$/=$&/'
zzz=1abc2efg3hij4=
```

Achten Sie hier bitte besonders auf die Art der **in Klammern** gesetzten regulären Ausdrücke und wie die gefundenen Zeichenketten bei der Textersetzung mit verschiedenen Werkzeugen genutzt werden.

Diese regulären Ausdrücke können in einigen Editoren auch zur Cursor-Verschiebung und Textersetzung verwendet werden.

Der Back-Slash "\" am Ende der Zeile in der Shell-Befehlszeile schützt das Newline als Leerraum-Zeichen und erlaubt die Fortsetzung der Shell-Befehlszeileneingabe in die nächste Zeile.

Bitte lesen Sie alle zugehörigen Handbuchseiten, um diese Befehle zu erlernen.

1.6.4 Globale Ersetzungen mit regulären Ausdrücken

Der Befehl ed(1) ersetzt hier alle Vorkommen von "VON_REGEX" in "datei" durch "ZU_TEXT".

```
$ ed datei <<EOF
,s/VON_REGEX/ZU_TEXT/g
w
q
EOF
```

Der Befehl `sed(1)` ersetzt hier alle Vorkommen von `"VON_REGEX"` in `"datei"` durch `"ZU_TEXT"`.

```
$ sed -i -e 's/VON_REGEX/ZU_TEXT/g' datei
```

Der Befehl `vim(1)` ersetzt hier durch Verwendung der `ex(1)`-Befehle alle Vorkommen von `"VON_REGEX"` in `"datei"` durch `"ZU_TEXT"`.

```
$ vim +%s/VON_REGEX/ZU_TEXT/gc ' +w' '+q' datei
```

Tipp

Die `"c"`-Markierung in dem obigen Beispiel gewährleistet die interaktive Ersetzung jedes Vorkommnisses.

Mehrere Dateien (`"datei1"`, `"datei2"` und `"datei3"`) können auf ähnlichem Wege mittels regulärer Ausdrücke mit `vim(1)` oder `perl(1)` verarbeitet werden.

```
$ vim +argdo %s/VON_REGEX/ZU_TEXT/ge|update' '+q' datei1 datei2 datei3
```

Tipp

Die Option `"e"` in dem obigen Beispiel verhindert, dass der `"No match"`-Fehler die Zuweisung unterbricht.

```
$ perl -i -p -e 's/VON_REGEX/ZU_TEXT/g;' datei1 datei2 datei3
```

In dem `perl(1)`-Beispiel steht `"-i"` für direkte Bearbeitung der jeweiligen Zieldatei und `"-p"` für implizites Abarbeiten aller angegebenen Dateien mittels Schleifen.

Tipp

Durch die Verwendung von `"-i.bak"` statt `"-i"` wird jede Originaldatei gesichert, dabei wird `".bak"` zum Dateinamen hinzugefügt. Dies erlaubt eine einfachere Wiederherstellung bei Fehlern in komplexen Situationen.

Anmerkung

`ed(1)` und `vim(1)` gehören zu den **BRE**, `perl(1)` zu den **ERE**.

1.6.5 Extrahieren von Daten aus einer Textdatei-Tabelle

Wir nehmen an, eine Textdatei namens `"DPL"` enthält die Namen einiger Debian-Projektleiter der Jahre vor 2004 sowie deren Ernennungsdatum in einem durch Leerzeichen getrennten Format:

```
Ian      Murdock   August  1993
Bruce    Perens    April   1996
Ian      Jackson   January 1998
Wichert  Akkerman   January 1999
Ben      Collins    April   2001
Bdale    Garbee     April   2002
Martin   Michlmayr  March   2003
```

Tipp

Lesen Sie ["Eine kurze Geschichte von Debian"](#) bezüglich der neuesten [Vergangenheit zum Debian-Projektleiter](#).

Awk wird oft verwendet, um Daten aus dieser Art von Dateien zu extrahieren.

Probieren Sie zum Beispiel folgendes:

```
$ awk '{ print $3 }' <DPL          # Monat der Ernennung
August
April
January
January
April
April
March
$ awk '($1=="Ian") { print }' <DPL      # DPL namens Ian
Ian      Murdock      August  1993
Ian      Jackson      January 1998
$ awk '($2=="Perens") { print $3,$4 }' <DPL # Wann Perens ernannt wurde
April 1996
```

Auch Shells wie Bash können verwendet werden, um diese Art von Dateien zu verarbeiten.

Probieren Sie zum Beispiel folgendes:

```
$ while read first last month year; do
    echo $month
done <DPL
... ergibt die gleiche Ausgabe wie beim ersten Awk-Beispiel
```

Hier verwendet der Builtin-Befehl `read` Zeichen aus `"IFS"` (interne Feldseparatoren), um Zeilen in einzelne Wörter aufzusplitten.

Wenn Sie `"IFS"` in `":"` ändern, können Sie auf nette Weise `"/etc/passwd"` mit der Shell verarbeiten.

```
$ oldIFS="$IFS"          # den alten Wert speichern
$ IFS=':'
$ while read user password uid gid rest_of_line; do
    if [ "$user" = "bozo" ]; then
        echo "$user's ID is $uid"
    fi
done < /etc/passwd
bozo's ID is 1000
$ IFS="$oldIFS"          # den alten Wert wiederherstellen
```

(Wenn Awk dies erledigen soll, verwenden Sie `"FS=':'"`, um den Feldseparator festzulegen.)

IFS wird auch von der Shell verwendet, um die Ergebnisse von Parameterexpansion, Befehlsersetzung und arithmetischer Expansion aufzusplitten. Diese werden nicht innerhalb von doppelten oder einfachen Anführungszeichen dargestellt. Der Standardwert von IFS ist die Kombination von `<Space>`, `<Tab>` und `<Newline>`.

Seien Sie vorsichtig bei der Verwendung dieser Shell-IFS-Tricks. Verrückte Dinge passieren, wenn die Shell einige Teile des Skripts als eigene **Eingabe** interpretiert.

```
$ IFS=":"                # ":" und "," als IFS verwenden
$ echo IFS=$IFS, IFS="$IFS" # echo ist ein Bash-Builtin
IFS= , IFS=;,
$ date -R                # eine einfache Befehlsausgabe
Sat, 23 Aug 2003 08:30:15 +0200
$ echo $(date -R)         # Unter-Shell --> Eingabe zur Haupt-Shell
Sat 23 Aug 2003 08 30 36 +0200
$ unset IFS               # IFS auf den Standardwert zurücksetzen
$ echo $(date -R)
Sat, 23 Aug 2003 08:30:50 +0200
```

1.6.6 Skript-Schnipsel für die Befehlsweiterleitung

Die folgenden Befehlsketten tun nette Dinge als Teil einer Weiterleitung:

Skript-Schnipsel (in einer Zeile eingeben)	Auswirkung des Befehls
<code>find /usr -print</code>	findet alle Dateien in <code>"/usr"</code>
<code>seq 1 100</code>	gibt die Zahlen 1 bis 100 aus
<code> xargs -n 1 <befehl></code>	führt <code><befehl></code> wiederholt aus, jeweils mit jedem Element aus der Weiterleitung als Argument
<code> xargs -n 1 echo</code>	splittet durch Leerraum getrennte Elemente aus der Weiterleitung in einzelne Zeilen auf
<code> xargs echo</code>	führt alle Zeilen aus der Weiterleitung in eine Zeile zusammen
<code> grep -e <regex_muster></code>	extrahiert Zeilen aus der Weiterleitung, die das Regex-Suchmuster enthalten
<code> grep -v -e <regex_muster></code>	extrahiert Zeilen aus der Weiterleitung, die nicht das Regex-Suchmuster enthalten
<code> cut -d: -f3 -</code>	extrahiert das dritte Feld aus der Weiterleitung, separiert durch <code>:</code> (z.B. aus der <code>passwd</code> -Datei)
<code> awk '{ print \$3 }'</code>	extrahiert das dritte Feld aus der Weiterleitung, separiert durch Leerraum-Zeichen
<code> awk -F'\t' '{ print \$3 }'</code>	extrahiert das dritte Feld aus der Weiterleitung, separiert durch Tab
<code> col -bx</code>	entfernt Backspace und expandiert Tabs zu Leerzeichen
<code> expand -</code>	expandiert Tabs
<code> sort uniq</code>	sortiert Inhalte und entfernt Duplikate
<code> tr 'A-Z' 'a-z'</code>	konvertiert Großschreibung in Kleinschreibung
<code> tr -d '\n'</code>	führt mehrere Zeilen in eine zusammen
<code> tr -d '\r'</code>	entfernt CR (carriage return/Wagenrücklauf)
<code> sed 's/^/# /'</code>	fügt <code>"#"</code> zum Anfang jeder Zeile hinzu
<code> sed 's/\.ext//g'</code>	entfernt <code>".ext"</code>
<code> sed -n -e 2p</code>	gibt die zweite Zeile aus
<code> head -n 2 -</code>	gibt die ersten zwei Zeilen aus
<code> tail -n 2 -</code>	gibt die letzten zwei Zeilen aus

Tabelle 1.26: Liste von Skript-Schnipseln für die Befehlsweiterleitung

Ein einzeliliges Shellskript kann durch Verwendung von `find(1)` und `xargs(1)` über Schleifen viele Dateien in Folge abarbeiten und so ziemlich komplexe Aufgaben erledigen. Lesen Sie dazu Abschnitt [10.1.5](#) und Abschnitt [9.3.9](#).

Wenn die Verwendung des interaktiven Shell-Modus zu kompliziert wird, ziehen Sie bitte in Erwägung, ein Shellskript zu schreiben (weiteres dazu in Abschnitt [12.1](#)).

Kapitel 2

Debian-Paketmanagement

Anmerkung

Dieses Kapitel geht davon aus, dass Bullseye die aktuelle stabile Veröffentlichung ist.

[Debian](#) ist eine Organisation Freiwilliger, die **konsistente** Distributionen vorkompilierter Binärpakete von freier Software erstellt und sie über ihr Archiv verteilt.

Das [Debian-Archiv](#) wird von [vielen Spiegel-Servern](#) via HTTP und FTP bereitgestellt. Es ist auch als [CD-ROM/DVD](#) erhältlich.

Debians Paketmanagement-System, ermöglicht dem Nutzer (**wenn es korrekt genutzt wird**), **konsistente Sätze von Binärpaketen** auf dem System zu installieren. Derzeit sind für die amd64-Architektur 62716 Pakete verfügbar.

Das Debian-Paketmanagement-System hat eine reichhaltige Geschichte und bietet eine Auswahl an Frontend-Benutzerprogrammen, die genutzt werden können sowie an verschiedenen Varianten des Archivzugriffs. Derzeit empfehlen wir folgendes:

- `apt(8)` für alle interaktiven Befehlszeilen-Operationen, inklusive Paketinstallationen und -entfernungen sowie Distributions-Upgrades; verfügbar seit Debian Jessie (Debian 8);
- `apt-get(8)` für Paketverwaltungsoperationen mittels Skripten; auch eine Ausweichlösung für den Fall, dass `apt` nicht verfügbar ist (z.B. auf älteren Debian-Systemen);
- `aptitude(8)` für ein interaktives Text-Interface zur Verwaltung der installierten Pakete und für die Suche nach verfügbaren Paketen.

2.1 Grundvoraussetzungen für das Debian-Paketmanagement

2.1.1 Paketkonfiguration

Hier einige wichtige Punkte für die Paketkonfiguration auf einem Debian-System:

- Die händische Konfiguration durch den Systemadministrator wird respektiert. In anderen Worten: das Paketkonfigurationssystem führt keine aufdringliche Konfiguration durch, nur weil es einfach wäre.
 - Jedes Paket enthält sein eigenes Konfigurationsskript mit einer standardisierten Benutzerschnittstelle namens `debconf(7)`, um Sie bei der Erstinstallation des Pakets zu unterstützen.
 - Debian-Entwickler versuchen ihr Bestes, um mittels der Paketkonfigurationsskripte für einen fehlerfreien Ablauf der Paketaktualisierungen zu sorgen.
-

Paket	Popcon	Größe	Beschreibung
apt	V:868, I:999	4299	Advanced Packaging Tool (APT), Frontend für dpkg, das "http"-, "ftp"- und "file"-Methoden für den Zugriff auf das Archiv beherrscht (dazu gehören apt-, apt-get- und apt-cache-Befehle)
aptitude	V:80, I:431	4249	aptitude(8): interaktiver Terminal-basierter Paketmanager
tasksel	V:36, I:975	393	Werkzeug, um Programmgruppen (Tasks) zur Installation auszuwählen (Frontend für APT)
unattended-upgrades	V:324, I:447	325	Erweiterungspaket für APT, um die automatische Installation von Sicherheits-Aktualisierungen zu ermöglichen
dselect	V:3, I:32	2591	Terminal-basierter Paketmanager (früher Standard, Frontend für APT und andere alte Zugriffsmethoden)
dpkg	V:925, I:999	6856	Paketmanagement-System für Debian
synaptic	V:47, I:320	7873	grafischer Paketmanager (GNOME-Frontend für APT)
apt-utils	V:339, I:996	1162	APT-Hilfsprogramme: apt-extracttemplates(1), apt-ftparchive(1) und apt-sortpkgs(1)
apt-listchanges	V:369, I:851	421	Benachrichtigungswerkzeug für die Änderungshistorie von Paketen
apt-listbugs	V:7, I:12	465	Auflistung kritischer Fehler vor jeder APT-Installation
apt-file	V:18, I:78	90	APT-Paketsuch-Werkzeug - Befehlszeilen-Interface
apt-rdepends	V:0, I:6	40	rekursive Auflistung von Paketabhängigkeiten

Tabelle 2.1: Liste von Debians Paketmanagement-Werkzeugen

- Dem Systemadministrator steht die volle Funktionalität der paketierte Software zur Verfügung, aber Funktionen mit Sicherheitsrisiken sind standardmäßig deaktiviert.
- Wenn Sie von Hand einen Dienst aktivieren und so Sicherheitsrisiken hervorrufen, sind Sie selbst für die Beherrschung der Risiken verantwortlich.
- Esoterische Konfigurationen könnten manuell durch den Systemadministrator aktiviert worden sein. Dies kann Behinderungen mit beliebigen, grundlegenden Systemkonfigurations-Programmen verursachen.

2.1.2 Grundsätzliche Vorsichtsmaßnahmen



Warnung

Installieren Sie nicht Pakete aus einer wahllosen Mischung von Debian-Suiten (Stable, Testing, Unstable, Experimental). Sie können dadurch die Paketkonsistenz beschädigen und deren Korrektur erfordert ein tiefes Verständnis der Systemverwaltung wie Compiler-ABI, Bibliotheks-Versionen, Interpreter-Funktionalitäten usw.

Als [Neuling](#) unter den Debian-Systemadministratoren sollten Sie bei der **Stable**-Veröffentlichung bleiben und nur Sicherheitsaktualisierungen einspielen. Das soll heißen, dass Sie einige der folgenden, eigentlich gültigen Aktionen als Vorsichtsmaßnahme besser vermeiden sollten, bis Sie das Debian-System gut verstehen. Hier einige Erinnerungstützen:

- Fügen Sie nicht **testing** oder **unstable** in Ihre `"/etc/apt/sources.list"` ein.
- Vermischen Sie in Ihrer `"/etc/apt/sources.list"` nicht das Standard-Debian mit anderen Nicht-Debian-Archiven wie Ubuntu.
- Erstellen Sie keine `"/etc/apt/preferences"`-Datei.
- Ändern Sie nicht über Konfigurationsdateien das Standardverhalten der Paketmanagement-Werkzeuge, wenn Sie die vollständigen Auswirkungen nicht kennen.

- Installieren Sie nicht irgendwelche Pakete mit `"dpkg -i <irgendein_paket>"`.
- Installieren Sie niemals irgendwelche Pakete mit `"dpkg --force-all -i <irgendein_paket>"`.
- Löschen oder verändern Sie keine Dateien in `"/var/lib/dpkg/"`.
- Überschreiben Sie keine Systemdateien, indem Sie Software-Programme installieren, die direkt aus den Quellen übersetzt wurden.
 - Installieren Sie solche Pakete, falls nötig, in `"/usr/local"` oder `"/opt"`.

Die nicht-kompatiblen Effekte, die obige Aktionen in dem Debian-Paketmanagement-System verursachen können, führen im schlimmsten Falle zu einem nicht mehr verwendbaren System.

Ein ernsthafter Debian-Systemadministrator, der missionskritische Server betreibt, sollte besondere Vorsicht walten lassen.

- Installieren Sie nicht irgendwelche Pakete von Debian, die Sicherheitsaktualisierungen beinhalten, ohne sie unter sicheren Bedingungen sorgfältig mit Ihrer speziellen Konfiguration getestet zu haben.
 - Sie als Systemadministrator sind am Ende für Ihr System verantwortlich.
 - Die lange Stabilitäts-Historie des Debian-Systems ist für sich alleine keine Garantie.

2.1.3 Leben mit den ewigen Aktualisierungen

Trotz meiner obigen Warnungen weiß ich, dass viele Leser dieses Dokuments den Wunsch haben, die `Testing`- oder `Unstable`-Suite von Debian als Haupt-Betriebssystem für ihre **selbst-administrierte Desktop-Umgebung** zu nutzen. Dies liegt daran, dass sie trotz allem sehr gut funktionieren, regelmäßig aktualisiert werden und immer die neuesten Funktionalitäten bieten.



Achtung

Für Ihre **Produktions-Server** wird die `Stable`-Suite inklusive der Sicherheitsaktualisierungen empfohlen. Dasselbe gilt für Desktop-Rechner, für die nur eingeschränkt administrative Betreuung gewährleistet ist.

Alles was Sie tun müssen ist, den Wert für die Distribution in `"/etc/apt/sources.list"` auf den Namen der Suite (`"testing"` oder `"unstable"`) bzw. auf den Codenamen (`"bookworm"` oder `"sid"`) zu ändern. Und schon haben Sie ein Leben mit **ewigen Aktualisierungen**.

Die Verwendung von `Testing` oder `Unstable` bereitet **viel Spaß**, bringt aber auch einige Risiken mit sich. Auch wenn die `Unstable`-Suite die meiste Zeit sehr stabil erscheint, hat es in der Vergangenheit einige Paketprobleme in `Testing` und `Unstable` gegeben, und einige davon waren nicht ganz einfach zu lösen. So etwas könnte **ziemlich schmerzlich** für Sie sein. Manchmal haben Sie für mehrere Wochen ein beschädigtes Paket oder fehlende Funktionalitäten.

Hier einige Ideen, wie Sie eine schnelle und einfache Systemwiederherstellung bei Fehlern in Debian-Paketen gewährleisten können:

- Erstellen Sie ein **Dual-boot**-fähiges System, das Debians `Stable`-Suite auf einer anderen Partition installiert hat.
- Halten Sie die Installations-CD griffbereit, um davon das **Rettungssystem (Rescue mode)** starten zu können.
- Ziehen Sie in Betracht, `apt-listbugs` zu installieren, um vor einer Paketaktualisierung relevante Informationen aus der [Debian-Fehlerdatenbank \(BTS\)](#) einsehen zu können.
- Lernen Sie genug über die Infrastruktur des Paketsystems, um das Problem umgehen zu können.
- Erzeugen Sie eine chroot- oder ähnliche Umgebung und lassen Sie dort das neueste System im Voraus testweise laufen (lesen Sie dazu Abschnitt [9.10](#)).

Falls Sie keine dieser Vorsichtsmaßnahmen durchführen können, sind Sie möglicherweise nicht bereit für die `Testing`- oder `Unstable`-Suite!

Die [Erleuchtung](#), die wir Ihnen mit den folgenden Informationen bieten, bewahrt Sie vor der ewigen [karmischen](#) Qual der Upgrade-[Hölle](#) und ermöglicht Ihnen, das Debian-[Nirvana](#) zu erreichen.

2.1.4 Grundlagen über das Debian-Archiv

Lassen Sie uns das [Debian-Archiv](#) aus der Sicht eines Benutzers betrachten.

Tipp

Die offiziellen Richtlinien für das Debian-Archiv sind im [Debian Policy-Handbuch, Kapitel 2 - The Debian Archive](#) festgeschrieben.

Für typischen HTTP-Zugriff wird das Archiv in der Datei `/etc/apt/sources.list` wie folgt definiert, hier z.B. für das aktuelle **Stable**-System (= **Bullseye**):

```
deb http://deb.debian.org/debian/ bullseye main contrib non-free
deb-src http://deb.debian.org/debian/ bullseye main contrib non-free

deb http://security.debian.org/ bullseye/updates main contrib
deb-src http://security.debian.org/ bullseye/updates main contrib
```

Hier neige ich dazu, den Codenamen `"bullseye"` zu verwenden statt dem Namen der Suite (`stable`), um Überraschungen zu vermeiden, wenn das nächste **Stable**-Release veröffentlicht wird.

Die Bedeutung von `/etc/apt/sources.list` ist in `sources.list(5)` beschrieben; hier die wichtigsten Punkte:

- Die `"deb"`-Zeile definiert die Quelle für Binärpakete.
- Die `"deb-src"`-Zeile definiert die Quelle für Quellpakete.
- Das erste Argument ist die Wurzel-URL für das Debian-Archiv.
- Das zweite Argument ist der Name der Distribution: entweder der Name der Suite oder der Codename.
- Das dritte und die darauf folgenden Argumente sind eine Liste gültiger Namen für Bereiche im Debian-Archiv.

Die `"deb-src"`-Zeilen können ohne Probleme weggelassen werden (oder auskommentiert, indem ein `"#"` an den Anfang der Zeile gestellt wird), wenn nur `aptitude` verwendet wird, was nicht auf Metadaten bezüglich der Quellpakete zugreift. Dies beschleunigt die Aktualisierungen der Paket-Metadaten. Die URL kann u.a. `"http://"`, `"ftp://"` oder `"file://"` sein.

Tipp

Falls in dem obigen Beispiel `"sid"` statt `"bullseye"` benutzt wird, ist die Zeile `"deb: http://security.debian.org/ ..."` für Sicherheitsaktualisierungen in `/etc/apt/sources.list` nicht nötig (weil es kein Archiv für Sicherheitsaktualisierungen für `"Sid"` (**Unstable**) gibt).

Hier ist eine Liste der URL von Debian-Archiv-Seiten und Suite- oder Codenamen, die in der Konfigurationsdatei verwendet werden.



Achtung

Nur die reine **Stable**-Veröffentlichung mit Sicherheitsaktualisierungen bietet beste Stabilität. Zum größten Teil die **Stable**-Veröffentlichung zu verwenden, aber vermischt mit einigen Paketen aus **Testing** oder **Unstable**, birgt ein höheres Risiko, als reines **Unstable** zu nutzen (aufgrund von unpassenden Bibliotheksversionen usw.). Wenn Sie wirklich die neueste Version einiger Programme unter **Stable** benötigen, verwenden Sie bitte Pakete der Dienste [bullseye-updates](#) und <http://backports.debian.org> (lesen Sie dazu Abschnitt [2.7.4](#)). Diese Dienste müssen mit besonderer Sorgfalt genutzt werden.

Archiv-URL	Suite-Name (Codename)	Zweck
http://deb.debian.org/debian/	stable (bullseye)	Stable-(Bullseye-)Veröffentlichung
http://deb.debian.org/debian/	testing (bookworm)	Testing-(Bookworm-)Veröffentlichung
http://deb.debian.org/debian/	unstable (sid)	Unstable-(Sid-)Veröffentlichung
http://deb.debian.org/debian/	experimental	Experimental-Vorveröffentlichung (optional, nur für Entwickler)
http://deb.debian.org/debian/	stable-proposed-updates	Aktualisierungen für die nächste Stable-Zwischenveröffentlichung (optional)
http://security.debian.org/	stable/updates	Sicherheitsaktualisierungen für die Stable-Veröffentlichung (wichtig)
http://security.debian.org/	testing/updates	Sicherheitsaktualisierungen für die Testing-Veröffentlichung (wichtig)
http://deb.debian.org/debian/	bullseye-updates	Kompatible Aktualisierungen für Spam-Filter, IM-Client-Programme usw. für Bullseye
http://deb.debian.org/debian/	bullseye-backports	Neuere rückportierte Pakete für Bullseye (optional)

Tabelle 2.2: Liste von Debian-Archiv-Seiten

**Achtung**

Sie sollten grundsätzlich nur eine der Suiten `stable`, `testing` oder `unstable` in der `"deb"`-Zeile integrieren. Wenn Sie irgendeine Kombination von `stable`, `testing` und `unstable` in der `"deb"`-Zeile verwenden, bremst dies die APT-Programme aus, wobei am Ende aber nur das neueste Archiv verwendet wird. Eine Auflistung mehrerer Einträge macht Sinn, wenn die `"/etc/apt/preferences"`-Datei genutzt wird und dort klare Richtlinien festgelegt sind (lesen Sie dazu Abschnitt 2.7.3).

Tipp

Für das Debian-System mit den Stable- und Testing-Veröffentlichungen ist es eine gute Idee, wie in obigem Beispiel Zeilen mit `"http://security.debian.org/"` in der `"/etc/apt/sources.list"` zu integrieren, um so die Sicherheitsaktualisierungen zu aktivieren.

Anmerkung

Sicherheitsprobleme im Stable-Archiv werden vom Debian-Sicherheits-Team behoben. Dies war immer ziemlich gründlich und zuverlässig. Sicherheitsprobleme im Testing-Archiv werden unter Umständen durch das Debian-Testing-Sicherheits-Team behoben. Aus [verschiedenen Gründen](#) funktioniert dies aber nicht so gründlich wie für Stable und Sie müssen möglicherweise warten, bis die Pakete mit den Fehlerkorrekturen von Unstable nach Testing migriert sind. Sicherheitsprobleme in Unstable werden durch die jeweiligen Paketbetreuer behoben. Aktiv betreute Unstable-Pakete sind für gewöhnlich in einem ziemlich guten Zustand, da Sicherheitskorrekturen von Upstream mit einfließen. In der [Debian Sicherheits-FAQ](#) finden Sie Informationen, wie Debian mit Sicherheitslücken umgeht.

Bereich	Anzahl der Pakete	Kriterien der Paketkomponenten
main	61595	DFSG-konform und keine Abhängigkeiten zu non-free
contrib	349	DFSG-konform, aber mit Abhängigkeiten zu non-free
non-free	772	nicht DFSG-konform

Tabelle 2.3: Liste der Debian-Archiv-Bereiche

Die Paketanzahl in obiger Liste gilt für die amd64-Architektur. Der `main`-Bereich stellt das Debian-System dar (lesen Sie dazu Abschnitt 2.1.5).

Die Struktur des Debian-Archivs können am besten erforschen, indem Sie mit Ihrem Browser die jeweilige Archiv-URL mit einem angehängten `dists` oder `pool` besuchen.

Die Distribution wird auf zwei Arten referenziert, über die Suite oder über den [Codennamen](#). Das Wort "Distribution" wird in vielen Dokumentationen synonym zum Namen der Suite verwendet. Die Beziehung zwischen Suite und Codename kann wie folgt zusammengefasst werden:

Zeitraum	Suite = Stable	Suite = Testing	Suite = Unstable
nach der Veröffentlichung von Bullseye	Codename = Bullseye	Codename = Bookworm	Codename = Sid
nach der Veröffentlichung von Bookworm	Codename = Bookworm	Codename = Trixie	Codename = Sid

Tabelle 2.4: Zusammenhanag zwischen Suite und Codename

Die Historie der Codennamen ist beschrieben in der [Debian FAQ: Kapitel 6.2.1 Welche Codennamen wurden in der Vergangenheit verwendet?](#)

In der strengerer Debian-Archiv-Terminologie wird das Wort "Sektion" (section) speziell für die Kategorisierung der Pakete durch Anwendungen genutzt (obwohl der Begriff "main section" auch verwendet werden könnte, um den Bereich "main" im Debian-Archiv zu beschreiben).

Immer wenn ein Debian-Entwickler (DD, von Debian developer) ein Paket in das **Unstable**-Archiv hochlädt (via [incoming](#)), muss der DD dafür sorgen, dass das hochgeladene Paket mit dem aktuellen Satz der Pakete im **Unstable**-Archiv kompatibel ist.

Falls der DD diese Kompatibilität absichtlich verletzt (für wichtige Bibliotheksaktualisierungen usw.), gibt es gewöhnlich eine Ankündigung dazu auf der [Mailingliste debian-devel](#).

Bevor eine Gruppe von Paketen durch das Wartungsskript des Debian-Archivs von **Unstable** nach **Testing** verschoben wird, prüft das Wartungsskript nicht nur das Alter (mindestens rund 10 Tage alt) und den Status der veröffentlichungskritischen Fehler (RC bugs) für die Pakete, sondern versucht auch die Kompatibilität der Pakete mit dem aktuellen Satz der Pakete im **Testing**-Archiv sicherzustellen. Dieser Prozess macht **Testing** sehr aktuell und nutzbar.

Durch den abgestuften, vom Release-Team gesteuerten Archiv-Freeze-Prozess ist das **Testing**-Archiv bereits gereift, um es auf diesem Wege - mit einigen manuellen Eingriffen - vollständig konsistent und fehlerfrei zu machen. Dann wird die neue **Stable**-Veröffentlichung erzeugt, indem der Codename für das alte **Testing**-Archiv dem neuen **Stable**-Archiv zugewiesen wird; anschließend wird ein neuer Codename für das neue **Testing**-Archiv erstellt. Die anfänglichen Inhalte des neuen **Testing** entsprechen exakt dem des neu veröffentlichten **Stable**.

Sowohl das **Unstable**- wie auch das **Testing**-Archiv können aus unterschiedlichen Gründen unter vorübergehenden Störungen leiden:

- Upload eines beschädigten Pakets in das Archiv (meistens bei **Unstable**);
- Verzögerung, bis die neuen Pakete im Archiv akzeptiert werden (meistens bei **Unstable**);
- Timing-Probleme bei der Archiv-Synchronisation (sowohl bei **Testing** wie auch bei **Unstable**);
- Manuelle Eingriffe in das Archiv wie das Entfernen eines Pakets (überwiegend bei **Testing**).

Falls Sie sich also jemals entscheiden, diese Archive zu verwenden, sollten Sie in der Lage sein, diese Arten von Störungen zu beheben oder zu umgehen.

**Achtung**

Für die ersten Monate nach einer neuen Stable-Veröffentlichung sollten die meisten Desktop-Benutzer das Stable-Archiv inklusive der Sicherheitsaktualisierungen verwenden, selbst wenn sie normalerweise Unstable oder Testing benutzen. In dieser Übergangszeit sind sowohl das Unstable- wie auch das Testing-Archiv für die meisten Leute nicht gut. Das System ist in dieser Zeit nur schwer in lauffähigem Zustand zu halten, da Unstable Fluten von gravierenden Hochrüstungen der Kernpakete ertragen muss, und auch Testing ist nicht von Nutzen, weil sein Inhalt überwiegend dem von Stable entspricht, aber ohne die Unterstützung für Sicherheitsaktualisierungen ([Debian Testing-Security-Ankündigung 12/2008](#)). Nach einigen Monaten könnte das Unstable-Archiv wieder nutzbar sein, wenn Sie vorsichtig sind.

Tipp

Wenn Sie das Testing-Archiv beobachten, stellen Sie fest, dass ein durch die Entfernung eines Pakets verursachtes Problem normalerweise umgangen wird, indem man das entsprechende Paket aus Unstable installiert, welches zwecks Behebung des Fehlers hochgeladen wurde.

Näheres über Begriffsdefinitionen zum Debian-Archiv finden Sie im [Debian Policy-Handbuch](#):

- ["Sections"](#) (Sektionen);
- ["Priorities"](#) (Prioritäten);
- ["Base system"](#) (Grundsystem);
- ["Essential packages"](#) (grundlegende Pakete).

2.1.5 Debian ist zu 100% freie Software

Debian besteht aus folgenden Gründen zu 100% aus freier Software:

- Debian installiert standardmäßig nur freie Software und respektiert so die Freiheit des Benutzers.
- Debian stellt in `main` nur freie Software bereit.
- Debian empfiehlt, nur freie Software aus `main` zu verwenden.
- Kein Paket in `main` hängt von Paketen in `non-free` oder `contrib` ab bzw. empfiehlt diese.

Einige Leute fragen sich, ob sich folgende Fakten widersprechen oder nicht:

- "Debian wird zu 100% frei bleiben" (der erste Punkt des [Debian-Gesellschaftsvertrags](#)).
- Debian-Server beherbergen etliche `non-free`- und `contrib`-Pakete.

Sie widersprechen sich nicht, und zwar aus folgenden Gründen:

- Das Debian-System ist zu 100% frei und seine Pakete werden von Debian-Servern im `main`-Bereich vorgehalten.
- Pakete außerhalb des Debian-Systems befinden sich auf den Debian-Servern in den Bereichen `non-free` und `contrib`.

Dies wird auch präzise unter Punkt 4 und 5 des [Debian-Gesellschaftsvertrags](#) erklärt:

- Unsere Prioritäten sind unsere Anwender und Freie Software

- Wir orientieren uns an den Bedürfnissen unserer Anwender und der Gemeinschaft für Freie Software. Deren Interessen stehen an erster Stelle. Wir werden unsere Nutzer bei ihrer Arbeit mit den verschiedensten Rechnerumgebungen unterstützen. Wir haben nichts gegen unfreie Arbeiten, die darauf abzielen, auf Debian-Systemen verwendet zu werden oder versuchen, eine Gebühr von Personen, die solche Arbeiten erstellen oder verwenden, einzufordern. Wir erlauben anderen, Distributionen zu erstellen, die das Debian-System und andere Arbeiten enthalten, ohne dafür irgendwelche Gebühren zu erheben. Um diese Ziele zu fördern, bieten wir ein integriertes System von hoher Qualität an, das die gerade beschriebene Nutzung nicht durch rechtliche Einschränkungen verhindert.
- Arbeiten, die nicht unseren Standards für Freie Software genügen
 - Wir wissen, dass einige unserer Anwender unbedingt Arbeiten einsetzen müssen, die nicht den Debian-Richtlinien für Freie Software entsprechen. Für solche Arbeiten haben wir die Bereiche `contrib` und `non-free` in unserem Archiv eingerichtet. Die Pakete in diesen Bereichen sind nicht Bestandteil des Debian-Systems, wurden aber trotzdem für den Einsatz mit Debian vorbereitet. Wir empfehlen den CD-Herstellern, die jeweiligen Lizenzbestimmungen der Pakete in diesen Bereichen zu studieren und selbst zu entscheiden, ob sie die Pakete mit ihren CDs verteilen dürfen. Obwohl unfreie Arbeiten nicht Bestandteil von Debian sind, unterstützen wir ihren Einsatz und bieten Infrastruktur für nicht-freie Pakete an (z.B. unsere Fehlerdatenbank und die Mailinglisten).

Benutzer sollten sich der Risiken bewußt sein, die durch die Verwendung von Paketen aus `non-free` und `contrib` entstehen:

- fehlende Freiheit bei diesen Software-Paketen;
- keine Unterstützung von Debian für solche Software-Pakete (Debian kann keine vernünftige Unterstützung für Software bieten, bei der nicht auf den Quellcode zugegriffen werden kann);
- Verunreinigung Ihres zu 100% freien Debian-Systems.

Die [Debian-Richtlinien für Freie Software \(DFSG\)](#) sind der Freie-Software-Standard für [Debian](#). Debian interpretiert "Software" im weitesten Sinne inklusive Dokumentation, Firmware, Logos und künstlerische Arbeiten in einem Paket. Aufgrund dessen sind Debians Freie-Software-Standards sehr streng.

Um diese strengen Freie-Software-Standards für `main` erreichen zu können, hat Debian damals Software-Pakete von Mozilla (wie Firefox, Thunderbird und Seamonkey) [abgewandelt](#), indem deren Logo und einige künstlerische Arbeiten entfernt wurden und sie als Iceweasel, Icedove bzw. Iceape veröffentlicht. Diese Pakete haben mit der Veröffentlichung von Debian Stretch (Debian 9) ihre alten Namen wiederbekommen, nachdem die Lizenzprobleme behoben wurden.

Typische `non-free`- und `contrib`-Pakete enthalten frei verteilbare Pakete der folgenden Typen:

- Dokumentationspakete unter der [GNU-Lizenz für freie Dokumentation](#) mit unveränderlichen Abschnitten wie diejenigen für GCC und Make (meistens zu finden in der Sektion `non-free/doc`).
- Firmware-Pakete, die Binärdaten ohne Quelltext enthalten, wie diejenigen, die in Abschnitt [9.9.6](#) als nicht-frei aufgeführt sind (meistens zu finden in der Sektion `non-free/kernel`).
- Spiele- und Schriftarten-Pakete mit Einschränkungen für die kommerzielle Verwendung und/oder inhaltliche Veränderung.

Bitte beachten Sie, dass die Anzahl der Pakete in `non-free` und `contrib` weniger als 2% der Pakete in `main` beträgt. Den Zugriff auf die `non-free`- und `contrib`-Bereiche des Archivs zu ermöglichen, verschleiert nicht die Quellen der Pakete. Die interaktive Verwendung von `aptitude(8)` im Vollbildmodus erlaubt Ihnen die vollständige Ansicht und Kontrolle darüber, welche Pakete aus welchen Bereichen installiert sind, um Ihr System so frei zu halten, wie Sie möchten.

2.1.6 Paketabhängigkeiten

Das Debian-System stellt einen konsistenten Satz von Binärpaketen bereit und verwendet dafür einen Deklarationsmechanismus für versionierte Binärabhängigkeiten über Felder in der `control`-Datei. Hier eine leicht vereinfachte Definition dieser Felder:

- "Depends" / Hängt ab von

- Depends deklariert eine absolute Abhängigkeit und alle Pakete, die in diesem Feld aufgelistet sind, müssen zur selben Zeit oder im Voraus installiert sein.
- "Pre-Depends" / Hängt ab (vorher) von
 - Pre-Depends ist ähnlich wie Depends, nur dass die vollständige Installation der aufgelisteten Pakete im Voraus erforderlich ist.
- "Recommends" / Empfiehlt
 - Recommends deklariert eine starke, aber nicht absolute Abhängigkeit. Die meisten Benutzer würden das Paket nicht installiert haben wollen, solange nicht auch alle in diesem Feld aufgelisteten Pakete installiert sind.
- "Suggests" / Schlägt vor
 - Suggests deklariert eine schwache Abhängigkeit. Viele Benutzer dieses Pakets würden davon profitieren, auch die in diesem Feld aufgelisteten Pakete zu installieren, aber selbst ohne diese bietet das Paket vernünftige Funktionalität.
- "Enhances" / Wertet auf
 - Enhances deklariert eine schwache Abhängigkeit wie Suggests, nur funktioniert sie in der entgegengesetzten Richtung.
- "Breaks" / Beschädigt
 - Breaks deklariert eine Paket-Inkompatibilität, gewöhnlich mit einer Versionsangabe. Grundsätzlich lösen Sie etwaige Probleme, indem Sie alle in diesem Feld aufgelisteten Pakete auf den neuesten Stand bringen.
- "Conflicts" / Kollidiert
 - Conflicts deklariert eine absolute Inkompatibilität. Alle in diesem Feld aufgelisteten Pakete müssen entfernt werden, um dieses Paket zu installieren.
- "Replaces" / Ersetzt
 - Replaces wird deklariert, wenn Dateien, die von diesem Paket installiert werden, andere Dateien in den aufgelisteten Paketen ersetzen.
- "Provides" / Stellt bereit
 - Provides wird deklariert, wenn dieses Paket alle Dateien und Funktionalitäten der hier aufgelisteten Pakete bereitstellt.

Anmerkung

Bitte beachten Sie, dass die gleichzeitige Deklaration von "Provides", "Conflicts" und "Replaces" für ein virtuelles Paket eine vernünftige Konfiguration ist. So wird sichergestellt, dass nur ein reelles Paket, das dieses virtuelle Paket bereit stellt, zur jeweiligen Zeit installiert sein kann.

Die offizielle Definition inklusive der Quellpaket-Abhängigkeiten finden Sie im [Debian Policy-Handbuch: Kapitel 7 - Declaring relationships between packages](#).

2.1.7 Die Ereignisabfolge für das Paketmanagement

Hier folgt eine vereinfachte Ereignisabfolge für das APT-Paketmanagement.

- **Aktualisierung der Paketlisten / Update** ("apt update", "aptitude update" oder "apt-get update"):
 1. Archiv-Metadaten vom Archiv-Server beziehen;
 2. Neuaufbau und Aktualisierung der lokalen Metadaten für die Verwendung durch APT.
- **Paketaktualisierung / Upgrade** ("apt upgrade" und "apt full-upgrade" oder "aptitude safe-upgrade" und "aptitude full-upgrade" oder "apt-get upgrade" und "apt-get dist-upgrade"):
 1. Auswahl der Installationskandidat-Version für alle installierten Pakete, was üblicherweise die jeweils letzte verfügbare Version ist (in Abschnitt 2.7.3 finden Sie Infos, wie Sie Ausnahmen festlegen);
 2. Auflösung der Paketabhängigkeiten;
 3. Ausgewählte Binärpakete vom Archiv-Server beziehen, falls die jeweilige Installationskandidat-Version sich von der installierten Version unterscheidet;
 4. Entpacken der heruntergeladenen Binärpakete;
 5. Ausführen des **preinst**-Skripts;
 6. Installieren der Binärdateien;
 7. Ausführen des **postinst**-Skripts.
- **Installieren / Install** ("apt install ...", "aptitude install ..." oder "apt-get install ..."):
 1. Auswählen der gewünschten Pakete;
 2. Auflösung der Paketabhängigkeiten;
 3. Beziehen der ausgewählten Binärpakete vom Archiv-Server;
 4. Entpacken der heruntergeladenen Binärpakete;
 5. Ausführen des **preinst**-Skripts;
 6. Installieren der Binärdateien;
 7. Ausführen des **postinst**-Skripts.
- **Entfernen / Remove** ("apt remove ...", "aptitude remove ..." oder "apt-get remove ..."):
 1. Auswählen der gewünschten Pakete;
 2. Auflösung der Paketabhängigkeiten;
 3. Ausführen des **prerm**-Skripts;
 4. Löschen der installierten Dateien **außer** den Konfigurationsdateien;
 5. Ausführen des **postrm**-Skripts.
- **Entfernen inkl. der Konfigurationsdateien / Purge** ("apt purge", "aptitude purge ..." oder "apt-get purge ..."):
 1. Auswählen der gewünschten Pakete;
 2. Auflösung der Paketabhängigkeiten;
 3. Ausführen des **prerm**-Skripts;
 4. Löschen der installierten Dateien **inklusive** der Konfigurationsdateien;
 5. Ausführen des **postrm**-Skripts.

Um des Gesamtbildes willen habe ich hier absichtlich technische Details übersprungen.

2.1.8 Erste Hilfe bei Paketmanagement-Problemen

Sie sollten die offizielle Dokumentation lesen. Das erste Dokument ist hierbei die Debian-spezifische Datei `"/usr/share/doc/<paketname>/"`. Weitere Dokumentation in `"/usr/share/doc/<paketname>/"` sollten Sie ebenfalls konsultieren. Wenn Sie die Shell wie in Abschnitt 1.4.2 beschrieben eingerichtet haben, geben Sie folgendes ein:

```
$ cd <paketname>
$ pager README.Debian
$ mc
```

Sie müssen unter Umständen das entsprechende Dokumentationspaket installieren (hängen Sie dazu `-doc` an den Namen an), um die Dokumentation verfügbar zu haben.

Wenn Sie Probleme mit einem bestimmten Paket haben, sollten Sie unbedingt zuerst die Seiten der [Debian-Fehlerdatenbank \(BTS\)](#) besuchen.

Website	Befehl
Homepage der Debian-Fehlerdatenbank (BTS)	<code>sensible-browser "http://bugs.debian.org/"</code>
Fehlerberichte gegen ein bestimmtes Paket	<code>sensible-browser "http://bugs.debian.org/<paketname>"</code>
Fehlerbericht mit einer bekannten Fehlernummer	<code>sensible-browser "http://bugs.debian.org/<fehlernummer>"</code>

Tabelle 2.5: Liste wichtiger Websites zur Lösung von Paketproblemen

Suchen Sie bei [Google](#) mit Suchbegriffen, die `"site:debian.org"`, `"site:wiki.debian.org"`, `"site:lists.debian.org"` usw. enthalten.

Wenn Sie einen Fehlerbericht einreichen, nutzen Sie bitte den Befehl `reportbug(1)`.

2.2 Grundlegende Paketmanagement-Operationen

Paketmanagement-Operationen können im Debian-System mit vielen, auf APT aufbauenden Werkzeugen durchgeführt werden. Wir beschreiben hier drei grundlegende Paketmanagement-Werkzeuge: `apt`, `apt-get` / `apt-cache` und `aptitude`.

Für Paketmanagement-Operationen inklusive Installation oder Aktualisierung der Paket-Metadaten benötigen Sie root-Privilegien.

2.2.1 apt contra apt-get / apt-cache contra aptitude

Obwohl `aptitude` ein tolles interaktives Werkzeug ist, das auch der Autor überwiegend verwendet, sollten Sie einige warnende Fakten kennen:

- `aptitude` wird nicht für das Upgrade eines Debian-Stable-Systems (Hochrüstung von einer Stable-Version zur nächsten) empfohlen.
 - Dafür empfehlen wir `"apt full-upgrade"` oder `"apt-get dist-upgrade"`. Lesen Sie dazu den [Fehlerbericht #411280](#).
- Manchmal empfiehlt `aptitude` beim System-Upgrade von Testing- oder Unstable-Systemen die massenweise Entfernung von Paketen.
 - Diese Situation hat schon manchen Systemadministrator erschreckt. Aber: keine Panik.
 - Dies wurde scheinbar überwiegend dadurch verursacht, dass eine Schieflage bei den Versionsnummern von Paketen bestand, die von Metapaketten wie `gnome-core` abhängen oder diese empfehlen.

- Sie können dies beheben, indem Sie "Noch ausstehende Aktionen abbrechen" im Menü von `aptitude` wählen, `aptitude` beenden und `apt full-upgrade` verwenden.

Die Befehle `apt-get` und `apt-cache` sind sehr **grundlegende** APT-basierte Paketmanagement-Werkzeuge.

- `apt-get` und `apt-cache` bieten lediglich eine Befehlszeilen-Oberfläche.
- `apt-get` eignet sich sehr gut für die **großen Systemhochrüstungen** von einer Debian-Veröffentlichung auf die nächste, usw.
- `apt-get` bietet einen **robusten** Mechanismus zur Auflösung von Paketabhängigkeiten.
- `apt-get` ist weniger anspruchsvoll bezüglich der Hardware-Ressourcen. Es verbraucht weniger Speicher und läuft schneller.
- `apt-cache` bietet eine **grundlegende**, auf regulären Ausdrücken basierende Suche über Paketname und -beschreibung.
- `apt-get` und `apt-cache` können über `/etc/apt/preferences` mehrere Versionen des gleichen Pakets verwalten, aber dies ist ziemlich mühselig.

Das `apt`-Programm ist eine Befehlszeilenschnittstelle für das Paketmanagement. Es ist vom Grundsatz her ein Wrapper-Skript für `apt-get`, `apt-cache` und ähnliche Programme und als Bedienoberfläche für den Benutzer gedacht; einige für die interaktive Nutzung zweckmäßige Optionen sind standardmäßig aktiviert.

- `apt` bietet eine nette Fortschrittsanzeige bei der Paketinstallation mittels `apt install`.
- Standardmäßig **entfernt** `apt` die zwischengespeicherten `.deb`-Dateien nach der erfolgreichen Installation.

Tipp

Benutzern wird empfohlen, den neuen `apt(8)`-Befehl zu verwenden, der für die **interaktive** Nutzung durch den Benutzer gedacht ist. In Shell-Skripten sollten Sie `apt-get(8)` und `apt-cache(8)` einsetzen.

Der Befehl `aptitude` ist das **vielseitigste** APT-basierte Paketmanagement-Werkzeug.

- `aptitude` bietet eine interaktive textbasierte Benutzeroberfläche mit Vollbildschirm-Ansicht.
- `aptitude` enthält auch eine Befehlszeilen-Benutzerschnittstelle.
- `aptitude` eignet sich sehr gut für das **tägliche interaktive Paketmanagement**, wie die Überprüfung installierter Pakete und die Suche nach verfügbaren Paketen.
- `aptitude` ist anspruchsvoller bezüglich der Hardware-Ressourcen. Es verbraucht mehr Speicher und läuft geringfügig langsamer.
- `aptitude` bietet eine **erweiterte**, auf regulären Ausdrücken basierende Suche über alle Paket-Metadaten.
- `aptitude` kann mehrere Versionen des gleichen Pakets verwalten, ohne dabei `/etc/apt/preferences` zu verwenden, und es ist sehr intuitiv.

2.2.2 Grundlegende Paketmanagement-Operationen auf der Befehlszeile

Hier einige grundlegende Paketmanagement-Operationen auf der Befehlszeile, die `apt(8)`, `aptitude(8)` und `apt-get(8)` / `apt-cache(8)` verwenden.

Anmerkung

Obwohl der `aptitude`-Befehl reichhaltige Funktionen mit sich bringt, wie z.B. seinen erweiterten Paketabhängigkeitsauflöser, hat diese Komplexität einige Rückentwicklungen verursacht (oder könnte sie auch in Zukunft verursachen), wie z.B. [Fehler #411123](#), [Fehler #514930](#) und [Fehler #570377](#). Im Zweifel bevorzugen Sie bitte die `apt-`, `apt-get-` und `apt-cache-`Befehle.

apt-Syntax	aptitude-Syntax	apt-get / apt-cache-Syntax	Beschreibung
apt update	aptitude update	apt-get update	Aktualisieren der Paket-Metadaten
apt install foo	aptitude install foo	apt-get install foo	Installieren der Installationskandidat-Version von "foo" inklusive seiner Abhängigkeiten
apt upgrade	aptitude safe-upgrade	apt-get upgrade	Installieren der Installationskandidat-Version aller installierten Pakete, ohne irgendwelche anderen Pakete zu entfernen
apt full-upgrade	aptitude full-upgrade	apt-get dist-upgrade	Installieren der Installationskandidat-Version aller installierten Pakete und Entfernen anderer Pakete, falls nötig
apt remove foo	aptitude remove foo	apt-get remove foo	Entfernen des Pakets "foo" bei gleichzeitiger Erhaltung seiner Konfigurationsdateien
apt autoremove	Nicht verfügbar	apt-get autoremove	Entfernen aller automatisch installierten, nicht mehr benötigten Pakete
apt purge foo	aptitude purge foo	apt-get purge foo	Vollständiges Entfernen des Pakets "foo" und Löschung seiner Konfigurationsdateien
apt clean	aptitude clean	apt-get clean	Vollständiges Leeren des lokalen Depots heruntergeladener Paketdateien
apt autoclean	aptitude autoclean	apt-get autoclean	Leeren des lokalen Depots heruntergeladener Paketdateien für alte Pakete
apt show foo	aptitude show foo	apt-cache show foo	Anzeigen von detaillierten Informationen über das Paket "foo"
apt search <regex>	aptitude search <regex>	apt-cache search <regex>	Suchen nach Paketen, auf die der reguläre Ausdruck <regex> zutrifft
Nicht verfügbar	aptitude why <regex>	Nicht verfügbar	Erklären, warum die Pakete, auf die der reguläre Ausdruck <regex> zutrifft, installiert werden sollen
Nicht verfügbar	aptitude why-not <regex>	Nicht verfügbar	Erklären, warum die Pakete, auf die der reguläre Ausdruck <regex> zutrifft, nicht installiert werden sollen
Nicht verfügbar	aptitude search '~i!~M'	apt-mark showmanual	Händisch installierte Paket auflisten

Tabelle 2.6: Grundlegende Paketmanagement-Operationen mit apt(8), aptitude(8) und apt-get(8) / apt-cache(8)

Anmerkung

Da `apt` / `apt-get` und `aptitude` den Paket-Status für automatisch installierte Pakete (lesen Sie dazu Abschnitt 2.5.5) seit der Lenny-Veröffentlichung gemeinsam verwalten, können Sie diese beiden Programme ohne größere Probleme im Wechsel verwenden (weiteres dazu unter [Fehler #594490](#)).

Der Befehl `aptitude why <regex>` kann mehr Informationen ausgeben, wenn `-v` verwendet wird (`aptitude -v why <regex>`). Ähnliche Informationen erhalten Sie mit `apt rdepends <paket>` oder `apt-cache rdepends <paket>`.

Wenn `aptitude` im Befehlszeilenmodus gestartet wird und Probleme wie z.B. Paketkonflikte anzeigt, können Sie im Nachhinein in die interaktive Vollbildschirm-Ansicht wechseln, indem Sie am Prompt die Taste `e` drücken.

Befehlsoptionen können direkt im Anschluß an den `aptitude`-Befehl angegeben werden.

Befehlsoptionen	Beschreibung
<code>-s</code>	das Resultat des Befehls simulieren
<code>-d</code>	nur Herunterladen, aber nicht installieren/aktualisieren
<code>-D</code>	kurze Erläuterungen vor automatischen Installationen und Entfernungen anzeigen

Tabelle 2.7: Erwähnenswerte Befehlsoptionen für `aptitude`(8)

Weitere Informationen finden Sie unter `aptitude(8)` und im `aptitude`-Benutzerhandbuch in `/usr/share/doc/aptitude/README`.

Tipp

Das `dselect`-Paket ist immer noch verfügbar; es war in früheren Veröffentlichungen das bevorzugte interaktive Paketmanagement-Werkzeug mit Vollbildschirm-Ansicht.

2.2.3 Interaktive Nutzung von `aptitude`

Für interaktives Paketmanagement starten Sie `aptitude` von der Konsolen-Shell aus im interaktiven Modus mit:

```
$ sudo aptitude -u
Password:
```

Die lokale Kopie der Archivinformationen wird aktualisiert und die Paketliste in der Vollbildschirm-Ansicht angezeigt. `Aptitude` legt seine Konfiguration in `~/.aptitude/config` ab.

Tipp

Wenn Sie `root`'s Konfiguration verwenden wollen statt der des unprivilegierten Benutzers, verwenden Sie in obigem Ausdruck `sudo -H aptitude ...` statt `sudo aptitude ...`.

Tipp

`Aptitude` merkt automatisch **bestimmte Aktionen** vor, wenn es interaktiv gestartet wird. Wenn Sie dies nicht möchten, können Sie sie über das Menü zurücksetzen: `Aktionen` → `Nicht abgeschlossene Aktionen abbrechen`.

Taste	Tastaturbefehl
F10 oder Strg-t	Menü
?	Hilfe für Tastaturkürzel anzeigen (ausführlichere Auflistung)
F10 → Hilfe → Handbuch	Benutzerhandbuch anzeigen
u	Paketinformationen (Metadaten) aktualisieren
+	Paket zur Aktualisierung (upgrade) oder Installation (install) vormerken
-	Paket zum Entfernen (remove) vormerken (Konfigurationsdateien erhalten)
—	Paket zum vollständigen Entfernen (purge) vormerken (Konfigurationsdateien löschen)
=	Paket auf Zurückhalten (hold) setzen
U	Alle aktualisierbaren Pakete markieren (die gleiche Funktion wie full-upgrade)
g	Das Herunterladen und Installieren aller markierten Pakete starten
q	Aktuelle Ansicht schließen und Änderungen speichern
x	Aktuelle Ansicht schließen und Änderungen verwerfen
Enter	Informationen über ein Paket anzeigen
C	Änderungsprotokoll (Changelog) eines Pakets anzeigen
l	Ändern des Filters für die Paketanzeige
/	Suche nach dem ersten Treffer
\	Wiederholen der letzten Suche

Tabelle 2.8: Liste der Tastaturkürzel für aptitude

2.2.4 Tastaturkürzel von aptitude

Es folgen einige erwähnenswerte Tastenkürzel, um in der Vollbildschirm-Ansicht den Status von Paketen abzufragen und "geplante Aktionen" vorzumerken.

Die Angabe des Dateinamens auf der Befehlszeile sowie das Eingabefeld nach dem Drücken von "l" und "/" nutzen aptitudes Regex (regulären Ausdruck) wie unten beschrieben. Damit aptitudes regulärer Ausdruck exakt auf einen Paketnamen zutrifft, können Sie eine Zeichenkette verwenden, die mit ~n beginnt, gefolgt von dem Paketnamen.

Tipp

Im interaktiven Modus müssen Sie "U" drücken, um alle installierten Pakete auf die **Installationskandidat-Version** zu aktualisieren. Andernfalls werden nur die markierten Pakete und solche mit versionierten Abhängigkeiten zu diesen auf die **Installationskandidat-Version** aktualisiert.

2.2.5 Paketansichten in aptitude

In der interaktiven Vollbildschirm-Ansicht von aptitude(8) werden Pakete in der Paketliste wie im folgenden Beispiel angezeigt:

```
idA  libsmclient          -2220kB 3.0.25a-1 3.0.25a-2
```

Diese Zeile hat die folgenden Bedeutungen (von links nach rechts):

- die Markierung für den aktuellen Zustand (der erste Buchstabe);
- die Markierung für die geplante Aktion (der zweite Buchstabe);
- die Markierung für "automatisch installiert" (der dritte Buchstabe);
- der Paketname;
- die voraussichtliche Änderung beim verwendeten Platz auf der Festplatte, ausgelöst durch die geplante Aktion;

- die aktuelle Version des Pakets;
- die Installationskandidat-Version des Pakets.

Tipp

Eine vollständige Liste der Markierungen finden Sie am unteren Ende der **Hilfe**-Ansicht, die Ihnen durch Drücken von "?" angezeigt wird.

Die **Installationskandidat-Version** wird aufgrund der aktuellen lokalen Voreinstellungen ausgewählt (lesen Sie dazu `apt_preferences(5)` und Abschnitt 2.7.3).

Im Menü unter **Ansichten** sind verschiedene Arten von Paketansichten verfügbar:

Ansicht	Status	Beschreibung der Ansicht
Paketansicht	Gut	laut Tabelle 2.10 (Standardeinstellung)
Empfehlungen überprüfen	Gut	Pakete auflisten, die von installierten Paketen empfohlen werden, aber noch nicht installiert sind
Einfache Paketansicht	Gut	Pakete ohne Kategorisierung anzeigen (für Verwendung mit Regex)
Debtags-Browser	Sehr nützlich	Pakete sortiert nach der Kategorisierung in ihren <code>debtags</code> -Einträgen anzeigen
Kategorie-Browser	Veraltet	Pakete sortiert nach ihrer Kategorie anzeigen (nicht mehr empfohlen, verwenden Sie stattdessen den <code>Debtags-Browser</code>)

Tabelle 2.9: Liste der Ansichten für aptitude

Anmerkung

Bitte helfen Sie uns, [Pakete mit debtags zu versehen](#)!

Die Standard-Paketansicht kategorisiert Pakete ähnlich wie `dselect` plus einiger zusätzlicher Funktionalitäten.

Kategorie	Beschreibung der Ansicht
Aktualisierbare Pakete	Pakete organisiert nach Sektion → Bereich → Paket
Neue Pakete	„
Installierte Pakete	„
Nicht installierte Pakete	„
Veraltete und lokal erstellte Pakete	„
Virtuelle Pakete	Pakete mit der gleichen Funktionalität auflisten
Tasks (Programmgruppen)	Pakete mit verschiedenen Funktionalitäten auflisten, die grundsätzlich für eine bestimmte Aufgabe benötigt werden

Tabelle 2.10: Die Kategorisierung von Standard-Paketansichten

Tipp

Die **Tasks**-Ansicht kann verwendet werden, um aus verschiedenen Paketen für eine Aufgabe auszuwählen.

2.2.6 Optionen für Suchmethoden mit aptitude

Aptitude bietet verschiedene Optionen, damit Sie über seine Regex-Formel (regulären Ausdruck) nach Paketen suchen können.

- Shell-Befehlszeile:
 - `"aptitude search <aptitude_regex>"` zur Anzeige von Installationsstatus, Paketname und Kurzbeschreibung passender Pakete
 - `"aptitude show <paketname>"` zur Anzeige detaillierter Informationen für ein Paket
- Interaktive Vollbildschirm-Ansicht:
 - `"l"` zur Einschränkung der Paketansicht auf zutreffende Pakete
 - `"/"` für die Suche nach einem passenden Paket
 - `"\"` für die Rückwärts-Suche nach einem passenden Paket
 - `"n"` zum Finden des nächsten Treffers
 - `"N"` zum Finden des nächsten Treffers in umgekehrter Richtung

Tipp

Die obige Zeichenkette `<paketname>` muss exakt auf den Paketnamen zutreffen, außer sie beginnt mit `"~"`, um anzuzeigen, dass es ein regulärer Ausdruck ist.

2.2.7 Aptitudes Regex-Formel

Die Regex-Formel (regulärer Ausdruck) ist ähnlich wie bei `mutt` ein **erweiterter regulärer Ausdruck (ERE)** (lesen Sie dazu Abschnitt 1.6.2); die Bedeutung der `aptitude`-spezifischen speziellen Regelerweiterungen ist im folgenden erklärt:

- Der Regex-Teil ist der gleiche erweiterte reguläre Ausdruck (**ERE**) wie in Unix-typischen Textverarbeitungswerkzeugen und nutzt `"^"`, `"."`, `"*"`, `"$"` usw., wie es auch `egrep(1)`, `awk(1)` und `perl(1)` tun.
- Der Typ einer Abhängigkeit zur Festlegung von Wechselbeziehungen zwischen Paketen ist einer aus der folgenden Liste: `depends` (hängt ab von), `predepends` (hängt ab (vorher) von), `recommends` (empfiehlt), `suggests` (schlägt vor), `conflicts` (kollidiert mit), `replaces` (ersetzt), `provides` (stellt bereit).
- Der Typ der Standardabhängigkeit ist `"depends"`.

Tipp

Wenn `<regex_muster>` ein Null-String ist, geben Sie `"~T"` direkt hinter dem Befehl an.

Hier einige Abkürzungen:

- `"~P<name>" == "~Dprovides:<name>"`
- `"~C<name>" == "~Dconflicts:<name>"`
- `"...~W name" == "(...|name)"`

Benutzer, denen `mutt` geläufig ist, werden sich auch hier schnell zurechtfinden, da `mutt` die Inspiration für die Syntax des regulären Ausdrucks war. Lesen Sie `"SEARCHING, LIMITING, AND EXPRESSIONS"` im `"Benutzerhandbuch"` unter `"/usr/share/doc/`

Anmerkung

Seit der Lenny-Version von `aptitude(8)` kann für die Suche über reguläre Ausdrücke statt der alten **kurzen Form** (wie z.B. `"~b"`) auch die neue **lange Form** (entsprechend `"?broken"`) verwendet werden. Leerzeichen `" "` werden jetzt als Regex-Abschluß-Zeichen gewertet, zusätzlich zu dem Tilde-Zeichen `"~"`. Weitere Details zur Syntax der neuen **langen Form** finden Sie im Benutzerhandbuch.

Beschreibung der erweiterten Regel	Regex-Formel
Treffer auf Paketname	<code>~n<regex_name></code>
Treffer auf Paketbeschreibung	<code>~d<regex_beschreibung></code>
Treffer auf Task-Name	<code>~t<regex_task></code>
Treffer auf Debtage	<code>~G<regex_debtage></code>
Treffer auf Paketbetreuer	<code>~m<regex_betreuer></code>
Treffer auf Paketsektion	<code>~s<regex_sektion></code>
Treffer auf Paketversion	<code>~V<regex_version></code>
Treffer auf Archiv	<code>~A{bullseye,bookworm,sid}</code>
Treffer auf Paket-Herkunft	<code>~O{debian,...}</code>
Treffer auf Priorität	<code>~p{extra,important,optional,required,standard}</code>
Treffer auf essentielle Pakete	<code>~E</code>
Treffer auf virtuelle Pakete	<code>~v</code>
Treffer auf neue Pakete	<code>~N</code>
Treffer auf noch nicht abgeschlossene Aktion	<code>~a{install,upgrade,downgrade,remove,purge,hold,keep}</code>
Treffer auf installierte Pakete	<code>~i</code>
Treffer auf Pakete mit der A -Markierung (automatisch installierte Pakete)	<code>~M</code>
Treffer auf Pakete ohne A -Markierung (vom Administrator manuell zur Installation ausgewählte Pakete)	<code>~i!~M</code>
Treffer auf installierte und aktualisierbare Pakete	<code>~U</code>
Treffer auf entfernte Pakete, deren Konfigurationsdateien aber noch vorhanden sind	<code>~C</code>
Treffer auf Pakete, die entfernt oder vollständig inklusive der Konfigurationsdateien entfernt wurden oder entfernt werden können	<code>~g</code>
Treffer auf Pakete, die eine beschädigte Abhängigkeit melden	<code>~b</code>
Treffer auf Pakete, die eine beschädigte Abhängigkeit vom Typ <code><typ></code> melden	<code>~B<typ></code>
Treffer auf Pakete, die eine Abhängigkeit vom Typ <code><typ></code> definiert haben	<code>~D[<typ>:]<muster></code>
Treffer auf Pakete, die eine beschädigte Abhängigkeit vom Typ <code><typ></code> definiert haben	<code>~DB[<typ>:]<muster></code>
Treffer auf Pakete, zu denen auf <code><muster></code> passende Pakete die Abhängigkeit <code><typ></code> definiert haben	<code>~R[<typ>:]<muster></code>
Treffer auf Pakete, zu denen auf <code><muster></code> passende Pakete die beschädigte Abhängigkeit <code><typ></code> definiert haben	<code>~RB[<typ>:]<muster></code>
Treffer auf Pakete, von denen andere installierte Pakete abhängig sind	<code>~R~i</code>
Treffer auf Pakete, von denen keine anderen installierten Pakete abhängig sind	<code>!~R~i</code>
Treffer auf Pakete, von denen andere installierte Pakete abhängig sind oder empfohlen werden	<code>~R~i ~Rrecommends:~i</code>
Treffer auf Pakete, die auf <code><muster></code> passen, mit gefilterter Version	<code>~S filter <muster></code>
Treffer auf alle Pakete (true/wahr)	<code>~T</code>
Treffer auf kein Paket (false/unwahr)	<code>~F</code>

Tabelle 2.11: Liste von aptitudes Regex-Formeln

2.2.8 Abhängigkeitsauflösung bei aptitude

Die Auswahl eines Pakets in `aptitude` bringt nicht nur Pakete mit, die in seiner `Depends`:-Liste stehen, sondern auch solche in `Recommends`:-, falls die Einstellung im Menü unter `F10` → `Optionen` → `Einstellungen` → `Abhängigkeitsbehandlung` entsprechend gesetzt ist. Diese automatisch installierten Pakete werden automatisch entfernt, wenn sie laut `aptitude` nicht mehr benötigt werden.

Der Schalter, der das `auto install`-Verhalten von `aptitude` steuert, kann auch über den Befehl `apt-mark(8)` aus dem `apt`-Paket verändert werden.

2.2.9 Protokollierung der Paketaktivitäten

Sie können vergangene Paketaktivitäten in den Protokolldateien nachkontrollieren.

Datei	Inhalt
<code>/var/log/dpkg.log</code>	Protokollierung der Operationen auf <code>dpkg</code> -Ebene für alle Paketaktivitäten
<code>/var/log/apt/term.log</code>	Protokollierung der grundlegenden APT-Aktivitäten
<code>/var/log/aptitude</code>	Protokollierung der Aktivitäten des <code>aptitude</code> -Befehls

Tabelle 2.12: Protokolldateien für Paketaktivitäten

In der Realität ist es nicht so einfach, aus diesen Protokollen auf die Schnelle aussagekräftige Informationen herauszuziehen. Abschnitt [9.2.10](#) gibt Ihnen hilfreiche Infos, wie Sie dies einfacher bewältigen können.

2.3 Beispiele für aptitude-Operationen

Hier ein paar Beispiele für `aptitude(8)`-Operationen.

2.3.1 Auflisten von Paketen mit Regex-Suche auf den Paketnamen

Der folgende Befehl listet Pakete auf, bei denen die Regex-Suche nach dem Paketnamen zutrifft:

```
$ aptitude search '~n(pam|nss).*ldap'
p libnss-ldap - NSS module for using LDAP as a naming service
p libpam-ldap - Pluggable Authentication Module allowing LDAP interfaces
```

Dies ist für Sie ziemlich praktisch, um den exakten Namen eines Pakets zu finden.

2.3.2 Durchsuchen mit der Regex-Suche

Wenn Sie den regulären Ausdruck `~dipv6` in der `Neuen einfachen Paketansicht` im `l`-Dialog eingeben, werden die angezeigten Pakete auf diejenigen eingeschränkt, bei denen der Ausdruck auf die Paketbeschreibung zutrifft, und Sie können deren Informationen interaktiv durchsuchen.

2.3.3 Entfernte Pakete endgültig löschen

Sie können alle verbliebenen Konfigurationsdateien entfernter Pakete endgültig löschen.

Überprüfen Sie die Resultate des folgenden Befehls:

```
# aptitude search '~c'
```

Wenn Sie glauben, dass die aufgelisteten Pakete vollständig entfernt werden können, führen Sie folgenden Befehl aus:

```
# aptitude purge '~c'
```

Sie möchten das gleiche vielleicht im interaktiven Modus durchführen, um eine detailgenaue Kontrolle über den Vorgang zu haben:

Sie geben den regulären Ausdruck `"~c"` unter der "Neuen Paketansicht" in dem `"l"`-Dialog ein. Dadurch werden die angezeigten Pakete auf solche eingeschränkt, auf die der reguläre Ausdruck zutrifft, d.h. "gelöscht, aber nicht vollständig inklusive der Konfigurationsdateien entfernt". All diese, auf den regulären Ausdruck zutreffenden Pakete können angezeigt werden, indem Sie auf der höchsten Ebene der Anzeigehierarchie (z.B. "Nicht installierte Pakete") `"["` drücken.

Dann drücken Sie auf einem Eintrag in der höchsten Anzeigehierarchie (z.B. auf "Nicht installierte Pakete") `"]"`. Nur Pakete, auf die der reguläre Ausdruck passt, werden auf diese Art vollständig inklusive der Konfigurationsdateien entfernt. Sie können jedes dieser Pakete interaktiv von der vollständigen Entfernung ausschließen, indem Sie auf dem entsprechenden Paket `"="` drücken.

Diese Methode ist ziemlich praktisch und funktioniert für viele andere Befehlskürzel.

2.3.4 Automatisch/manuell-Installationsstatus bereinigen

Hier beschreibe ich, wie Sie den automatisch/manuell-Installationsstatus bereinigen können (nach der Verwendung von anderen Paketinstallationsprogrammen o.ä.):

1. Starten Sie `aptitude` im interaktiven Modus als root.
2. Drücken Sie `"u"`, `"U"`, `"f"` und `"g"`, um die Paketliste sowie die Pakete zu aktualisieren.
3. Drücken Sie `"l"` und setzen Sie den Anzeigefilter auf `"~i(~R~i|~Rrecommends:~i)"`; drücken Sie dann `"M"` mit der Markierung auf `"Installierte Pakete"`, um deren Status auf `"Automatisch installiert"` zu ändern.
4. Drücken Sie `"l"`, setzen Sie den Anzeigefilter auf `"~prequired|~pimportant|~pstandard|~E"` und drücken Sie `"m"` mit der Markierung auf `"Installierte Pakete"`, um deren Status auf `"Manuell installiert"` zu ändern.
5. Drücken Sie `"l"`, setzen Sie den Anzeigefilter auf `"~i!~M"` und entfernen Sie nicht genutzte Pakete, indem Sie `"-"` auf jedem davon drücken (wenn die Markierung auf `"Installierte Pakete"` steht, können Sie mit `"["` die Liste aufklappen, so dass alle Pakete sichtbar werden).
6. Drücken Sie `"l"` um den Anzeigefilter auf `"~i"` zu setzen; drücken Sie dann `"m"`, während die Markierung auf `"Tasks"` steht, um diese Pakete auf `"Manuell installiert"` zu setzen.
7. Beenden Sie `aptitude`.
8. Führen Sie `"apt-get -s autoremove|less"` als root aus, um zu überprüfen, welche Pakete derzeit von `apt-get` als ungenutzt gemeldet werden.
9. Starten Sie `aptitude` erneut im interaktiven Modus und markieren Sie Pakete, bei denen dies nötig ist, mit `"m"` als `"Manuell installiert"`.
10. Führen Sie erneut `"apt-get -s autoremove|less"` als root aus, um erneut zu überprüfen, ob bei `ENTFERNT` nur die gewünschten Pakete enthalten sind.
11. Führen Sie `"apt-get autoremove|less"` als root aus, um nicht genutzte Pakete automatisch zu entfernen.

Die Aktion `"m"` mit der Markierung auf `"Tasks"` ist eine optionale Aktion, um die Situation massenhaft zu entfernender Pakete in der Zukunft zu vermeiden.

2.3.5 Systemweite Hochrüstung

Anmerkung

Beim Wechsel auf eine neue Veröffentlichung sollten Sie eine Neuinstallation in Erwägung ziehen, auch wenn Debian wie unten beschrieben auch auf die neue Version hochgerüstet werden kann. So haben Sie eine Chance, Müll, der sich über die Dauer der Zeit angesammelt hat, loszuwerden und bekommen die beste Kombination der aktuellsten Pakete. Natürlich sollten Sie ein vollständiges System-Backup machen und an einem sicheren Platz ablegen (lesen Sie dazu Abschnitt 10.2), bevor Sie neu installieren. Ich empfehle für einen sanften Übergang eine Dual-Boot-Installation unter Verwendung separater Partitionen.

Sie können eine systemweite Hochrüstung auf eine neue Veröffentlichung durchführen, indem Sie die Inhalte der Datei `"/etc/apt/sources.list"` ändern, so dass sie auf die neue Veröffentlichung zeigt; führen Sie danach `"apt update; apt dist-upgrade"` aus.

Um von `stable` nach `testing` bzw. `unstable` zu aktualisieren, ersetzen Sie `"bullseye"` in dem `"/etc/apt/sources.list"` Beispiel aus Abschnitt 2.1.4 durch `"bookworm"` bzw. `"sid"`.

In der Realität könnte es vielleicht einige Komplikationen aufgrund von Problemen mit Paketübergängen geben, meistens wegen Paketabhängigkeiten. Je größer die Unterschiede bei der Hochrüstung sind, um so wahrscheinlicher werden Sie gravierendere Probleme bekommen. Für die Hochrüstung des alten `Stable` auf das neue `Stable` nach dessen Veröffentlichung können Sie die neuen [Veröffentlichungshinweise \(Release Notes\)](#) lesen. Folgen Sie exakt der dort beschriebenen Prozedur, um die möglichen Probleme zu minimieren.

Wenn Sie sich entscheiden, vor der offiziellen Freigabe der nächsten Veröffentlichung von `Stable` zu `Testing` zu wechseln, gibt es keine [Veröffentlichungshinweise](#), die Ihnen helfen könnten. Die Unterschiede zwischen `Stable` und `Testing` könnten seit der letzten `Stable`-Veröffentlichung ziemlich groß geworden sein und die Situation für eine Hochrüstung recht komplex machen.

Sie sollten vorbeugende Schritte für das vollständige Upgrade durchführen, indem Sie aktuellste Informationen von den Mailinglisten sammeln, und Ihren gesunden Menschenverstand einsetzen.

1. Lesen Sie die vorherigen "Veröffentlichungshinweise (Release Notes)".
 2. Machen Sie eine vollständige Sicherung (Backup) von Ihrem System (speziell Daten und Konfigurationsdateien).
 3. Halten Sie ein boot-fähiges Medium bereit für den Fall eines beschädigten Bootloaders.
 4. Informieren Sie die Benutzer des Systems rechtzeitig vorher.
 5. Zeichnen Sie die Aktivitäten während des Upgrades mit `script(1)` auf.
 6. Wenden Sie `"unmarkauto"` für erforderliche Pakete an, also z.B. `"aptitude unmarkauto vim"`, um solche Pakete als "Manuell installiert" zu markieren und zu verhindern, dass sie entfernt werden.
 7. Minimieren Sie die Anzahl installierter Pakete, um die Wahrscheinlichkeit von Paketkonflikten zu reduzieren, entfernen Sie z.B. Desktop-Task-Pakete.
 8. Entfernen Sie die Datei `"/etc/apt/preferences"` (um apt-pinning zu deaktivieren).
 9. Versuchen Sie, das Upgrade in mehreren Schritten durchzuführen: `oldstable` → `stable` → `testing` → `unstable`.
 10. Aktualisieren Sie die Datei `"/etc/apt/sources.list"`, so dass sie nur auf das neue Archiv verweist, und führen Sie `"aptitude update"` aus.
 11. Installieren Sie neue **Kern-Pakete** separat im Voraus, z.B. `"aptitude install perl"` (optional).
 12. Führen Sie `"apt-get -s dist-upgrade"` aus, um die Auswirkungen im Voraus abschätzen zu können.
 13. Führen Sie als letztes den Befehl `"apt-get dist-upgrade"` aus.
-

**Achtung**

Es ist nicht klug, bei Hochrüstung einer `Stable`-Veröffentlichung auf ein anderes `Stable` ein oder mehrere Debian-Hauptveröffentlichungen zu überspringen (also z.B. von Debian 5.0 direkt auf 7.0 hochzurüsten und dabei 6.0 zu überspringen).

**Achtung**

In früheren "Veröffentlichungshinweisen" wurden für GCC, Linux-Kernel, `initrd`-Werkzeuge, Glibc, Perl, APT-Werkzeuge usw. spezielle Vorsichtsmaßnahmen bei systemweiten Upgrades erwähnt.

Informationen zu täglichen Upgrades in `Unstable` finden Sie in Abschnitt [2.4.3](#).

2.4 Erweiterte Paketmanagement-Operationen

2.4.1 Erweiterte Paketmanagement-Operationen auf der Befehlszeile

Hier eine Liste weiterer Paketmanagement-Operationen, die `aptitude` nicht unterstützt, weil es auf einer anderen Ebene arbeitet oder weil ihm die nötigen Funktionen fehlen.

Anmerkung

Für ein Paket mit der [multi-arch](#)-Funktionalität müssen Sie unter Umständen bei einigen Befehlen den Architekturnamen mit angeben. Verwenden Sie zum Beispiel `dpkg -L libglib2.0-0:amd64`, um die Inhalte des Pakets `libglib2.0-0` für die `amd64`-Architektur anzuzeigen.

**Achtung**

Grundlegende Paketwerkzeuge wie `dpkg -i ...` und `debi ...` sollten vom Systemadministrator mit Vorsicht eingesetzt werden. Erforderliche Paketabhängigkeiten werden dabei nicht automatisch aufgelöst. `Dpkg`'s Befehlszeilenoption `--force-all` und ähnliche (schauen Sie in `dpkg(1)`) dürfen nur von Experten eingesetzt werden. Sie zu verwenden ohne deren Auswirkungen vollständig zu verstehen könnte das komplette System beschädigen.

Bitte beachten Sie folgendes:

- Alle System-Konfigurations- und Installationsbefehle müssen von `root` ausgeführt werden.
- Anders als `aptitude`, das reguläre Ausdrücke verwendet (mehr dazu in Abschnitt [1.6.2](#)), nutzen andere Paketmanagement-Befehle Suchmuster wie Shell-Globs (Details in Abschnitt [1.5.6](#)).
- Für `apt-file(1)` aus dem `apt-file`-Paket muss zuvor `apt-file update` ausgeführt werden.
- `configure-debian(8)` aus dem `configure-debian`-Paket führt `dpkg-reconfigure(8)` als Backend im Hintergrund aus.
- `dpkg-reconfigure(8)` führt Paketskripte aus, die `debconf(1)` als Backend im Hintergrund verwenden.
- Die Befehle `apt-get build-dep`, `apt-get source` und `apt-cache showsrc` erfordern einen `deb-src`-Eintrag in `/etc/apt/sources.list`.
- `dget(1)`, `debuild(1)` und `debi(1)` erfordern die Installation des `devscripts`-Pakets.
- Infos zur Prozedur des (Neu-)Paketierens mittels `apt-get source` finden Sie in Abschnitt [2.7.13](#).
- Der `make-kpkg`-Befehl erfordert die Installation des `kernel-package`-Pakets (lesen Sie Abschnitt [9.9](#)).
- Grundsätzliche Informationen über das Paketieren finden Sie in Abschnitt [12.11](#).

Befehl	Aktion
COLUMNS=120 dpkg -l <paketnamen_muster>	Status eines installierten Pakets für einen Fehlerbericht auflisten
dpkg -L <paketname>	Inhalte eines installierten Pakets auflisten
dpkg -L <paketname> egrep '/usr/share/man/man.*/.+'	Handbuchseiten (manpages) für ein installiertes Paket auflisten
dpkg -S <dateinamen_muster>	Installierte Pakete auflisten, die auf das Muster passende Dateien enthalten
apt-file search <dateinamen_muster>	Pakete im Archiv auflisten, die auf das Muster passende Dateien enthalten
apt-file list <paketnamen_muster>	Inhalte von auf das Muster passenden Paketen im Archiv auflisten
dpkg-reconfigure <paketname>	Dieses Paket neu konfigurieren
dpkg-reconfigure -p=low <paketname>	Dieses Paket mit detaillierten Fragen neu konfigurieren
configure-debian	Pakete über das Vollbildschirm-Menü neu konfigurieren
dpkg --audit	System nach teilweise installierten Paketen durchsuchen
dpkg --configure -a	Alle teilweise installierten Pakete konfigurieren
apt-cache policy <binärpaketname>	Verfügbare Version, Priorität und Archivinformationen eines Binärpakets anzeigen
apt-cache madison <paketname>	Verfügbare Version und Archivinformationen eines Pakets anzeigen
apt-cache showsrc <binärpaketname>	Quellpaket-Informationen eines Binärpakets anzeigen
apt-get build-dep <paketname>	Zum Bau eines Pakets benötigte Pakete installieren
aptitude build-dep <paketname>	Zum Bau eines Pakets benötigte Pakete installieren
apt-get source <paketname>	Ein Quellpaket herunterladen (aus dem Standardarchiv)
dget <URL für dsc-Datei>	Ein Quellpaket herunterladen (aus einem anderen Archiv)
dpkg-source -x <paketname>_<version>-<debian_version> -<source_archive>.tar.gz und "/*.debian.tar.gz" / "/*.diff.gz")	Einen Quellcodebaum aus einem Satz von Quellpaketen erstellen
debuild binärdatei	Paket(e) aus einem lokalen Quellcodebaum bauen
make-kpkg kernel_image	Ein Kernelpaket aus einem Kernel-Quellcodebaum bauen
make-kpkg --initrd kernel_image	Ein Kernelpaket mit aktivierter initramfs aus einem Kernel-Quellcodebaum bauen
dpkg -i <paketname>_<version>-<debian_version>_<arch>.deb	Ein lokales Paket in das System installieren
apt install /pfad/zu/<paketname>.deb	Ein lokales Paket in das System installieren und dabei versuchen, die Abhängigkeiten automatisch aufzulösen
debi <paketname>_<version>-<debian_version>_<arch>.dsc	Lokale(s) Paket(e) in das System installieren
dpkg --get-selections '*'> selection.txt	dpkg-Paketauswahlinformationen sichern
dpkg --set-selections <selection.txt	dpkg-Paketauswahlinformationen setzen
echo <paketname> hold dpkg --set-selections	dpkg-Paketauswahlinformationen eines Pakets auf hold (halten) setzen (gleichbedeutend mit "aptitude hold <paketname>")

Tabelle 2.13: Liste erweiterter Paketmanagement-Operationen

2.4.2 Verifizierung von installierten Paketdateien

Die Installation von `debsums` ermöglicht die Verifizierung installierter Pakete über MD5sum-Werte aus der Datei `"/var/lib/dpkg/` mittels `debsums(1)`. Details darüber, wie MD5sum arbeitet, finden Sie in Abschnitt [10.3.5](#).

Anmerkung

Da die MD5sum-Datenbank durch Eindringlinge gefälscht werden könnte, ist `debsums(1)` als Werkzeug für die Systemsicherheit nur von begrenztem Nutzen. Es eignet sich aber gut dafür, lokale Veränderungen durch den Administrator oder durch Beschädigungen aufgrund von Fehlern des Speichermediums zu erkennen.

2.4.3 Absicherungen für den Fall von Paketproblemen

Viele Benutzer bevorzugen die **Unstable**-Veröffentlichung des Debian-Systems wegen seiner neuen Funktionen und Pakete. Dies macht deren System jedoch anfällig für kritische Paketfehler.

Die Installation des `apt-listbugs`-Pakets schützt Ihr System vor kritischen Fehlern, indem bei Upgrades durch das APT-System automatisch die Debian-Fehlerdatenbank (BTS) bezüglich kritischer Fehler abgefragt wird.

Die Installation des `apt-listchanges`-Pakets stellt Ihnen wichtige Neuigkeiten aus `"NEWS.Debian"` zur Verfügung, wenn Upgrades durch das APT-System durchgeführt werden.

2.4.4 Durchsuchen der Paket-Metadaten

Obwohl der Besuch der Debian-Site <https://packages.debian.org/> heutzutage einfache Möglichkeiten bietet, die Paket-Metadaten zu durchsuchen, wollen wir uns auch die traditionellen Wege anschauen.

Die Befehle `grep-dctrl(1)`, `grep-status(1)` und `grep-available(1)` können verwendet werden, um jegliche Datei zu durchsuchen, die das grundsätzliche Format einer control-Datei für ein Debian-Paket hat.

`"dpkg -S <dateinamen_muster>"` kann verwendet werden, um durch `dpkg` installierte Pakete zu finden, die auf das Suchmuster passende Dateinamen enthalten. Hiermit werden jedoch keine Dateien gefunden, die durch Skripte der Paketbetreuer erzeugt wurden.

Wenn Sie eine tiefergehende Suche der `dpkg`-Metadaten benötigen, müssen Sie den Befehl `"grep -e regex_muster *"` im Verzeichnis `"/var/lib/dpkg/info/"` ausführen. So können Sie Wörter finden, die in Paketskripten und Texten für Abfragen bei der Installation auftauchen.

Wenn Sie Paketabhängigkeiten rekursiv abfragen möchten, sollten Sie `apt-rdepends(8)` verwenden.

2.5 Internas des Debian-Paketmanagements

Lassen Sie uns betrachten, wie das Debian-Paketmanagement-System intern funktioniert. Dies sollte Ihnen dabei helfen, eigene Lösungen für gewisse Paketprobleme zu finden.

2.5.1 Archiv-Metadaten

Metadaten-Dateien für die jeweilige Distribution sind auf jedem Debian-Spiegel unter `"dist/<codename>"` abgelegt, z.B. auf `"http://deb.debian.org/debian/"`. Sie können die Archivstruktur mit einem Webbrowser durchsuchen. Es gibt sechs Arten von wichtigen Metadaten:

Im aktuellen Archiv sind diese Metadaten als komprimierte und differenzielle Dateien abgelegt, um Netzwerkverkehr zu reduzieren.

Datei	Speicherort	Inhalt
Release	Wurzelverzeichnis der Distribution	Archivbeschreibung und Integritätsinformationen
Release.gpg	Wurzelverzeichnis der Distribution	Signaturdatei für die "Release"-Datei, mit dem Archiv-Schlüssel signiert
Contents-<architektur>	Wurzelverzeichnis der Distribution	Liste aller Dateien für alle Pakete in dem entsprechenden Archiv
Release	Wurzelverzeichnis aller Distributions-/Bereichs-/Architektur-Kombinationen	Archivbeschreibung, die zur Festlegung von apt_preferences(5) verwendet wird
Packages	Wurzelverzeichnis aller Distributions-/Bereichs-/Binärarchitektur-Kombinationen	vereinigte debian/control für Binärpakete
Sources	Wurzelverzeichnis aller Distributions-/Bereichs-/Quellen-Kombinationen	vereinigte debian/control für Quellpakete

Tabelle 2.14: Inhalt der Metadaten des Debian-Archivs

2.5.2 "Release"-Datei im Wurzelverzeichnis und Authentizität

Tipp

Die "Release"-Datei im Wurzelverzeichnis wird verwendet, um die Archive im **Secure-APT**-System zu signieren.

Jede Suite im Debian-Archiv hat im Wurzelverzeichnis eine "Release"-Datei, z.B. "<http://deb.debian.org/debian/dists>" wie hier:

```
Origin: Debian
Label: Debian
Suite: unstable
Codename: sid
Date: Sat, 14 May 2011 08:20:50 UTC
Valid-Until: Sat, 21 May 2011 08:20:50 UTC
Architectures: alpha amd64 armel hppa hurd-i386 i386 ia64 kfreebsd-amd64 kfreebsd-i386 mips ←
               mipsel powerpc s390 sparc
Components: main contrib non-free
Description: Debian x.y Unstable - Not Released
MD5Sum:
bdc8fa4b3f5e4a715dd0d56d176fc789 18876880 Contents-alpha.gz
9469a03c94b85e010d116aeeab9614c0 19441880 Contents-amd64.gz
3d68e206d7faa3aded660dc0996054fe 19203165 Contents-armel.gz
...
```

Anmerkung

Hier finden Sie den Grund dafür, warum ich in Abschnitt 2.1.4 "Suite" und "Codename" verwende. Die "Distribution" wird benutzt, um sowohl auf "Suite" wie auch auf "Codename" zu verweisen. Alle Namen der Archiv-Bereiche, die von dem Archiv angeboten werden, sind unter "Components" aufgelistet.

Die Integrität der "Release"-Datei im Wurzelverzeichnis wird über die kryptografische Infrastruktur namens [SecureApt](#) verifiziert:

- Die kryptografische Signaturdatei "Release.gpg" wird aus der authentischen "Release"-Datei im Wurzelverzeichnis und dem geheimen Debian-Archiv-Schlüssel erzeugt.

- Der öffentliche Debian-Archiv-Schlüssel kann in `"/etc/apt/trusted.gpg"` abgelegt werden:
 - automatisch mittels Installation des Schlüsselrings über das aktuellste `base-files`-Paket, oder
 - manuell über das `gpg`- oder `apt-key`-Werkzeug mit dem [aktuellsten öffentlichen Archiv-Schlüssel, veröffentlicht auf ftp-master.debian.org](http://ftp-master.debian.org).
- Das **Secure-APT**-System verifiziert kryptografisch die Integrität der heruntergeladenen "Release"-Datei im Wurzelverzeichnis über diese "Release.gpg"-Datei und den öffentlichen Debian-Archiv-Schlüssel in `"/etc/apt/trusted.gpg"`.

Die Integrität all der "Packages"- und "Sources"-Dateien wird über MD5sum-Werte in der "Release"-Datei im Wurzelverzeichnis verifiziert. Die Integrität der Paketdateien wird über MD5sum-Werte in den "Packages"- und "Sources"-Dateien verifiziert. Lesen Sie dazu `debsums(1)` und Abschnitt [2.4.2](#).

Da die kryptografische Signaturverifizierung viel CPU-intensiver ist als die Berechnung von MD5sum-Werten, bietet die Verwendung von MD5sum-Werten für die Pakete bei gleichzeitiger Nutzung einer kryptografischen Signatur für die "Release"-Datei im Wurzelverzeichnis [einen guten Kompromiss zwischen Sicherheit und Performance](#) (weiteres in Abschnitt [10.3](#)).

2.5.3 "Release"-Dateien im Archivverzeichnis

Tipp

Die "Release"-Dateien im Archivverzeichnis werden zur Festlegung von `apt_preferences(5)` genutzt.

Es gibt "Release"-Dateien in allen Archivverzeichnissen, die über "deb"-Zeilen in `"/etc/apt/sources.list"` definiert sind, z.B. `"http://deb.debian.org/debian/dists/unstable/main/binary-amd64/Release"` oder `"http://deb.debian.org/debian/dists/sid/main/binary-amd64/Release"`, wie hier:

```
Archive: unstable
Origin: Debian
Label: Debian
Component: main
Architecture: amd64
```



Achtung

Für "Archive:" werden im [Debian-Archiv](#) Suite-Namen ("stable", "testing", "unstable", ...) verwendet, im [Ubuntu-Archiv](#) jedoch Codenamen ("trusty", "xenial", "artful", ...).

Für einige Archive (wie `experimental` und `bullseye-backports`), die Pakete enthalten, welche nicht automatisch installiert werden sollten, z.B. `"http://deb.debian.org/debian/dists/experimental/main/binary-amd64/Release"` existiert eine zusätzliche Zeile, wie hier:

```
Archive: experimental
Origin: Debian
Label: Debian
NotAutomatic: yes
Component: main
Architecture: amd64
```

Bitte beachten Sie, dass für normale Archive ohne `"NotAutomatic: yes"` der Wert für die Pin-Priorität 500 ist, während für besondere Archive mit `"NotAutomatic: yes"` dieser Wert 1 ist (lesen Sie dazu `apt_preferences(5)` und Abschnitt [2.7.3](#)).

2.5.4 Empfangen der Metadaten für ein Paket

Um APT-Werkzeuge wie `aptitude`, `apt-get`, `synaptic`, `apt-file`, `auto-apt` ... zu verwenden, müssen wir die lokalen Kopien der Metadaten, die die Debian-Archivinformationen enthalten, aktualisieren. Diese lokalen Kopien haben die folgenden Dateinamen, passend zu den in `/etc/apt/sources.list` festgelegten Namen für `Distribution`, `Bereich` und `Architektur` (weiteres in Abschnitt 2.1.4):

- `/var/lib/apt/lists/deb.debian.org_debian_dists_<distribution>_Release`
- `/var/lib/apt/lists/deb.debian.org_debian_dists_<distribution>_Release.gpg`
- `/var/lib/apt/lists/deb.debian.org_debian_dists_<distribution>_<area>_binary-<architektur>`
- `/var/lib/apt/lists/deb.debian.org_debian_dists_<distribution>_<bereich>_source_Sources`
- `/var/cache/apt/apt-file/deb.debian.org_debian_dists_<distribution>_Contents-<architektur>`
(für `apt-file`)

Die ersten vier Typen obiger Dateien werden von allen entsprechenden APT-Befehlen gemeinsam genutzt und mittels `apt-get update` oder `aptitude update` aktualisiert. Die `"Packages"`-Metadaten werden aktualisiert, wenn eine `"deb"`-Zeile in `/etc/apt/sources.list` existiert. Die `"Sources"`-Metadaten werden aktualisiert, wenn eine `"deb-src"`-Zeile in `/etc/apt/sources.list` existiert.

Die `"Packages"`- und `"Sources"`-Metadaten enthalten einen `"Filename:"`-Eintrag, der auf den Speicherort der Binär- und Quellpakete verweist. Derzeit sind diese Pakete in dem Verzeichnisbaum unterhalb von `"pool/"` abgelegt zwecks einfacher Übergänge zwischen den Veröffentlichungen.

Lokale Kopien der `"Packages"`-Metadaten können mit Hilfe von `aptitude` interaktiv durchsucht werden. Der spezialisierte Suchbefehl `grep-dctrl(1)` kann lokale Kopien der `"Packages"`- und `"Sources"`-Metadaten durchsuchen.

Lokale Kopien von `"Contents-<architektur>"`-Metadaten können mittels `apt-file update` aktualisiert werden und ihr Speicherort unterscheidet sich von den vier anderen. Lesen Sie dazu `apt-file(1)`. (`auto-apt` nutzt standardmäßig andere Speicherorte für die lokale Kopie von `"Contents-<architektur>.gz"`.)

2.5.5 Der Paketstatus für APT

Zusätzlich zu den von außerhalb empfangenen Metadaten speichert das APT-Werkzeug seit Lenny seine lokal erzeugten Installationsstatus-Informationen in der Datei `/var/lib/apt/extended_states`, die von allen APT-Programmen genutzt wird, um automatisch installierte Pakete zu verfolgen.

2.5.6 Der Paketstatus von aptitude

Zusätzlich zu den von außerhalb empfangenen Metadaten speichert der `aptitude`-Befehl seine lokal erzeugten Installationsstatus-Informationen in der Datei `/var/lib/aptitude/pkgstates`, die nur von `aptitude` selbst genutzt wird.

2.5.7 Lokale Kopien der empfangenen Pakete

Alle über den APT-Mechanismus von außerhalb empfangenen Pakete werden im Verzeichnis `/var/cache/apt/archives` abgelegt, bis sie gelöscht werden.

Sie können diese Richtlinie zur Entfernung von zwischengespeicherten Dateien bei `aptitude` unter `"Optionen"` → `"Einstellung festlegen"`; über das Menü unter `"Aktionen"` → `"Paketcache komplett leeren"` oder `"Nur veraltete Paketdateien löschen"` können Sie dies auch händisch erzwingen.

Pakettyp	Namensstruktur
Binärpaket (auch deb)	<package-name>_<upstream-version>-<debian-version>_<arch>
Binärpaket für den debian-installer (auch udeb)	<package-name>_<upstream-version>-<debian-version>_<arch>.udeb
Quellpaket (Quellcode der Originalprogrammierer)	<package-name>_<upstream-version>-<debian-version>.orig.tar.gz
1.0-Quellpaket (Debian-Änderungen)	<package-name>_<upstream-version>-<debian-version>.diff.gz
3.0 (quilt)-Quellpaket (Debian-Änderungen)	<package-name>_<upstream-version>-<debian-version>.debdiff.gz
Quellpaket (Beschreibung)	<package-name>_<upstream-version>-<debian-version>.dsc

Tabelle 2.15: Namensstruktur von Debian-Paketen

2.5.8 Debian-Paketdateinamen

Die Dateien von Debian-Paketen haben eine bestimmte Namensstruktur:

Tipp

Hier sind nur die grundlegenden Quellpaketformate beschrieben. Lesen Sie weitere Details in `dpkg-source(1)`.

Namenskomponente	zu verwendende Zeichen (regex)	Existenz
<paketname>	[a-z,A-Z,0-9,.,+,-]+	erforderlich
<epoche>:	[0-9]+:	optional
<upstream-version>	[a-z,A-Z,0-9,.,+,-,:]+	erforderlich
<debian.version>	[a-z,A-Z,0-9,.,+,-,~]+	optional

Tabelle 2.16: In den einzelnen Komponenten von Debian-Paketnamen zu verwendende Zeichen

Anmerkung

Sie können die Reihenfolge von Paketversionen (welche Versionsnummer höher ist) mit `dpkg(1)` überprüfen, z.B. mittels `"dpkg --compare-versions 7.0 gt 7.~pre1 ; echo $?"`.

Anmerkung

Der [debian-installer \(d-i\)](#) nutzt `udeb` als Dateinamenerweiterung für seine Binärpakete statt des normalen `deb`. Ein `udeb`-Paket ist ein reduziertes `deb`-Paket, in dem einige nicht-essentielle Bestandteile wie Dokumentation entfernt wurden, um Speicherplatz zu sparen; dazu wurden die Anforderungen an die Paketrichtlinien etwas gelockert. Sowohl `deb`- wie auch `udeb`-Pakete verwenden die gleiche Paketstruktur. Das "u" steht für micro (sehr klein).

2.5.9 Der dpkg-Befehl

`dpkg(1)` ist das Werkzeug, das beim Debian-Paketmanagement auf der untersten Ebene arbeitet. Es ist sehr leistungsfähig und muss mit Vorsicht verwendet werden.

Beim Installieren eines Pakets "`<paketname>`" arbeitet `dpkg` folgende Schritte ab:

1. die `deb`-Datei auspacken (gleichbedeutend mit `"ar -x"`);
2. ausführen von `"<paketname>.preinst"` mittels `debconf(1)`;
3. installieren des Paketinhalts in das System (gleichbedeutend zu `"tar -x"`);

4. ausführen von "`<paketname>.postinst`" mittels `debconf(1)`.

Das `debconf`-System bietet eine standardisierte Benutzer-Schnittstelle mit Unterstützung für I18N und L10N (Näheres in Kapitel 8).

Datei	Beschreibung des Inhalts
<code>/var/lib/dpkg/info/<paketname>.conf</code>	File des Konfigurationsdateien (durch den Benutzer veränderbar)
<code>/var/lib/dpkg/info/<paketname>.list</code>	Liste von Dateien und Verzeichnissen, die durch das Paket installiert werden
<code>/var/lib/dpkg/info/<paketname>.md5sums</code>	Liste der MD5-Hash-Werte für Dateien, die durch das Paket installiert werden
<code>/var/lib/dpkg/info/<paketname>.preinst</code>	Paket-Skript, das vor der Paketinstallation ausgeführt wird
<code>/var/lib/dpkg/info/<paketname>.postinst</code>	Paket-Skript, das nach der Paketinstallation ausgeführt wird
<code>/var/lib/dpkg/info/<paketname>.prerm</code>	Paket-Skript, das vor der Paketentfernung ausgeführt wird
<code>/var/lib/dpkg/info/<paketname>.postrm</code>	Paket-Skript, das nach der Paketentfernung ausgeführt wird
<code>/var/lib/dpkg/info/<paketname>.conf.d</code>	File-Skript für das <code>debconf</code> -System
<code>/var/lib/dpkg/alternatives/<paketname></code>	alternative-Informationen, die durch den <code>update-alternatives</code> -Befehl genutzt werden
<code>/var/lib/dpkg/available</code>	availability-Informationen (Verfügbarkeit) für alle Pakete
<code>/var/lib/dpkg/diversions</code>	die diversions-Informationen, die durch <code>dpkg(1)</code> genutzt und durch <code>dpkg-divert(8)</code> gesetzt werden
<code>/var/lib/dpkg/statoverride</code>	die stat-override-Informationen, die durch <code>dpkg(1)</code> genutzt und durch <code>dpkg-statoverride(8)</code> gesetzt werden
<code>/var/lib/dpkg/status</code>	Status-Informationen für alle Pakete
<code>/var/lib/dpkg/status-old</code>	Backup der ersten Generation von " <code>var/lib/dpkg/status</code> "
<code>/var/backups/dpkg.status*</code>	Backup der zweiten Generation und ältere von " <code>var/lib/dpkg/status</code> "

Tabelle 2.17: Erwähnenswerte Dateien, die durch `dpkg` erzeugt werden

Die "`status`"-Datei wird auch von Werkzeugen wie `dpkg(1)`, "`dselect update`" und "`apt-get -u dselect-upgrade`" verwendet.

Der spezialisierte Suchbefehl `grep-dctrl(1)` kann lokale Kopien der "`status`"- und "`available`"-Metadaten durchsuchen.

Tipp

In der `debian-installer`-Umgebung wird der Befehl `udpkg` benutzt, um `udeb`-Pakete zu öffnen. `udpkg` ist eine reduzierte Variante des `dpkg`-Befehls.

2.5.10 Der `update-alternatives`-Befehl

Das Debian-System hat Mechanismen, um Programme, die sich bei der Funktionalität ein wenig überlappen, friedlich nebeneinander zu installieren, und zwar mittels `update-alternatives(1)`. Zum Beispiel können Sie den `vi`-Befehl so einrichten, dass er `vim` ausführt, wenn Sie sowohl das `vim` wie auch das `nvi`-Paket installiert haben.

```
$ ls -l $(type -p vi)
lrwxrwxrwx 1 root root 20 2007-03-24 19:05 /usr/bin/vi -> /etc/alternatives/vi
$ sudo update-alternatives --display vi
...
$ sudo update-alternatives --config vi
Auswahl      Befehl
-----
      1      /usr/bin/vim
*+      2      /usr/bin/nvi
```

Drücken Sie die Eingabetaste, um die aktuelle Wahl[*] beizubehalten, oder geben Sie die Auswahlnummer ein: 1 ↵

Das Debian-alternatives-System setzt seine Auswahl als symbolischer Link in `"/etc/alternatives/".` Der Auswahlprozess nutzt die entsprechende Datei in `"/var/lib/dpkg/alternatives/".`

2.5.11 Der dpkg-statoverride-Befehl

Stat overrides, bereitgestellt durch den `dpkg-statoverride(8)`-Befehl, sind ein Weg, `dpkg(1)` mitzuteilen, dass bei Installation eines Pakets für eine **Datei** ein anderer Eigentümer oder andere Berechtigungen verwendet werden sollen. Falls `"- -update"` angegeben wurde und die Datei bereits existiert, werden direkt der neue Eigentümer oder neue Berechtigungen gesetzt.



Achtung

Die direkte Abänderung von Benutzer oder Berechtigungen durch den Systemadministrator (über die Befehle `chmod` oder `chown`) bei einer **Datei**, die zu einem Paket gehört, wird bei der nächsten Aktualisierung des Pakets wieder aufgehoben.

Anmerkung

Ich verwende hier den Begriff **Datei**, jedoch kann dies jegliches Objekt im Dateisystem sein, dass durch `dpkg` behandelt wird, inklusive Verzeichnissen, Gerätedateien usw.

2.5.12 Der dpkg-divert-Befehl

Diversions, bereitgestellt durch den `dpkg-divert(8)`-Befehl, bieten die Möglichkeit, `dpkg(1)` zu zwingen, eine Datei nicht an Ihren Standardort zu installieren, sondern an einen umgelenkten Speicherort (**diverted**). `dpkg-divert` ist zur Verwendung durch Paketverwaltungs-Skripte gedacht. Die beiläufige Verwendung durch den Systemadministrator ist nicht mehr zeitgemäß.

2.6 Wiederherstellung eines beschädigten Systems

Bei einem `unstable`-System sollte der Administrator darauf vorbereitet sein, das System im Falle einer Beschädigung des Paketmanagements wiederherstellen zu können.



Achtung

Einige Methoden, die hier beschrieben werden, beinhalten ein hohes Risiko. Sie wurden gewarnt!

2.6.1 Inkompatibilität mit alter Benutzerkonfiguration

Wenn ein Desktop-GUI-Programm nach signifikanten Upgrades von Upstream Instabilitäten erkennen lässt, sollten Sie Behinderungen durch alte lokal erstellte Konfigurationsdateien als Ursache in Betracht ziehen. Falls das Programm unter einem neu erstellten Benutzerkonto stabil läuft, ist diese Hypothese bestätigt. (Dies ist ein Fehler beim Paketieren und wird für gewöhnlich durch den Paketersteller vermieden.)

Um das Programm wieder stabil zu bekommen, sollten Sie die entsprechenden lokalen Konfigurationsdateien beiseite schaffen und das GUI-Programm neu starten. Sie müssen möglicherweise die Inhalte alter Konfigurationsdateien lesen, um die Konfigurationsinformationen später wiederherstellen zu können. (Löschen Sie sie nicht zu früh.)

2.6.2 Mehrere Pakete mit überlappenden Dateien

Paketmanagementsysteme für das Debian-Archiv wie `aptitude(8)` oder `apt-get(1)` versuchen erst gar nicht, Pakete mit überlappenden Dateien über Paketabhängigkeiten zu installieren (weiteres in Abschnitt 2.1.6).

Fehler durch einen Paketbetreuer oder die Einrichtung von inkonsistent gemischten Archivquellen durch den Systemadministrator (lesen Sie Abschnitt 2.7.2) könnten Situationen mit falsch definierten Paketabhängigkeiten hervorrufen. Wenn Sie in solch einer Situation ein Paket mit überlappenden Dateien mittels `aptitude(8)` oder `apt-get(1)` installieren, stellt `dpkg(1)`, das das Paket entpackt, sicher, dass ein Fehler an das aufrufende Programm zurückgegeben wird, ohne dass vorhandene Dateien überschrieben werden.



Achtung

Die Verwendung von Paketen aus Drittquellen bringt durch Betreuerskripte ein erhebliches Risiko für das System mit sich, da diese mit `root`-Privilegien ausgeführt werden und jegliche Aktion auf dem System ausführen können. Der `dpkg(1)`-Befehl schützt lediglich vor dem Überschreiben beim Entpacken.

Sie können solche beschädigten Installationen umgehen, indem Sie vorher das alte betroffene Paket `<altes-paket>` entfernen:

```
$ sudo dpkg -P <altes-paket>
```

2.6.3 Behebung von Problemen aufgrund von beschädigtem Paketskript

Wenn ein Befehl in einem Paketskript aus irgendeinem Grund einen Fehler zurückgibt und das Skript aufgrund des Fehlers abgebrochen wird, beendet das Paketmanagement die Aktion und es bleiben halb-installierte Pakete zurück. Enthält ein Paket Programmierfehler in seinen Skripten zum Löschen des Pakets, könnte es sein, dass das Paket nicht mehr entfernt werden kann und ziemlich unangenehm wird.

Wenn das Paket `"<paketname>"` ein Problem mit den Skripten hat, sollten Sie in folgenden Paketskripten nachschauen:

- `"/var/lib/dpkg/info/<paketname>.preinst"`
- `"/var/lib/dpkg/info/<paketname>.postinst"`
- `"/var/lib/dpkg/info/<paketname>.prerm"`
- `"/var/lib/dpkg/info/<paketname>.postrm"`

Editieren Sie die angebotenen Paketskripte von Grund auf mittels folgender Techniken:

- deaktivieren Sie die problematische Zeile, indem Sie ein `"#"` voranstellen;
- erzwingen Sie, dass das Skript einen Rückgabewert für Erfolg zurück gibt, indem Sie der fraglichen Zeile `"|| true"` anhängen.

Konfigurieren Sie alle halb-installierten Pakete mit folgendem Befehl:

```
# dpkg --configure -a
```

2.6.4 Systemrettung mit dem `dpkg`-Befehl

Da `dpkg` ein Paketwerkzeug ist, das auf der untersten Ebene arbeitet, kann es auch in sehr schlechten Situationen wie einem nicht startfähigen System und ohne Netzwerkverbindung noch arbeiten. Wir gehen hier davon aus, dass das Paket `foo` beschädigt war und ersetzt werden muss.

Sie könnten im `cache`-Verzeichnis `"/var/cache/apt/archives/"` zwischengespeicherte Kopien von älteren, fehlerfreien Versionen des Pakets `foo` finden. (Falls nicht, können Sie diese auch aus dem Archiv unter <https://snapshot.debian.org/> herunterladen oder von einem funktionierenden System herüber kopieren.)

Ist das System boot-fähig, könnte es unter Umständen möglich sein, das Paket mit folgendem Befehl zu installieren:

```
# dpkg -i /pfad/zum/foo_<alte_version>_<arch>.deb
```

Tipp

Wenn die Beschädigung des Systems nur gering ist, könnten Sie das ganze System möglicherweise mit dem APT-System auf eine ältere Version zurückrüsten, wie in Abschnitt [2.7.10](#) beschrieben.

Kann Ihr System überhaupt nicht mehr von Festplatte gestartet werden, suchen Sie nach anderen Wegen, das System zu starten:

1. Booten Sie das System, indem Sie den Rettungsmodus der debian-installer-CD verwenden.
2. Binden Sie die Festplattenpartitionen des nicht mehr zu startenden Systems in `"/target"` ein.
3. Installieren Sie wie folgt eine ältere Version des Pakets `foo`:

```
# dpkg --root /target -i /pfad/zum/foo_<alte_version>_<arch>.deb
```

Dies funktioniert sogar, wenn der `dpkg`-Befehl auf der Festplatte beschädigt ist.

Tipp

Jegliches GNU/Linux-System, das von einem anderen System auf der Festplatte oder von Live-GNU/Linux-CD, bootfähigem USB-Stick oder per Netboot gestartet wird, kann auf ähnliche Art zur Rettung des beschädigten Systems verwendet werden.

Falls der Versuch, ein Paket auf diese Art zu installieren, aufgrund von verletzten Abhängigkeiten fehlschlägt, können Sie als letzten Ausweg Abhängigkeiten mit den `dpkg`-Optionen `"--ignore-depends"`, `"--force-depends"` und weiteren überschreiben. Dabei müssen Sie besondere Sorgfalt darauf verwenden, die Abhängigkeiten später korrekt wiederherzustellen. Details finden Sie in `dpkg(8)`.

Anmerkung

Falls Ihr System ernsthaft beschädigt ist, sollten ein vollständiges Backup aller Daten an einem sicheren Ort ablegen (siehe Abschnitt [10.2](#)) und eine saubere Neuinstallation durchführen. Dies ist weniger zeitaufwändig und führt am Ende zu einem besseren Ergebnis.

2.6.5 Paketauswahldaten wiederherstellen

Falls `"/var/lib/dpkg/status"` aus irgendeinem Grund beschädigt wird, verliert das Debian-System die Daten über ausgewählte Pakete und kommt dadurch erheblich zu Schaden. Sie finden alte `"/var/lib/dpkg/status"`-Dateien in `"/var/lib/dpkg/backup/"` oder `"/var/backups/dpkg.status.*"`.

Es könnte eine gute Idee sein, `"/var/backups/"` auf einer separaten Partition abzulegen, da dieses Verzeichnis viele wichtige Systemdaten enthält.

Bei ernststen Beschädigungen empfehle ich eine frische Neuinstallation, nachdem eine Sicherung der Daten durchgeführt wurde. Sogar wenn alles in `"/var/"` verloren ist, können Sie trotzdem noch etliche Informationen aus Verzeichnissen in `"/usr/share/doc/"` wiederherstellen, um Ihre neue Installation zu lenken:

Installieren Sie ein minimales (Desktop-)System.

```
# mkdir -p /pfad/zum/alten/system
```

Binden Sie die alten Systempartitionen in `"/pfad/zum/alten/system/"` ein.

```
# cd /pfad/zum/alten/system/usr/share/doc
# ls -l >~/ls1.txt
# cd /usr/share/doc
# ls -l >>~/ls1.txt
# cd
# sort ls1.txt | uniq | less
```

Jetzt haben Sie eine Liste mit Paketnamen, die Sie installieren können. (Es könnten auch einige Zeichenfolgen darunter sein, die keine Paketnamen sind, wie z.B. "texmf".)

2.7 Tipps für das Paketmanagement

2.7.1 Wie Sie Debian-Pakete auswählen

Mit `aptitude` können Sie nach Paketen suchen, die Ihren Anforderungen entsprechen, entweder über die Paketbeschreibung oder über die Liste in "Tasks".

Wenn Sie auf mehr als zwei ähnliche Pakete stoßen und sich fragen, welches Sie installieren sollen, ohne dabei nach dem "trial and error"-Verfahren (ausprobieren und schauen, welches das richtige ist) vorgehen zu müssen, sollten Sie ein wenig Ihren **gesunden Menschenverstand** benutzen. Die folgenden Punkte können als gute Hinweise auf passende Pakete angesehen werden:

- Essentiell: ja > nein;
- Bereich: main > contrib > non-free;
- Priorität: erforderlich > wichtig > standard > optional > extra;
- Tasks: Pakete, die in Tasks wie "Desktop-Umgebungen" aufgelistet sind;
- Pakete, die über eine Paketabhängigkeit ausgewählt werden (z.B. `python2.4` über `python`);
- Popcon: höhere Einstufung bei den vote- und install-Werten;
- Changelog (Änderungsprotokoll): regelmäßige Aktualisierungen durch den Betreuer;
- BTS (Fehlerdatenbank): keine veröffentlichungskritischen Fehler (RC-Bugs, d.h. keine Fehler mit Schweregrad critical, grave oder serious);
- BTS (Fehlerdatenbank): Betreuer reagiert auf Fehlerberichte;
- BTS (Fehlerdatenbank): höhere Anzahl von kürzlich behobenen Fehlern;
- BTS (Fehlerdatenbank): niedrigere Anzahl von offenen Fehlern mit Schweregrad (Severity) verschieden von wishlist.

Debian ist ein Freiwilligenprojekt mit verteiltem Entwicklungsmodell, dessen Archiv viele Pakete unterschiedlicher Zielsetzung und Qualität enthält. Sie müssen Ihre eigene Entscheidung treffen, was Sie damit anfangen.

2.7.2 Pakete aus gemischten Paketquellen



Achtung

Die Installation von Paketen aus gemischten Paketquellen wird von der offiziellen Debian-Distribution nicht unterstützt, außer für ein paar bestimmte Archivkombinationen, wie z.B. `stable` mit [Sicherheits-Updates](#) und [bullseye-updates](#).

Hier ein Beispiel von Befehlen, über die man spezielle neuere Upstream-Versionen von Paketen aus `Unstable` verwendet, während ansonsten `Testing` genutzt wird:

1. Ändern Sie die Datei `"/etc/apt/sources.list"` vorübergehend in `"unstable"`.
2. Führen Sie `"aptitude update"` aus.
3. Führen Sie `"aptitude install <packetname>"` aus.
4. Stellen Sie die Originalversion von `"/etc/apt/sources.list"` für `testing` wieder her.
5. Führen Sie `"aptitude update"` aus.

Bei diesem manuellen Vorgehen erstellen Sie keine `"/etc/apt/preferences"`-Datei und müssen sich auch keine Sorgen über apt-pinning machen. Allerdings ist dies sehr mühsam.

**Achtung**

Wenn Sie gemischte Paketquellen verwenden, müssen Sie die Kompatibilität selbst sicherstellen, da Debian diese in solchem Falle nicht garantieren kann. Falls Paketinkompatibilitäten existieren, könnten Sie Ihr System beschädigen. Sie müssen in der Lage sein, diese technischen Anforderungen zu beurteilen. Die Verwendung von gemischten Quellen zufällig ausgewählter Archive ist eine absolut optionale Operation und nicht zu empfehlen.

Die grundsätzlichen Regeln für die Installation von Paketen aus unterschiedlichen Archiven sind wie folgt:

- Nicht-binäre Pakete (`"Architecture: all"`) sind recht **unproblematisch** zu installieren:
 - Dokumentationspakete: keine speziellen Anforderungen;
 - Pakete von Interpreter-Programmen: ein kompatibler Interpreter muss verfügbar sein.
- Bei Binärpaketen (nicht `"Architecture: all"`) gibt es gewöhnlich viele Stolpersteine und ihre Installation könnte **problematisch** sein:
 - Kompatibilität zu Bibliotheksversionen (inklusive `"libc"`);
 - Kompatibilität zu damit in Verbindung stehenden Hilfswerkzeugen;
 - Kernel [ABI](#)-Kompatibilität;
 - C++ [ABI](#)-Kompatibilität;
 - ...

Anmerkung

Um die Installation eines Pakets **unproblematischer** zu machen, existieren möglicherweise einige nicht-freie Binärprogramm-Pakete mit vollständig statisch gelinkten Bibliotheken. Sie sollten diese immer auf [ABI](#)-Kompatibilitätsprobleme usw. kontrollieren.

Anmerkung

Die Installation von Binärpaketen aus offiziell nicht unterstützten Archiven ist grundsätzlich eine schlechte Idee (außer zur kurzfristigen Vermeidung von beschädigten Paketen). Dies gilt sogar, wenn Sie apt-pinning verwenden (lesen Sie dazu Abschnitt [2.7.3](#)). Sie sollten chroot oder ähnliche Techniken in Betracht ziehen (weiteres in Abschnitt [9.10](#)), um Programme aus verschiedenen Archiven laufen zu lassen.

2.7.3 Installationskandidat-Version beeinflussen

Ohne eine `"/etc/apt/preferences"`-Datei wählt das APT-System basierend auf dem Versionseintrag die letzte verfügbare Version als **Installationskandidat-Version** aus. Dies ist der normale Weg und die empfohlene Verwendung des APT-Systems. Alle offiziell unterstützten Archivkombinationen erfordern keine `"/etc/apt/preferences"`-Datei, da bei Archiven, die nicht als automatische Quelle für Aktualisierungen empfohlen werden, die Einstellung **NotAutomatic** gesetzt ist und diese entsprechend behandelt werden.

Tipp

Die Regel zum Vergleich der Versionsnummern (welche Versionsnummer größer ist) kann z.B. über `"dpkg --compare-versions ver1.1 gt ver1.1~1; echo $?"` verifiziert werden (näheres unter `dpkg(1)`).

Wenn Sie regelmäßig Pakete aus gemischten Quellen installieren (siehe Abschnitt 2.7.2), können Sie diese komplizierten Operationen automatisieren, indem Sie eine Datei `"/etc/apt/preferences"` mit entsprechenden Einträgen erstellen; so beeinflussen Sie die Auswahlregel für die **Installationskandidat-Version**; dies ist in `apt_preferences(5)` beschrieben. Dies Verfahren wird **apt-pinning** genannt.



Warnung

Die Verwendung von apt-pinning durch einen unerfahrenen Benutzer wird sicher große Probleme hervorrufen. Sie sollten die Verwendung von apt-pinning vermeiden, außer es ist unbedingt erforderlich.



Achtung

Wenn Sie apt-pinning verwenden, müssen Sie die Kompatibilität der Pakete selbst sicherstellen, da Debian diese in einem solchen Fall nicht garantieren kann. Apt-pinning ist ein absolut optionales Verfahren und nicht zu empfehlen.



Achtung

Für die `apt_preferences(5)`-Regeln werden die Release-Dateien im Archiv-Wurzelverzeichnis verwendet (näheres in Abschnitt 2.5.3). Daher funktioniert apt-pinning bei [normalen Debian-Archiven](#) und [Debian-Archiven für Sicherheitsaktualisierungen](#) nur mit dem "Suite"-Namen. (Dies ist anders als bei [Ubuntu-Archiven](#).) Zum Beispiel können wir `"Pin: release a=unstable"` in `"/etc/apt/preferences"` verwenden, aber nicht `"Pin: release a=sid"`.



Achtung

Wenn Sie Debian-fremde Archive als Teil von apt-pinning verwenden, sollten Sie kontrollieren, wofür diese gedacht sind und ob sie glaubwürdig sind. So sind zum Beispiel Ubuntu und Debian nicht dazu gedacht, miteinander vermischt zu werden.

Anmerkung

Selbst wenn Sie keine `"/etc/apt/preferences"`-Datei erstellen, können Sie auch ohne apt-pinning ziemlich komplexe Systemoperationen durchführen (lesen Sie Abschnitt 2.6.4 und Abschnitt 2.7.2).

Hier eine vereinfachte Beschreibung der Technik hinter **apt-pinning**:

Das APT-System wählt für ein **Upgrade** das Paket mit der höchsten Pin-Priorität aus den verfügbaren (in `"/etc/apt/sources.list"` definierten) Paketquellen als **Installationskandidat-Version** aus. Wenn die Pin-Priorität des Pakets größer als 1000 ist, wird die

Pin-Priorität	apt-pinning-Effekte auf das Paket
1001	das Paket installieren, auch wenn dies ein Downgrade des Pakets bedeutet
990	wird als Standardwert für das Archiv der Zielveröffentlichung verwendet
500	wird als Standardwert für das normale Archiv verwendet
100	wird als Standardwert für Archive mit NotAutomatic und ButAutomaticUpgrades verwendet
100	wird für das installierte Paket verwendet
1	wird als Standardwert für Archive mit NotAutomatic verwendet
-1	das Paket niemals installieren , selbst wenn es empfohlen wird

Tabelle 2.18: Liste erwähnenswerter Pin-Prioritäts-Werte für **apt-pinning**

Einschränkung für **Upgrades** nicht beachtet, um ein Downgrade (eine Zurückrüstung auf eine ältere Version) zu ermöglichen (siehe Abschnitt 2.7.10).

Werte für die Pin-Priorität eines jeden Pakets werden über Einträge in der `/etc/apt/preferences`-Datei festgelegt oder verwenden deren Standardwert.

Die gewünschte **Zielveröffentlichung** kann über verschiedene Methoden festgelegt werden:

- `/etc/apt/apt.conf`-Konfigurationsdatei mit einer `APT::Default-Release "stable";`-Zeile;
- Befehlszeilen-Option, z.B. `apt-get install -t testing irgendein-paket`.

Die **NotAutomatic**- und **ButAutomaticUpgrades**-Archive werden durch Archiv-Server definiert, die in ihren Release-Dateien im Archiv-Wurzelverzeichnis (siehe Abschnitt 2.5.3) sowohl `NotAutomatic: yes` wie auch `ButAutomaticUpgrades: yes` gesetzt haben. Das **NotAutomatic**-Archiv wird über Server definiert, die in ihren Release-Dateien lediglich `NotAutomatic: yes` gesetzt haben.

Die **apt-pinning-Situation** von `<package>` aus mehreren Archivquellen wird mittels `apt-cache policy <package>` angezeigt.

- Eine Zeile beginnend mit `Package pin:` listet die Paketversion von **pin** auf, wenn die Zuordnung direkt über `<package>` definiert ist, z.B. `Package pin: 0.190`.
- Es existiert keine Zeile mit `Package pin:`, wenn keine Zuordnung direkt über `<package>` definiert ist.
- Der Pin-Prioritäts-Wert, der direkt zu `<package>` gehört, ist rechts neben den Versionseinträgen aufgelistet, z.B. `0.181 700`.
- `0` wird rechts neben den Versionseinträgen aufgelistet, wenn keine Zuordnung direkt mit `<package>` definiert ist, z.B. `0.181 0`.
- Die Pin-Prioritäts-Werte von Archiven (definiert als `Package: *` in der `/etc/apt/preferences`-Datei) sind links von den Archivpfaden aufgelistet, z.B. `100 http://deb.debian.org/debian/ bullseye-backports/main Packages`.

2.7.4 Aktualisierungen und Backports

Es gibt [bullseye-updates](#)- und [backports.debian.org](#)-Archive, die aktualisierte Pakete für **stable** (**bullseye**) bereitstellen.

Um diese Archive zu nutzen, listen Sie alle erforderlichen Archive in der `/etc/apt/sources.list`-Datei auf, wie hier:

```
deb http://deb.debian.org/debian/ bullseye main contrib non-free
deb http://security.debian.org/ bullseye/updates main contrib
deb http://deb.debian.org/debian/ bullseye-updates main contrib non-free
deb http://deb.debian.org/debian/ bullseye-backports main contrib non-free
```

Es ist nicht nötig, einen Wert für die Pin-Priorität in `/etc/apt/preferences` zu setzen. Wenn neuere Pakete verfügbar werden, werden mit der Standardkonfiguration passende Upgrades bereitgestellt (siehe Abschnitt 2.5.3).

- Alle installierten veralteten Pakete werden auf die neuen von `bullseye-updates` aktualisiert.
- Nur manuell installierte veraltete Pakete von `bullseye-backports` werden auf die neuen Versionen von `bullseye-backports` aktualisiert.

Wann immer Sie ein Paket namens "`<paketname>`" inklusive seiner Abhängigkeiten von `bullseye-backports` manuell installieren möchten, verwenden Sie den folgenden Befehl, wobei Sie die Zielveröffentlichung mit der Option "`-t`" festlegen:

```
$ sudo apt-get install -t bullseye-backports <paketname>
```

2.7.5 Über "Recommends" installierte Pakete blockieren

Wenn Sie verhindern möchten, dass bestimmte Pakete über "Recommends"-Abhängigkeiten (Empfohlen) installiert werden, müssen Sie die Datei `/etc/apt/preferences` erstellen und explizit all diese Pakete ganz oben in der Datei auflisten, wie in diesem Beispiel:

```
Package: <paket-1>
Pin: version *
Pin-Priority: -1

Package: <paket-2>
Pin: version *
Pin-Priority: -1
```

2.7.6 Nutzen von Testing mit einigen Paketen aus Unstable

Hier ein Beispiel für eine **apt-pinning**-Technik, um neuere Upstream-Versionen einzelner Pakete aus `Unstable` regelmäßig aktualisieren zu können und ansonsten `Testing` zu nutzen. Sie listen alle benötigten Archive in der `/etc/apt/sources.list`-Datei auf, wie hier:

```
deb http://deb.debian.org/debian/ testing main contrib non-free
deb http://deb.debian.org/debian/ unstable main contrib non-free
deb http://security.debian.org/ testing/updates main contrib
```

Richten Sie die `/etc/apt/preferences`-Datei wie folgt ein:

```
Package: *
Pin: release a=unstable
Pin-Priority: 100
```

Wenn Sie mit dieser Konfiguration ein Paket namens "`<paketname>`" aus dem `Unstable`-Archiv inklusive seiner Abhängigkeiten installieren möchten, führen Sie folgenden Befehl aus, wobei über die Option "`-t`" die Zielveröffentlichung angepasst wird (die Pin-Priorität von `Unstable` wird 990):

```
$ sudo apt-get install -t unstable <paketname>
```

Mit dieser Konfiguration wird die normale Ausführung von `apt-get upgrade` und `apt-get dist-upgrade` (oder `aptitude safe-upgrade` und `aptitude full-upgrade`) Pakete, die aus `Testing` installiert wurden, unter Verwendung des derzeitigen `Testing`-Archivs aktualisieren; Pakete, die aus `Unstable` installiert wurden, werden unter Verwendung des derzeitigen `Unstable`-Archivs aktualisiert.



Achtung

Achten Sie darauf, dass Sie nicht den "testing"-Eintrag aus der `/etc/apt/sources.list`-Datei entfernen. Ohne diesen "testing"-Eintrag aktualisiert das APT-System alle Pakete unter Verwendung des (neueren) `Unstable`-Archivs.

Tipp

Ich bearbeite für gewöhnlich die `/etc/apt/sources.list`-Datei direkt nach obigen Befehlen, um das `unstable`-Archiv auszukommentieren. Dies verhindert, dass der Prozess zur Aktualisierung der Paketdaten aufgrund von zu vielen Einträgen in der `/etc/apt/sources.list`-Datei sehr lange dauert; allerdings können dadurch Pakete, die aus `Unstable` installiert wurden, nicht unter Verwendung des derzeitigen `Unstable`-Archivs aktualisiert werden.

Tipp

Wenn `Pin-Priority: 1` statt `Pin-Priority: 100` in der `/etc/apt/preferences`-Datei eingetragen wird, werden bereits installierte Pakete, die den Pin-Prioritäts-Wert 100 haben, nicht mittels dem `Unstable`-Archiv aktualisiert, selbst wenn der `testing`-Eintrag in `/etc/apt/sources.list` entfernt wird.

Möchten Sie bestimmte Pakete in `Unstable` automatisch ohne vorangestelltes `-t unstable` auf aktuellem Stand halten, müssen Sie die `/etc/apt/preferences`-Datei erstellen und ganz oben in der Datei all diese Pakete wie folgt auflisten:

```
Package: <paket-1>
Pin: release a=unstable
Pin-Priority: 700
```

```
Package: <paket-2>
Pin: release a=unstable
Pin-Priority: 700
```

Dadurch wird die Pin-Priorität für jedes dieser Pakete spezifisch gesetzt. Um zum Beispiel immer die aktuellste `Unstable`-Version dieser `Debian Reference` in Englisch zu installieren, sollten Sie folgende Einträge in Ihrer `/etc/apt/preferences`-Datei haben:

```
Package: debian-reference-en
Pin: release a=unstable
Pin-Priority: 700

Package: debian-reference-common
Pin: release a=unstable
Pin-Priority: 700
```

Tipp

Diese `apt-pinning`-Technik funktioniert sogar, wenn Sie normalerweise `Stable` verwenden. Dokumentationspakete konnten nach meiner Erfahrung bis jetzt immer problemlos aus dem `Unstable`-Archiv installiert werden.

2.7.7 Nutzen von `Unstable` mit einigen Paketen aus `Experimental`

Hier ein anderes Beispiel der **apt-pinning**-Technik, um neuere Upstream-Versionen einzelner Pakete aus `Experimental` zu integrieren und ansonsten `Testing` zu nutzen. Sie listen alle benötigten Archive in der `/etc/apt/sources.list`-Datei auf, wie hier:

```
deb http://deb.debian.org/debian/ unstable main contrib non-free
deb http://deb.debian.org/debian/ experimental main contrib non-free
deb http://security.debian.org/ testing/updates main contrib
```

Die Standard-Pin-Priorität für das `Experimental`-Archiv ist immer 1 ($\ll 100$), da es ein Archiv mit gesetztem **NotAutomatic** ist (siehe Abschnitt 2.5.3). Es ist nicht nötig, den Wert für die Pin-Priorität explizit in der `/etc/apt/preferences`-Datei zu setzen, um das `Experimental`-Archiv nutzen zu können (außer wenn Sie möchten, dass bestimmte Pakete daraus beim nächsten Upgrade automatisch verwendet werden sollen).

2.7.8 Automatisches Herunterladen und Aktualisieren von Paketen

Das `apt`-Paket enthält ein eigenes cron-Skript (`/etc/cron.daily/apt`), um den automatischen Download von Paketen zu unterstützen. Durch die Installation des `unattended-upgrades`-Pakets kann dieses Skript erweitert werden, so dass auch die automatische Aktualisierung von Paketen durchgeführt wird. Sie können dies über Parameter in `/etc/apt/apt.conf.d/02backu` und `/etc/apt/apt.conf.d/50unattended-upgrades` noch weiter anpassen, wie in `/usr/share/doc/unattended-upgrades` beschrieben.

Das `unattended-upgrades`-Paket ist hauptsächlich für Sicherheits-Updates auf `Stable`-Systemen gedacht. Wenn das Risiko, ein vorhandenes `Stable`-System über einen automatischen Upgrade-Prozess zu beschädigen, geringer ist als das, welches von einem Angreifer ausgeht, der über Ausnutzung einer Sicherheitslücke das System schädigt, sollten Sie ein automatisches Upgrade in Erwägung ziehen, das diese Lücke über ein Sicherheits-Upgrade schließt. Nutzen Sie dazu folgende Konfigurationsparameter:

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::Unattended-Upgrade "1";
```

Wenn Sie ein `Unstable`-System laufen haben, sollten Sie kein automatisches Upgrade verwenden, da sonst mit Sicherheit eines Tages das System beschädigt würde. Aber sogar für solch ein `Unstable`-System möchten Sie vielleicht in Vorbereitung für ein interaktives Upgrade die Pakete schon mal automatisch herunterladen, um Zeit zu sparen. Verwenden Sie dazu folgende Konfigurationsparameter:

```
APT::Periodic::Update-Package-Lists "1";
APT::Periodic::Download-Upgradeable-Packages "1";
APT::Periodic::Unattended-Upgrade "0";
```

2.7.9 Die Download-Bandbreite für APT einschränken

Falls Sie die Download-Bandbreite für APT einschränken möchten, sagen wir z.B. auf 800Kib/sec (=100kiB/sec), sollten Sie APT mit folgenden Konfigurationsparametern einrichten:

```
APT::Acquire::http::DL-Limit "800";
```

2.7.10 Downgrade im Notfall



Achtung

Ein Downgrade wird von Debian aufgrund seines Designs nicht offiziell unterstützt. Es sollte nur als Teil einer Notfall-Rettungsmaßnahme durchgeführt werden. Davon einmal abgesehen ist bekannt, dass es in den meisten Fällen gut funktioniert. Bei kritischen Systemen sollten Sie nach der Wiederherstellung alle wichtigen Daten auf dem System sichern und dann das System von Grund auf neu installieren.

Sie könnten Glück haben und ein Downgrade auf eine ältere Archiv-Version durch Manipulation der **Installationskandidat-Versionen** hinbekommen (lesen Sie dazu Abschnitt 2.7.3). Dies ist die tippfaule Alternative zu vielen langwierigen Befehlen der Art `dpkg -i <beschädigtes-paket>_<alte-version>.deb` (näheres dazu in Abschnitt 2.6.4).

Suchen Sie Zeilen in der `/etc/apt/sources.list`-Datei wie die folgende, die `unstable` referenzieren:

```
deb http://deb.debian.org/debian/ sid main contrib non-free
```

Ersetzen Sie sie durch die folgende, um stattdessen auf `testing` zu verweisen:

```
deb http://deb.debian.org/debian/ bookworm main contrib non-free
```

Richten Sie die `/etc/apt/preferences`-Datei wie folgt ein:

```
Package: *  
Pin: release a=testing  
Pin-Priority: 1010
```

Führen Sie `apt-get update; apt-get dist-upgrade` aus, um ein Downgrade aller Pakete im System zu erzwingen. Entfernen Sie diese spezielle `/etc/apt/preferences`-Datei nach dem Downgrade.

Tipp

Es ist eine gute Idee, so viele Pakete wie möglich zu löschen (nicht vollständig inklusive der Konfigurationsdateien entfernen!), um die Wahrscheinlichkeit von Abhängigkeitsproblemen zu minimieren. Sie müssen bei einem solchen Downgrade unter Umständen einige Pakete von Hand entfernen oder installieren. Linux-Kernel, Bootloader, udev, PAM, APT und Netzwerk-bezogene Pakete sowie deren Konfigurationsdateien erfordern besondere Aufmerksamkeit.

2.7.11 Wer hat das Paket hochgeladen?

Obwohl der Betreuername, der in `/var/lib/dpkg/available` und `/usr/share/doc/package_name/changelog` aufgelistet ist, Informationen darüber gibt, "wer hinter den Paketaktivitäten steht", sind Informationen über den Uploader (die Person, die das Paket wirklich hochgeladen hat) ein wenig verschleiert. `who-uploads(1)` aus dem `devscripts`-Paket identifiziert die realen Uploader von Debian-Quellpaketen.

2.7.12 Das `equivs`-Paket

Wenn Sie ein Programm aus den Quellen übersetzen möchten, um das entsprechende Debian-Paket zu ersetzen, ist es das beste, ein echtes lokal gebautes Paket nach Debian-Art zu erstellen (`*.deb`) und dafür ein privates Archiv zu nutzen.

Haben Sie sich jedoch entschieden, ein Programm aus den Quellen zu kompilieren und es stattdessen in `/usr/local` zu installieren, können Sie möglicherweise `equivs` als letzte Rettung einsetzen, um die fehlenden Paketabhängigkeiten zu erfüllen:

```
Package: equivs  
Priority: optional  
Section: admin  
Description: Umgehung der Abhängigkeiten zwischen Debian-Paketen  
Dieses Paket enthält ein Werkzeug zum Erstellen einfacher Debian-Pakete.  
Typischerweise enthalten diese Pakete nur Informationen über  
Abhängigkeiten, sie können aber auch normal installierte Dateien enthalten,  
wie dies bei anderen Paketen der Fall ist.  
.  
Eine Anwendungsmöglichkeit ist die Erstellung von Metapaketten. Der einzige  
Zweck solcher Pakete ist es, Abhängigkeiten und Konflikte mit anderen Paketen  
festzulegen, so dass diese automatisch installiert, aktualisiert beziehungsweise  
entfernt werden.  
.  
Eine weitere Anwendung ist es, die Abhängigkeitsprüfung zu umgehen: indem  
dpkg mit der Annahme arbeitet, eine bestimmte Paketversion sei installiert,  
wenn diese Version tatsächlich nicht installiert ist, kann man Fehler in den  
Abhängigkeiten anderer Pakete umgehen. (Bitte reichen Sie für solche Fehler  
dennoch einen Fehlerbericht ein.)
```

2.7.13 Ein Paket auf das Stable-System portieren

Für partielle Upgrades eines Stable-Systems ist das Neubauen von Paketen innerhalb ihrer Umgebung unter Verwendung der Quellpakete sinnvoll. Sie vermeiden so riesige Upgrades aufgrund von Paketabhängigkeiten.

Fügen Sie folgende Einträge zur `/etc/apt/sources.list` des Stable-Systems hinzu:

```
deb-src http://http.de.debian.org/debian unstable main contrib non-free
```

Installieren Sie für das Kompilieren erforderliche Pakete und laden Sie das Quellpaket wie folgt herunter:

```
# apt-get update
# apt-get dist-upgrade
# apt-get install fakeroot devscripts build-essential
# apt-get build-dep foo
$ apt-get source foo
$ cd foo*
```

Aktualisieren Sie einige Pakete aus der Werkzeugkette wie z.B. `dpkg` und `debhelper` aus Paketquellen für zurück-portierte Pakete, falls diese für die Rückportierung erforderlich sind.

Führen Sie folgendes aus:

```
$ dch -i
```

Erhöhen Sie die Paketversion, z.B. indem Sie ein `"+bp1"` in `"debian/changelog"` anhängen.

Bauen Sie die Pakete und installieren Sie sie wie folgt im System:

```
$ debuild
$ cd ..
# debi foo*.changes
```

2.7.14 Proxy-Server für APT

Obwohl das Spiegeln von vollständigen Sektionen des Debian-Archivs Plattenplatz und Netzwerkbandbreite verschwendet, ist es der Einsatz eines lokalen Proxy-Servers für APT wert in Erwägung gezogen zu werden, wenn Sie viele Systeme im [LAN](#) (Netzwerk) administrieren. APT kann konfiguriert werden, einen generischen Web-(http-)Proxy-Server wie `squid` (siehe auch Abschnitt [6.10](#)) zu verwenden, wie in `apt.conf(5)` und `"/usr/share/doc/apt/examples/configure-index.gz"` beschrieben. Sie können die Umgebungsvariable `"$http_proxy"` nutzen, um die Proxy-Server-Einstellungen aus der `"/etc/apt/apt` Datei zu überschreiben.

Es gibt Proxy-Hilfsprogramme, die für das Debian-Archiv spezialisiert sind. Sie sollten die Fehlerdatenbank (BTS) überprüfen, bevor Sie sie verwenden.

Paket	Popcon	Größe	Beschreibung
approx	V:0, I:0	6317	zischenspeichernder Proxy-Server für Debian-Archiv-Dateien (kompiliertes OCaml -Programm)
apt-cacher	V:0, I:0	289	zischenspeichernder Proxy-Server für Debian-Paket- und Quell-Dateien (Perl-Programm)
apt-cacher-ng	V:5, I:5	1488	zischenspeichernder Proxy für die Verteilung von Software-Paketen (kompiliertes C++-Programm)

Tabelle 2.19: Liste von Proxy-Hilfsprogrammen speziell für das Debian-Archiv



Achtung

Sollte Debian einmal seine Archivstruktur umorganisieren, benötigen diese spezialisierten Proxy-Hilfsprogramme eventuell eine Code-Änderung durch den Paketbetreuer und könnten eine Zeit lang nicht funktionsfähig sein. Generische Web-(http-)Proxy-Server sind andererseits robuster und können einfacher mit solchen Änderungen umgehen.

2.7.15 Kleines öffentliches Paketarchiv

Tipp

Das manuelle Aufsetzen eines Paket-Archivs ist kompliziert. Es gibt mehrere Werkzeuge zur Verwaltung von Paketdepots. Eine [umfassende Liste](#) ist online verfügbar.

Hier eine Demo der Erstellung eines kleinen öffentlichen Paketarchivs, das mit dem modernen **secure-APT-System** (siehe Abschnitt [2.5.2](#)) kompatibel ist. Wir gehen einmal von folgenden Dingen aus:

- Kontoname: "foo"
- Rechnername (host name): "www.example.com"
- Erforderliche Pakete: apt-utils, gnupg, und andere
- URL: "http://www.example.com/~foo/" (→ "/home/foo/public_html/index.html")
- Architektur der Pakete: "amd64"

Erzeugen Sie wie folgt einen APT-Archiv-Schlüssel für den Benutzer Foo auf Ihrem System:

```
$ ssh foo@www.example.com
$ gpg --gen-key
...
$ gpg -K
...
sec 1024D/3A3CB5A6 2008-08-14
uid                               Foo (ARCHIVE KEY) <foo@www.example.com>
ssb 2048g/6856F4A7 2008-08-14
$ gpg --export -a 3A3CB5A6 >foo.public.key
```

Publizieren Sie die Archivschlüssel-Datei "foo.public.key" mit der Schlüssel-ID "3A3CB5A6".

Erzeugen Sie wie folgt einen Archivbaum namens "Origin: Foo":

```
$ umask 022
$ mkdir -p ~/public_html/debian/pool/main
$ mkdir -p ~/public_html/debian/dists/unstable/main/binary-amd64
$ mkdir -p ~/public_html/debian/dists/unstable/main/source
$ cd ~/public_html/debian
$ cat > dists/unstable/main/binary-amd64/Release << EOF
Archive: unstable
Version: 4.0
Component: main
Origin: Foo
Label: Foo
Architecture: amd64
EOF
$ cat > dists/unstable/main/source/Release << EOF
Archive: unstable
Version: 4.0
Component: main
Origin: Foo
Label: Foo
Architecture: source
EOF
$ cat > aptftp.conf << EOF
APT::FTPArchive::Release {
    Origin "Foo";
    Label "Foo";
}
```

```

Suite "unstable";
Codename "sid";
Architectures "amd64";
Components "main";
Description "Public archive for Foo";
};
EOF
$ cat >aptgenerate.conf <<EOF
Dir::ArchiveDir ".";
Dir::CacheDir ".";
TreeDefault::Directory "pool/";
TreeDefault::SrcDirectory "pool/";
Default::Packages::Extensions ".deb";
Default::Packages::Compress ". gzip bzip2";
Default::Sources::Compress "gzip bzip2";
Default::Contents::Compress "gzip bzip2";

BinDirectory "dists/unstable/main/binary-amd64" {
    Packages "dists/unstable/main/binary-amd64/Packages";
    Contents "dists/unstable/Contents-amd64";
    SrcPackages "dists/unstable/main/source/Sources";
};

Tree "dists/unstable" {
    Sections "main";
    Architectures "amd64 source";
};
EOF

```

Sie können wiederkehrende Updates der APT-Archiv-Inhalte auf Ihrem Server automatisieren, indem Sie `dupload` konfigurieren.

Platzieren Sie die Paketdateien in `~/foo/public_html/debian/pool/main/`, indem Sie `dupload -t foo changes_f` ausführen, während `~/ .dupload.conf` folgendes enthält:

```

$cfg{'foo'} = {
    fqdn => "www.example.com",
    method => "scpb",
    incoming => "/home/foo/public_html/debian/pool/main",
    # The dinstall on ftp-master sends emails itself
    dinstall_runs => 1,
};

$cfg{'foo'}{postupload}{'changes'} = "
echo 'cd public_html/debian ;
apt-ftparchive generate -c=aptftp.conf aptgenerate.conf;
apt-ftparchive release -c=aptftp.conf dists/unstable >dists/unstable/Release ;
rm -f dists/unstable/Release.gpg ;
gpg -u 3A3CB5A6 -bao dists/unstable/Release.gpg dists/unstable/Release' |
ssh foo@www.example.com 2>/dev/null ;
echo 'Package archive created!'";

```

Das **postupload**-Hook-Skript, das von `dupload(1)` angestoßen wurde, erzeugt aktualisierte Archivdateien für jeden Upload. Mit folgenden Befehlen können Sie dieses kleine öffentliche Archiv zu der `apt`-Zeile auf Ihrem Client-System hinzufügen:

```

$ sudo bash
# echo "deb http://www.example.com/~foo/debian/ unstable main" \
  >> /etc/apt/sources.list
# apt-key add foo.public.key

```

Tipp

Wenn das Archiv in dem lokalen Dateisystem abgelegt ist, können Sie stattdessen auch "deb file:///home/foo/debian/ ..." verwenden.

2.7.16 Aufzeichnen und Kopieren der Systemkonfiguration

Mit folgenden Befehlen erstellen Sie eine lokale Kopie des Paket- und Debconf-Auswahlstatus:

```
# dpkg --get-selections '*' > selection.dpkg
# debconf-get-selections > selection.debconf
```

Hier sorgt "*" dafür, dass "selection.dpkg" auch Einträge für "purge" (vollständig inklusive der Konfigurationsdateien entfernt) enthält.

Sie können diese zwei Dateien auf einen anderen Computer übertragen und dort zur Installation verwenden:

```
# dselect update
# debconf-set-selections < myselection.debconf
# dpkg --set-selections < myselection.dpkg
# apt-get -u dselect-upgrade # oder dselect install
```

Wenn Sie darüber nachdenken, viele Server in einem Cluster mit nahezu identischer Konfiguration zu verwalten, sollten Sie die Verwendung spezialisierter Pakete wie **fai** zur Verwaltung des ganzen Systems in Betracht ziehen.

2.7.17 Konvertieren oder Installieren eines Binärpakets mit alien

alien(1) erlaubt die Konvertierung von Binärpaketen verschiedener Formate wie rpm (Red Hat), slp (Stampede), tgz (Slackware) und pkg (Solaris) in ein deb-Paket für Debian. Wenn Sie also ein Paket von einer fremden Linux-Distribution nutzen möchten, können Sie alien verwenden, um das Paket Ihrer Wahl umzuwandeln und zu installieren. alien unterstützt auch LSB-Pakete.

**Warnung**

alien(1) darf nicht benutzt werden, um essentielle Systempakete wie sysvinit, libc6, libpam-modules usw. zu ersetzen. In der Praxis sollte alien(1) nur für reine Binärpakete aus **non-free** verwendet werden, die LSB-konform oder statisch gelinkt sind. Bei freier Software sollten Sie deren Quellpakete benutzen, um echte Debian-Pakete zu erzeugen.

2.7.18 Extrahieren eines Pakets ohne dpkg

Die Inhalte des "dpkg*.deb"-Pakets können in allen [Unix-ähnlichen](#) Umgebungen mit den Standardwerkzeugen ar(1) und tar(1) auch ohne dpkg(1) extrahiert werden:

```
# ar x /pfad/zu/dpkg_<version>_<arch>.deb
# ls
total 24
-rw-r--r-- 1 bozo bozo 1320 2007-05-07 00:11 control.tar.gz
-rw-r--r-- 1 bozo bozo 12837 2007-05-07 00:11 data.tar.gz
-rw-r--r-- 1 bozo bozo 4 2007-05-07 00:11 debian-binary
# mkdir control
# mkdir data
# tar xvzf control.tar.gz -C control
# tar xvzf data.tar.gz -C data
```


Die Inhalte anderer `*.deb`-Pakete können Sie dem `dpkg-deb(1)`-Befehl aus dem oben genannten `dpkg*.deb`-Paket entpacken; oder verwenden Sie `ar(1)` und das neuere GNU `tar(1)` mit Unterstützung für den `xz(1)`-Dekompressionsalgorithmus, ähnlich dem obigen Beispiel.

Sie können anschließend den Paketinhalt z.B. mit dem `mc`-Befehl durchsuchen.

2.7.19 Weitere Lektüre zum Paketmanagement

Folgende Dokumentation bietet sich an, um mehr über das Paketmanagement zu lernen.

- Primäre Dokumentation zum Paketmanagement:
 - `aptitude(8)`, `dpkg(1)`, `tasksel(8)`, `apt(8)`, `apt-get(8)`, `apt-config(8)`, `apt-key(8)`, `sources.list(5)`, `apt.conf(5)` und `apt_preferences(5)`;
 - `"/usr/share/doc/apt-doc/guide.html/index.html"` und `"/usr/share/doc/apt-doc/offline.html/index.html"` aus dem `apt-doc`-Paket;
 - `"/usr/share/doc/aptitude/html/en/index.html"` aus dem `aptitude-doc-en`-Paket.
 - Offizielle und detaillierte Dokumentation zum Debian-Archiv:
 - ["Debian Policy Manual Chapter 2 - The Debian Archive"](#)
 - ["Debian Entwickler-Referenz, Kapitel 4 - Ressourcen für Debian-Entwickler, 4.6 Das Debian-Archiv"](#)
 - ["Die Debian GNU/Linux-FAQ, Kapitel 6 - Die Debian FTP-Archive"](#)
 - Anleitung zum Bau eines Debian-Pakets für Debian-Benutzer:
 - ["Debian-Leitfaden für Neue Paketbetreuer"](#) (veraltet)
 - ["Debian-Leitfaden für Paketbetreuer"](#)
-

Kapitel 3

Die Systeminitialisierung

Als Systemadministrator sollten Sie grob wissen, wie das Debian-System gestartet und konfiguriert wird. Obwohl die genauen Details in den Quelldateien der installierten Pakete und deren Dokumentation zu finden sind, ist dies für die meisten von uns ein bisschen viel.

Ich habe mein bestes gegeben, um einen schnellen Überblick über die wichtigsten Themen des Debian-Systems und dessen Konfiguration als Referenz für Sie bereitzustellen, basierend auf aktuellem und früherem Wissen von mir und anderen. Da das Debian-System sich ständig verändert, könnte sich die Situation teilweise verändert haben. Bevor Sie irgendwelche Änderungen an dem System vornehmen, sollten Sie die aktuellste Dokumentation der jeweiligen Pakete zu Rate ziehen.

Tipp

bootup(7) beschreibt den System-Boot-Prozess basierend auf `systemd` (derzeitiges Debian-System).

Tipp

boot(7) beschreibt den System-Boot-Prozess basierend auf UNIX System V Release 4 (älteres Debian-System).

3.1 Ein Überblick über den Bootstrap-Prozess

Das Computer-System durchläuft verschiedene Phasen des [Bootstrap-Prozesses](#) vom Einschalten bis zur Bereitstellung des funktionalen Betriebssystems an den Benutzer.

Der Einfachheit halber beschränke ich meine Betrachtung auf die weit verbreitete PC-Plattform mit einer Standardinstallation.

Der typische Bootstrap-Prozess ist wie eine 4-stufige Rakete. Jede Stufe übergibt die Systemkontrolle an die jeweils nachfolgende Stufe:

- Abschnitt [3.1.1](#)
- Abschnitt [3.1.2](#)
- Abschnitt [3.1.3](#)
- Abschnitt [3.1.4](#)

Natürlich können diese unterschiedlich konfiguriert werden. Wenn Sie zum Beispiel Ihren eigenen Kernel kompilieren, werden Sie unter Umständen den Schritt mit dem Mini-Debian-System überspringen. Gehen Sie daher nicht davon aus, dass dies alles in Ihrem Fall zutrifft, solange Sie es nicht selbst überprüft haben.

Anmerkung

Bei ungewöhnlichen Computer-Plattformen wie dem SUN- oder dem Macintosh-System könnten das BIOS im ROM und die Partitionen auf der Festplatte sich stark von dem hier beschriebenen unterscheiden (Abschnitt 9.5.2). Bitte suchen Sie in solch einem Fall an anderer Stelle nach Plattform-spezifischer Dokumentation.

3.1.1 Stufe 1: das BIOS

Das **BIOS** ist die erste Stufe des Boot-Prozesses und wird durch das Einschalten des Rechners aufgerufen. Es liegt im **Nur-Lese-Speicher (read only memory/ROM)** und wird von einer speziellen Speicheradresse ausgeführt, auf die die CPU durch den Einschaltvorgang verwiesen wird.

Das BIOS führt die grundsätzliche Initialisierung der Hardware durch (**POST: power on self test** / Selbsttest nach dem Einschalten) und übergibt die Systemkontrolle an die nächste Stufe. Das BIOS wird üblicherweise mit der Hardware geliefert.

Im BIOS-Startbildschirm wird für gewöhnlich angezeigt, welche Taste(n) Sie drücken müssen, um das BIOS-Setup zu erreichen, in dem das Verhalten des BIOS konfiguriert werden kann. Typisch hierfür sind F1, F2, F10, Esc, Eingf und Entf. Wenn Ihr BIOS-Startbildschirm durch eine hübsche grafische Anzeige versteckt wird, müssen Sie eventuell eine oder mehrere Tasten wie z.B. Esc drücken, um diese Grafik auszublenden. Die hierzu benötigten Tasten sind je nach Hardware sehr unterschiedlich.

Die Hardware und die Reihenfolge des Codes, der durch das BIOS gestartet wird, kann über das BIOS-Setup ausgewählt werden. Typischerweise werden die ersten paar Sektoren des ersten gefundenen, vorgewählten Gerätes (Festplatte, Diskette, CD-ROM, ...) in den Speicher geladen und dieser initiale Code wird dann ausgeführt. Das kann folgendes sein:

- ein Bootloader-Code;
- der Kernel-Code eines Ursprungs-Betriebssystems wie z.B. **FreeDOS**;
- der Kernel-Code des Ziel-Betriebssystems, falls er in diesen kleinen Speicherbereich passt.

Gewöhnlich wird das System von der angegebenen Partition der primären Festplatte gestartet. Die ersten zwei Sektoren der Festplatte enthalten auf normalen PC-Systemen den **Master Boot Record (MBR)**. Die Partitionsinformationen inklusive der Boot-Auswahl sind am Ende des MBR gespeichert. Der in der ersten Stufe vom BIOS ausgeführte Bootloader-Code liegt im Rest des MBR.

3.1.2 Stufe 2: der Bootloader

Der **Bootloader** ist die zweite Stufe des Boot-Prozesses und wird durch das BIOS gestartet. Er lädt das System-Kernel-Image und das **initrd**-Image in den Speicher und übergibt diesen die Kontrolle. Das **initrd**-Image ist ein Abbild des Wurzeldateisystems und seine Unterstützung hängt von dem verwendeten Bootloader ab.

Das Debian-System nutzt normalerweise den Linux-Kernel als Standard-System-Kernel. Das **initrd**-Image des aktuellen 2.6/3.x-Linux-Kernels ist technisch gesehen ein **initramfs-Image** (initial RAM filesystem). Das Basis-**initrd**-Image ist ein komprimiertes **cpio**-Archiv der Dateien im Wurzeldateisystem. Der Kernel kann sehr früh im Boot-Prozess (bevor dieses **initrd**-Image geladen wird) ein Update des Microcodes durchführen. Dies wird ermöglicht durch das kombinierte **initrd**-Image, das aus dem Binär-Blob des Microcodes und dem daran angehängten Basis-**initrd**-Image besteht.

Tipp

Sie können den Inhalt der **initrd**-Image-Datei mittels **lsinitramfs(8)** und **unmkinitramfs(8)** aus dem Paket **initramfs-tools-core** untersuchen. Details finden Sie unter <https://wiki.debian.org/initramfs>.

Eine Standardinstallation des Debian-Systems auf der PC-Plattform legt den ersten Teil des GRUB-Bootloaders (stage 1) im **MBR** ab. Es sind allerdings viele weitere Bootloader und Konfigurationsoptionen verfügbar.

Paket	Popcon	Größe	initrd	Bootloader	Beschreibung
grub-legacy	V:0, I:2	735	Unterstützt	GRUB Legacy	Intelligenter Bootloader, der Festplattenpartitionen und Dateisysteme wie vfat, ext3 ...unterstützt.
grub-pc	V:28, I:774	533	Unterstützt	GRUB 2	Intelligenter Bootloader, der Festplattenpartitionen und Dateisysteme wie vfat, ext4 ...unterstützt. (Debian-Standard)
grub-rescue-pc	V:0, I:1	6367	Unterstützt	GRUB 2	Dies ist das boot-fähige Rettungs-Image von GRUB 2 (CD und Diskette) (PC/BIOS-Version).
lilo	V:0, I:2	697	Unterstützt	Lilo	Beruh auf den Sektorbereichen der Daten auf der Festplatte (alt).
syslinux	V:4, I:48	343	Unterstützt	Isolinux	Unterstützt das ISO9660-Dateisystem. Dies wird von Boot-CDs verwendet.
syslinux	V:4, I:48	343	Unterstützt	Syslinux	Unterstützt das MSDOS-Dateisystem FAT . Dies wird von Boot-Disketten verwendet.
loadlin	V:0, I:1	90	Unterstützt	Loadlin	Das gewünschte Betriebssystem wird aus dem laufenden FreeDOS-/MSDOS-System heraus gestartet.
mbr	V:0, I:7	50	Nicht unterstützt	MBR von Neil Turton	Dies ist freie Software, die den MSDOS-MBR ersetzt. Unterstützt nur Festplattenpartitionen.

Tabelle 3.1: Liste der Bootloader

**Warnung**

Spielen Sie nicht mit Bootloadern herum, ohne boot-fähige Rettungsmedien (USB-Stick, CD, Diskette) zur Hand zu haben, die von Images im `grub-rescue-pc`-Paket erstellt wurden. Damit können Sie Ihr System auch ohne funktionsfähigen Bootloader auf der Festplatte starten.

Bei GRUB Legacy ist die Datei zur Konfiguration des Menüs unter `"/boot/grub/menu.lst"` abgelegt. Sie enthält zum Beispiel Einträge wie den folgenden:

```
title          Debian GNU/Linux
root           (hd0,2)
kernel        /vmlinuz root=/dev/hda3 ro
initrd        /initrd.img
```

Bei GRUB 2 ist die Datei zur Konfiguration des Menüs unter `"/boot/grub/grub.cfg"` gespeichert. Sie wird automatisch durch `"/usr/sbin/update-grub"` unter Verwendung von Vorlagen aus `"/etc/grub.d/*"` und Einstellungen aus `"/etc/default/grub"` erzeugt. Die Einträge sehen aus wie folgt:

```
menuentry "Debian GNU/Linux" {
    set root=(hd0,3)
    linux /vmlinuz root=/dev/hda3
    initrd /initrd.img
}
```

Die GRUB-Parameter in diesen Beispielen haben folgende Bedeutungen:

Anmerkung

Der Wert der Partitionsnummer, wie er von GRUB Legacy verwendet wird, ist um 1 kleiner als der normal durch den Linux-Kernel und andere Werkzeuge genutzte. Das GRUB-2-Programm behebt dieses Problem.

GRUB-Parameter	Bedeutung
root	Verwendung der dritten Partition auf der primären Festplatte, welche bei GRUB Legacy über "(hd0, 2)" angegeben wird und bei GRUB 2 über "(hd0, 3)".
kernel	Verwendung des Kernels unter "/vmlinuz" mit den Parametern "root=/dev/hda3 ro"
initrd	Verwendung des initrd/initramfs -Images unter "/initrd.img"

Tabelle 3.2: Bedeutung der GRUB-Parameter

Tipp

Die [UUID](#)-Kennung (lesen Sie dazu Abschnitt [9.5.3](#)) kann statt dem Gerätenamen (z.B. "/dev/hda3") genutzt werden, um ein block-orientiertes Gerät eindeutig zu identifizieren, z.B. mittels "root=UUID=81b289d5-4341-4003-9602-e254a17ac232 ro".

Tipp

Bei der Nutzung von [GRUB](#) sind die Kernel-Boot-Parameter in /boot/grub/grub.cfg festgelegt. Auf Debian-Systemen sollten Sie /boot/grub/grub.cfg jedoch nicht direkt verändern. Ändern Sie stattdessen den GRUB_CMDLINE_LINUX_DEFAULT-Wert in /etc/default/grub und führen Sie dann update-grub(8) aus, um /boot/grub/grub.cfg zu aktualisieren.

Tipp

Sie können über einen Bootloader einen weiteren Bootloader starten, diese Technik nennt man [Chain loading](#).

Weitere Infos finden Sie unter "info grub" und grub-install(8).

3.1.3 Stufe 3: das Mini-Debian-System

Das Mini-Debian-System ist die dritte Stufe des Boot-Prozesses und wird durch den Bootloader gestartet. Es lässt den System-Kernel mit seinem eigenen Wurzeldateisystem im Speicher laufen. Dies ist ein optionaler, vorbereitender Schritt des Boot-Prozesses.

Anmerkung

Der Begriff "Mini-Debian-System" wurde von dem Autor erfunden, um diese dritte Stufe des Boot-Prozesses in diesem Dokument zu beschreiben. Dieses System wird normalerweise [initrd](#)- oder [initramfs](#)-System genannt. Ein ähnliches System wird im Speicher auch durch den [Debian Installer](#) verwendet.

"/init" wird als erstes Programm aus diesem Wurzeldateisystem im Speicher ausgeführt. Es ist ein Programm, das den Kernel im Userspace initialisiert und die Kontrolle an die nächste Stufe übergibt. Dieses Mini-Debian-System bietet Flexibilität für den Boot-Prozess, um zum Beispiel Kernel-Module vor dem Hauptteil des Boot-Prozesses hinzuzufügen oder um das Wurzeldateisystem als verschlüsseltes Dateisystem einzubinden.

- "/init" ist ein Shell-Skript, wenn das [initramfs](#) durch [initramfs-tools](#) erstellt wurde.
 - Sie können diesen Teil des Boot-Prozesses unterbrechen, um eine root-Shell zu bekommen, indem Sie "break=init" usw. zu den Kernel-Boot-Parametern hinzufügen. Informationen zu weiteren Unterbrechungsmöglichkeiten finden Sie im "/init"-Skript. Diese Shell-Umgebung ist ausgeklügelt genug, um eine gute Überprüfung der Hardware Ihrer Maschine zu ermöglichen.

- Die verfügbaren Befehle in diesem Mini-Debian-System gehen auf ein GNU-Werkzeug namens `busybox(1)` zurück und werden auch hauptsächlich von diesem bereitgestellt.
- `"/init"` ist ein binäres `systemd`-Programm, wenn das `initramfs` durch `dracut` erstellt wurde.
- Befehle in diesem Mini-Debian-System sind in ihrer Funktionalität auf die `systemd(1)`-Umgebung reduziert.

**Achtung**

Sie müssen die Option `"-n"` für den `mount`-Befehl verwenden, wenn Sie sich im Nur-Lese-Wurzeldateisystem befinden.

3.1.4 Stufe 4: das normale Debian-System

Das normale Debian-System ist die vierte Stufe des Boot-Prozesses und wird von dem Mini-Debian-System gestartet. Der System-Kernel des Mini-Debian-Systems läuft in dieser Umgebung weiter. Das verwendete Wurzeldateisystem wird von dem im Arbeitsspeicher umgeschwenkt zu dem auf der echten Festplatte.

Das Programm `init` wird als erstes Programm mit `PID=1` ausgeführt und erledigt die eigentliche Hauptarbeit beim Booten, das Starten verschiedener Programme. Der Standardpfad zum `init`-Programm ist `"/sbin/init"`, aber er kann über einen Kernel-Boot-Parameter wie `"init=/pfad/zum/init-programm"` auch geändert werden.

Das Standard-Init-Programm hat sich im Laufe der Zeit verändert:

- Vor Squeeze verwendete Debian das einfache [SysV](#)-artige Init.
- Debian Wheezy verbesserte das SysV-ähnliche Init durch Sortierung der Boot-Reihenfolge mittels LSB-Header und parallelen Start der Boot-Skripte.
- Debian Jessie schwenkt um auf [systemd](#), ein ereignisbasiertes Init-System mit paralleler Initialisierung.

Tipp

Mittels `"ps -p 1 -f"` können Sie überprüfen, welcher `init`-Befehl letztlich auf Ihrem System verwendet wird.

Tipp

`"/sbin/init"` wurde nach Debian Jessie ein symbolischer Link auf `"/lib/systemd/systemd"`.

Tipp

Unter [Debian Wiki: BootProcessSpeedup](#) finden Sie aktuelle Tipps zur Beschleunigung des Boot-Prozesses.

3.2 Systemd-Init

Dieser Abschnitt beschreibt, wie das System durch das `systemd(1)`-Programm mit der `PID=1` (also dem `init`-Prozess) gestartet wird.

Der `systemd`-Init-Prozess wird - basierend auf den Unit-Konfigurationsdateien (siehe `systemd.unit(5)`) - in mehrere parallele Prozesse aufgespalten; diese Konfigurationsdateien sind in deklarativem Stil geschrieben, im Unterschied zu dem prozeduralen Stil von SysV. Diese können von einer Reihe von Dateisystempfaden geladen werden (siehe `systemd-system.conf(5)`), die hier aufgeführt sind:

Paket	Popcon	Größe	Beschreibung
systemd	V:810, I:916	15998	Ereignis-basierter init(8)-Daemon für gleichzeitige Ausführung (Alternative zu sysvinit)
systemd-sysv	V:802, I:914	138	die Handbuchseiten und Links, die nötig sind, um sysvinit durch systemd zu ersetzen
systemd-cron	V:1, I:1	143	systemd -Units, um die Funktionalitäten des cron -Daemons sowie von anacron bereitzustellen
init-system-helpers	V:675, I:930	131	Hilfsprogramme, um zwischen sysvinit und systemd umschalten zu können
initscripts	V:91, I:323	176	Skripte zur Initialisierung und zum Herunterfahren des Systems
sysvinit-core	V:7, I:9	276	System-V-ähnliche init(8)-Werkzeuge
sysv-rc	V:183, I:335	81	System-V-ähnlicher Mechanismus zum Wechsel des Runlevels
sysvinit-utils	V:494, I:999	79	System-V-ähnliche Werkzeuge (startpar(8) , bootlogd(8) , ...)
lsb-base	V:881, I:999	49	Zur Linux Standard Base 3.2 konforme init-Skript-Funktionalität
insserv	V:210, I:330	150	Werkzeug, um die Boot-Reihenfolge unter Verwendung von LSB-konformen init.d-Skript-Abhängigkeiten zu organisieren
uswsusp	V:3, I:8	714	Werkzeuge, um das durch Linux bereitgestellte Userspace-Software-Suspend zu verwenden
kexec-tools	V:1, I:8	278	Werkzeug für kexec(8) -Neustarts (Warmstarts)
systemd-bootchart	V:0, I:1	128	Performance-Analyseprogramm für den Boot-Prozess
bootchart2	V:0, I:0	94	Performance-Analyseprogramm für den Boot-Prozess
pybootchartgui	V:0, I:0	177	Performance-Analyseprogramm für den Boot-Prozess (Visualisierung)
mingetty	V:0, I:3	38	getty(8) nur für die Konsole
mgetty	V:0, I:1	315	Intelligenter getty(8) -Ersatz für Modems

Tabelle 3.3: Liste von Boot-Hilfsprogrammen für das Debian-System

- `"/lib/systemd/system"`: Standard-Konfigurationsdateien des Betriebssystems
- `"/etc/systemd/system"`: Konfigurationsdateien zur Systemadministration, welche die Standard-Konfigurationsdateien des Betriebssystems überschreiben
- `"/run/systemd/system"`: zur Laufzeit generierte Konfigurationsdateien, welche die fest installierten Konfigurationsdateien überschreiben

Deren Abhängigkeiten untereinander sind durch die Regeln `"Wants="`, `"Requires="`, `"Before="`, `"After="`, ... (siehe `"MAPPING OF UNIT PROPERTIES TO THEIR INVERSES"` in `systemd.unit(5)`) definiert. Die Ressourcen-Steuerung ist ebenfalls festgelegt (siehe `systemd.resource-control(5)`).

Die Endung der Unit-Konfigurationsdateien definiert ihren Typ wie folgt:

- ***.service** beschreibt einen Prozess (Dienst), der von `systemd` gesteuert und überwacht wird. Siehe `systemd.service(5)`.
- ***.device** beschreibt ein Gerät, das im `sysfs(5)` als `udev(7)`-Gerätedatei abgebildet ist. Siehe `systemd.device(5)`.
- ***.mount** beschreibt einen Einbindungspunkt im System, der von `systemd` gesteuert und überwacht wird. Siehe `systemd.mount(5)`.
- ***.automount** beschreibt einen automatischen Einbindungspunkt im System, der von `systemd` gesteuert und überwacht wird. Siehe `systemd.automount(5)`.
- ***.swap** beschreibt ein Swap-Device oder eine Swap-Datei (zum Auslagern von Arbeitsspeicher auf eine Festplatte), die von `systemd` gesteuert und überwacht wird. Siehe `systemd.swap(5)`.
- ***.path** beschreibt einen Pfad im Dateisystem, der von `systemd` zum Zwecke der pfad-basierten Aktivierung überwacht wird. Siehe `systemd.path(5)`.
- ***.socket** beschreibt einen Socket, der von `systemd` zum Zwecke der socket-basierten Aktivierung gesteuert und überwacht wird. Siehe `systemd.socket(5)`.
- ***.timer** beschreibt einen Timer (Zeitgeber), der von `systemd` zum Zwecker der timer-basierten Aktivierung gesteuert und überwacht wird. Siehe `systemd.timer(5)`.
- ***.slice** verwaltet Ressourcen über `cgroups(7)`. Siehe `systemd.slice(5)`.
- ***.scope** wird programmgesteuert über die Busschnittstellen von `systemd` erzeugt, um Systemprozesse zu verwalten. Siehe `systemd.scope(5)`.
- ***.target** fasst andere Unit-Konfigurationsdateien zu Gruppen zusammen, um Synchronisierungspunkte für den Startprozess zu erstellen. Siehe `systemd.target(5)`.

Während des Systemstarts versucht der `systemd`-Prozess, das Target `"/lib/systemd/system/default.target` (normalerweise ein symbolischer Link auf `"graphical.target"`) zu starten. Als erstes werden dabei einige spezielle Target-Units (siehe `systemd.special(7)`) wie `"local-fs.target"`, `"swap.target"` und `"cryptsetup.target"` aktiviert, um die Dateisysteme einzubinden. Dann werden über die Abhängigkeiten weitere Target-Units aktiviert. Details finden Sie in `bootup(7)`.

`systemd` enthält Funktionalitäten, um die Rückwärtskompatibilität zu SysV zu gewährleisten. Boot-Skripte im SysV-Stil in `"/etc/init.d/rc[0123456S].d/[KS]<name>"` werden immer noch abgearbeitet, und `telinit(8)`-Befehle werden in Aktivierungsanforderungen für `systemd`-Units übersetzt.

**Achtung**

Die emulierten Runlevel 2 bis 4 verweisen über symbolische Links alle auf des gleiche `"multi-user.target"`.

3.2.1 Der Rechnername

Der Kernel verwaltet den Rechnernamen (**hostname**) des Systems. Die durch `systemd-hostnamed.service` gestartete System-Unit setzt beim Systemstart den Rechnernamen auf den in `"/etc/hostname"` festgelegten Wert. Diese Datei sollte **nur** den Rechnernamen des Systems enthalten, nicht einen vollqualifizierten Domännennamen.

Um den derzeitigen Rechnernamen auszugeben, führen Sie `hostname(1)` ohne ein Argument aus.

3.2.2 Das Dateisystem

Die Optionen zum Einbinden normaler Festplatten- und Netzwerkdateisysteme werden in `"/etc/fstab"` festgelegt. Siehe `fstab(5)` und Abschnitt 9.5.7.

Die Konfiguration verschlüsselter Dateisysteme ist in `"/etc/crypttab"` abgelegt. Siehe `crypttab(5)`.

Software-RAID mit `mdadm(8)` wird in `"/etc/mdadm/mdadm.conf"` konfiguriert. Siehe `mdadm.conf(5)`.



Warnung

Bei jedem Systemstart werden nach dem Einbinden aller Dateisysteme temporäre Dateien in `"/tmp"`, `"/var/lock"` und `"/var/run"` gelöscht.

3.2.3 Initialisierung der Netzwerkschnittstellen

Auf modernen Debian-Desktop-Systemen mit `systemd` erfolgt die Initialisierung von Netzwerkschnittstellen für die Loopback-Schnittstelle `lo` typischerweise durch `networking.service` und für andere Schnittstellen durch `NetworkManager.service`.

Details zur Konfiguration finden Sie in Kapitel 5.

3.2.4 Die Kernel-Meldungen

Die angezeigten Fehlermeldungen des Kernels auf der Konsole können über einen Schwellwert gefiltert werden:

```
# dmesg -n3
```

Wert	Name	Bedeutung
0	KERN_EMERG	System ist unbenutzbar
1	KERN_ALERT	es ist unverzüglich eine Aktion erforderlich
2	KERN_CRIT	Zustände mit kritischen Fehlern
3	KERN_ERR	Zustände mit Fehlern
4	KERN_WARNING	Zustände mit Warnungen
5	KERN_NOTICE	normale, aber erwähnenswerte Zustände
6	KERN_INFO	rein informativ
7	KERN_DEBUG	Nachrichten zur Fehlersuche/-eingrenzung

Tabelle 3.4: Liste der Schwellwerte zur Filterung von Kernel-Fehler-Meldungen

3.2.5 Die Systemmeldungen

Unter `systemd` werden sowohl Kernel- wie auch Systemmeldungen durch den Journal-Dienst `systemd-journald.service` (a.k.a `journald`) protokolliert, entweder in Form von Binärdaten unterhalb von `"/var/log/journal"` oder in flüchtigen Binärdaten in `"/run/log/journal/"`. Diese binären Protokolldaten können mit dem Befehl `journalctl(1)` abgefragt werden.

Unter `systemd` hat das Protokollprogramm `rsyslogd(8)` sein Verhalten geändert: es liest jetzt die flüchtigen Binär-Logdaten (statt der Daten aus `"/dev/log"`, wie es vor `systemd` Standardeinstellung war) und erstellt traditionelle dauerhaft gespeicherte ASCII-Logdateien.

Die Systemmeldungen können über `"/etc/default/rsyslog"` und `"/etc/rsyslog.conf"` sowohl für die Protokoll-dateien wie auch für die Bildschirmanzeige angepasst werden. Lesen Sie dazu `rsyslogd(8)` und `rsyslog.conf(5)` sowie auch Abschnitt [9.2.2](#).

3.2.6 Systemmanagement unter systemd

`systemd` bietet nicht nur das eigentliche `init`-System zum Starten des Systems, sondern auch Funktionalitäten zum Systemma-nagement wie Logdateien, Anmeldedaten, Netzwerkverbindungen usw.

Das Systemmanagement unter `systemd(1)` läuft über eine Reihe von Befehlen:

- der Befehl `systemctl(1)` steuert den System- und Dienste-Manager (für die Konsolenoberfläche, CLI);
- der Befehl `systemsdm(1)` steuert den System- und Dienste-Manager (für die grafische Bedienoberfläche, GUI);
- der Befehl `journalctl(1)` bedient Anfragen an das Journalsystem (Logdaten);
- der Befehl `loginctl(1)` steuert den Anmeldemanager;
- der Befehl `systemd-analyze(1)` analysiert die Performance beim Booten des Systems.

Hier eine Liste typischer `systemd`-Management-Befehle. Bezüglich der exakten Bedeutungen lesen Sie bitte die zugehörigen Handbuchseiten.

In obigen Beispielen kann `"$unit"` für einen einzelnen Unit-Namen stehen (ein Anhang wie `.service` oder `.target` ist dabei optional), oder auch für die Angabe mehrerer Units (über Suchmuster im Shell-Stil wie `"**"`, `"?"` oder `"["`], die `fnmatch(3)` verwenden; diese werden auf die primären Namen aller Units angewandt, die derzeit in den Arbeitsspeicher geladen sind).

Befehlen zum Ändern des Systemstatus wird typischerweise ein `"sudo"` vorangestellt, um die nötigen administrativen Rechte anzufordern.

Die Ausgabe von `"systemctl status $unit | $PID | $device"` nutzt farbige Punkte ("**●**"), um den Unit-Status kompakt zusammenzufassen.

- Ein weisser "**●**" steht für "inaktiv" oder "deaktiviert".
- Ein roter "**●**" steht für "fehlgeschlagen" oder "Fehler".
- Ein grüner "**●**" steht für "aktiv", "wird neu geladen" oder "wird aktiviert".

3.2.7 Anpassen von systemd

Bei einer Standardinstallation werden viele Netzwerkdienste (siehe Kapitel [6](#)) von `systemd` durch `network.target` als Daemon-Prozess gestartet. `"sshd"` ist hier keine Ausnahme. Als Beispiel dafür, wie man so etwas anpassen kann, wollen wir zeigen, wie Sie `"sshd"` so ändern, dass er nur auf Anfrage (on-demand) startet.

Als erstes deaktivieren Sie die entsprechende Dienst-Unit:

```
$ sudo systemctl stop sshd.service
$ sudo systemctl mask sshd.service
```

Der klassische Weg zur on-demand-Aktivierung von Sockets führte früher über den `inetd`-Superserver. Unter `systemd` kann dies über das Hinzufügen von `*.socket` und `*.service` Unit-Konfigurationsdateien erreicht werden.

Eine `sshd.socket` anlegen zur Spezifizierung eines Sockets, der auf Anfragen überwacht wird:

Tätigkeit	Typ	Befehl
grafische Oberfläche (GUI) für den Service-Manager	GUI	"systemadm" (aus dem systemd-ui-Paket)
Unit-Konfiguration aller Targets auflisten	Unit	"systemctl list-units --type=target"
Unit-Konfiguration aller Dienste (Services) auflisten	Unit	"systemctl list-units --type=service"
Typen aller Unit-Konfigurationen auflisten	Unit	"systemctl list-units --type=help"
Alle Socket-Units im Arbeitsspeicher auflisten	Unit	"systemctl list-sockets"
Alle Timer-Units im Arbeitsspeicher auflisten	Unit	"systemctl list-timers"
"\$unit" starten	Unit	"systemctl start \$unit"
"\$unit" stoppen	Unit	"systemctl stop \$unit"
Dienst-spezifische Konfiguration neu laden	Unit	"systemctl reload \$unit"
"\$unit" stoppen und neu starten	Unit	"systemctl restart \$unit"
"\$unit" starten und alle anderen stoppen	Unit	"systemctl isolate \$unit"
Zur grafischen Oberfläche wechseln (GUI-System)	Unit	"systemctl isolate graphical"
Zur Konsolenoberfläche wechseln (Mehrbenutzer-CLI-System)	Unit	"systemctl isolate multi-user"
Zur Rettungssystem-Oberfläche wechseln (Einzelbenutzer-CLI-System)	Unit	"systemctl isolate rescue"
Kill-Signal an "\$unit" senden	Unit	"systemctl kill \$unit"
Prüfen, ob "\$unit" aktiv ist	Unit	"systemctl is-active \$unit"
Prüfen, ob "\$unit" fehlgeschlagen ist	Unit	"systemctl is-failed \$unit"
Status von "\$unit \$PID device" prüfen	Unit	"systemctl status \$unit \$PID \$device"
Eigenschaften von "\$unit \$job" anzeigen	Unit	"systemctl show \$unit \$job"
Fehlgeschlagene "\$unit" zurücksetzen (Reset)	Unit	"systemctl reset-failed \$unit"
Abhängigkeiten aller Unit-Dienste auflisten	Unit	"systemctl list-dependencies --all"
Auf dem System installierte Unit-Dateien auflisten	Unit-Datei	"systemctl list-unit-files"
"\$unit" aktivieren (symbolischen Link hinzufügen)	Unit-Datei	"systemctl enable \$unit"
"\$unit" deaktivieren (symbolischen Link entfernen)	Unit-Datei	"systemctl disable \$unit"
"\$unit" zum Starten bereit machen (symbolischen Link auf "/dev/null" entfernen)	Unit-Datei	"systemctl unmask \$unit"
"\$unit" am Starten hindern (symbolischen Link auf "/dev/null" hinzufügen)	Unit-Datei	"systemctl mask \$unit"
Aktuelles default-Target abrufen	Unit-Datei	"systemctl get-default"
default-Target auf "graphical" setzen (grafische Oberfläche, GUI)	Unit-Datei	"systemctl set-default graphical"
default-Target auf "multi-user" setzen (Konsolenoberfläche, CLI)	Unit-Datei	"systemctl set-default multi-user"
Job-Umgebungseinstellungen anzeigen	Umgebung	"systemctl show-environment"
Job-Umgebungseinstellung "variable" auf "wert" setzen	Umgebung	"systemctl set-environment variable=wert"
Job-Umgebungseinstellung	Umgebung	"systemctl unset-environment

```
[Unit]
Description=SSH Socket for Per-Connection Servers

[Socket]
ListenStream=22
Accept=yes

[Install]
WantedBy=sockets.target
```

Und eine `sshd@.service` als Dienste-Datei passend zu `sshd.socket`:

```
[Unit]
Description=SSH Per-Connection Server

[Service]
ExecStart=-/usr/sbin/sshd -i
StandardInput=socket
```

Dann muss der Dienst neu geladen werden:

```
$ sudo systemctl daemon-reload
```

3.3 Das udev-System

Für Linux-Kernel der 2.6-Reihe und neuer bietet das [udev-System](#) Mechanismen für automatische Hardware-Erkennung und -initialisierung (lesen Sie dazu [udev\(7\)](#)). Nach der Erkennung eines Gerätes durch den Kernel startet das udev-System einen User-Prozess. Dieser verwendet Informationen aus dem [sysfs](#)-Dateisystem (Näheres in Abschnitt [1.2.12](#)), lädt über den Befehl [modprobe\(8\)](#) benötigte Kernel-Module, die die Hardware unterstützen (Details in Abschnitt [3.3.1](#)), und erstellt die zugehörigen Geräteknoten (device nodes).

Tipp

Falls `"/lib/modules/<kernel-version>/modules.dep"` mittels `depmod(8)` aus irgendeinem Grund nicht korrekt erstellt wurde, könnte es beim Laden der Module durch das udev-System Probleme geben. Führen Sie `"depmod -a"` aus, um dies zu beheben.

Die Namen der Geräteknoten können über udev-Regel-Dateien in `"/etc/udev/rules.d/"` konfiguriert werden. Aktuelle Standardregeln neigen dazu, dynamisch generierte Namen zu erzeugen, was (außer bei CD- und Netzwerkgeräten) dazu führt, dass sich die Gerätenamen von Mal zu Mal ändern. Indem Sie Ihre eigenen Regeln hinzufügen (ähnlich denen für CD- und Netzwerkgeräte), können Sie auch für andere Geräte wie z.B. USB-Memory-Sticks fest zugeordnete Gerätenamen vergeben. Lesen Sie dazu ["Writing udev rules"](#) oder `"/usr/share/doc/udev/writing_udev_rules/index.html"`.

Da das udev-System immer ein wenig im Wandel ist, überlasse ich die Details anderen Dokumenten und beschränke mich hier auf das Nötigste.

Tipp

Für die Regeln zum Einbinden von Dateisystemen in `"/etc/fstab"` müssen Geräteknoten nicht fest zugeordnet sein. Sie können auch [UUIDs](#) verwenden, um Geräte einzubinden, statt der Gerätenamen wie `"/dev/sda"`. Lesen Sie dazu Abschnitt [9.5.3](#).

3.3.1 Die Kernel-Modul-Initialisierung

Das `modprobe(8)`-Programm erlaubt es, einen laufenden Linux-Kernel über einen User-Prozess zu konfigurieren, indem Kernel-Module hinzugefügt und entfernt werden. Das `udev`-System (Näheres in Abschnitt 3.3) automatisiert dessen Aufruf, um bei der Initialisierung des Kernel-Moduls zu helfen.

Es gibt Module, die nicht zu bestimmter Hardware gehören, sowie spezielle Hardware-Treibermodule wie die folgenden, die im Voraus geladen werden müssen, indem Sie in die Datei `"/etc/modules"` eingetragen werden (Details in `modules(5)`):

- [TUN/TAP](#)-Module, die ein virtuelles Point-to-Point Netzwerkgerät (TUN) und ein virtuelles Ethernet-Netzwerkgerät (TAP) bereitstellen;
- [netfilter](#)-Module, die Netfilter-Firewall-Funktionalitäten bereitstellen (lesen Sie dazu `iptables(8)` und Abschnitt 5.10);
- [watchdog timer](#)-Treibermodule.

Die Konfigurationsdateien für das `modprobe(8)`-Programm sind unterhalb des `"/etc/modprobes.d/"`-Verzeichnisses abgelegt, wie in `modprobe.conf(5)` beschrieben. (Falls Sie vermeiden möchten, dass einige Kernel-Module automatisch geladen werden, sollten Sie erwägen, diese in die Datei `"/etc/modprobes.d/blacklist"` einzutragen.)

Die Datei `"/lib/modules/<version>/modules.dep"` (erzeugt durch das Programm `depmod(8)`) beschreibt Abhängigkeiten zwischen den Modulen; diese Abhängigkeiten werden von `modprobe(8)` genutzt.

Anmerkung

Wenn Sie Probleme beim Laden von Modulen feststellen, entweder während des Systemstarts oder beim Nachladen mit `modprobe(8)`, kann `"depmod -a"` diese Probleme möglicherweise durch Neuerstellung der `"modules.dep"`-Datei beheben.

Der Befehl `modinfo(8)` zeigt Informationen über ein Linux-Kernel-Modul an.

Das `lsmod(8)`-Programm formatiert den Inhalt von `"/proc/modules"` zu einer hübschen Ausgabe, um anzuzeigen, welche Kernel-Module gerade geladen sind.

Tipp

Sie können die Hardware in Ihrem System exakt identifizieren. Lesen Sie dazu Abschnitt 9.4.3.

Tipp

Möglicherweise wollen Sie Hardware während des Systemstarts konfigurieren, um bestimmte erwartete Hardware-Funktionalitäten zu aktivieren. Näheres finden Sie in Abschnitt 9.4.4.

Tipp

Unterstützung für spezielle Geräte können Sie unter Umständen durch Neukompilieren des Kernels hinzufügen. Details finden Sie in Abschnitt 9.9.

Kapitel 4

Authentifizierung

Wenn eine Person (oder ein Programm) Zugriff auf das System erlangen möchte, wird über die Authentifizierung kontrolliert, ob der Identität vertraut werden kann.

**Warnung**

Konfigurationsfehler von PAM können Sie aus Ihrem eigenen System aussperren. Sie müssen eine Rettungs-CD zur Hand haben oder eine alternative Boot-Partition einrichten. Um das System wiederherzustellen, booten Sie das System damit und korrigieren Sie alles Nötige von dort.

**Warnung**

Dieses Kapitel ist überholt, da es auf dem in 2013 veröffentlichten Debian 7.0 (wheezy) basiert.

4.1 Normale Unix-Authentifizierung

Die normale Unix-Authentifizierung wird über das Modul `pam_unix(8)` von [PAM \(Pluggable Authentication Modules\)](#) bereitgestellt. Dessen drei wichtige Konfigurationsdateien (mit durch `”:` getrennten Einträgen) sind:

Datei	Berechtigung	Benutzer	Gruppe	Beschreibung
<code>/etc/passwd</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	(bereinigte) Informationen zum Benutzerkonto
<code>/etc/shadow</code>	<code>-rw-r-----</code>	<code>root</code>	<code>shadow</code>	geschützte Informationen zum Benutzerkonto
<code>/etc/group</code>	<code>-rw-r--r--</code>	<code>root</code>	<code>root</code>	Informationen zur Gruppe

Tabelle 4.1: Wichtige Konfigurationsdateien für `pam_unix(8)`

`”/etc/passwd”` enthält Folgendes:

```
...
benutzer1:x:1000:1000:Name von Benutzer1,,,:/home/benutzer1:/bin/bash
benutzer2:x:1001:1001:Name von Benutzer2,,,:/home/benutzer2:/bin/bash
...
```

Wie in `passwd(5)` beschrieben haben die durch `”:` separierten Einträge dieser Datei folgende Bedeutungen:

- Benutzername (Login-Name);
- Eintrag zur Passwortspezifikation;
- numerische Benutzer-ID;
- numerische Gruppen-ID;
- Name oder Kommentarfeld;
- Home-Verzeichnis des Benutzers;
- optionale Angabe des Befehlsinterpreters für den Benutzer.

Der zweite Eintrag von `/etc/passwd` wurde früher für das verschlüsselte Passwort benutzt. Seit der Einführung von `/etc/shadow` enthält er die Passwortspezifikation:

Inhalt	Bedeutung
(leer)	Konto ohne Passwort
x	das verschlüsselte Passwort ist in <code>/etc/shadow</code> abgelegt
*	kein Login für dieses Konto
!	kein Login für dieses Konto

Tabelle 4.2: Inhalt des zweiten Eintrags in `/etc/passwd`

`/etc/shadow` enthält Folgendes:

```
...
benutzer1:$1$Xop0FYH9$IfxyQwBe9b8tiyIkt2P4F/:13262:0:99999:7:::
benutzer2:$1$vXGZLVbS$ElyErNf/agUDsm1DehJMS/:13261:0:99999:7:::
...
```

Wie in `shadow(5)` beschrieben haben die durch `:` separierten Einträge dieser Datei folgende Bedeutungen:

- Benutzername (Login-Name);
- verschlüsseltes Passwort (das `1` am Anfang zeigt die Verwendung der MD5-Verschlüsselung an; `***` steht für "kein Login");
- Datum der letzten Passwortänderung, ausgedrückt als Anzahl der Tage seit dem 1. Januar 1970;
- Anzahl der Tage, die der Benutzer vor der erneuten Änderung des Passworts warten muss;
- Anzahl der Tage, nach denen der Benutzer das Passwort ändern muss;
- Anzahl der Tage vor dem Verfall des Passworts, während derer der Benutzer vor dem Passwortverfall gewarnt wird;
- Anzahl der Tage, während derer das Passwort noch akzeptiert wird, obwohl es abgelaufen ist;
- Datum, an dem das Konto abläuft, ausgedrückt als Anzahl der Tage seit dem 1. Januar 1970.
- ...

`/etc/group` enthält Folgendes:

```
gruppe1:x:20:benutzer1,benutzer2
```

Wie in `group(5)` beschrieben haben die durch `:` separierten Einträge dieser Datei folgende Bedeutungen:

- Gruppenname;
- verschlüsseltes Passwort (nicht wirklich benutzt);

- numerische Gruppen-ID;
- durch „;“ getrennte Liste von Benutzernamen.

Anmerkung

„/etc/gshadow“ bietet für „/etc/group“ ähnliche Funktionalität wie „/etc/shadow“, diese wird aber nicht wirklich genutzt.

Anmerkung

Die aktuelle Gruppenmitgliedschaft eines Benutzers kann dynamisch angepasst werden, wenn eine Zeile mit „auth optional pam_group.so“ zu „/etc/pam.d/common-auth“ hinzugefügt wird und wenn dies in „/etc/security/group.conf“ gesetzt ist. Lesen Sie dazu pam_group(8).

Anmerkung

Das base-passwd-Paket enthält eine verbindliche Liste von Benutzern und Gruppen: „/usr/share/doc/base-passwd/users-and-groups.html“.

4.2 Verwalten von Konten- und Passwortinformationen

Hier einige erwähnenswerte Befehle zur Verwaltung von Konteninformationen:

Befehl	Funktion
getent passwd <benutzername>	Konteninformationen von <benutzername> anzeigen
getent shadow <benutzername>	durch shadow geschützte Konteninformationen von <benutzername> anzeigen
getent group <gruppenname>	Gruppeninformationen von <gruppenname> anzeigen
passwd	Passwort für das Konto verwalten
passwd -e	Einmal-Passwort für die Kontenaktivierung setzen
chage	Alterungsinformationen des Passworts verwalten

Tabelle 4.3: Liste von Befehlen zur Verwaltung von Konteninformationen

Sie benötigen für einige Funktionen root-Privilegien. Lesen Sie crypt(3) für Informationen zur Passwort- und Datenverschlüsselung.

Anmerkung

Auf Systemen, die mit PAM und NSS eingerichtet sind, wie der Debian-Salsa-Maschine, wird der Inhalt der lokalen Dateien „/etc/passwd“, „/etc/group“ und „/etc/shadow“ auf dem System unter Umständen nicht aktiv verwendet. Obige Befehle sind aber auch in solchen Umgebungen gültig.

4.3 Ein gutes Passwort

Bei der Erstellung eines Kontos während der Systeminstallation oder mit dem passwd(1)-Befehl sollten Sie ein [gutes Passwort](#) auswählen, das aus mindestens sechs bis acht Zeichen besteht und ein oder mehrere Zeichen aus folgenden Gruppen enthält (gemäß passwd(1)):

- Kleinbuchstaben;

- Ziffern zwischen 0 und 9;
- Satzzeichen.



Warnung

Wählen Sie keine Wörter für das Passwort, die erraten werden könnten: Kontoname, Sozialversicherungsnummer, Telefonnummer, Adresse, Geburtstag, Name von Familienmitgliedern oder Haustieren, Wörter, die in Lexika auftauchen, einfache Zeichenfolgen wie "12345" oder "qwerty", ...alle diese sind eine schlechte Wahl für ein Passwort.

4.4 Verschlüsselte Passwörter erstellen

Es gibt mehrere unabhängige Werkzeuge, um [verschlüsselte Passwörter mit Salz](#) zu erzeugen:

Paket	Popcon	Größe	Befehl	Funktion
whois	V:35, I:393	364	<code>mkpasswd</code>	best-ausgestattetes Frontend für die <code>crypt(3)</code> -Bibliothek
openssl	V:794, I:993	1465	<code>openssl passwd</code>	Passwort-Hashes berechnen (OpenSSL). <code>passwd(1ssl)</code>

Tabelle 4.4: Liste der Werkzeuge zur Passwörterzeugung

4.5 PAM und NSS

Moderne [Unix-ähnliche](#) Systeme wie Debian stellen dem lokalen Systemadministrator die Mechanismen [PAM \(Pluggable Authentication Modules\)](#) und [NSS \(Name Service Switch\)](#) zur Systemkonfiguration bereit. Deren Funktionen können wie folgt zusammengefasst werden:

- PAM bietet flexible Authentifizierungsmechanismen, die von Anwendungen genutzt werden, und integriert daher auch den Austausch von Passwortdaten.
- NSS besitzt einen flexiblen Name-Service-Mechanismus, der oft von der [C-Standard-Bibliothek](#) genutzt wird, um die Benutzer- und Gruppennamen für Programme wie `ls(1)` und `id(1)` einzuholen.

Diese PAM- und NSS-Systeme müssen konsistent konfiguriert sein.

Hier einige erwähnenswerte Pakete von PAM- und NSS-Systemen:

- Der "Linux-PAM System Administrators' Guide" in `libpam-doc` ist zum Erlernen der PAM-Konfiguration unerlässlich.
- Zum Erlernen der NSS-Konfiguration ist der Abschnitt "System Databases and Name Service Switch" in `glibc-doc-reference` unerlässlich.

Anmerkung

Eine ausführlichere und aktuellere Liste bekommen Sie mit dem Befehl `"aptitude search 'libpam-|libnss-'"`. Die Abkürzung NSS kann auch "Network Security Service" bedeuten, was etwas anderes ist als "Name Service Switch".

Anmerkung

PAM ist der grundlegendste Weg, um Umgebungsvariablen für jegliche Programme mit einem systemweiten Standardwert vorzubelegen.

Paket	Popcon	Größe	Beschreibung
libpam-modules	V:807, I:999	1032	Pluggable Authentication Modules (grundlegende Dienste)
libpam-ldap	I:12	249	Pluggable Authentication Module, das LDAP-Schnittstellen erlaubt
libpam-cracklib	I:16	115	Pluggable Authentication Module, um Unterstützung für cracklib zu aktivieren
libpam-systemd	V:474, I:853	573	Pluggable Authentication Module zur Registrierung von Nutzersitzungen für logind
libpam-doc	I:1	1044	Pluggable Authentication Modules (Dokumentation in html- und Textform)
libc6	V:935, I:999	12771	GNU-C-Bibliothek: Programmbibliothek, die auch einen "Name Service Switch"-Dienst bereitstellt
glibc-doc	I:11	3161	GNU-C-Bibliothek: Handbuchseiten
glibc-doc-reference	I:5	12740	GNU-C-Bibliothek: Referenzhandbuch im info-, pdf- und html-Format (nicht-frei)
libnss-mdns	I:526	150	NSS-Modul für Multicast-DNS-Namensauflösung
libnss-ldap	I:11	265	NSS-Modul, um LDAP als Namensdienst zu verwenden
libnss-ldapd	I:14	153	NSS-Modul, um LDAP als Namensdienst zu verwenden (neue Abspaltung von libnss-ldap)

Tabelle 4.5: Liste von PAM- und NSS-Systemen

Unter [systemd](#) ist das Paket [libpam-systemd](#) installiert, um die Login-Informationen der Benutzer zu verwalten; dazu werden die Nutzersitzungen in der [systemd](#)-Kontrollgruppen-Hierarchie für [logind](#) registriert. Details finden Sie in [systemd-logind\(8\)](#), [logind.conf\(5\)](#) und [pam_systemd\(8\)](#).

4.5.1 Konfigurationsdateien, auf die PAM und NSS zugreifen

Hier einige erwähnenswerte Konfigurationsdateien, die von PAM und NSS genutzt werden:

Einschränkungen bei der Passwortauswahl sind über die PAM-Module [pam_unix\(8\)](#) und [pam_cracklib\(8\)](#) implementiert. Diese können über deren Argumente konfiguriert werden.

Tipp

PAM-Module verwenden den Anhang ".so" für ihre Dateinamen.

4.5.2 Modernes zentralisiertes Systemmanagement

Ein [Lightweight Directory Access Protocol \(LDAP\)](#)-Server erlaubt modernes zentralisiertes Systemmanagement und somit die Administrierung vieler Unix-ähnlicher und nicht-Unix-Systeme über das Netzwerk. Die quelloffene Implementation des Lightweight Directory Access Protocol ist [OpenLDAP](#).

Der LDAP-Server stellt die Konteninformationen auf Debian-Systemen durch die Nutzung von PAM und NSS über die [libpam-ldap](#)- und [libnss-ldap](#)-Pakete bereit. Verschiedene Aktionen sind nötig, um dies zu aktivieren. (Ich verwende dieses Setup nicht, daher stammen diese Informationen komplett aus zweiter Hand. Bitte beachten Sie das beim Lesen dieses Abschnitts.)

- Richten Sie einen zentralisierten LDAP-Server ein, indem Sie ein Programm wie den eigenständigen LDAP-Daemon [slapd\(8\)](#) starten.
 - Ändern Sie die PAM-Konfigurationsdateien im Verzeichnis `/etc/pam.d/`, um `"pam_ldap.so"` statt dem Standard `"pam_unix.so"` zu verwenden.
 - Debian nutzt `/etc/pam_ldap.conf` als Konfigurationsdatei für [libpam-ldap](#) und `/etc/pam_ldap.secret` als Datei zur Speicherung des root-Passworts.
-

Konfigurationsdatei	Funktion
/etc/pam.d/<programmname>	Einrichtung der PAM-Konfiguration für das Programm <programmname>; lesen Sie dazu pam(7) und pam.d(5)
/etc/nsswitch.conf	Einrichtung der NSS-Konfiguration mit Einträgen für die jeweiligen Dienste; lesen Sie dazu nsswitch.conf(5)
/etc/nologin	Einschränkung der Benutzeranmeldung über das pam_nologin(8)-Modul
/etc/securetty	Einschränkung des tty für den root-Zugriff durch das pam_securetty(8)-Modul
/etc/security/access.conf	Zugriffsbeschränkungen setzen über das pam_access(8)-Modul
/etc/security/group.conf	Einschränkungen für Gruppen setzen über das pam_group(8)-Modul
/etc/security/pam_env.conf	Umgebungsvariablen setzen über das pam_env(8)-Modul
/etc/environment	Zusätzliche Umgebungsvariablen setzen über das pam_env(8)-Modul mit dem Argument "readenv=1"
/etc/default/locale	Setzen der Locale (Gebietsschema) über das pam_env(8)-Modul mit dem Argument "readenv=1 envfile=/etc/default/locale" (Debian)
/etc/security/limits.conf	Beschränkungen für Ressourcen (ulimit, core, ...) setzen über das pam_limits(8)-Modul
/etc/security/time.conf	Zeitbeschränkungen setzen über das pam_time(8)-Modul
/etc/systemd/logind.conf	systemd Login-Manager-Konfiguration setzen (siehe logind.conf(5) and systemd-logind.service(8))

Tabelle 4.6: Liste von Konfigurationsdateien, auf die PAM und NSS zugreifen

- Ändern Sie die NSS-Konfiguration in der Datei `/etc/nsswitch.conf`, um `"ldap"` statt dem Standard (`"compat"` oder `"file"`) zu verwenden.
 - Debian nutzt `/etc/libnss-ldap.conf` als Konfigurationsdatei für `libnss-ldap`.
- Richten Sie es so ein, dass `libpam-ldap` [SSL- \(oder TLS-\)](#)Verbindungen verwendet, um die Passwortsicherheit zu gewährleisten.
- Richten Sie es so ein, dass `libnss-ldap` [SSL- \(oder TLS-\)](#)Verbindungen verwendet, um die Unversehrtheit der Daten sicherzustellen (auch wenn dies zusätzlichen LDAP-Netzwerk-Overhead verursacht).
- Sie sollten `nscd(8)` lokal ausführen, um alle Antworten auf LDAP-Anfragen zwischenspeichern; dies reduziert den LDAP-Netzwerkverkehr.

Lesen Sie die Dokumentation in `pam_ldap.conf(5)` und `/usr/share/doc/libpam-doc/html/` aus dem `libpam-doc`-Paket sowie `info libc 'Name Service Switch'`, bereitgestellt durch das Paket `glibc-doc`.

Ähnlich zu diesem Verfahren können Sie auch alternative zentralisierte Systeme mit anderen Methoden einrichten:

- Verflechtung von Benutzer- und Gruppeninformationen mit Windows-Systemen
 - Zugriff auf [Windows Domain](#)-Dienste mittels der Pakete `winbind`- und `libpam_winbind`;
 - Lesen Sie dazu `winbindd(8)` und [Integrating MS Windows Networks with Samba](#).
- Verflechtung von Benutzer- und Gruppeninformationen mit Unix-ähnlichen Systemen
 - Zugriff auf [NIS \(ursprünglich YP genannt\)](#) oder [NIS+](#) mittels dem `nis`-Paket;
 - Lesen Sie dazu das [Linux NIS\(YP\)/NIS/NIS+ HOWTO](#).

4.5.3 "Warum unterstützt GNU su nicht die wheel-Gruppe"

Dies ist eine bekannte Redewendung am Ende der alten `info su`-Seite von Richard M. Stallman. Aber keine Sorge: der aktuelle `su`-Befehl in Debian nutzt PAM, so dass man die Erlaubnis zur Nutzung von `su` auf die `root`-Gruppe beschränken kann, indem die Zeile mit `"pam_wheel.so"` in `/etc/pam.d/su` aktiviert wird.

4.5.4 Schärfere Passwortregeln

Die Installation des `libpam-cracklib`-Pakets ermöglicht Ihnen, schärfere Regeln für Passwörter vorzuschreiben, indem Sie zum Beispiel aktive Zeilen wie die folgenden in `/etc/pam.d/common-password` eintragen:

Für Squeeze:

```
password required pam_cracklib.so retry=3 minlen=9 difok=3
password [success=1 default=ignore] pam_unix.so use_authtok nullok md5
password requisite pam_deny.so
password required pam_permit.so
```

4.6 Andere Möglichkeiten zur Zugriffskontrolle

Anmerkung

Abschnitt 9.3.15 enthält Informationen, wie Sie die Kernel-Funktion [Secure attention key \(SAK\)](#) einschränken.

4.6.1 sudo

Das Programm `sudo(8)` wurde entwickelt, um einem Systemadministrator die Möglichkeit zu geben, Benutzern eingeschränkte root-Privilegien zu gewähren sowie um die Aktivitäten rund um das root-Konto zu protokollieren. `sudo` benötigt nur das Passwort eines normalen Benutzers. Installieren Sie das `sudo`-Paket und aktivieren Sie es, indem Sie passende Optionen in `/etc/sudoers` setzen. Konfigurationsbeispiele finden Sie unter `/usr/share/doc/sudo/examples/sudoers` und in Abschnitt 1.1.12.

Die Art, wie ich `sudo` auf meinem Einzelbenutzersystem verwende (lesen Sie Abschnitt 1.1.12), soll mich selbst vor meiner eigenen Dummheit schützen. Ich persönlich denke, dass die Verwendung von `sudo` eine bessere Alternative zur dauerhaften Nutzung des root-Kontos ist. Der folgende Befehl ändert zum Beispiel den Eigentümer von `<irgendeine_datei>` in `<mein_name>`:

```
$ sudo chown <mein_name> <irgendeine_datei>
```

Wenn Sie das root-Passwort kennen (was bei Benutzern, die sich ihr Debian-System selbst installieren, immer der Fall ist), können Sie natürlich jeden Befehl von jeglichem Benutzerkonto aus ausführen, indem Sie `su -c` verwenden.

4.6.2 PolicyKit

[PolicyKit](#) ist eine Komponente des Betriebssystems zur Kontrolle von systemweiten Privilegien auf Unix-ähnlichen Systemen.

Neuere GUI-Anwendungen sind nicht dafür entwickelt, als privilegierte Prozesse zu laufen. Sie kommunizieren mit privilegierten Prozessen über PolicyKit, um administrative Operationen durchführen zu können.

Auf Debian-Systemen beschränkt PolicyKit solche Operationen auf Benutzerkonten, die der `sudo`-Gruppe angehören.

Lesen Sie dazu `polkit(8)`.

4.6.3 SELinux

[Security-Enhanced Linux \(SELinux\)](#) ist ein Rahmenwerk, das die Privilegien gegenüber dem gewöhnlichen Unix-ähnlichen Sicherheitsmodell (welches auf [Mandatory access control \(MAC\)](#)-Regeln beruht) verschärft. Unter bestimmten Bedingungen kann dabei die Leistungsfähigkeit des root-Kontos eingeschränkt sein.

4.6.4 Den Zugriff auf einige Server-Dienste einschränken

Aus Gründen der Systemsicherheit ist es eine gute Idee, so viele Server-Programme wie möglich zu deaktivieren. Dies ist besonders für Netzwerk-Server kritisch. Ungenutzte Server, die entweder direkt als [Daemon](#) oder über den [Super-Server](#) aktiviert sind, müssen als Sicherheitsrisiko angesehen werden.

Viele Programme wie `sshd(8)` verwenden PAM-basierte Zugriffskontrollen. Es gibt viele Möglichkeiten, um den Zugriff auf Server-Dienste einzuschränken:

- Konfigurationsdateien: `"/etc/default/<programmname>";`
- Dienste-Unit-Konfiguration für den [Daemon](#);
- [PAM \(Pluggable Authentication Modules\)](#);
- `"/etc/inetd.conf"` für den [Super-Server](#);
- `"/etc/hosts.deny"` und `"/etc/hosts.allow"` für [TCP-Wrapper](#) (`tcpd(8)`);
- `"/etc/rpc.conf"` für [Sun RPC](#);
- `"/etc/at.allow"` und `"/etc/at.deny"` für `atd(8)`;
- `"/etc/cron.allow"` und `"/etc/cron.deny"` für `crontab(1)`;
- [Netzwerk-Firewall](#) aus der [netfilter](#)-Infrastruktur.

Lesen Sie dazu Abschnitt [3.2.6](#), Abschnitt [4.5.1](#) und Abschnitt [5.10](#).

Tipp

[Sun-RPC](#)-Dienste müssen für [NFS](#) und andere RPC-basierte Programme aktiviert sein.

Tipp

Falls Sie Probleme mit dem Zugriff von außen auf ein aktuelles Debian-System haben, kommentieren Sie blockierende Konfigurationselemente wie `"ALL: PARANOID"` in `"/etc/hosts.deny"` aus, falls solche existieren. (Aber sie müssen vorsichtig sein bezüglich der Sicherheitsrisiken, die durch solche Aktionen entstehen.)

4.7 Sicherheit der Authentifizierung

Anmerkung

Die Informationen hier sind für Ihre Sicherheitsbedürfnisse **möglicherweise nicht passend**, aber sie sollten für den Anfang **eine gute Basis** sein.

4.7.1 Sicheres Passwort im Internet

Viele beliebte Transport-Layer-Dienste kommunizieren Nachrichten inklusive der Passwort-Authentifizierung im Klartext. Es ist eine sehr schlechte Idee, Passwörter im Klartext über das wilde Internet zu übertragen, wo es abgehört werden kann. Sie können diese Dienste über ["Transport Layer Security"](#) (TLS) oder seinen Vorgänger ["Secure Sockets Layer"](#) (SSL) betreiben, um die vollständige Kommunikation inklusive des Passworts über die Verschlüsselung zu schützen.

Die Verschlüsselung kostet CPU-Zeit. Als CPU-freundliche Alternative können Sie die Kommunikation im Klartext lassen und nur das Passwort verschlüsseln; verwenden Sie dazu ein sicheres Authentifizierungsprotokoll wie ["Authenticated Post Office Protocol"](#) (APOP) für POP oder ["Challenge-Response Authentication Mechanism MD5"](#) (CRAM-MD5) für SMTP und IMAP. (Um E-Mail-Nachrichten über das Internet von Ihrem Mail-Client-Programm zum Mail-Server zu übertragen, ist es derzeit verbreitet, den neuen Message-Submission-Port 587 zu nutzen statt dem traditionellen SMTP-Port 25, um zu vermeiden, dass Port 25 vom Netzwerk-Provider blockiert wird, während Sie sich über CRAM-MD5 authentifizieren.)

Name des unsicheren Dienstes	Port	Name des sicheren Dienstes	Port
www (http)	80	https	443
smtp (Mail)	25	ssmtp (smtps)	465
ftp-data	20	ftps-data	989
ftp	21	ftps	990
telnet	23	telnets	992
imap2	143	imaps	993
pop3	110	pop3s	995
ldap	389	ldaps	636

Tabelle 4.7: Liste von unsicheren und sicheren Diensten und Ports

4.7.2 Secure Shell (sichere Shell)

Das [Secure-Shell-\(SSH-\)](#)Programm bietet sichere verschlüsselte Kommunikation zwischen zwei nicht vertrauenswürdigen Rechnern über ein unsicheres Netzwerk mittels sicherer Authentifizierung. Es besteht aus dem [OpenSSH-Client](#) (ssh(1)) und dem [OpenSSH-Daemon](#) (sshd(8)). SSH kann genutzt werden, um mittels der Port-Forwarding-Funktionalität (Port-Weiterleitung) eine unsichere Protokoll-Kommunikation wie POP oder X gesichert durch das Internet zu tunneln.

Der Client versucht, sich selbst über eine Host-basierte Authentifizierung gegenüber dem Server zu identifizieren; dazu können verschiedene Verfahren angewandt werden: Public Key Authentication (über einen öffentlichen Schlüssel), Challenge-Response Authentication (es wird eine Aufgabe gestellt, für die die andere Seite die Lösung liefern muss) oder Passwort-Authentifizierung. Die Nutzung der Public Key Authentication ermöglicht eine Anmeldung aus der Ferne ohne Passwort. Lesen Sie dazu Abschnitt 6.9.

4.7.3 Zusätzliche Sicherheitsmaßnahmen für das Internet

Sogar wenn Sie sichere Dienste wie [Secure Shell \(SSH\)](#) und [Point-to-point Tunneling Protocol \(PPTP\)](#) verwenden, bestehen trotzdem noch Chancen für die Einbrecher mittels Brute-Force-Attacks zum Erraten von Passwörtern usw. über das Internet. Die Nutzung von Firewall-Richtlinien (mehr dazu in Abschnitt 5.10) zusammen mit den folgenden Sicherheitswerkzeugen kann die Situation weiter verbessern.

Paket	Popcon	Größe	Beschreibung
knockd	V:0, I:3	102	kleiner Port-Knock-Daemon (horcht auf spezielle Anklopf-Sequenzen für Ports) knockd(1) und -Client konck(1)
fail2ban	V:112, I:123	2092	IP-Adressen sperren, die vielfache Authentifizierungsfehler verursachen
libpam-shield	V:0, I:0	115	Ferne Angreifer aussperren, die versuchen, Passwörter zu erraten

Tabelle 4.8: Liste von Werkzeugen, die zusätzliche Sicherheitsmaßnahmen ermöglichen

4.7.4 Sichern des root-Passworts

Um zu verhindern, dass Leute auf Ihre Maschine mit root-Privilegien zugreifen, müssen Sie folgende Aktionen durchführen:

- physikalischen Zugriff auf die Festplatte unterbinden;
- das BIOS abriegeln und verhindern, dass von Wechseldatenträgern gebootet wird;
- ein Passwort für interaktive Sitzungen von GRUB vergeben;
- das Editieren des GRUB-Menüs verhindern.

Mit physikalischem Zugriff auf die Festplatte ist das Zurücksetzen des root-Passworts relativ leicht; dies erfordert folgende Schritte:

1. Bauen Sie die Festplatte in einen PC ein, der von CD gebootet werden kann.
2. Booten Sie das System mit einer Rettungs-CD (Debian Boot-Disk, Knoppix-CD, GRUB-CD, ...).
3. Binden Sie die root-Partition mit Lese-/Schreibberechtigung ein.
4. Editieren Sie `/etc/passwd` auf der root-Partition und verändern Sie den zweiten Eintrag für das `root`-Konto, so dass dieser leer ist.

Wenn Sie beim Booten Schreibzugriff auf den GRUB-Menüeintrag für `grub - rescue - pc` haben (lesen Sie dazu Abschnitt [3.1.2](#)), ist es mit folgenden Schritten sogar noch einfacher:

1. Booten Sie das System, nachdem Sie die Kernel-Parameter etwa wie folgt geändert haben: `root=/dev/hda6 rw init=/bin/sh`.
2. Editieren Sie `/etc/passwd` und verändern Sie den zweiten Eintrag für das `root`-Konto, so dass dieser leer ist.
3. Starten Sie das System neu.

Die root-Shell des Systems ist jetzt ohne Passwort zugänglich.

Anmerkung

Sobald jemand Zugriff auf die root-Shell hat, kann er auf alles auf dem System zugreifen und jegliche Passwörter auf dem System zurücksetzen. Mehr noch, er könnte die Passwörter für alle Benutzerkonten kompromittieren, indem Brute-Force-Werkzeuge zum Knacken von Passwörtern wie `john` oder `crack` eingesetzt werden (Näheres in Abschnitt [9.4.11](#)). Diese geknackten Passwörter könnten dann dazu verwendet werden, um andere Systeme zu kompromittieren.

Die einzige vernünftige Software-Lösung, um all diese Bedenken auszuräumen, ist die Verwendung einer Software-verschlüsselten root-Partition (oder `/etc`-Partition) mittels [dm-crypt](#) und `initramfs` (lesen Sie dazu Abschnitt [9.8](#)). Sie benötigen dann allerdings immer ein Passwort, um das System zu booten.

Kapitel 5

Netzwerkconfiguration

Tipp

Bezüglich einer grundsätzlichen Anleitung zum Thema Netzwerk unter GNU/Linux lesen Sie den [Linux Network Administrators Guide](#).

Tipp

Bezüglich einer aktuellen Anleitung für Debian zum Thema Netzwerk lesen Sie [Debian Administratorhandbuch — Konfigurieren des Netzwerks](#).

**Warnung**

Statt des traditionellen Namensschemas für die Schnittstellen ("eth0", "eth1", "wlan0", ...) verwendet das neue [systemd](#) "verlässlich vorhersagbare Namen (Predictable Network Interface Names)" wie "enp0s25".

**Warnung**

Dieses Kapitel ist überholt, da es auf dem in 2013 veröffentlichten Debian 7.0 (Wheezy) basiert.

Tipp

Obwohl dieses Dokument immer noch das alte ifconfig(8) mit IPv4 für die Netzwerkconfigurations-Beispiele verwendet, wechselte Debian mit der Wheezy-Veröffentlichung zu ip(8) mit IPv4 + IPv6. Änderungsvorschläge, um dieses Dokument auf aktuellen Stand zu bringen, sind gerne gesehen.

Tipp

Unter [systemd](#) kann [networkd](#) für die Netzwerkverwaltung genutzt werden; lesen Sie dazu [systemd-networkd\(8\)](#).

5.1 Die elementare Netzwerkinfrastruktur

Lassen Sie uns einen Blick auf die elementare Netzwerkinfrastruktur eines modernen Debian-Systems werfen:

Pakete	Popcon	Größe	Art	Beschreibung
ifupdown	V:587, I:991	217	config::ifupdown	standardisiertes Werkzeug zum Aktivieren und Deaktivieren des Netzwerks (Debian-spezifisch)
ifplugd	V:3, I:18	217	„	kabelgebundenes Netzwerk automatisch verwalten
ifupdown-extra	V:0, I:1	106	„	Netzwerk-Testskript zur Aufwertung des „ifupdown“-Pakets
ifmetric	V:0, I:1	37	„	Routenmetrik für eine Netzwerkschnittstelle setzen
guessnet	V:0, I:0	422	„	Mapping-Skript zur Aufwertung des „ifupdown“-Pakets mittels der „/etc/network/interfaces“-Datei
ifscheme	V:0, I:0	59	„	Mapping-Skript zur Aufwertung des „ifupdown“-Pakets
network-manager	V:358, I:440	14957	config::NM	NetworkManager (Daemon): das Netzwerk automatisch verwalten
network-manager-gnome	V:132, I:372	5540	„	NetworkManager (GNOME-Frontend)
wicd	I:24	36(*)	config::wicd	Manager für kabelgebundenes und kabelloses Netzwerk (Metapaket)
wicd-cli	V:0, I:1	60(*)	„	Manager für kabelgebundenes und kabelloses Netzwerk (Befehlszeilen-Programm)
wicd-curses	V:0, I:3	176(*)	„	Manager für kabelgebundenes und kabelloses Netzwerk (Curses-basiertes Programm)
wicd-daemon	V:19, I:26	992(*)	„	Manager für kabelgebundenes und kabelloses Netzwerk (Daemon)
wicd-gtk	V:15, I:25	576(*)	„	Manager für kabelgebundenes und kabelloses Netzwerk (GTK+-Programm)
iptables	V:300, I:993	2520	config::Netfilter	Administrationswerkzeuge für Paketfilterung und NAT (Netfilter)
iproute2	V:672, I:926	2867	config::iproute2	iproute2 , IPv6 und andere erweiterte Netzwerkkonfiguration: ip(8), tc(8) usw.
ifrename	V:0, I:3	125	„	Netzwerkschnittstellen basierend auf verschiedenen statischen Kriterien umbenennen: ifrename(8)
ethtool	V:102, I:261	597	„	Eigenschaften von Ethernet-Geräten anzeigen oder ändern
iputils-ping	V:234, I:997	113	test::iproute2	Erreichbarkeit eines fernen Rechners über das Netzwerk testen, entweder mittels Rechnername oder IP-Adresse (iproute2)
iputils-arping	V:8, I:127	55	„	Erreichbarkeit eines fernen Rechners über das Netzwerk mittels seiner ARP -Adresse testen
iputils-tracepath	V:4, I:60	72	„	Netzwerkpfad zu einem fernen Rechner verfolgen
net-tools	V:234, I:634	991	config::net-tools	NET-3 Netzwerkprogramm-Sammlung (net-tools , IPv4-Netzwerkkonfiguration): ifconfig(8) usw.
inetutils-ping	V:0, I:1	359	test::net-tools	Erreichbarkeit eines fernen Rechners über das Netzwerk testen, entweder mittels Rechnername oder IP-Adresse (altes System, GNU)
arping	V:2, I:29	77	„	Erreichbarkeit eines fernen Rechners über das Netzwerk mittels seiner ARP -Adresse testen (altes System)
traceroute	V:63, I:936	159	„	Netzwerkpfad zu einem fernen Rechner verfolgen (altes System, Konsolenprogramm)
isc-dhcp-client	V:231, I:979	686	config::low-level	DHCP-Client
wpasupplicant	V:332, I:507	3436	„	clientseitige Unterstützung für WPA und WPA2 (IEEE 802.11i)
wpaui	V:0, I:2	781	„	Qt-GUI-Programm für wpa_supplicant
wireless-tools	V:188, I:254	297	„	Werkzeuge zum Bearbeiten der Linux Wireless Extensions
ppp	V:206, I:474	1054	„	PPP-/PPPoE-Verbindung mit chat
pppoeconf	V:0, I:8	192	config::helper	Konfigurations-Hilfswerkzeug für PPPoE-Verbindungen
pppconfig	V:1, I:2	801	„	Konfigurations-Hilfswerkzeug für PPP-Verbindungen mit chat
pppconf	V:0, I:5	540	„	Konfigurations-Hilfswerkzeug für PPP-Verbindungen

5.1.1 Die Auflösung des Rechnernamens

Die Auflösung des Rechnernamens (hostname) wird derzeit auch durch den [NSS-\(Name-Service-Switch-\)](#)Mechanismus unterstützt. Die Auflösung läuft wie folgt ab:

1. Die `"/etc/nsswitch.conf"`-Datei mit Einträgen wie `"hosts: files dns"` bestimmt die Reihenfolge der Rechnernamensauflösung. (Dies ersetzt die alte Funktionalität der `"order"`-Einträge in `"/etc/host.conf"`.)
2. Als erstes wird in diesem Beispiel die `files`-Methode aufgerufen. Wenn der Rechnername in der `"/etc/hosts"`-Datei gefunden wird, werden alle gültigen Adressen für den Rechner ausgegeben und die Abfrage wird beendet. (Die `"/etc/host.conf"`-Datei enthält `"multi on"`.)
3. Dann wird die `dns`-Methode aufgerufen. Wenn der Rechnername über das [Internet Domain Name System \(DNS\)](#) (definiert über die Datei `"/etc/resolv.conf"`) gefunden wird, werden alle dafür gültigen Adressen ausgegeben und die Abfrage wird beendet.

Die `"/etc/hosts"`-Datei sieht zum Beispiel so aus:

```
127.0.0.1 localhost
127.0.1.1 <rechnername>

# The following lines are desirable for IPv6 capable hosts
# (die folgenden Zeilen sind für IPv6-fähige Rechner wünschenswert).
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
ff02::3  ip6-allhosts
```

Jede Zeile beginnt mit einer [IP-Adresse](#) und dahinter steht jeweils der zugeordnete [Rechnername](#).

Die IP-Adresse `127.0.1.1` in der zweiten Zeile dieses Beispiels ist auf einigen anderen Unix-ähnlichen Systemen möglicherweise nicht vorhanden. Der [Debian Installer](#) erstellt diesen Eintrag für Systeme ohne feste IP-Adresse als provisorische Lösung für einige Software-Produkte (z.B. GNOME), wie in [Fehler #719621](#) dokumentiert.

Der Eintrag `<rechnername>` entspricht dem in `"/etc/hostname"` festgelegten Rechnernamen.

Auf Systemen mit einer festen IP-Adresse sollte allerdings diese feste IP-Adresse statt der `127.0.1.1` verwendet werden.

Bei Systemen mit einer festen IP-Adresse und einem [voll qualifizierten Domain-Namen \(FQDN\)](#), bereitgestellt durch das [Domain Name System \(DNS\)](#), sollte `<rechnername>.<domain-name>` verwendet werden statt nur `<rechnername>`.

Die Datei `"/etc/resolv.conf"` ist eine statische Datei, falls das Paket `resolvconf` nicht installiert ist. Falls das Paket installiert ist, ist dies ein symbolischer Link. In beiden Fällen enthält es Informationen zur Initialisierung der Namensauflösungs-Routinen. Wenn das DNS zum Beispiel über die IP `"192.168.11.1"` erreichbar ist, enthält sie Folgendes:

```
nameserver 192.168.11.1
```

Das `resolvconf`-Paket macht `"/etc/resolv.conf"` zu einem symbolischen Link und verwaltet ihren Inhalt automatisch über die Hook-Skripte.

Bei PC-Arbeitsplatz-Rechnern in einer typischen LAN-Umgebung kann der Rechnername zusätzlich zu den grundlegenden `files`- und `dns`-Methoden auch über Multicast DNS (mDNS, [Zeroconf](#)) aufgelöst werden:

- [Avahi](#) stellt ein Rahmenwerk für Multicast-DNS-Diensteabfragen auf Debian-Systemen bereit.
- Es ist ein Äquivalent zu [Apple Bonjour / Apple Rendezvous](#).
- Das `libnss-mdns`-Plugin-Paket bietet Rechnernamensauflösung via mDNS für die GNU Name-Service-Switch-(NSS-)Funktionalität der GNU C-Bibliothek (glibc).

- Die Datei `/etc/nsswitch.conf` sollte Einträge wie `hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4` enthalten.
- Rechnernamen, die mit der [Pseudo-Top-Level Domain \(TLD\) ".local"](#) enden, werden aufgelöst.
- Die mDNS IPv4 link-lokale Multicast-Adresse `"224.0.0.251"` oder ihr IPv6-Äquivalent `"FF02::FB"` wird verwendet, um DNS-Abfragen für einen auf `".local"` endenden Namen durchzuführen.

Rechnernamensauflösung über das veraltete [NETBios over TCP/IP](#), das von älteren Windows-Systemen verwendet wurde, kann über die Installation des Pakets `winbind` realisiert werden. Die `/etc/nsswitch.conf`-Datei sollte Einträge wie `hosts: files mdns4_minimal [NOTFOUND=return] dns mdns4 wins` enthalten, um diese Funktionalität zu aktivieren. (Moderne Windows-Systeme verwenden normalerweise die `dns`-Methode zur Rechnernamensauflösung.)

Anmerkung

Die [Ausweitung generischer Top-Level-Domains \(gTLD\)](#) im [Domain-Name-System](#) ist in Arbeit. Achten Sie bei Auswahl von Domain-Namen, die nur im lokalen Netzwerk verwendet werden sollen, auf [Namenskollisionen](#).

5.1.2 Der Netzwerkschnittstellename

Der Netzwerkschnittstellename, z.B. `eth0`, wird jeglicher Hardware im Linux-Kernel bei der Erkennung durch den User-Space-Konfigurationsmechanismus `udev` zugewiesen (lesen Sie dazu [Abschnitt 3.3](#)). Der Netzwerkschnittstellename wird in `ifup(8)` und `interfaces(5)` als **physical interface** (physikalische Schnittstelle) bezeichnet.

Um über [MAC-Adresse](#) (und anderes) sicherzustellen, dass jede Netzwerkschnittstelle bei jedem Neustart dauerhaft die gleiche Bezeichnung hat, gibt es eine Regeldatei `/etc/udev/rules.d/70-persistent-net.rules`. Diese Datei wird automatisch durch das `/lib/udev/write_net_rules`-Programm generiert, unter Umständen ausgeführt von der `"persistent-net"` Regeldatei. Sie können sie editieren, um die Namensregeln zu ändern.



Achtung

Wenn Sie die `/etc/udev/rules.d/70-persistent-net.rules`-Regeldatei verändern, müssen Sie für jede Regel eine eigene Zeile verwenden und die Buchstaben in der [MAC-Adresse](#) müssen Kleinbuchstaben sein. Wenn Sie zum Beispiel Einträge für "FireWire device" und "PCI device" in dieser Datei finden, möchten Sie vielleicht das Gerät unter "PCI device" mit `eth0` benennen und es als primäre Netzwerkschnittstelle konfigurieren.

5.1.3 Der Netzwerkbereich für das LAN

Wir wollen uns an die IPv4 32-Bit-Adressbereiche erinnern, die durch die [rfc1918](#) für jede Klasse zur Verwendung in [Local Area Networks \(LANs\)](#) reserviert sind. Diese Adressen werden bestimmt nicht mit irgendwelchen Adressen im Internet kollidieren.

Klasse	Netzwerkadressen	Netzmaske	Netzmaske /Bits	Anzahl der Unter-netzwerke
A	10.x.x.x	255.0.0.0	/8	1
B	172.16.x.x — 172.31.x.x	255.255.0.0	/16	16
C	192.168.0.x — 192.168.255.x	255.255.255.0	/24	256

Tabelle 5.2: Liste der Netzwerkbereiche

Anmerkung

Wenn eine dieser Adressen einem Rechner zugewiesen ist, kann dieser Rechner das Internet nicht direkt erreichen, sondern muss ein Gateway verwenden, der als Proxy für verschiedene Dienste dient, oder er nutzt [Network Address Translation \(NAT\)](#). Ein Breitband-Router nutzt üblicherweise NAT für das Anwender-Netzwerk.

5.1.4 Unterstützung für Netzwerkgeräte

Der größte Teil verfügbarer Netzwerk-Hardware wird durch das Debian-System unterstützt; es gibt einige Geräte, die laut [DFSG](#) nicht-freie Firmware für den Betrieb erfordern. Lesen Sie dazu Abschnitt [9.9.6](#).

5.2 Moderne Netzwerkkonfiguration für Arbeitsplatzsysteme

Auf modernen Debian-Desktop-Systemen mit `systemd` erfolgt die Initialisierung von Netzwerkschnittstellen für die Loopback-Schnittstelle `lo` typischerweise durch `networking.service` und für andere Schnittstellen durch `NetworkManager.service`.

Debian-Systeme der Version 6.0 (Squeeze) und später können Netzwerkverbindungen über Software-Daemons wie [NetworkManager \(NM\)](#) (`network-manager` und zugehörige Pakete) oder [Wicd](#) (`wicd` und zugehörige Pakete) verwalten.

- Sie haben ihre eigenen grafischen [GUI](#)- und Befehlszeilen-Programme als Bedienoberfläche.
- Sie haben ihre eigenen [Daemons](#) als Unterbau.
- Sie erlauben eine einfache Verbindung Ihres Systems mit dem Internet.
- Sie ermöglichen eine problemlose Verwaltung von kabelgebundenen und kabellosen Netzwerkkonfigurationen.
- Sie erlauben uns, das Netzwerk unabhängig vom althergebrachten `ifupdown` zu konfigurieren.

Anmerkung

Verwenden Sie diese automatischen Netzwerkkonfigurations-Werkzeuge nicht für Server. Sie sind primär für die Nutzung auf Arbeitsplatzrechnern oder Laptops gedacht.

Diese modernen Werkzeuge müssen korrekt konfiguriert werden, um Konflikte mit dem `ifupdown`-Paket und seiner Konfigurationsdatei `/etc/network/interfaces` zu vermeiden.

Anmerkung

Einige Funktionalitäten dieser automatischen Netzwerkkonfigurations-Werkzeuge könnten unter Rückentwicklungen leiden. Sie sind nicht so robust wie das altbewährte `ifupdown`. Überprüfen Sie die Fehlerberichte zu [network-manager](#) und [wicd](#) bezüglich aktueller Probleme oder Einschränkungen.

5.2.1 Grafische Netzwerkkonfigurations-Werkzeuge

Offizielle Dokumentation für NM und Wicd unter Debian ist in `/usr/share/doc/network-manager/README.Debian` respektive `/usr/share/doc/wicd/README.Debian` verfügbar.

Grundsätzlich läuft die Netzwerkkonfiguration für Arbeitsplatzsysteme wie folgt ab:

1. Fügen Sie den Benutzer, der sich am Arbeitsplatz anmeldet, z.B. `foo`, mit folgendem Befehl zur Gruppe `netdev` hinzu (alternativ kann dies in modernen Arbeitsplatz-Umgebungen wie GNOME oder KDE auch automatisch über [D-bus](#) erledigt werden):
-

```
$ sudo adduser foo netdev
```

2. Halten Sie die Konfiguration in `"/etc/network/interfaces"` so einfach wie hier:

```
auto lo
iface lo inet loopback
```

3. Starten Sie NM bzw. Wicd mit einem der folgenden Befehle neu:

```
$ sudo /etc/init.d/network-manager restart
```

```
$ sudo /etc/init.d/wicd restart
```

4. Konfigurieren Sie Ihr Netzwerk über die grafische GUI-Oberfläche.

Anmerkung

Um Konflikte mit `ifupdown` zu vermeiden, werden nur Schnittstellen, die **nicht** in `"/etc/network/interfaces"` aufgelistet sind, von NM oder Wicd verwaltet.

Tipp

Wenn Sie die Fähigkeiten von NM erweitern möchten, suchen Sie nach entsprechenden Plugin-Modulen und zusätzlichen Paketen wie `network-manager-openconnect`, `network-manager-openvpn-gnome`, `network-manager-pptp-gnome`, `mobile-broadband-provider-info`, `gnome-bluetooth` usw. Das gleiche gilt für Wicd.



Achtung

Diese automatischen Netzwerkkonfigurations-Werkzeuge sind unter Umständen nicht kompatibel mit esoterischen Konfigurationen des althergebrachten `ifupdown` in `"/etc/network/interfaces"`, wie denen in Abschnitt 5.6 und Abschnitt 5.7. Überprüfen Sie die Fehlerberichte zu [network-manager](#) und [wicd](#) bezüglich aktueller Probleme oder Einschränkungen.

5.3 Moderne Netzwerkkonfiguration ohne grafische Oberfläche

Unter [systemd](#) kann das Netzwerk stattdessen in `/etc/systemd/network/` konfiguriert werden. Lesen Sie dazu `systemd-resolved(8)`, `resolved.conf(5)` und `systemd-networkd(8)`.

Dies ermöglicht eine moderne Netzwerkkonfiguration auch ohne grafische Oberfläche.

Eine DHCP-Client-Konfiguration kann durch Erzeugen von `"/etc/systemd/network/dhcp.network"` eingerichtet werden, z.B. mit:

```
[Match]
Name=en*
```

```
[Network]
DHCP=yes
```

Eine statische Netzwerkkonfiguration richten Sie über `"/etc/systemd/network/static.network"` ein, wie hier:

```
[Match]
Name=en*
```

```
[Network]
Address=192.168.0.15/24
Gateway=192.168.0.1
```

5.4 Die Netzwerkverbindung und -konfiguration nach altbewährtem System

Wenn die Methode aus Abschnitt 5.2 Ihren Ansprüchen nicht genügt, sollten Sie die altbewährte Methode für Netzwerkverbindung und -konfiguration verwenden, die erheblich einfachere Werkzeuge nutzt.

Die altbewährte Netzwerkverbindung ist für jede einzelne Methode unterschiedlich (Näheres in Abschnitt 5.5).

Es gibt unter Linux zwei Arten von Programmen für die Arbeiten im Untergrund der Netzwerkkonfiguration (lesen Sie dazu Abschnitt 5.8.1):

- Die alten [net-tools](#)-Programme (`ifconfig(8)`, ...) stammen vom Linux NET-3 Netzwerk-System. Die meisten davon sind mittlerweile überholt.
- Neue [Linux iproute2](#)-Programme (`ip(8)`, ...) bilden das aktuelle Linux-Netzwerk-System.

Obwohl diese Basis-Netzwerk-Programme sehr leistungsfähig sind, ist Ihre Anwendung etwas umständlich. Daher wurden übergeordnete Konfigurationssysteme entwickelt.

Das `ifupdown`-Paket ist der De-Facto-Standard für diese übergeordneten Netzwerkkonfigurationssysteme unter Debian. Es ermöglicht Ihnen die Aktivierung des Netzwerks z.B. über ein einfaches `ifup eth0`. Seine Konfigurationsdatei ist `/etc/network/` und deren typischen Inhalte sind Folgende:

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp
```

Das `resolvconf`-Paket wurde als Ergänzung zum `ifupdown`-System erstellt, um eine problemlose Neukonfiguration der Netzwerkadressauflösung durch automatisches Neuerstellen der Konfigurationsdatei `/etc/resolv.conf` zu ermöglichen. Die meisten Netzwerkkonfigurationspakete in Debian wurden entsprechend angepasst, so dass Sie jetzt das `resolvconf`-Paket verwenden (lesen Sie auch `/usr/share/doc/resolvconf/README.Debian`).

Hilfsskripte zum `ifupdown`-Paket wie `ifplugd`, `guessnet`, `ifscheme` usw. wurden erstellt, um die dynamische Konfiguration von Netzwerkumgebungen (wie z.B. die eines mobilen PC an kabelgebundenen LAN) zu automatisieren. Sie sind zwar relativ schwierig zu benutzen, funktionieren aber gut mit dem vorhandenen `ifupdown`-System.

Diese Hilfsskripte werden im Detail und mit Beispielen in Abschnitt 5.6 und Abschnitt 5.7) beschrieben.

5.5 Die Netzwerkverbindungs-Methoden (nach altbewährtem System)



Achtung

Die Methoden zum Testen der Netzwerkverbindung in diesem Abschnitt sind für Testzwecke gedacht. Sie sollten sie nicht direkt für die täglichen Netzwerkverbindungen nutzen. Es wird empfohlen, stattdessen `NM`, `Wicd` oder das `ifupdown`-Paket zu verwenden (Näheres in Abschnitt 5.2 und Abschnitt 5.6).

Typische Netzwerkverbindungsmethoden und Verbindungspfade für einen PC können wie folgt zusammengefasst werden:

Hier eine Zusammenfassung der Konfigurationsskripte für die verschiedenen Verbindungsmethoden:

Die Abkürzungen bei Netzwerkverbindungen haben folgende Bedeutungen:

Anmerkung

WAN-Verbindungsdienste über TV-Kabel werden grundsätzlich mit DHCP oder PPPoE ausgeliefert, diejenigen über ADSL und FTTP mit PPPoE. Sie müssen Ihren ISP kontaktieren, um die exakten Konfigurationsanforderungen der WAN-Verbindung zu erfragen.

PC	Verbindungsmethoden	Verbindungspfad
Serieller Port (ppp0)	PPP	⇔ Modem ⇔ Telefonverbindung ⇔ Einwahl-Zugangspunkt ⇔ Internet-Anbieter
Ethernet-Port (eth0)	PPPoE/DHCP/Statisch	⇔ Breitband-Modem ⇔ Breitband-Dienst ⇔ Breitband-Zugangspunkt ⇔ Internet-Anbieter
Ethernet-Port (eth0)	DHCP/Statisch	⇔ LAN ⇔ Breitband-Router mit Network Address Translation (NAT) (⇔ Breitband-Modem ...)

Tabelle 5.3: Liste von Netzwerkverbindungsmethoden und Verbindungspfaden

Verbindungsmethoden	Konfiguration	Backend-(Unterbau-)Paket
PPP	pppconfig zur Erzeugung von deterministischer Chat-Verbindung	pppconfig, ppp
PPP (alternativ)	wvdialconf zur Erzeugung von heuristischer Chat-Verbindung	ppp, wvdial
PPPoE	pppoeconf zur Erzeugung von deterministischer Chat-Verbindung	pppoeconf, ppp
DHCP	definiert in "/etc/dhcp/dhclient.conf"	isc-dhcp-client
statische IP (IPv4)	definiert in "/etc/network/interfaces"	iproute2 oder net-tools (veraltet)
statische IP (IPv6)	definiert in "/etc/network/interfaces"	iproute2

Tabelle 5.4: Liste der Netzwerkverbindungskonfigurationen

Abkürzung	Bedeutung
POTS	plain old telephone service (alter reiner Telefondienst)
BB	Breitband
BB-service	z.B. Digital Subscriber Line (DSL), TV-Kabel oder Fiber To The Premises (FTTP, Glasfaser)
BB-modem	z.B. DSL-Modem , Kabel-Modem oder Optical Network Terminal (ONT) für Glasfaser
LAN	Local Area Network (lokales Netzwerk)
WAN	Wide Area Network (großräumiges Netzwerk, z.B. das Internet)
DHCP	Dynamic Host Configuration Protocol
PPP	Point-to-Point Protocol
PPPoE	Point-to-Point Protocol over Ethernet
ISP	Internet Service Provider (Internet-Diensteanbieter)

Tabelle 5.5: Liste der Abkürzungen bei Netzwerkverbindungen

Anmerkung

Wenn ein Breitband-Router verwendet wird, um eine Heim-Netzwerkumgebung aufzubauen, werden PCs im LAN mittels [Network Address Translation \(NAT\)](#) durch den Breitband-Router mit dem WAN (Internet) verbunden. In solchen Fällen haben die Netzwerkschnittstellen der PCs im LAN entweder eine statische IP-Adresse oder werden via DHCP vom Breitband-Router bedient. Die Internet-Verbindung des Breitband-Routers muss entsprechend den Anweisungen des ISP konfiguriert werden.

5.5.1 Die DHCP-Verbindung über Ethernet

Das typische moderne Netzwerk für Heim und kleine Unternehmen (LAN) wird über einen Breitband-Router mit dem Internet (WAN) verbunden. Das LAN hinter dem Router wird üblicherweise über einen [Dynamic Host Configuration Protocol \(DHCP\)](#)-Server, der auf dem Router läuft, verwaltet.

Installieren Sie einfach das Paket `isc-dhcp-client`, um das durch [DHCP](#) verwaltete Ethernet zu nutzen.

Lesen Sie dazu auch `dhclient.conf(5)`.

5.5.2 Statische IP-Verbindung über Ethernet

Es gibt keine speziellen Anforderungen, um das Ethernet mit statischer IP zu nutzen.

5.5.3 Die PPP-Verbindung mit pppconfig

Das Konfigurationsskript `pppconfig` konfiguriert die [PPP](#)-Verbindung interaktiv, indem einfach die folgenden Daten abgefragt werden:

- Telefonnummer;
- Name des ISP;
- Passwort vom ISP;
- Anschlußgeschwindigkeit;
- Modemanschluß für die Kommunikation;
- Authentifizierungsmethode.

Datei	Funktion
<code>/etc/ppp/peers/<isp-name></code>	Durch <code>pppconfig</code> erzeugte Konfigurationsdatei für <code>pppd</code> , spezifisch für den ISP <code><isp-name></code>
<code>/etc/chatscripts/<isp-name></code>	Durch <code>pppconfig</code> erzeugte Konfigurationsdatei für <code>chat</code> , spezifisch für den ISP <code><isp-name></code>
<code>/etc/ppp/options</code>	Grundlegende Ausführungsparameter für <code>pppd</code>
<code>/etc/ppp/pap-secret</code>	Authentifizierungsdaten für PAP (unsicher)
<code>/etc/ppp/chap-secret</code>	Authentifizierungsdaten für CHAP (höhere Sicherheit)

Tabelle 5.6: Liste der Konfigurationsdateien für eine [PPP](#)-Verbindung mit `pppconfig`

**Achtung**

Werden die Befehle `pon` und `poff` ohne weitere Argumente aufgerufen, wird automatisch "provider" als "`<isp-name>`" genutzt.

Sie können die Konfiguration testen, indem Sie die zugrundeliegenden Konfigurationswerkzeuge wie folgt aufrufen:

```
$ sudo pon <isp-name>
...
$ sudo poff <isp-name>
```

Lesen Sie dazu auch `"/usr/share/doc/ppp/README.Debian.gz"`.

5.5.4 Die alternative PPP-Verbindung mit wvdialconf

Ein anderer Ansatz, `pppd(8)` zu verwenden, ist der Aufruf über `wvdial(1)` aus dem gleichnamigen Paket. Statt dass `pppd` den `chat(8)`-Prozess zwecks Einwahl und Aushandlung der Verbindung aufruft, übernimmt `wvdial` die Einwahl und den anfänglichen Verbindungsaufbau selbst und startet dann `pppd` für den Rest.

Das Konfigurationsskript `wvdialconf` konfiguriert die PPP-Verbindung interaktiv, indem folgende Daten abgefragt werden:

- Telefonnummer;
- Name des ISP;
- Passwort vom ISP;

`wvdial` stellt in den meisten Fällen erfolgreich die Verbindung her und verwaltet automatisch eine Liste mit den Authentifizierungsdaten.

Datei	Funktion
/etc/ppp/peers/wvdial	Durch <code>wvdialconf</code> erzeugte Konfigurationsdatei für <code>pppd</code> , spezifisch für <code>wvdial</code>
/etc/wvdial.conf	Durch <code>wvdialconf</code> erzeugte Konfigurationsdatei
/etc/ppp/options	Grundlegende Ausführungsparameter für <code>pppd</code>
/etc/ppp/pap-secret	Authentifizierungsdaten für PAP (unsicher)
/etc/ppp/chap-secret	Authentifizierungsdaten für CHAP (höhere Sicherheit)

Tabelle 5.7: Liste der Konfigurationsdateien für eine PPP-Verbindung mit `wvdialconf`

Sie können die Konfiguration testen, indem Sie die zugrundeliegenden Konfigurationswerkzeuge wie folgt aufrufen:

```
$ sudo wvdial
...
$ sudo killall wvdial
```

Lesen Sie dazu auch `wvdial(1)` und `wvdial.conf(5)`.

5.5.5 Die PPPoE-Verbindung mit pppoeconf

Wenn Ihr ISP Ihnen eine PPPoE-Verbindung anbietet und Sie sich entscheiden, den PC direkt mit dem WAN (Internet) zu verbinden, muss das Netzwerk Ihres PCs mit PPPoE (PPP over Ethernet) konfiguriert werden. Das Konfigurationsskript `pppoeconf` konfiguriert die PPPoE-Verbindung durch interaktive Abfragen.

Die Konfigurationsdateien sind folgende:

Sie können die Konfiguration testen, indem Sie die zugrundeliegenden Konfigurationswerkzeuge wie folgt aufrufen:

```
$ sudo /sbin/ifconfig eth0 up
$ sudo pon dsl-provider
...
$ sudo poff dsl-provider
$ sudo /sbin/ifconfig eth0 down
```

Lesen Sie dazu auch `"/usr/share/doc/pppoeconf/README.Debian"`.

Datei	Funktion
/etc/ppp/peers/dsl-provider	Durch <code>pppoeconf</code> erzeugte Konfigurationsdatei für <code>pppd</code> , spezifisch für <code>pppoe</code>
/etc/ppp/options	Grundlegende Ausführungsparameter für <code>pppd</code>
/etc/ppp/pap-secret	Authentifizierungsdaten für PAP (unsicher)
/etc/ppp/chap-secret	Authentifizierungsdaten für CHAP (höhere Sicherheit)

Tabelle 5.8: Liste der Konfigurationsdateien für eine PPPoE-Verbindung mit `pppoeconf`

5.6 Die grundlegende Netzwerkkonfiguration mit `ifupdown` (nach altbewährtem System)

Zur Einrichtung eines traditionellen **TCP/IP-Netzwerks** wird auf einem Debian-System das `ifupdown`-Paket als hochentwickeltes Werkzeug verwendet. Es gibt zwei typische Fälle:

- Auf Systemen mit einer **dynamischen IP-Adresse** wie mobilen PCs sollten Sie das TCP/IP-Netzwerk **mit** dem `resolvconf`-Paket einrichten; dies ermöglicht Ihnen, Ihre Netzwerkkonfiguration einfach auf eine andere Situation umzuschalten (Näheres in Abschnitt 5.6.4).
- Auf Systemen mit einer **statischen IP-Adresse** wie z.B. auf Servern sollten Sie das TCP/IP-Netzwerk **ohne** das `resolvconf`-Paket einrichten und so das System einfach halten (Weiteres finden Sie in Abschnitt 5.6.5).

Diese traditionellen Methoden der Einrichtung sind sehr nützlich, wenn Sie erweiterte Konfigurationen benötigen; hier folgen weitere Details.

Das `ifupdown`-Paket stellt ein standardisiertes Rahmenwerk für hochentwickelte Netzwerkkonfiguration im Debian-System bereit. In diesem Abschnitt lernen wir die grundlegende Netzwerkkonfiguration mittels `ifupdown` mit vereinfachten Erklärungen und vielen typischen Beispielen kennen.

5.6.1 Die Befehlssyntax in vereinfachter Form

Das `ifupdown`-Paket enthält zwei Befehle: `ifup(8)` und `ifdown(8)`. Sie bieten hochentwickelte Netzwerkkonfiguration, gesteuert durch die Konfigurationsdatei `"/etc/network/interfaces"`.

Befehl	Aktion
<code>ifup eth0</code>	die Netzwerkschnittstelle <code>eth0</code> mit der Konfiguration für <code>eth0</code> aktivieren, falls ein Eintrag für <code>"iface eth0"</code> existiert.
<code>ifdown eth0</code>	die Netzwerkschnittstelle <code>eth0</code> mit der Konfiguration für <code>eth0</code> deaktivieren, falls ein Eintrag für <code>"iface eth0"</code> existiert.

Tabelle 5.9: Liste der Befehle zur grundlegenden Netzwerkkonfiguration mit `ifupdown`

Warnung

Verwenden Sie keine der im Untergrund arbeitenden Werkzeuge wie `ifconfig(8)` und `ip(8)`, um Schnittstellen zu konfigurieren, die sich im Status **up** (aktiv) befinden.

Anmerkung

Es gibt keinen Befehl `ifupdown`.

5.6.2 Der grundlegende Aufbau von `"/etc/network/interfaces"`

Der Aufbau von `"/etc/network/interfaces"` (auch in `interfaces(5)` beschrieben) kann wie folgt zusammengefasst werden:

Eintrag	Bedeutung
<code>"auto <name>"</code>	die Schnittstelle <name> während des Systemstarts aktivieren
<code>"allow-auto <name>"</code>	"
<code>"allow-hotplug <name>"</code>	die Schnittstelle <name> aktivieren, wenn der Kernel ein Hotplug-Ereignis von der Schnittstelle empfängt
Zeilen beginnend mit <code>"iface <konfig-name> ..."</code>	Festlegung der Netzwerkkonfiguration <konfig-name>
Zeilen beginnend mit <code>"mapping <muster-für-name>"</code>	Festlegung des Wertes zum Mapping von <konfig_name> für eine auf <muster-für-name> passende Schnittstelle
eine mit <code>"#"</code> beginnende Zeile	wird als Kommentar ignoriert (Kommentare am Ende einer Zeile werden nicht unterstützt)
eine mit <code>"\"</code> endende Zeile	Konfiguration in der nächsten Zeile fortsetzen

Tabelle 5.10: Liste der Einträge in `"/etc/network/interfaces"`

Einträge beginnend mit **iface** haben folgenden Aufbau:

```
iface <konfig-name> <addressfamilie> <methodenname>
<option1> <wert1>
<option2> <wert2>
...
```

Bei Basiskonfigurationen werden **mapping**-Einträge nicht verwendet; es wird stattdessen der Name der Schnittstelle als Konfigurationsname genutzt (lesen Sie dazu Abschnitt [5.7.5](#)).



Warnung

Geben Sie keine doppelten "iface"-Einträge für eine Netzwerkschnittstelle in `"/etc/network/interfaces"` an.

5.6.3 Die loopback-Netzwerkschnittstelle

Folgender Eintrag in der Datei `"/etc/network/interfaces"` aktiviert die loopback-Netzwerkschnittstelle `lo` während des Systemstarts (über den **auto**-Eintrag).

```
auto lo
iface lo inet loopback
```

Dieser Eintrag existiert immer in `"/etc/network/interfaces"`.

5.6.4 Die über DHCP bediente Netzwerkschnittstelle

Nachdem das System gemäß Abschnitt [5.5.1](#) vorbereitet wurde, wird eine über DHCP verwaltete Netzwerkschnittstelle mit einem Eintrag wie dem folgenden in `"/etc/network/interfaces"` konfiguriert:

```
allow-hotplug eth0
iface eth0 inet dhcp
```

Wenn der Linux-Kernel die physikalische Schnittstelle `eth0` detektiert, führt die Zeile **allow-hotplug** dazu, dass `ifup` automatisch die Schnittstelle aktiviert, und aufgrund des **iface**-Eintrags wird DHCP zur Konfiguration verwendet.

5.6.5 Die Netzwerkschnittstelle mit statischer IP-Adresse

Eine Netzwerkschnittstelle mit statischer (fester) IP-Adresse wird über einen Eintrag wie dem folgenden in `/etc/network/interfaces` konfiguriert:

```
allow-hotplug eth0
iface eth0 inet static
    address 192.168.11.100
    netmask 255.255.255.0
    gateway 192.168.11.1
    dns-domain example.com
    dns-nameservers 192.168.11.1
```

Detektiert der Linux-Kernel die physikalische Schnittstelle `eth0`, aktiviert `ifup` aufgrund der **allow-hotplug**-Zeile automatisch die Schnittstelle und nutzt gemäß dem **iface**-Eintrag die statische IP-Adresse für die Konfiguration.

Dabei gehe ich von folgenden Annahmen aus:

- IP-Adressbereich für das LAN-Netzwerk: `192.168.11.0 - 192.168.11.255`
- IP-Adresse des Gateways: `192.168.11.1`
- IP-Adresse des PCs: `192.168.11.100`
- Das `resolvconf`-Paket: installiert
- Domain-Name: `"example.com"`
- IP-Adresse des DNS-Servers: `192.168.11.1`

Falls das `resolvconf`-Paket nicht installiert ist, muss die Konfiguration des DNS manuell durchgeführt werden, indem `/etc/resolv.conf` wie folgt editiert wird:

```
nameserver 192.168.11.1
domain example.com
```



Achtung

Die in diesem Beispiel verwendeten IP-Adressen können nicht zwingend direkt kopiert werden. Sie müssen Adressen verwenden, die zu Ihrer jeweiligen Netzwerksituation passen.

5.6.6 Die Grundlagen von kabellosen LAN-Schnittstellen

Das [Wireless LAN \(kurz WLAN\)](#) bietet schnelle kabellose Verbindungen über eine Breitspektrum-Kommunikation in unlicenzierten Funkfrequenzbändern, basierend auf den [IEEE 802.11](#)-Standards.

Die WLAN-Schnittstellen sind nahezu normale Ethernet-Schnittstellen, erfordern aber bei der Initialisierung einige Daten für Netzwerk-ID und Verschlüsselung. Die entsprechenden Netzwerkkonfigurations-Werkzeuge sind identisch mit denen für Ethernet-Schnittstellen, nur dass sich die Namen der Schnittstellen ein bisschen unterscheiden, sie heißen z.B. `eth1`, `wlan0`, `ath0`, `wifi0` ..., je nachdem, welche Kernel-Treiber zum Einsatz kommen.

Tipp

`wmaster0` ist das Master-Gerät, das als internes Gerät nur von [SoftMAC](#) mit der neuen [mac80211 API-of-Linux](#)-Implementierung genutzt wird.

Hier einige für WLAN relevante Schlüsselbegriffe:

Die Entscheidung, welches Protokoll letztlich verwendet wird, wird für gewöhnlich eingeschränkt durch die Fähigkeiten des verwendeten Wireless-Routers.

Abkürzung	vollständiger Begriff	Bedeutung
NWID	Netzwerk-ID	16-Bit-Netzwerk-ID, die von alten (pre-802.11 / Vor-802.11-Standard-Implementierung) WaveLAN-Netzwerken genutzt wird (stark veraltet)
(E)SSID	(Extended) Service Set Identifier	Netzwerkname der Wireless Access Points (AP / drahtloser Zugangspunkt) zum Aufbau eines integrierten 802.11 Wireless-LAN, Domain-ID
WEP, (WEP2)	Wired Equivalent Privacy	64-Bit- (128-Bit-) Wireless-Verschlüsselungsstandard der ersten Generation mit 40-Bit-Schlüssel (veraltet)
WPA	Wi-Fi Protected Access	Wireless-Verschlüsselungsstandard der zweiten Generation (überwiegend 802.11i), kompatibel mit WEP
WPA2	Wi-Fi Protected Access 2	Wireless-Verschlüsselungsstandard der dritten Generation (vollständig 802.11i), nicht kompatibel mit WEP

Tabelle 5.11: Liste der Abkürzungen für WLAN

5.6.7 Die Wireless-LAN-Schnittstelle mit WPA/WPA2

Zur Unterstützung des neuen WPA/WPA2 müssen Sie das `wpa_supplicant`-Paket installieren.

Falls die IP-Adressen der WLAN-Verbindung per [DHCP](#) verwaltet werden, muss der Eintrag in der `"/etc/network/interfaces"`-Datei wie folgt aussehen:

```
allow-hotplug ath0
iface ath0 inet dhcp
wpa-ssid homezone
# der hexadezimale PSK ist mit einer Klartext-Passphrase verschlüsselt
wpa-psk 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f
```

Lesen Sie dazu auch `"/usr/share/doc/wpa_supplicant/README.modes.gz"`.

5.6.8 Die Wireless-LAN-Schnittstelle mit WEP

Zur Unterstützung des alten WEP müssen Sie das `wireless-tools`-Paket installieren. (Unter Umständen unterstützt Ihr Router nur diese unsichere Verschlüsselungsmethode, aber das ist immer noch besser als gar keine Verschlüsselung zu verwenden.)



Achtung

Bitte beachten Sie, dass der Netzwerkverkehr in einem WLAN mit WEP durch andere mitgeschrieben werden könnte.

Falls die IP-Adressen der WLAN-Verbindung per [DHCP](#) verwaltet werden, muss der Eintrag in der `"/etc/network/interfaces"`-Datei wie folgt aussehen:

```
allow-hotplug eth0
iface eth0 inet dhcp
wireless-ssid Home
wireless-key1 0123-4567-89ab-cdef
wireless-key2 12345678
```

```
wireless-key3 s:password
wireless-defaultkey 2
wireless-keymode open
```

Lesen Sie dazu `"/usr/share/doc/wireless-tools/README.Debian"`.

5.6.9 Die PPP-Verbindung

Sie müssen die PPP-Verbindung zunächst wie oben in Abschnitt 5.5.3 beschrieben einrichten. Fügen Sie dann folgenden Eintrag für das primäre PPP-Gerät `ppp0` in die `"/etc/network/interfaces"`-Datei ein:

```
iface ppp0 inet ppp
provider <isp-name>
```

5.6.10 Die alternative PPP-Verbindung

Sie müssen die alternative PPP-Verbindung mit `wvdial` zunächst wie in Abschnitt 5.5.4) einrichten. Fügen Sie dann folgenden Eintrag für das primäre PPP-Gerät `ppp0` in die `"/etc/network/interfaces"`-Datei ein:

```
iface ppp0 inet wvdial
```

5.6.11 Die PPPoE-Verbindung

PCs, die über PPPoE direkt mit dem WAN (Internet) verbunden sind, müssen mittels PPPoE-Verbindung (wie in Abschnitt 5.5.5 beschrieben) eingerichtet werden. Fügen Sie dann folgenden Eintrag für das primäre PPPoE-Gerät `eth0` in die `"/etc/network/interfaces"`-Datei ein:

```
allow-hotplug eth0
iface eth0 inet manual
pre-up /sbin/ifconfig eth0 up
up ifup ppp0=dsl
down ifdown ppp0=dsl
post-down /sbin/ifconfig eth0 down
# Folgendes wird nur intern verwendet:
iface dsl inet ppp
provider dsl-provider
```

5.6.12 Der Netzwerkkonfigurations-Status von ifupdown

Die Datei `"/etc/network/run/ifstate"` speichert die **vorgesehenen** Netzwerkkonfigurations-Status für alle derzeit aktiven, durch das `ifupdown`-Paket verwalteten Netzwerkschnittstellen. Unglücklicherweise listet die `"/etc/network/run/ifstate"`-Datei eine Schnittstelle selbst dann als aktiv auf, wenn diese von `ifupdown` nicht aktiviert werden konnte.

Solange die Ausgabe des Befehls `ifconfig(8)` für eine Schnittstelle keine Zeile wie in folgenden Beispiel enthält, kann die Schnittstelle nicht als Teil eines [IPV4-Netzwerks](#) genutzt werden:

```
inet addr:192.168.11.2 Bcast:192.168.11.255 Mask:255.255.255.0
```

Anmerkung

Für das Ethernet-Gerät, das mittels PPPoE verbunden ist, enthält die `ifconfig(8)`-Ausgabe keine solche Zeile wie in obigem Beispiel.

5.6.13 Die grundlegende Netzwerk-Neukonfiguration

Wenn Sie versuchen, eine Schnittstelle wie z.B. `eth0` neu zu konfigurieren, müssen Sie sie zunächst mit dem Befehl **"sudo ifdown eth0"** deaktivieren. So wird der Eintrag für `eth0` aus der `"/etc/network/run/ifstate"`-Datei entfernt. (Dies könnte zu Fehlermeldungen führen, falls `eth0` nicht aktiviert ist oder vorher unpassend konfiguriert war. Allerdings ist es offensichtlich unproblematisch, dies auf einfachen Arbeitsplatzsystemen für Einzelbenutzer so durchzuführen.)

Sie können jetzt die Inhalte von `"/etc/network/interfaces"` wie für die Neukonfiguration der Schnittstelle `eth0` erforderlich anpassen.

Aktivieren Sie dann wieder die `eth0`-Schnittstelle mit dem Befehl **"sudo ifup eth0"**.

Tipp

Sie können eine einfache Neuinitialisierung der Netzwerkschnittstelle mittels **"sudo ifdown eth0;sudo ifup eth0"** durchführen.

5.6.14 Das ifupdown-extra-Paket

Das Paket `ifupdown-extra` bietet einfache Netzwerk-Verbindungstests zur Verwendung mit dem `ifupdown`-Paket:

- Der Befehl `network-test(1)` kann von der Shell aus genutzt werden.
- Die automatischen Skripte werden bei jeder `ifup`-Befehlsausführung gestartet.

Der `network-test`-Befehl befreit Sie von der Verwendung umständlicher Basisbefehle zur Analyse von Netzwerkproblemen. Die automatischen Skripte werden in `"/etc/network/*/"` installiert und bieten Folgendes:

- prüfen der Verbindung des Netzkabels;
- Überprüfung auf mehrfache Verwendung einer IP-Adresse;
- Einrichtung der statischen Routen des Systems basierend auf den Definitionen in `"/etc/network/routes"`;
- Prüfung, ob der Netzwerk-Gateway erreichbar ist;
- Aufzeichnung der Ergebnisse in der Datei `"/var/log/syslog"`.

Diese syslog-Aufzeichnungen sind für die Administration von Netzwerkproblemen auf fernen Systemen sehr nützlich.

Tipp

Das automatische Verhalten des `ifupdown-extra`-Pakets ist über `"/etc/default/network-test"` konfigurierbar. Einige dieser automatischen Tests können den Systemstart ein wenig verlangsamen, da die Antworten auf [ARP](#)-Anfragen eine gewisse Zeit benötigen.

5.7 Die erweiterte Netzwerkkonfiguration mit ifupdown (nach altbewährtem System)

Die Funktionalität des `ifupdown`-Pakets kann mit entsprechendem fortgeschrittenen Wissen über das oben in Abschnitt [5.6](#) beschriebene Maß hinaus erweitert werden.

Die hier beschriebenen Funktionalitäten sind absolut optional. Ich für meinen Teil, der ich faul und ein Minimalist bin, habe mich nur selten darum bemüht, sie zu nutzen.

**Achtung**

Falls Sie mit obigen Informationen aus Abschnitt [5.6](#) keine Netzwerkverbindung einrichten konnten, werden Sie mit den hier folgenden Informationen die Situation nur noch schlechter machen.

5.7.1 Das ifplugd-Paket

Das `ifplugd`-Paket ist ein älteres Werkzeug für automatische Netzwerkkonfiguration, das nur Ethernet-Verbindungen verwalten kann. Es behebt Probleme mit abgezogenen/neu eingesteckten Ethernet-Kabeln bei mobilen PCs usw. Falls Sie [NetworkManager](#) oder [Wicd](#) (lesen Sie dazu Abschnitt 5.2) installiert haben, benötigen Sie dieses Paket nicht.

Dieses Paket läßt einen [Daemon](#) laufen und ersetzt **auto**- oder **allow-hotplug**-Funktionalitäten (Näheres in Tabelle 5.10) und aktiviert entsprechende Schnittstellen bei deren Anschluß an das Netzwerk.

Hier eine Anleitung, wie Sie das `ifplugd`-Paket für die interne Ethernet-Schnittstelle, z.B. `eth0`, verwenden:

1. Entfernen Sie folgende Einträge in `/etc/network/interfaces`: `auto eth0` oder `allow-hotplug eth0`.
2. Folgende Einträge müssen in `/etc/network/interfaces` erhalten bleiben: `iface eth0 inet ...` und `mapping ...`.
3. Installieren Sie das Paket `ifplugd`
4. Führen Sie `sudo dpkg-reconfigure ifplugd` aus.
5. Wählen Sie `eth0` bei der Abfrage "Zu überwachende Schnittstellen, die immer (statisch) vorhanden sind".

Jetzt funktioniert die Netzwerk-Neukonfiguration wie gewünscht.

- Nach dem Einschalten oder bei Erkennung neuer Hardware wird die Schnittstelle nicht selbstständig aktiviert.
 - Schneller Boot-Prozess ohne das lange Warten auf die DHCP-Zeitüberschreitung.
 - Keine seltsame Aktivierung von Schnittstellen ohne korrekte IPv4-Adresse (lesen Sie auch Abschnitt 5.6.12).
- Nach Erkennung des eingesteckten Ethernet-Kabels wird die Schnittstelle aktiviert.
- Einige Zeit nach dem Abziehen des Ethernet-Kabels wird die Schnittstelle automatisch deaktiviert.
- Nach dem Einstecken eines anderen Ethernet-Kabels wird die Schnittstelle (für die neue Netzwerkumgebung passend) aktiviert.

Tipp

Über die Argumente für den `ifplugd(8)`-Befehl kann dessen Verhalten, wie z.B. die Verzögerung bis zur Neukonfiguration der Schnittstelle, angepasst werden.

5.7.2 Das ifmetric-Paket

Das `ifmetric`-Paket ermöglicht es uns, die Metrik von Routen sogar für DHCP zu manipulieren.

Mit folgendem Vorgehen setzen Sie die `eth0`-Schnittstelle als bevorzugt gegenüber der `wlan0`-Schnittstelle:

1. Installieren Sie das Paket `ifmetric`.
2. Fügen Sie eine Zeile mit `metric 0` direkt unterhalb von `iface eth0 inet dhcp` in `/etc/network/interfaces` ein.
3. Fügen Sie eine Zeile mit `metric 1` direkt unterhalb von `iface wlan0 inet dhcp` in `/etc/network/interfaces` ein.

Das `"metric 0"` steht für die Route mit der höchsten Priorität und ist der Standardwert. Der größere `metric`-Wert steht für Routen mit niedrigeren Prioritäten. Die IP-Adresse der aktivierten Schnittstelle mit dem niedrigsten `metric`-Wert wird die bevorzugte Adresse. Details finden Sie unter `ifmetric(8)`.

5.7.3 Die virtuelle Schnittstelle

Mit einer einzigen physikalischen Ethernet-Schnittstelle können mehrere virtuelle Schnittstellen mit verschiedenen IP-Adressen konfiguriert werden. Der übliche Ansatz ist hierbei, eine Schnittstelle mit mehreren IP-Subnetzen zu verbinden. Zum Beispiel ist das IP-Adress-basierte virtuelle Web-Hosting mit einer einzigen Netzwerkschnittstelle eine solche Anwendung.

Gehen wir zum Beispiel einmal von folgendem aus:

- Eine einzelne Ethernet-Schnittstelle in Ihrem Rechner ist mit einem Ethernet-Hub verbunden (nicht mit dem Breitband-Router).
- Der Ethernet-Hub ist sowohl mit dem Internet wie auch mit dem LAN verbunden.
- Das LAN-Netzwerk nutzt das Subnetz 192.168.0.x/24.
- Ihr Rechner nutzt eine per DHCP verwaltete IP-Adresse mit der physikalischen Schnittstelle `eth0` für das Internet.
- Der Rechner nutzt 192.168.0.1 mit der virtuellen Schnittstelle `eth0:0` für das LAN.

Folgende Einträge in `/etc/network/interfaces` konfigurieren Ihr Netzwerk:

```
iface eth0 inet dhcp
metric 0
iface eth0:0 inet static
address 192.168.0.1
netmask 255.255.255.0
network 192.168.0.0
metric 1
```



Achtung

Obwohl dieses Konfigurationsbeispiel mit [Network Address Translation \(NAT\)](#) über [netfilter/iptables](#) (lesen Sie dazu Abschnitt [5.10](#)) günstige Router für das LAN mit nur einer einzigen Schnittstelle ermöglicht, hat solch ein Setup keine wirklichen Firewall-Fähigkeiten. Sie sollten zwei physikalische Schnittstellen mit NAT nutzen, um das lokale Netzwerk gegen das Internet abzusichern.

5.7.4 Die erweiterte Befehlssyntax

Das `ifupdown`-Paket bietet erweiterte Netzwerkkonfigurationen über **Netzwerkkonfigurations**-Namen und **Netzwerkschnittstellen**-Namen. Ich verwende eine Bezeichnungsweise, die sich leicht von der in `ifup(8)` und `interfaces(5)` unterscheidet.

Bezeichnung in der Handbuchseite	Meine Bezeichnung	Beispiele im folgenden Text	Beschreibung
Name der physikalischen Schnittstelle	Name der Netzwerk-schnittstelle	lo, eth0, <schnittstellen-name>	Name, der durch den Linux-Kernel (mittels <code>udev</code> -Mechanismus) vergeben wird
Name der logischen Schnittstelle	Name der Netzwerk-konfiguration	config1, config2, <konfigurations-name>	Namensbezeichnung gemäß iface in <code>/etc/network/interfaces</code>

Tabelle 5.12: Liste verschiedener Bezeichnungsweisen für Netzwerkschnittstellen

Die Befehle zur grundlegenden Netzwerkkonfiguration in Abschnitt [5.6.1](#) erfordern es, dass der **Netzwerkkonfigurations**-Name aus dem **iface**-Eintrag mit dem Name der **Netzwerkschnittstelle** in `/etc/network/interfaces` übereinstimmt.

Die erweiterten Befehle zur Netzwerkkonfiguration erlauben die Trennung von **Netzwerkkonfigurations**-Name und **Netzwerkschnittstellen**-Name in `/etc/network/interfaces` wie folgt:

Befehl	Aktion
<code>ifup eth0=config1</code>	die Netzwerkschnittstelle <code>eth0</code> mit der Konfiguration <code>config1</code> aktivieren
<code>ifdown eth0=config1</code>	die Netzwerkschnittstelle <code>eth0</code> mit der Konfiguration <code>config1</code> deaktivieren
<code>ifup eth0</code>	die Netzwerkschnittstelle <code>eth0</code> mit der über den mapping -Eintrag gewählten Konfiguration aktivieren
<code>ifdown eth0</code>	die Netzwerkschnittstelle <code>eth0</code> mit der über den mapping -Eintrag gewählten Konfiguration deaktivieren

Tabelle 5.13: Liste erweiterter Befehle zur Netzwerkkonfiguration mit `ifupdown`

5.7.5 Der mapping-Eintrag

Um Verwirrungen zu vermeiden, haben wir die Erklärung des **mapping**-Eintrags in der `"/etc/network/interfaces"`-Datei im Abschnitt 5.6.2 übersprungen. Dieser Eintrag hat folgende Syntax:

```
mapping <muster-für-schnittstellename>
script <skript-name>
map <skript-eingabewert1>
map <skript-eingabewert2>
map ...
```

Dieses ermöglicht erweiterte Funktionalitäten für die `"/etc/network/interfaces"`-Datei, indem die Wahl der Konfiguration über das mittels `<skript-name>` festgelegte mapping-Skript automatisiert wird.

Wir wollen einmal dieser Programmausführung folgen:

```
$ sudo ifup eth0
```

Wenn `"<muster-für-schnittstellename>"` auf `"eth0"` zutrifft, wird automatisch die Ausführung des folgenden Befehls ausgelöst, um `eth0` zu konfigurieren:

```
$ sudo ifup eth0=$(echo -e '<skript-eingabewert1> \n <skript-eingabewert2> \n ...' | <skript-name> eth0)
```

Hierbei sind die Zeilen mit `"map"` optional und können wiederholt verwendet werden.

Anmerkung

Das Muster für den **mapping**-Eintrag funktioniert wie die Shell-Dateinamen-Globs (Näheres in Abschnitt 1.5.6).

5.7.6 Die manuell umschaltbare Netzwerkkonfiguration

Hier erklären wir, wie Sie manuell zwischen mehreren Netzwerkkonfigurationen wechseln können, ohne die `"/etc/network/interfaces"`-Datei umzuschreiben (wie es in Abschnitt 5.6.13 beschrieben ist).

Für alle Netzwerkkonfigurationen, auf die Sie zugreifen müssen, erstellen Sie separate Einträge in `"/etc/network/interfaces"` wie folgt:

```
auto lo
iface lo inet loopback

iface config1 inet dhcp

iface config2 inet static
address 192.168.11.100
netmask 255.255.255.0
```

```
gateway 192.168.11.1
dns-domain example.com
dns-nameservers 192.168.11.1

iface pppoe inet manual
pre-up /sbin/ifconfig eth0 up
up ifup ppp0=dsl
down ifdown ppp0=dsl
post-down /sbin/ifconfig eth0 down

# Folgendes wird nur intern verwendet:
iface dsl inet ppp
provider dsl-provider

iface pots inet ppp
provider provider
```

Bitte beachten Sie, dass der **Netzwerkkonfigurations**-Name, also das Kürzel hinter **iface**, nicht identisch ist mit dem **Netzwerkschnittstellen**-Namen. Auch gibt es keine **auto**- oder **allow-hotplug**-Einträge, über die die Netzwerkschnittstelle `eth0` beim Auftreten entsprechender Ereignisse automatisch gestartet würde.

Jetzt sind Sie bereit für den Wechsel zwischen den verschiedenen Konfigurationen.

Verbinden Sie den PC mit einem per DHCP verwalteten LAN-Netzwerk. Sie aktivieren die **Netzwerkschnittstelle** (die physikalische Schnittstelle) `eth0`, indem Sie ihr wie folgt den **Netzwerkkonfigurations**-Namen (den logischen Schnittstellennamen) `config1` zuweisen:

```
$ sudo ifup eth0=config1
Password:
...
```

Die Schnittstelle `eth0` wird aktiviert, per DHCP konfiguriert und mit dem LAN verbunden.

```
$ sudo ifdown eth0=config1
...
```

Die Schnittstelle `eth0` wird deaktiviert und vom LAN getrennt.

Nun verbinden Sie den PC mit einem LAN-Netzwerk mit statischen IP-Adressen. Sie aktivieren die **Netzwerkschnittstelle** `eth0`, indem Sie ihr wie folgt den **Netzwerkkonfigurations**-Namen `config2` zuweisen:

```
$ sudo ifup eth0=config2
...
```

Die Schnittstelle `eth0` wird aktiviert, mit einer statischen IP-Adresse konfiguriert und mit dem LAN verbunden. Über die zusätzlichen `dns-*`-Parameter werden die Inhalte von `/etc/resolv.conf` festgelegt. Die Verwaltung von `/etc/resolv.conf` funktioniert aber besser, wenn das Paket `resolvconf` installiert ist.

```
$ sudo ifdown eth0=config2
...
```

Die Schnittstelle `eth0` wird erneut deaktiviert und vom LAN getrennt.

Nun verbinden Sie den PC mit einem Anschluß des Breitband-Modems, das per PPPoE bedient wird. Sie aktivieren die **Netzwerkschnittstelle** `eth0`, indem Sie ihr wie folgt den **Netzwerkkonfigurations**-Namen `pppoe` zuweisen:

```
$ sudo ifup eth0=pppoe
...
```

Die Schnittstelle `eth0` wird aktiviert und über die direkte PPPoE-Verbindung zum ISP konfiguriert.

```
$ sudo ifdown eth0=pppoe
...
```

Die Schnittstelle `eth0` wird erneut deaktiviert und vom LAN getrennt.

Zu guter Letzt verbinden Sie den PC ohne LAN oder Breitband-Modem via Modem und Telefonanschluß (POTS) mit dem Internet. Sie aktivieren die **Netzwerkschnittstelle** `ppp0`, indem Sie ihr wie folgt den **Netzwerkkonfigurations**-Namen `pots` zuweisen:

```
$ sudo ifup ppp0=pots
...
```

Die Schnittstelle `ppp0` wird aktiviert und via PPP mit dem Internet verbunden.

```
$ sudo ifdown ppp0=pots
...
```

Die Schnittstelle `ppp0` wird deaktiviert und die Verbindung zum Internet getrennt.

Sie sollten die `/etc/network/run/ifstate`-Datei bezüglich des aktuellen `ifupdown`-Netzwerkkonfigurations-Status kontrollieren.



Warnung

Falls Sie mehrere Netzwerkschnittstellen in Ihrem PC haben, müssen Sie unter Umständen die Zahlen von `eth*`, `ppp*` usw. anpassen.

5.7.7 Skriptverarbeitung mit dem `ifupdown`-System

Das `ifupdown`-System führt automatisch Skripte aus, die in `/etc/network/*` installiert sind; dabei werden Umgebungsvariablen an die Skripte exportiert.

Umgebungsvariable	Übergebener Wert
<code>"\$IFACE"</code>	physikalischer Name (Schnittstellename) der zu verarbeitenden Schnittstelle
<code>"\$LOGICAL"</code>	logischer Name (Konfigurationsname) der zu verarbeitenden Schnittstelle
<code>"\$ADDRFAM"</code>	<addressfamilie> der Schnittstelle
<code>"\$METHOD"</code>	<methodenname> der Schnittstelle (z.B. "statisch")
<code>"\$MODE"</code>	"start", wenn von <code>ifup</code> ausgeführt; "stop", wenn von <code>ifdown</code> ausgeführt
<code>"\$PHASE"</code>	wie " <code>\$MODE</code> ", aber mit feinerer Auflösung, unterscheidet die Phasen <code>pre-up</code> , <code>post-up</code> , <code>pre-down</code> und <code>post-down</code>
<code>"\$VERBOSITY"</code>	gibt an, ob " <code>--verbose</code> " genutzt wird (ausführliche Protokollausgabe); Wert = 1 falls genutzt, Wert = 0 falls nicht
<code>"\$PATH"</code>	Befehlssuchpfad: z.B. <code>/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin</code>
<code>"\$IF_<OPTION>"</code>	Wert für die entsprechende Option im iface -Eintrag

Tabelle 5.14: Liste der Umgebungsvariablen, die dem `ifupdown`-System übergeben werden

Hierbei wird die Umgebungsvariable `"$IF_<OPTION>"` aus dem Namen für die zugehörige Option (z.B. <option1> oder <option2>) gebildet; dazu wird `"$IF_"` vorangestellt, alle Buchstaben werden in Großbuchstaben umgewandelt, Bindestriche in Unterstriche geändert und nicht-alphanumerische Zeichen werden entfernt.

Tipp

Lesen Sie Abschnitt [5.6.2](#) bezüglich weiterer Details zu <addressfamilie>, <methodenname>, <option> usw.

Das `ifupdown-extra`-Paket (Näheres in Abschnitt 5.6.14) verwendet diese Umgebungsvariablen, um die Funktionalität des `ifupdown`-Pakets zu erweitern. Das Paket `ifmetric` (lesen Sie dazu Abschnitt 5.7.2) installiert das Skript `/etc/network/if-up`, welches über die Variable `$IF_METRIC` die Metric festlegt. Auch das Paket `guessnet` (weitere Infos dazu in Abschnitt 5.7.8), das ein einfaches und leistungsfähiges Rahmenwerk zur automatischen Auswahl der Netzwerkkonfiguration über den mapping-Mechanismus bietet, nutzt diese Variablen.

Anmerkung

Weiterführende Beispiele für speziell angepasste Netzwerkkonfigurations-Skripte mittels dieser Umgebungsvariablen finden Sie in `/usr/share/doc/ifupdown/examples/*` sowie in den Paketen `ifscheme` und `ifupdown-scripts-zg2`. Diese zusätzlichen Skripte haben einige Überschneidungen mit den Funktionalitäten aus den Paketen `ifupdown-extra` und `guessnet`. Falls Sie diese zusätzlichen Skripte installieren, sollten Sie vorgenannte Skripte anpassen, um wechselseitige Störungen zu vermeiden.

5.7.8 Mapping mit guessnet

Statt der in Abschnitt 5.7.6 beschriebenen manuellen Auswahl der Konfiguration können Sie auch den Mapping-Mechanismus (Näheres in Abschnitt 5.7.5) verwenden, um die Netzwerkkonfiguration mit speziell auf Ihre Situation angepassten Skripten automatisch auszuwählen.

Der Befehl `guessnet-ifupdown(8)` aus dem `guessnet`-Paket wurde als Mapping-Skript entwickelt und bietet ein leistungsfähiges Rahmenwerk zum `ifupdown`-System.

- Sie legen für jeden **iface**-Eintrag Testbedingungen als Wert für die **guessnet**-Option fest.
- Der Mapping-Mechanismus wählt den ersten **iface**-Eintrag als Netzwerkkonfiguration aus, für den die Abfrage kein `ERROR`-Ergebnis (Fehler) zurückgibt.

Diese doppelte Verwendung der `/etc/network/interfaces`-Datei, einmal durch das mapping-Skript (`guessnet - ifupdown`) und dann noch durch die originale Netzwerk-Infrastruktur (`ifupdown`) hat keine negativen Nebenwirkungen, da durch die **guessnet**-Optionen nur zusätzliche Umgebungsvariablen an die vom `ifupdown`-System ausgeführten Skripte exportiert werden. Details dazu finden Sie in `guessnet-ifupdown(8)`.

Anmerkung

Falls mehrere **guessnet**-Optionen in `/etc/network/interfaces` benötigt werden, verwenden Sie Optionen, die mit **guessnet1**, **guessnet2** usw. beginnen, da das `ifupdown`-Paket es nicht erlaubt, die gleiche Zeichenkette am Anfang von Optionszeilen mehrfach zu verwenden.

5.8 Die Netzwerkkonfiguration auf unterster Ebene

5.8.1 iproute2-Befehle

Die `iproute2`-Befehle bieten vollwertige Funktionalität auf der untersten Ebene der Netzwerkkonfiguration. Hier eine Tabelle zur Gegenüberstellung von veralteten `net-tools`-Befehlen und neuen `iproute2`- und anderen Befehlen.

Lesen Sie `ip(8)` und das [IPROUTE2 Utility Suite Howto](#).

5.8.2 Sichere Basis-Netzwerkoperationen

Sie können die folgenden Netzwerkbefehle der untersten Ebene problemlos verwenden, da sie die Netzwerkkonfiguration nicht verändern:

net-tools (veraltet)	iproute2 usw. (neu)	Beeinflussung
ifconfig(8)	ip addr	Protokoll-Adresse (IP oder IPv6) eines Gerätes
route(8)	ip route	Eintrag in der Routing-Tabelle
arp(8)	ip neigh	ARP- oder NDISC-Cache-Eintrag
ipmaddr	ip maddr	Multicast-Adresse
iptunnel	ip tunnel	Tunnel über IP
nameif(8)	ifrename(8)	Netzwerkschnittstellen basierend auf MAC-Adressen benennen
mii-tool(8)	ethtool(8)	Einstellungen von Ethernet-Geräten

Tabelle 5.15: Gegenüberstellung von net-tools- und iproute2-Befehlen

Befehl	Beschreibung
ifconfig	Verbindungs- und Adressstatus von aktiven Schnittstellen anzeigen
ip addr show	Verbindungs- und Adressstatus von aktiven Schnittstellen anzeigen
route -n	Vollständige Routing-Tabelle mit numerischen Adressen anzeigen
ip route show	Vollständige Routing-Tabelle mit numerischen Adressen anzeigen
arp	Aktuellen Inhalt der ARP -Cache-Tabellen anzeigen
ip neigh	Aktuellen Inhalt der ARP -Cache-Tabellen anzeigen
plog	Logdaten des PPP-Daemons anzeigen
ping yahoo.com	Internet-Verbindung zu "yahoo.com" überprüfen
whois yahoo.com	Überprüfen, wer "yahoo.com" in der Domain-Datenbank registriert hat
traceroute yahoo.com	Verbindung zu "yahoo.com" durch das Internet verfolgen
tracpath yahoo.com	Verbindung zu "yahoo.com" durch das Internet verfolgen
mtr yahoo.com	Verbindung zu "yahoo.com" durch das Internet verfolgen (wiederholt)
dig [@dns-server.com] example.com [{a mx any}]	DNS -Einträge von "example.com" laut den Daten von "dns-server.com" auf einen "a"- , "mx"- oder "any"-Eintrag überprüfen
iptables -L -n	Paketfilter überprüfen
netstat -a	Alle offenen Ports finden
netstat -l --inet	Ports finden, die auf eine Verbindung warten
netstat -ln --tcp	TCP-Ports finden, die auf eine Verbindung warten (numerisch)
dlint example.com	DNS-Zonen-Informationen von "example.com" überprüfen

Tabelle 5.16: Liste von Basis-Netzwerkbefehlen

Tipp

Einige dieser Basisbefehle zur Netzwerkkonfiguration sind in `/sbin/` abgelegt. Sie müssen unter Umständen den vollständigen Pfad, wie z.B. `/sbin/ifconfig` angeben oder `/sbin` zur Variable `$PATH` in Ihrer `~/.bashrc`-Datei hinzufügen.

5.9 Netzwerkoptimierung

Die grundsätzliche Netzwerkoptimierung liegt außerhalb des Rahmens dieser Dokumentation. Ich erwähne hier nur Dinge, die für Anwender-typische Verbindungen passend sind.

Pakete	Popcon	Größe	Beschreibung
iftop	V:7, I:115	97	Informationen zur Bandbreitennutzung einer Netzwerkschnittstelle anzeigen
iperf	V:4, I:55	263	Werkzeug zur IP-Bandbreiten-Messung
ifstat	V:0, I:8	60	InterFace STATistics Monitoring (Netzwerkschnittstellen-Statistik/-Überwachung)
bmon	V:1, I:17	146	Portierbarer Bandbreitenmonitor und Geschwindigkeitsrechner
ethstatus	V:0, I:5	40	Skript, das schnell den Durchsatz eines Netzwerkgerätes messen kann
bing	V:0, I:1	80	Empirisch stochastischer Bandbreitentester
bwm-ng	V:2, I:17	90	Kleiner und einfacher konsolenbasierter Bandbreitenmonitor
ethstats	V:0, I:0	23	Konsolenbasierter Ethernet-Statistikmonitor
ipfm	V:0, I:0	78	Bandbreitenanalyse-Werkzeug

Tabelle 5.17: Liste von Werkzeugen zur Netzwerkoptimierung

5.9.1 Die optimale MTU finden

Der Wert für die [Maximum Transmission Unit \(MTU\)](#) kann experimentell mit `ping(8)` mittels der Option `-M do` ermittelt werden; dabei werden ICMP-Pakete mit einer Datengröße ab 1500 (mit einem Offset von 28 Byte für den IP+ICMP-Header) versendet, um herauszufinden, welche maximale Größe ohne IP-Fragmentierung möglich ist.

Probieren Sie zum Beispiel folgendes:

```
$ ping -c 1 -s $((1500-28)) -M do www.debian.org
PING www.debian.org (194.109.137.218) 1472(1500) bytes of data.
From 192.168.11.2 icmp_seq=1 Frag needed and DF set (mtu = 1454)

--- www.debian.org ping statistics ---
0 packets transmitted, 0 received, +1 errors
```

Versuchen Sie dann 1454 statt 1500.

Sie sehen, dass `ping(8)` bei einem Wert von 1454 erfolgreich beendet wird.

Dies ist das [Path MTU \(PMTU\) Discovery](#)-Verfahren ([RFC1191](#)) und der Befehl `tracpath(8)` kann dies automatisieren.

Tipp

Das obige Beispiel mit einem PMTU-Wert von 1454 stammt von meinem früheren FTTP-Provider, der [Asynchronous Transfer Mode \(ATM\)](#) als Backbone-Netzwerk verwendet und seine Kunden per [PPPoE](#) bedient. Der bei Ihnen aktuelle PMTU-Wert hängt von der jeweiligen Situation ab, z.B. habe ich bei meinem neuen FTTP-Provider einen Wert von 1500.

Zusätzlich zu diesen Grundregeln sollten Sie folgendes wissen:

Netzwerkumgebung	MTU	Argumentation
Einwahlverbindung (IP: PPP)	576	Standard
Ethernet-Verbindung (IP: DHCP oder fest)	1500	Standard und vorgegeben
Ethernet-Verbindung (IP: PPPoE)	1492 (=1500-8)	2 Byte für den PPP-Header und 6 Byte für den PPPoE-Header
Ethernet-Verbindung (Backbone des ISP: ATM. IP: DHCP oder fest)	1462 (=48*31-18-8)	Spekulation des Autors: 18 Byte für den Ethernet-Header, 8 Byte für den SAR-Trailer
Ethernet-Verbindung (Backbone des ISP: ATM. IP: PPPoE)	1454 (=48*31-8-18-8)	Unter " Optimal MTU configuration for PPPoE ADSL Connections " finden Sie nähere Infos.

Tabelle 5.18: Wesentliche Grundregeln für den optimalen MTU-Wert

- Jegliche Nutzung von Tunneling-Methoden ([VPN](#) usw.) kann aufgrund des Overheads den optimalen MTU-Wert reduzieren.
- Der MTU-Wert sollte den über die experimentelle Methode ermittelten PMTU-Wert nicht überschreiten.
- Ein größerer MTU-Wert ist grundsätzlich besser, wenn andere Einschränkungen greifen.

5.9.2 Setzen der MTU

Hier einige Beispiele, wie Sie den MTU-Wert von seinem Standard 1500 auf 1454 setzen:

Für DHCP (Näheres in Abschnitt [5.6.4](#)) können Sie den entsprechenden **iface**-Eintrag in `/etc/network/interfaces` wie folgt ersetzen:

```
iface eth0 inet dhcp
pre-up /sbin/ifconfig $IFACE mtu 1454
```

Bei Verwendung statischer IPs (wie in Abschnitt [5.6.5](#)) können Sie den entsprechenden **iface**-Eintrag in `/etc/network/interfaces` wie folgt anpassen:

```
iface eth0 inet static
address 192.168.11.100
netmask 255.255.255.0
gateway 192.168.11.1
mtu 1454
dns-domain example.com
dns-nameservers 192.168.11.1
```

Bei Verwendung von direktem PPPoE (wie in Abschnitt [5.5.5](#)) können Sie die entsprechende `mtu`-Zeile in `/etc/ppp/peers/dsl` durch folgendes ersetzen:

```
mtu 1454
```

Die [Maximum Segment Size](#) (MSS) wird als alternative Messmethode für die Paketgröße verwendet. Der Zusammenhang zwischen MSS und MTU ist wie folgt:

- $MSS = MTU - 40$ bei IPv4
- $MSS = MTU - 60$ bei IPv6

Anmerkung

Bei Netzwerkoptimierung mittels `iptables(8)` (lesen Sie dazu auch Abschnitt [5.10](#)) kann die Paketgröße über die MSS begrenzt werden; dies ist für einen Router nützlich. Lesen Sie den Abschnitt bezüglich `"TCPMSS"` in `iptables(8)`.

5.9.3 WAN-TCP-Optimierung

Der TCP-Durchsatz kann über die Anpassung von Parametern zur TCP-Puffergröße maximiert werden, wie es die Anleitungen ["TCP Tuning Guide"](#) und ["TCP Tuning"](#) für modernes WAN mit hoher Bandbreite und hoher Latenz beschreiben. Das soll hierzu genügen; die aktuellen Debian-Standardeinstellungen funktionieren für mein LAN mit seiner Verbindung zum sehr schnellen 1Gbps FFTP-Dienst sehr gut.

5.10 Die Netfilter-Infrastruktur

Netfilter stellt eine Infrastruktur für [Stateful Packet Inspection \(SPI, zustandsorientierte Paketüberprüfung\)](#) und [Network Address Translation \(NAT\)](#) über Module des [Linux-Kernels](#) (lesen Sie hierzu Abschnitt 3.3.1) zur Verfügung.

Pakete	Popcon	Größe	Beschreibung
iptables	V:300, I:993	2520	Administrationswerkzeuge für netfilter (iptables(8) für IPv4, ip6tables(8) für IPv6)
arptables	V:0, I:2	96	Administrationswerkzeuge für netfilter (arptables(8) für ARP)
ebtables	V:15, I:39	265	Administrationswerkzeuge für netfilter (ebtables(8) für Ethernet-Bridging-Betrieb)
iptstate	V:0, I:3	116	Fortlaufende Überwachung des netfilter -Status (ähnlich zu top(1))
shorewall-init	V:0, I:0	68	Initialisierung der Shoreline Firewall
shorewall	V:5, I:13	2458	Erzeugung von netfilter -Konfigurationsdateien für Shoreline Firewall
shorewall-lite	V:0, I:0	65	Erzeugung von netfilter -Konfigurationsdateien für Shoreline Firewall (abgespeckte Version)
shorewall6	V:1, I:2	779	Erzeugung von netfilter -Konfigurationsdateien für Shoreline Firewall (IPv6-Version)
shorewall6-lite	V:0, I:0	64	Erzeugung von netfilter -Konfigurationsdateien für Shoreline Firewall (abgespeckte IPv6-Version)

Tabelle 5.19: Liste von Firewall-Werkzeugen

Das vorherrschende Nutzerprogramm für [netfilter](#) ist iptables(8). Sie können [netfilter](#) von Hand interaktiv über die Shell konfigurieren, seinen Status mit iptables-save(8) sichern und beim Systemstart über ein Init-Skript mittels iptables-restore(8) wiederherstellen.

Konfigurations-Hilfsskripte wie [shorewall](#) vereinfachen diesen Prozess.

Sie finden Dokumentation unter <http://www.netfilter.org/documentation/> oder in `"/usr/share/doc/iptables/html/":`

- [Linux Networking-Concepts HOWTO](#)
- [Linux 2.4 Packet Filtering HOWTO](#)
- [Linux 2.4 NAT HOWTO](#)

Tipp

Obwohl für Linux 2.4 geschrieben, sind sowohl der iptables(8)-Befehl wie auch die Netfilter-Kernel-Funktionalität für die Linux-Kernel-Serien 2.6 und 3.x passend.

Kapitel 6

Netzwerkapplikationen

Nach dem Aufbau der Netzwerkverbindung (laut Kapitel 5) können Sie verschiedenste Netzwerkapplikationen nutzen.

Tipp

Bezüglich einer aktuellen Anleitung für Debian zum Thema Netzwerk-Infrastruktur lesen Sie das [Debian Administratorhandbuch —Netzwerk-Infrastruktur](#).

Tipp

Falls Sie "2-Schritt-Verifizierung" aktiviert haben, müssen Sie bei manchen Internet-Providern ein Anwendungspasswort beziehen, um aus Ihrem Anwendungsprogramm Zugriff auf POP- und SMTP-Dienste zu erhalten. Auch kann es erforderlich sein, dass Sie im voraus Ihre Host-IP-Adresse bestätigen müssen.

6.1 Webbrowser

Es gibt viele [Webbrowser](#)-Pakete, um über das [Hypertext Transfer Protocol](#) (HTTP-Protokoll) auf ferne Inhalte zuzugreifen:

Paket	Popcon	Größe	Art	Beschreibung des Webbrowsers
chromium	V:51, I:141	180040	X	Chromium (quelloffener Browser von Google)
firefox	V:13, I:20	205631	”	Firefox , (quelloffener Browser von Mozilla, nur in Debian Unstable verfügbar)
firefox-esr	V:217, I:437	198436	”	Firefox ESR , (Firefox-Variante mit erweitertem Support-Umfang, ESR)
epiphany-browser	V:4, I:24	3730	”	GNOME , HIG -konform, Epiphany
konqueror	V:18, I:100	20763	”	KDE , Konqueror
dillo	V:1, I:7	1536	”	Dillo (ressourcenschonender Browser, FLTK -basiert)
w3m	V:31, I:284	2289	Text	w3m
lynx	V:13, I:98	1948	”	Lynx
elinks	V:6, I:28	1767	”	ELinks
links	V:6, I:39	2249	”	Links (Nur-Text)
links2	V:1, I:15	5417	grafisch	Links (Konsolen-Grafik ohne X)

Tabelle 6.1: Liste der Webbrowser

6.1.1 Browser-Konfiguration

Manche Web-Browser erlauben es, über folgende spezielle URLs die Einstellungen zu kontrollieren:

- "about:"
- "about:config"
- "about:plugins"

Debian bietet viele Pakete mit freien Browser-Plugins im main-Bereich des Archivs. Sie ermöglichen über die Nutzung von [Java \(Software-Plattform\)](#) und [Flash](#) hinaus die Wiedergabe von [MPEG](#), [MPEG2](#), [MPEG4](#), [DivX](#), [Windows Media Video \(.wmv\)](#), [QuickTime \(.mov\)](#), [MP3 \(.mp3\)](#) und [Ogg/Vorbis](#). Auch Inhalte von Datenträgern wie DVDs, VCDs usw. werden zugänglich. Debian stellt im contrib- oder non-free-Bereich des Archivs auch Hilfsprogramme zur Installation von nicht-freien Browser-Paketen bereit.

Paket	Popcon	Größe	Bereich	Beschreibung
pepperflashplugin-nonfree	V:1, I:21	29	contrib	Pepper Flash-Player - Browser-Plugin
browser-plugin-freshplayer-pepperflash	I:9	1135	contrib	NPAPI-Plugin-Adapter für pepperflash (auf PPAPI-Host)

Tabelle 6.2: Liste von Browser-Plugin-Paketen

Tipp

Obwohl die Verwendung der oben genannten Debian-Pakete viel einfacher ist, können Browser-Plugins auch manuell aktiviert werden, indem die entsprechende "*.so"-Datei in das entsprechende Plugin-Verzeichnis (z.B. "/usr/lib/iceweasel/plugins/") installiert wird; anschließend muss der Browser neu gestartet werden.

Einige Webseiten lehnen Verbindungen aufgrund der user-agent-Angabe des Browsers ab. Sie können diese Situation umgehen, indem Sie die [user-agent-Angabe fälschen](#), zum Beispiel über das Hinzufügen der folgenden Zeile zu Benutzer-Konfigurationsdateien wie "~/.gnome2/epiphany/mozilla/epiphany/user.js" oder "~/.mozilla/firefox/*.default/user.js":

```
user_pref("general.useragent.override", "Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0) ↵";
```

Alternativ können Sie diese Variable hinzufügen und zurücksetzen, indem Sie "about:config" in die Adresszeile eingeben und einen Rechtsklick auf den angezeigten Inhalt ausführen.



Achtung

Eine gefälschte user-agent-Angabe kann möglicherweise [nachteilige Nebenwirkungen bei der Verwendung von Java](#) haben.

6.2 Das Mail-System



Achtung

Falls Sie vorhaben, den Mail-Server so einzurichten, dass Mails direkt über das Internet ausgetauscht werden, gibt es wahrscheinlich bessere Beschreibungen als dieses grundlegende Dokument.

Beim Mail-Verkehr arbeiten viele Server- und Client-Programme auf verschiedenen Rechnern zusammen. Von der Funktionalität her gibt es drei Arten von Mail-Agent-Programmen:

- Der Mail Transfer Agent ([MTA](#), Näheres in Abschnitt [6.3](#)) ist ein Programm für den Transfer von Mails zwischen verschiedenen Rechnern.
- Der Mail Delivery Agent ([MDA](#), Näheres in Abschnitt [6.6](#)) ist ein Programm zur Ablage von Mails innerhalb eines Rechners in die Mailbox der Benutzer.
- Der Mail User Agent (MUA, auch bekannt als [E-Mail-Client](#), Näheres in Abschnitt [6.4](#)) ist das Programm, mit dem Sie neue Mails erstellen und empfangene Mails anzeigen.

Anmerkung

Die folgenden Konfigurationsbeispiele passen nur für typische Arbeitsplatzrechner an Internetverbindungen, wie sie für normale Privatkunden üblich sind.

6.2.1 Grundlagen des E-Mail-Systems

Eine [E-Mail](#) besteht aus drei Komponenten: den Absender- und Empfänger-Informationen, wie sie bei einem normalen Brief auf dem Umschlag stehen würden, den Nachrichten-Kopfzeilen (Header) und der eigentlichen Nachricht (dem "Nachrichtenkörper").

Die "An"- und "Von"-Informationen ("To" und "From") auf dem "Umschlag" werden von [SMTP](#) genutzt, um die Nachricht zuzustellen. (Die "From"-Information auf dem Umschlag wird auch mit [Bounce address/Envelope sender](#), From_ usw. bezeichnet).

Die "An"- und "Von"-Informationen ("To" und "From") in den Nachrichten-Kopfzeilen werden vom [E-Mail-Client](#) (E-Mail-Programm) angezeigt. (Obwohl es sehr gängig ist, dass diese mit den Informationen vom "Umschlag" übereinstimmen, muss dies nicht immer der Fall sein.)

Das [E-Mail-Programm](#) (MUA) muss die Nachrichten-Kopfzeilen und den Nachrichtenkörper mittels der [Multipurpose Internet Mail Extensions \(MIME\)](#) auswerten, um den Datentyp und die Kodierung des Nachrichteninhalts korrekt zu bestimmen.

6.2.2 Grundlagen für moderne Mail-Dienste

Um Probleme mit Spam (unerwünschte und nicht angeforderte Nachrichten) zu minimieren, implementieren viele Internet-Diansteanbieter für Endbenutzer entsprechende Gegenmaßnahmen:

- Der Smarthost-Dienst für das Versenden von Nachrichten durch die Kunden nutzt den Message-Submission-Port 587 (gemäß [RFC 4409](#)) mit Passwort-Authentifizierung ([SMTP AUTH](#) gemäß [RFC 4954](#)).
- Verbindungen über den [SMTP](#)-Port 25 von deren internen Netzwerk-Rechnern zum Internet werden geblockt (außer die Mail-Ausgangsserver des Providers selbst).
- Verbindungen über den [SMTP](#)-Port 25 zum Mail-Eingangsserver des Providers von problematischen externen Netzwerk-Rechnern werden geblockt. (Verbindungen von Rechnern im dynamischen IP-Adressbereich, wie sie bei Einwahlverbindungen und anderen von Privatkunden genutzten Verbindungen verwendet werden, sind die ersten, die geblockt werden.)
- Zur [E-Mail-Filterung](#) werden großflächig [Anti-Spam-Techniken](#) wie [DomainKeys Identified Mail \(DKIM\)](#), [Sender Policy Framework \(SPF\)](#) und [Domain-based Message Authentication, Reporting and Conformance \(DMARC\)](#) eingesetzt.
- Unter Umständen können Sie zum Versand von Mails über einen Smarthost den [DomainKeys Identified Mail](#)-Dienst verwenden.
- Der Smarthost-Dienst überschreibt unter Umständen die Absenderadresse mit der Ihres Mail-Accounts.

Wenn Sie Ihr Mail-System konfigurieren oder Probleme bei der Auslieferung von E-Mails beheben möchten, sollten Sie diese neuen Einschränkungen mit in Betracht ziehen.

**Achtung**

Sie können nicht davon ausgehen, dass das Betreiben eines SMTP-Servers an einem Endverbraucher-Anschluß (und somit das Senden von Nachrichten direkt an einen fernen Rechner) zuverlässig funktioniert.

**Achtung**

Sie können nicht davon ausgehen, dass das Versenden von Mails an ferne Rechner mit einer zufälligen Absender-Mail-Adresse über einen eigenen Smarthost zuverlässig funktioniert.

**Achtung**

Eine E-Mail kann an jeglicher Stelle auf der Route zum Ziel ohne weiteren Hinweis verworfen werden. Der einzige Weg, um eine Mail zuverlässig bis zu ihrem Empfänger zu bringen, ist, sie so authentisch wie möglich zu gestalten.

Im Licht dieser feindseligen Internet-Situation und dieser Einschränkungen bieten einige unabhängige Internet-Mail-Provider wie Yahoo.com und Gmail.com einen sicheren Mail-Dienst an, der von überall aus dem Internet über [Transport Layer Security \(TLS\)](#) und seinen Vorgänger [Secure Sockets Layer \(SSL\)](#) genutzt werden kann:

- Der Smarthost-Dienst über Port 465 mit dem veralteten SMTP-over-SSL ([SMTPS](#)-Protokoll).
- Der Smarthost-Dienst über Port 587 mit [STARTTLS](#).
- Ankommende Nachrichten sind erreichbar über den TLS/POP3-Port 995 via [POP3](#).

Der Einfachheit halber gehe ich hier davon aus, dass sich der Smarthost auf "smtp.rechnername.dom" befindet, [SMTP-Authentifizierung](#) benötigt und den Message-Submission-Port 587 via [STARTTLS](#) verwendet.

6.2.3 Installationskonzept für Mail-Programme an Arbeitsplatz-Systemen

Die einfachste Variante einer Mail-Konfiguration ist, dass Mails zum Smarthost des Providers geschickt bzw. direkt durch den MUA (das Programm, mit dem Sie Ihre Mails lesen und schreiben, Näheres dazu in Abschnitt 6.4) vom POP3-Server des Providers abgeholt werden. Diese Art der Konfiguration ist üblich für komfortabel ausgestattete grafische Mail-Programme wie icedove(1) oder evolution(1). Wenn Sie Nachrichten basierend auf ihrem Typ filtern möchten, nutzen Sie die Filterfunktionen des Mail-Programms. In diesem Fall muss der lokale MTA (ein Dienst rein für den Transport von E-Mails, siehe Abschnitt 6.3) lediglich den Transport von Nachrichten lokal innerhalb des Rechners (wenn Absender und Empfänger sich auf dem gleichen Rechner befinden) durchführen.

Bitte beachten Sie, dass Debian ein Mehrbenutzersystem ist. Sogar wenn Sie der einzige Nutzer des Rechners sind, gibt es viele Programme, die als root laufen und die Ihnen Mails schicken könnten.

Ein alternativer Ansatz für die Konfiguration des Mail-Systems ist, dass die Nachrichten durch den lokalen MTA zum Smarthost des Providers geschickt bzw. durch den MDA (ein Dienst zum Zustellen von Mails, siehe Abschnitt 6.5) vom POP3-Server des Providers abgeholt werden. Wenn Sie die Nachrichten basierend auf ihrem Typ filtern möchten, nutzen Sie einen MDA mit Filterfunktion (lesen Sie dazu Abschnitt 6.6), um Mails in separate Mailboxen einzusortieren. Diese Art der Mail-Konfiguration ist üblich für einfache konsolenbasierte Mailprogramme wie mutt(1) oder mew(1), obwohl sie auch mit jedem anderen MUA (Abschnitt 6.4) möglich ist. Der lokale MTA (Abschnitt 6.3) muss hier sowohl den Transfer zum Smarthost des Providers wie auch den lokalen Transfer für Mails innerhalb des Rechners durchführen. Da Privatkundenrechner meist keinen gültigen FQDN (vollständigen, im Internet gültigen Namen einer Domain) haben, muss der lokale MTA so konfiguriert werden, dass der lokale Mail-Name in ausgehenden Nachrichten versteckt bzw. verändert wird, um Fehler bei der Auslieferung der Nachrichten zu vermeiden (lesen Sie dazu Abschnitt 6.3.3).

Tipp

Erwägen Sie, den MUA/MDA so konfigurieren, dass [Maildir](#) genutzt wird; das gibt Ihnen die Möglichkeit, die E-Mails irgendwo frei wählbar in Ihrem Heimatverzeichnis abzulegen.

6.3 Mail Transfer Agent (MTA)

Auf normalen Arbeitsplatzrechnern sind die `exim4-*`- oder `postfix`-Pakete eine populäre Wahl für den Mail Transfer Agent (MTA). Es liegt an Ihnen.

Paket	Popcon	Größe	Beschreibung
exim4-daemon-light	V:342, I:367	1493	Exim4 Mail Transfer Agent (MTA: Debian-Standard)
exim4-base	V:349, I:377	1704	Exim4-Dokumentation (Text-Variante) und allgemeine Dateien
exim4-doc-html	I:1	3662	Exim4-Dokumentation (html-Variante)
exim4-doc-info	I:1	624	Exim4-Dokumentation (info-Variante)
postfix	V:145, I:160	4182	Postfix Mail Transfer Agent (MTA: alternativ)
postfix-doc	I:9	4444	Postfix-Dokumentation (html- + Text-Variante)
sas12-bin	V:5, I:19	428	Cyrus SASL API Implementation (Ergänzung zu Postfix für SMTP AUTH)
cyrus-sas12-doc	I:1	575	Cyrus SASL - Dokumentation

Tabelle 6.3: Liste von grundlegenden Paketen für Mail Transfer Agents

Obwohl aufgrund des Popcon-Werts `exim4-*` populärer zu sein scheint als `postfix`, heißt das nicht, dass `postfix` unter Debian-Entwicklern nicht beliebt ist. Das Debian-Server-System nutzt sowohl `exim4` wie auch `postfix`. Eine [Analyse der Mail-Header](#) von Mailinglisten-Nachrichten prominenter Debian-Entwickler zeigt, dass beide MTAs gleich beliebt sind.

Die `exim4-*`-Pakete sind bekannt für einen sehr kleinen Speicherbedarf und eine hohe Flexibilität bei der Konfiguration. Das `postfix`-Paket ist als kompakt, schnell, einfach und sicher bekannt. Beide enthalten ausführliche Dokumentation und sind bezüglich Qualität und Lizenz gleich gut.

Es gibt eine große Auswahl von Paketen für Mail Transfer Agents (MTA) mit unterschiedlichen Funktionalitäten und Fokus im Debian-Archiv.

Paket	Popcon	Größe	Funktionalitäten und Fokus
exim4-daemon-light	V:342, I:367	1493	vollständig
postfix	V:145, I:160	4182	vollständig (Sicherheit)
exim4-daemon-heavy	V:7, I:8	1643	vollständig (flexibel)
sendmail-bin	V:14, I:15	1854	vollständig (nur, wenn Sie sich damit bereits auskennen)
nullmailer	V:7, I:10	479	eingeschränkt, kein lokaler Mail-Transfer
ssmtp	V:8, I:11	2	eingeschränkt, kein lokaler Mail-Transfer
courier-mta	V:0, I:0	2416	absolut vollständig (inklusive Web-Interface usw.)
masqmail	V:0, I:0	337	gering
esmtip	V:0, I:0	128	gering
esmtip-run	V:0, I:0	32	gering (Sendmail-Kompatibilitätserweiterung für <code>esmtip</code>)
msmtip	V:5, I:10	547	gering
msmtip-mta	V:3, I:4	86	gering (Sendmail-Kompatibilitätserweiterung für <code>msmtip</code>)

Tabelle 6.4: Liste von Mail-Transport-Agent-Paketen im Debian-Archiv

6.3.1 Die Konfiguration von exim4

**Achtung**

Die Konfiguration von exim4 zum Versenden von E-Mails über mehrere Absender-Mail-Adressen und mehrere dazugehörige Smarthosts ist alles andere als einfach. Bitte richten Sie für Systemprogramme wie `popcon` und `cron` in exim4 nur eine einzige E-Mail-Adresse ein; für Anwenderprogramme wie `mutt` jedoch können Sie `smtp` für eine Reihe von Quell-Mail-Adressen konfigurieren.

Für das Verschicken von Mails über das Internet via Smarthost (re-)konfigurieren Sie die `exim4-*`-Pakete wie folgt:

```
$ sudo /etc/init.d/exim4 stop
$ sudo dpkg-reconfigure exim4-config
```

Wählen Sie bei "Generelle E-Mail-Einstellungen" den Eintrag "Versand über Sendezentrale (Smarthost); Empfang mit SMTP oder Fetchmail".

Setzen Sie den "E-Mail-Name des Systems" auf seinen Standardwert, den FQDN (Näheres dazu in Abschnitt 5.1.1).

Bei "IP-Adressen, an denen eingehende SMTP-Verbindungen erwartet werden" wählen Sie den Standardwert "127.0.0.1 ; ::1".

Löschen Sie alle eingetragenen Werte bei der Abfrage von "Weitere Ziele, für die E-Mails angenommen werden sollen".

Löschen Sie alle eingetragenen Werte bei der Abfrage von "Rechner, für die E-Mails weitergeleitet werden (Relay)".

Setzen Sie "IP-Adresse oder Rechnername der Sendezentrale für ausgehende E-Mails" auf "smtp.hostname.dom:587".

Wählen Sie "<Nein>" bei der Frage "Lokalen E-Mail-Namen in ausgehenden E-Mails verbergen?". (Nutzen Sie stattdessen `/etc/email-addresses` wie in Abschnitt 6.3.3 beschrieben.)

Bei der Abfrage "DNS-Anfragen minimieren (Automatische Einwahl, Dial-on-Demand)?" gehen Sie wie folgt vor:

- Wählen Sie "Nein", wenn Ihr System während des Rechnerstarts mit dem Internet verbunden ist.
- Wählen Sie "Ja", wenn Ihr System während des Rechnerstarts **nicht** (oder nicht immer) mit dem Internet verbunden ist.

Setzen Sie den Wert für "Versandart bei lokaler E-Mail-Zustellung:" auf "Mbox-Format in /var/mail".

Bei der Frage "Einstellungen auf kleine Dateien aufteilen?" wählen Sie "<Ja>".

Erzeugen Sie Passworteinträge für den Smarthost, indem Sie `/etc/exim4/passwd.client` editieren:

```
$ sudo vim /etc/exim4/passwd.client
...
$ cat /etc/exim4/passwd.client
^smtp.*\.rechnername\.dom:benutzername@rechnername.dom:passwort
```

Starten Sie `exim4` mit folgendem Befehl:

```
$ sudo /etc/init.d/exim4 start
```

Der Rechnername in `/etc/exim4/passwd.client` sollte nicht der Alias-Name sein. Sie können den echten Rechnernamen wie folgt herausfinden:

```
$ host smtp.rechnername.dom
smtp.rechnername.dom is an alias for smtp99.rechnername.dom.
smtp99.rechnername.dom has address 123.234.123.89
```

Ich verwende reguläre Ausdrücke in `/etc/exim4/passwd.client`, um das Alias-Problem zu umgehen. SMTP AUTH funktioniert unter Umständen sogar, wenn der Internet-Diensteanbieter den Host (Rechner), auf den der Alias zeigt, verändert.

Sie können die `exim4`-Konfiguration händisch aktualisieren, indem Sie wie folgt vorgehen:

- Aktualisieren der `exim4`-Konfigurationsdateien in `"/etc/exim4/"`.
 - Erzeugen von `"/etc/exim4/exim4.conf.localmacros"`, um MACROs zu aktivieren und Editieren von `"/etc/exim4/exim4.conf"` (bei Installation ohne gesplittete Konfigurationsdateien)
 - Erzeugen neuer oder Editieren vorhandener Dateien in den Unterverzeichnissen von `"/etc/exim4/exim4.conf.d"` (bei Installation mit gesplitteten Konfigurationsdateien)
- Ausführen von `"invoke-rc.d exim4 reload"`.

Lesen Sie bitte die offizielle Anleitung unter `"/usr/share/doc/exim4-base/README.Debian.gz"` und `update-exim4.conf(8)`.



Achtung

Das Starten von `exim4` dauert ziemlich lange, wenn die Frage "DNS-Anfragen minimieren (Automatische Einwahl, Dial-on-Demand)?" mit "Nein" (dem Standardwert) beantwortet wurde und das System während dem Rechnerstart **nicht** mit dem Internet verbunden ist.



Warnung

Es ist nicht sicher, Klartext-Passwörter ohne Verschlüsselung zu verwenden, auch wenn Ihr Internet-Diensteanbieter dies erlaubt.

Tipp

Obwohl die Verwendung von [SMTP](#) mit [STARTTLS](#) auf Port 587 empfohlen wird, nutzen einige Provider immer noch das veraltete [SMTPS](#) (SSL auf Port 465). Exim4 ab Version 4.77 unterstützt dieses veraltete SMTPS-Protokoll sowohl als Client wie auch als Server.

Tipp

Falls Sie nach einem ressourcen-schonenden MTA für Ihren Rechner suchen, der die Einträge von `"/etc/aliases"` berücksichtigt, sollten Sie in Betracht ziehen, `exim4(8)` mit `"QUEUERUNNER='queueonly'"`, `"QUEUERUNNER='nodaemon'"` usw. in `"/etc/default/exim4"` zu konfigurieren.

6.3.2 Die Konfiguration von Postfix mit SASL

Für das Verschicken von Mails über das Internet via Smarthost sollten Sie zunächst die [Postfix-Dokumentation](#) und einschlägige dazugehörige Handbuchseiten lesen.

Befehl	Funktion
<code>postfix(1)</code>	Postfix-Steuerprogramm
<code>postconf(1)</code>	Postfix-Konfigurationswerkzeug
<code>postconf(5)</code>	Postfix-Konfigurationsparameter
<code>postmap(1)</code>	Wartung von Postfix-Lookup-Tabellen
<code>postalias(1)</code>	Wartung der Postfix-Alias-Datenbank

Tabelle 6.5: Liste von wichtigen Postfix-Handbuchseiten

Sie (re-)konfigurieren die `postfix`- und `sasl2-bin`-Pakete wie folgt:

```
$ sudo /etc/init.d/postfix stop
$ sudo dpkg-reconfigure postfix
```


Wählen Sie "Internet mit Smarthost".

Setzen Sie "SMTP-Relay-Server (leere Eingabe: keiner)" auf "[smtp.rechnername.dom]:587" und konfigurieren Sie ihn mit folgenden Befehlen:

```
$ sudo postconf -e 'smtp_sender_dependent_authentication = yes'
$ sudo postconf -e 'smtp_sasl_auth_enable = yes'
$ sudo postconf -e 'smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd'
$ sudo postconf -e 'smtp_sasl_type = cyrus'
$ sudo vim /etc/postfix/sasl_passwd
```

Erzeugen Sie Passworteinträge für den Smarthost:

```
$ cat /etc/postfix/sasl_passwd
[smtp.rechnername.dom]:587      benutzername:passwort
$ sudo postmap hash:/etc/postfix/sasl_passwd
```

Starten Sie postfix wie folgt:

```
$ sudo /etc/init.d/postfix start
```

Hierbei stellt die Verwendung von "[" und "]" im dpkg-reconfigure-Dialog und in "/etc/postfix/sasl_passwd" sicher, dass keine MX-Einträge abgefragt werden, sondern direkt der angegebene Rechnername genutzt wird. Lesen Sie dazu "Enabling SASL authentication in the Postfix SMTP client" in "/usr/share/doc/postfix/html/SASL_README.html".

6.3.3 Die Mail-Adress-Konfiguration

Es gibt mehrere [Dateien zur Konfiguration der Mail-Adresse](#) für Mail Transfer Agents, Mail Delivery Agents und Mail User Agents.

Datei	Funktion	Applikation
/etc/mailname	Standard-Rechnername für (ausgehende) Mails	Debian-spezifisch, mailname(5)
/etc/email-addresses	Änderung des Rechnernamens für ausgehende Mails	exim(8)-spezifisch, exim4-config_files(5)
/etc/postfix/generic	Änderung des Rechnernamens für ausgehende Mails	postfix(1)-spezifisch, aktiviert nach Ausführung des postmap(1)-Befehls
/etc/aliases	Kontonamen-Alias für eingehende Mails	Grundlegend, aktiviert nach Ausführung des newaliases(1)-Befehls

Tabelle 6.6: Liste von Konfigurationsdateien für Mail-Adressen

Der **mailname** in der Datei "/etc/mailname" ist normalerweise ein voll-qualifizierter Domain-Name (FQDN), der sich auf eine der IP-Adressen des Rechners auflösen lässt. Bei Privatkundenrechnern, die keinen solchen FQDN-Namen mit einer aufzulösenden IP-Adresse haben, setzen Sie diesen **mailname** auf den Wert von "hostname -f". (Dies ist eine sichere Wahl und funktioniert sowohl für exim4-* wie auch für postfix.)

Tipp

Der Inhalt von "/etc/mailname" wird von vielen nicht-MTA-Programmen für deren Standardverhalten genutzt. Für mutt setzen Sie die "hostname"- und "from"-Variablen in der Datei ~/.muttrc, um den **mailname**-Wert zu überschreiben. Für Programme im devscripts-Paket, wie bts(1) und dch(1), exportieren Sie die Umgebungsvariablen "\$DEBFULLNAME" und "\$DEBEMAIL", um ihn zu überschreiben.

Tipp

Das Paket `popularity-contest` sendet normalerweise Mails vom `root`-Konto über den FQDN. Sie müssen `MAILFROM` in `/etc/popularity-contest.conf` setzen, wie in der Datei `/usr/share/popularity-contest/default.conf` beschrieben. Andernfalls wird Ihre Mail vom Smarthost-SMTP-Server abgelehnt werden. Obwohl lästig, ist dieser Ansatz sicherer als das Neuschreiben der Ursprungsadresse für alle Mails von `root` über den MTA und sollte für andere Daemons und cron-Skripte ebenfalls genutzt werden.

Wenn Sie den **mailname** auf den Wert von `"hostname -f"` setzen, kann die Änderung der Mail-Ursprungsadresse via MTA wie folgt realisiert werden:

- für `exim4(8)` über die Datei `"/etc/email-addresses"`, wie in `exim4-config_files(5)` beschrieben;
- für `postfix(1)` über die Datei `"/etc/postfix/generic"`, wie in `generic(5)` beschrieben.

Für `postfix` sind die folgenden zusätzlichen Schritte nötig:

```
# postmap hash:/etc/postfix/generic
# postconf -e 'smtp_generic_maps = hash:/etc/postfix/generic'
# postfix reload
```

Sie können die Mailadressen-Konfiguration wie folgt testen:

- `exim(8)` mit den Optionen `-brw`, `-bf`, `-bF`, `-bV`, ...;
- `postmap(1)` mit der Option `-q`.

Tipp

Exim wird zusammen mit mehreren Hilfsprogrammen wie `exiqgrep(8)` und `exipick(8)` ausgeliefert. Informationen über verfügbare Befehle finden Sie mittels `"dpkg -L exim4-base | grep man8/"`.

6.3.4 Grundlegende MTA-Operationen

Es gibt mehrere grundlegende MTA-Operationen. Einige könnten über die `sendmail(1)`-Kompatibilitätsschnittstelle ausgeführt werden.

Tipp

Es könnte eine gute Idee sein, alle Mails durch ein Skript in `"/etc/ppp/ip-up.d/*"` zustellen zu lassen.

6.4 Mail User Agent (MUA)

Wenn Sie Debian-bezogene Mailinglisten abonnieren, könnte es sinnvoll sein, einen MUA wie `mutt` oder `mew` zu verwenden, welche der De-Facto-Standard für Teilnehmer sind und sich wie dort erwartet verhalten.

6.4.1 Grundlegender MUA -- Mutt

Passen Sie `"~/ .muttrc"` wie folgt an, um `mutt` als Mail User Agent (MUA) in Kombination mit `vim` zu nutzen.

exim-Befehl	postfix-Befehl	Beschreibung
sendmail	sendmail	Mails von der Standardeingabe lesen und für die Zustellung vorbereiten (-bm)
mailq	mailq	Die Mail-Warteschlange auflisten mit Status und Queue-ID (-bp)
newaliases	newaliases	Initialisieren der Alias-Datenbank (-I)
exim4 -q	postqueue -f	Wartende Mails zustellen (-q)
exim4 -qf	postsuper -r ALL deferred; postqueue -f	Alle Mails zustellen
exim4 -qff	postsuper -r ALL; postqueue -f	Zurückgehaltene Mails ebenfalls zustellen
exim4 -Mg queue_id	postsuper -h queue_id	Eine durch ihre Queue-ID spezifizierte Nachricht zurückhalten
exim4 -Mrm queue_id	postsuper -d queue_id	Eine durch ihre Queue-ID spezifizierte Nachricht löschen
Nicht verfügbar	postsuper -d ALL	Alle Nachrichten löschen

Tabelle 6.7: Liste grundlegender MTA-Operationen

Paket	Popcon	Größe	Art
evolution	V:31, I:229	475	Programm mit grafischer X-GUI-Oberfläche (GNOME3, Groupware-Suite)
thunderbird	V:57, I:138	165180	Programm mit grafischer X-GUI-Oberfläche (GNOME2, abgewandelter Mozilla Thunderbird)
kmail	V:34, I:88	18011	Programm mit grafischer X-GUI-Oberfläche (KDE)
mutt	V:37, I:313	7056	Programm für textbasiertes Terminal, zu benutzen u.a. mit vim
mew	V:0, I:0	2325	Programm für textbasiertes Terminal, zu benutzen mit (x) emacs

Tabelle 6.8: Liste der Mail User Agents (MUA)

```
#
# Benutzer-spezifische Konfigurationsdatei mit Vorrang gegenüber /etc/Muttrc
#
# Quell-Mail-Adresse verändern
set use_from
set hostname=example.dom
set from="Vorname Nachname <benutzername@example.dom>"
set signature="~/signature"

# vim: "gq" zum Umformatieren von Zitaten
set editor="vim -c 'set tw=72 et ft=mail'"

# "mutt" zeigt den Posteingang, während "mutt -y" die Mailboxen auflistet
set mbox_type=Maildir           # qmail-Maildir-Format zur Erzeugung der mbox verwenden
set mbox=~/.Mail               # alle Mailboxen in $HOME/Mail/ ablegen
set spoolfile=+Inbox           # Neue Mails in $HOME/Mail/Inbox einliefern
set record=+Outbox             # Kopien ausgehender Mails in $HOME/Mail/Outbox ablegen
set postponed=+Postponed       # Zurückgestellte Mails in $HOME/Mail/postponed ablegen
set move=no                    # Inbox-Elemente nicht in mbox verschieben
set quit=ask-yes               # Nicht mit einem einfachen "q" beenden, sondern vorher ↵
    nachfragen
set delete=yes                 # Elemente beim Beenden ohne Nachfrage löschen
set fcc_clear                   # Kopien ausgehender Mails unverschlüsselt speichern

# Mailboxen in Maildir (automatische Aktualisierung)
mailboxes 'cd ~/.Mail; /bin/ls -l|sed -e 's/^/+/' | tr "\n" " "'
unmailboxes Maillog *.ev-summary

## Standardeinstellung:
#set index_format="%4C %Z %{b %d} %-15.15L (%4l) %s"
## Index von Antwortketten mit Sender (ausgeklappt)
set index_format="%4C %Z %{b %d} %-15.15n %?M?(&#03M)&(%4l)? %s"

## Standardeinstellung:
#set folder_format="%2C %t %N %F %2l %-8.8u %-8.8g %8s %d %f"
## Nur Ordernamen
set folder_format="%2C %t %N %f"
```

Fügen Sie folgendes zu `/etc/mailcap` oder `~/mailcap` hinzu, um HTML-Mails und MS-Word-Anhänge im Mailprogramm integriert anzuzeigen:

```
text/html; lynx -force_html %s; needsterminal;
application/msword; /usr/bin/antiword '%s'; copiousoutput; description="Microsoft Word Text ↵
"; nametemplate=%s.doc
```

Tipp

Mutt kann als [IMAP](#)-Client und Mailbox-Formatkonvertierer verwendet werden. Sie können Nachrichten mit `"t"`, `"T"` usw. markieren. Diese markierten Nachrichten können mit `;"C"` zwischen verschiedenen Mailboxen kopiert und mit `;"d"` in einem Rutsch gelöscht werden.

6.4.2 Erweiterter MUA -- Mutt + msmtplib

Mutt kann unter Verwendung von [msmtplib](#) so konfiguriert werden, dass er mehrere unterschiedliche Absender-Mail-Adressen mit mehreren dazugehörigen Smarthosts nutzen kann.

Tipp

msmtp ist ein sendmail-Emulator, der es erlaubt, parallel zu einem weiteren sendmail-Emulator installiert zu werden, der den `/usr/sbin/sendmail`-Befehl bereitstellt. Somit können Sie Ihre System-Mails bei `exim4` oder `postfix` belassen.

Wir nehmen als Beispiel an, dass 3 unterschiedliche E-Mail-Adressen unterstützt werden sollen:

- "My Name1 <myaccount1@gmail.com>"
- "My Name2 <myaccount2@gmail.com>"
- "My Name3 <myaccount3@example.org>"

Hier als Beispiel eine entsprechend angepasste `~/muttrc`, in der 3 Smarthosts mit 3 verschiedenen Absender-Mail-Adressen unterstützt werden:

```
set use_from
set from="My Name3 <myaccount3@example.org>"
set reverse_name
alternates myaccount1@gmail\|.com|myaccount1@gmail\|.com|myaccount3@example\|.org

# ...

# MACRO
macro compose "1" "<edit-from>^UMy Name1 \<myaccount1@gmail.com\>\n"
macro compose "2" "<edit-from>^UMy Name2 \<myaccount2@gmail.com\>\n"
macro compose "3" "<edit-from>^UMy Name3 \<myaccount3@example.org\>\n"

send2-hook '~f myaccount1@gmail.com' "set sendmail = '/usr/bin/msmtp --read-envelope-from'"
send2-hook '~f myaccount2@gmail.com' "set sendmail = '/usr/bin/msmtp --read-envelope-from'"
send2-hook '~f myaccount3@example.org' "set sendmail = '/usr/bin/msmtp --read-envelope-from" ↵
""

# ...
```

Wir installieren `msmtp-gnome` und erstellen `~/msmtprc` wie folgt:

```
defaults
logfile ~/.msmtp.log
domain myhostname.example.org
tls on
tls_starttls on
tls_certcheck on
tls_trust_file /etc/ssl/certs/ca-certificates.crt
auth on
port 587
auto_from

account myaccount1@gmail.com
host smtp.gmail.com
from myaccount1@gmail.com
user myaccount1@gmail.com

account myaccount2@gmail.com
host smtp.gmail.com
from myaccount2@gmail.com
user myaccount2@gmail.com

account myaccount3@example.org
host mail.example.org
```

```
from myaccount3@example.org
user myaccount3@example.org

account default : myaccount3@example.org
```

Fügen Sie dann die Passwortdaten dem Gnome-Schlüsselbund hinzu. Zum Beispiel:

```
$ secret-tool store --label=msmtp \
    host smtp.gmail.com \
    service smtp \
    user myaccount1@gmail.com
...
```

Tipp

Falls Sie den Gnome-Schlüsselbund nicht verwenden möchten, können Sie stattdessen das Paket `msmtp` installieren und einen Eintrag wie `"password geheim1234"` für jeden Mail-Account in `~/.msmtp.rc` hinzufügen. Lesen Sie dazu auch die [memtp-Dokumentation](#).

6.5 Programme zum Abrufen und Weiterleiten fern gespeicherter Mails

Statt einen MUA laufen zu lassen, der auf Mails auf einem fernen Server zugreift, und diese manuell zu verarbeiten, möchten Sie diesen Prozess vielleicht automatisieren, um alle Mails auf Ihren lokalen Rechner vor Ort geliefert zu bekommen. Die hier genannten Programme sind die richtigen für diesen Fall.

Obwohl `fetchmail`(1) der De-Facto-Standard für das Abrufen von fern gespeicherten Mails unter GNU/Linux war, bevorzugt der Autor jetzt `getmail`(1). Wenn Sie Mails bereits vor dem Herunterladen verwerfen möchten, um Bandbreite zu sparen, könnte auch `mailfilter` oder `mpop` für Sie nützlich sein. Welches Mail-Abrufwerkzeug Sie auch immer verwenden, es ist eine gute Idee, das System so zu konfigurieren, dass abgerufene Mails über eine Weiterleitung einem MDA wie `maildrop` übergeben werden.

Paket	Popcon	Größe	Beschreibung
fetchmail	V:5, I:17	814	Mail-Abrufprogramm (POP3, APOP, IMAP) (alt)
getmail	V:1, I:6	30	Mail-Abrufprogramm (POP3, IMAP4 und SDPS) (einfach, sicher und zuverlässig)
mailfilter	V:0, I:0	291	Mail-Abrufprogramm (POP3) mit Filterfunktion über reguläre Ausdrücke
mpop	V:0, I:0	400	Mail-Abrufprogramm (POP3) und MDA mit Filterfunktion

Tabelle 6.9: Liste von Programmen zum Abrufen und Weiterleiten von fern gespeicherten Mails

6.5.1 getmail-Konfiguration

Die Konfiguration von `getmail`(1) ist beschrieben in der [getmail-Dokumentation](#) (englisch). Hier ist mein Setup, um mehrere POP3-Konten als Benutzer abrufen zu können.

Erzeugen Sie `"/usr/local/bin/getmails"` wie hier:

```
#!/bin/sh
set -e
if [ -f $HOME/.getmail/running ]; then
    echo "getmail is already running ... (if not, remove $HOME/.getmail/running)" >&2
    pgrep -l "getmai[l]"
    exit 1
else
```

```

    echo "getmail has not been running ... " >&2
fi
if [ -f $HOME/.getmail/stop ]; then
    echo "do not run getmail ... (if not, remove $HOME/.getmail/stop)" >&2
    exit
fi
if [ "x$1" = "x-l" ]; then
    exit
fi
rcfiles="/usr/bin/getmail"
for file in $HOME/.getmail/config/* ; do
    rcfiles="$rcfiles --rcfile $file"
done
date -u > $HOME/.getmail/running
eval "$rcfiles $"
rm $HOME/.getmail/running

```

Führen Sie folgende Konfiguration aus:

```

$ sudo chmod 755 /usr/local/bin/getmails
$ mkdir -m 0700 $HOME/.getmail
$ mkdir -m 0700 $HOME/.getmail/config
$ mkdir -m 0700 $HOME/.getmail/log

```

Erzeugen Sie für jedes POP3-Konto eine Konfigurationsdatei "\$HOME/.getmail/config/pop3_name" wie hier:

```

[retriever]
type = SimplePOP3SSLRetriever
server = pop.example.com
username = pop3_name@example.com
password = <ihr_Passwort>

[destination]
type = MDA_external
path = /usr/bin/maildrop
unixfrom = True

[options]
verbose = 0
delete = True
delivered_to = False
message_log = ~/.getmail/log/pop3_name.log

```

Führen Sie folgende Konfiguration aus:

```

$ chmod 0600 $HOME/.getmail/config/*

```

Legen Sie fest, dass "/usr/local/bin/getmails" mittels cron(8) alle 15 Minuten ausgeführt wird; Sie erreichen dies, indem Sie "sudo crontab -e -u <benutzername>" ausführen und folgendes zum cron-Eintrag des Benutzers hinzufügen:

```

5,20,35,50 * * * * /usr/local/bin/getmails --quiet

```

Tipp

Probleme mit dem POP3-Zugriff müssen nicht zwingend von getmail kommen. Einige populäre freie POP3-Anbieter könnten das POP3-Protokoll missbrauchen und ihre Spam-Filter könnten nicht ideal eingestellt sein. Zum Beispiel könnten Sie Nachrichten direkt nach dem Empfangen des RETR-Kommandos löschen, statt zu warten, bis sie das DELE-Kommando bekommen haben oder Sie könnten Nachrichten in den Spam-Ordner in Quarantäne schicken. Um Schäden zu minimieren, sollten Sie die Konfiguration so wählen, dass Nachrichten, auf die zugegriffen wurde, archiviert und nicht gelöscht werden. Lesen Sie dazu auch ["Einige E-Mails wurden nicht heruntergeladen"](#) bei Gmail.

6.5.2 fetchmail-Konfiguration

Die Konfiguration von fetchmail(1) wird über `"/etc/default/fetchmail"`, `"/etc/fetchmailrc"` und `"$HOME/.fetchmailrc"` festgelegt. Lesen Sie das Beispiel in `"/usr/share/doc/fetchmail/examples/fetchmailrc.example"`.

6.6 Mail Delivery Agents (MDA) mit Filterfunktion

Die meisten MTA-Programme wie `postfix` und `exim4` fungieren als MDA (Mail Delivery Agent). Es gibt auch spezialisierte MDAs mit Filterfunktion.

Obwohl `procmail(1)` der De-Facto-Standard für MDAs mit Filterfunktion unter GNU/Linux war, bevorzugt der Autor jetzt `maildrop(1)`. Welches Filterwerkzeug Sie auch immer verwenden, es ist eine gute Idee, das System so zu konfigurieren, dass abgerufene, gefilterte Mails an eine [qmail-artige Maildir](#)-Struktur zur Speicherung übergeben werden.

Paket	Popcon	Größe	Beschreibung
procmail	V:40, I:277	300	MDA mit Filterfunktion (alt)
mailagent	V:0, I:5	1356	MDA mit Perl-Filterfunktion
maildrop	V:0, I:2	1141	MDA mit Structured Filtering Language

Tabelle 6.10: Liste von MDAs mit Filterfunktion

6.6.1 maildrop-Konfiguration

Die Konfiguration von `maildrop(1)` ist in der [maildropfilter-Dokumentation](#) beschrieben (englisch). Hier ein Konfigurationsbeispiel für `"$HOME/.mailfilter"`:

```
# Lokale Konfiguration
MAILROOT="$HOME/Mail"
# Setzen Sie dies auf den Inhalt von /etc/mailname
MAILHOST="example.dom"
logfile $HOME/.maildroplog

# Mit den Regeln wird der frühere Wert durch den späteren ersetzt.

# Mailinglisten-Nachrichten?
if (    /^Precedence:.*list/:h || /^Precedence:.*bulk/:h )
{
    # Regeln für Mailinglisten-Nachrichten
    # Standard-Mailbox für Nachrichten von Mailinglisten
    MAILBOX="Inbox-list"
    # Standard-Mailbox für Nachrichten von debian.org
    if ( /^(Sender|Resent-From|Resent-Sender): .*debian.org/:h )
    {
        MAILBOX="service.debian.org"
    }
    # Standard-Mailbox für Nachrichten von bugs.debian.org (BTS)
    if ( /^(Sender|Resent-From|Resent-sender): .*@bugs.debian.org/:h )
    {
        MAILBOX="bugs.debian.org"
    }
    # Mailbox für jede korrekt betreute Mailingliste mit "List-Id: foo"    # oder "List-Id: ←
    ...<foo.bar>"
    if ( /^List-Id: ([^<]*<)?([^>]*)>?/:h )
    {
        MAILBOX="$MATCH2"
    }
}
```



```
}
else
{
    # Regeln für Nachrichten, die nicht von Mailinglisten kommen
    # Standard-Posteingangs-Mailbox
    MAILBOX="Inbox-unusual"
    # lokale Nachrichten
    if ( /Envelope-to: .*@$MAILHOST/:h )
    {
        MAILBOX="Inbox-local"
    }
    # html-Nachrichten (zu 99% Spam)
    if ( /DOCTYPE html/:b ||\
        /^Content-Type: text\/html/ )
    {
        MAILBOX="Inbox-html"
    }
    # Blacklist-Regel für Spam-Nachrichten
    if ( /^X-Advertisement/:h ||\
        /^Subject:.*BUSINESS PROPOSAL/:h ||\
        /^Subject:.*URGENT.*ASISSTANCE/:h ||\
        /^Subject: *I NEED YOUR ASSISTANCE/:h )
    {
        MAILBOX="Inbox-trash"
    }
    # Whitelist-Regel für normale Nachrichten
    if ( /^From: .*@debian.org/:h ||\
        /^(Sender|Resent-From|Resent-Sender): .*debian.org/:h ||\
        /^Subject: .*(debian|bug|PATCH)/:h )
    {
        MAILBOX="Inbox"
    }
    # Whitelist-Regel für Nachrichten vom BTS
    if ( /^Subject: .*Bug#.*/:h ||\
        /^(To|Cc): .*@bugs.debian.org/:h )
    {
        MAILBOX="bugs.debian.org"
    }
    # Whitelist-Regel für getmails cron-Nachrichten
    if ( /^Subject: Cron .*getmails/:h )
    {
        MAILBOX="Inbox-getmails"
    }
}

# Auf Existenz von $MAILBOX prüfen
'test -d $MAILROOT/$MAILBOX'
if ( $RETURNCODE == 1 )
{
    # maildir-Mailbox für $MAILBOX erzeugen
    'maildirmake $MAILROOT/$MAILBOX'
}
# An maildir-$MAILBOX zustellen
to "$MAILROOT/$MAILBOX/"
exit
```

**Warnung**

Anders als procmail erzeugt maildrop fehlende maildir-Verzeichnisse nicht automatisch. Sie müssen sie vorab händisch mit maildirmake(1) erstellen, wie im obigen Beispiel von "\$HOME/.mailfilter".

6.6.2 procmail-Konfiguration

Hier eine ähnliche Konfiguration mit "\$HOME/.procmailrc" für procmail(1):

```
MAILDIR=$HOME/Maildir
DEFAULT=$MAILDIR/Inbox/
LOGFILE=$MAILDIR/Maillog
# Eindeutig unerwünschte Mails: in X-trash entsorgen und beenden
:0
* 1^0 ^X-Advertisement
* 1^0 ^Subject:.*BUSINESS PROPOSAL
* 1^0 ^Subject:.*URGENT.*ASISSTANCE
* 1^0 ^Subject: *I NEED YOUR ASSISTANCE
X-trash/

# Zustellen von Mailinglisten-Nachrichten
:0
* 1^0 ^Precedence:.*list
* 1^0 ^Precedence:.*bulk
* 1^0 ^List-
* 1^0 ^X-Distribution:.*bulk
{
:0
* 1^0 ^Return-path:.*debian-devel-admin@lists.debian.org.jp
jp-debian-devel/

:0
* ^Resent-Sender.*debian-user-request@lists.debian.org
debian-user/

:0
* ^Resent-Sender.*debian-devel-request@lists.debian.org
debian-devel/

:0
* ^Resent-Sender.*debian-announce-request@lists.debian.org
debian-announce

:0
mailing-list/
}

:0
Inbox/
```

6.6.3 mbox-Inhalte erneut zustellen

Wenn Ihr Heimatverzeichnis voll war und die Zustellung durch procmail(1) fehlgeschlagen ist, müssen Sie die Mails aus "/var/mail/" manuell in die einzelnen Mailboxen in Ihrem Heimatverzeichnis zustellen. Führen Sie folgendes aus, nachdem Sie in Ihrem Heimatverzeichnis Platz geschaffen haben:

```
# /etc/init.d/${MAILDAEMON} stop
```

```
# formail -s procmail </var/mail/<benutzername>
# /etc/init.d/${MAILDAEMON} start
```

6.7 POP3-/IMAP4-Server

Wenn Sie einen privaten Server im Netzwerk betreiben, sollten Sie erwägen, einen [POP3-/IMAP4](#)-Server laufen zu lassen, um Mails an Clients im Netzwerk zuzustellen.

Paket	Popcon	Größe	Art	Beschreibung
courier-pop	V:2, I:2	308	POP3	Courier Mail-Server - POP3-Server (nur Maildir-Format)
cyrus-pop3d	V:0, I:0	160	POP3	Cyrus Mail-System (POP3-Unterstützung)
courier-imap	V:3, I:4	589	IMAP	Courier Mail-Server - IMAP-Server (nur Maildir-Format)
cyrus-imapd	V:1, I:1	484	IMAP	Cyrus Mail-System (IMAP-Unterstützung)

Tabelle 6.11: Liste von POP3-/IMAP4-Servern

6.8 Der Print-Server und Hilfsprogramme

Im alten Unix-ähnlichen System war der BSD [Line Printer Daemon](#) Standard. Da das Standard-Druckausgabe-Format von freier Software auf Unix-artigen Systemen PostScript ist, wurde ein Filtersystem zusammen mit [Ghostscript](#) verwendet, um das Drucken auf nicht-PostScript-Druckern zu ermöglichen.

Mittlerweile ist [Common UNIX Printing System](#) (CUPS) der neue De-Facto-Standard. CUPS nutzt das [Internet Printing Protocol](#) (IPP). IPP wird jetzt auch von anderen Betriebssystemen wie Windows XP und Mac OS X unterstützt und ist der plattformübergreifende De-Facto-Standard für das Drucken von fern; es unterstützt bi-direktionale Kommunikation.

Das Standard-Format für druckbare Daten in Anwendungen auf dem Debian-System ist [PostScript \(PS\)](#), was eine Seitenbeschreibungssprache (Page Description Language) ist. Die Daten im PS-Format werden in den Ghostscript-PostScript-Interpreter eingespeist, um druckbare Daten spezifisch für den jeweiligen Drucker zu erzeugen. Lesen Sie dazu Abschnitt [11.4.1](#).

Dank der Dateiformat-abhängigen automatischen Konvertierungsfunktion des CUPS-Systems sollte die einfache Übergabe jeglicher Daten an den `lpr`-Befehl zur gewünschten Druckausgabe führen. (In CUPS kann `lpr` aktiviert werden, indem das `cups-bsd`-Paket installiert wird.)

Das Debian-System enthält einige erwähnenswerte Pakete für Print-Server und deren Hilfsprogramme:

Paket	Popcon	Größe	Port	Beschreibung
lpr	V:3, I:4	362	printer (515)	BSD <code>lpr/lpd</code> (Line Printer Daemon)
lprng	V:1, I:1	3064	”	” (Erweitert)
cups	V:140, I:395	1141	IPP (631)	Internet-Printing CUPS-Server
cups-client	V:56, I:454	493	”	System-V-Druckerbefehle für CUPS: <code>lp(1)</code> , <code>lpstat(1)</code> , <code>lpoptions(1)</code> , <code>cancel(1)</code> , <code>lpmove(8)</code> , <code>lpinfo(8)</code> , <code>lpadmin(8)</code> , ...
cups-bsd	V:36, I:385	122	”	BSD-Druckbefehle für CUPS: <code>lpr(1)</code> , <code>lpq(1)</code> , <code>lprm(1)</code> , <code>lpc(8)</code>
printer-driver-gutenprint	V:100, I:372	937	Nicht anwendbar	Druckertreiber für CUPS

Tabelle 6.12: Liste von Print-Servern und Hilfsprogrammen

Tipp

Sie können das CUPS-System konfigurieren, indem Sie in Ihrem Browser "<http://localhost:631/>" eingeben.

6.9 Der Server für Fernzugriff (SSH) und Hilfsprogramme

Die **Secure SHell** (SSH) ist der **sichere** Weg für Verbindungen über das Internet. Eine freie Version von SSH namens **OpenSSH** ist in Debian über die `openssh-client`- und `openssh-server`-Pakete verfügbar.

Paket	Popcon	Größe	Werkzeug	Beschreibung
openssh-client	V:803, I:996	4298	ssh(1)	Secure-Shell-Client
openssh-server	V:690, I:834	1567	sshd(8)	Secure-Shell-Server
ssh-askpass-fullscreen	V:0, I:0	42	ssh-askpass-fullscreen(1)	Fragt den Benutzer nach einer Passphrase für ssh-add (GNOME2)
ssh-askpass	V:3, I:34	106	ssh-askpass(1)	Fragt den Benutzer nach einer Passphrase für ssh-add (reines X)

Tabelle 6.13: Liste von Servern für Fernzugriff und Hilfsprogrammen

**Achtung**

Lesen Sie Abschnitt [4.7.3](#), falls Ihr SSH über das Internet erreichbar ist.

Tipp

Bitte nutzen Sie das Programm `screen(1)`, um dem Remote-Shell-Prozess die Chance zu geben, Verbindungsunterbrechungen zu überstehen (Weiteres dazu finden Sie in Abschnitt [9.1](#)).

6.9.1 Grundlagen von SSH

**Warnung**

Wenn Sie den OpenSSH-Server laufen lassen möchten, darf `/etc/ssh/sshd_not_to_be_run` nicht vorhanden sein.

SSH hat zwei Authentifizierungsprotokolle:

**Achtung**

Seien Sie vorsichtig bezüglich dieser Unterschiede, wenn Sie kein Debian-System verwenden.

Lesen Sie `/usr/share/doc/ssh/README.Debian.gz`, `ssh(1)`, `sshd(8)`, `ssh-agent(1)` und `ssh-keygen(1)` für weitere Details.

Hier einige wichtige Konfigurationsdateien:

SSH-Protokoll	SSH-Methode	Beschreibung
SSH-1	"RSAAuthentication"	Benutzerauthentifizierung basierend auf RSA-Identity-Keys
"	"RhostsAuthentication"	Rechnerauthentifizierung basierend auf ".rhosts" (unsicher, deaktiviert)
"	"RhostsRSAAuthentication"	Rechnerauthentifizierung basierend auf ".rhosts", kombiniert mit RSA-Host-Keys (deaktiviert)
"	"ChallengeResponseAuthentication"	RSA-Challenge-Response-Authentifizierung
"	"PasswordAuthentication"	Passwort-basierte Authentifizierung
SSH-2	"PubkeyAuthentication"	Benutzerauthentifizierung basierend auf Public Keys
"	"HostbasedAuthentication"	Rechnerauthentifizierung basierend auf "~/.rhosts" oder "/etc/hosts.equiv", kombiniert mit Public-Key-basierter Client-Host-Authentifizierung (deaktiviert)
"	"ChallengeResponseAuthentication"	Challenge-Response-Authentifizierung
"	"PasswordAuthentication"	Passwort-basierte Authentifizierung

Tabelle 6.14: Liste von SSH-Authentifizierungsprotokollen und -methoden

Konfigurationsdatei	Beschreibung
/etc/ssh/ssh_config	SSH-Client-Standardeinstellungen, lesen Sie ssh_config(5)
/etc/ssh/sshd_config	SSH-Server-Standardeinstellungen, lesen Sie sshd_config(5)
~/.ssh/authorized_keys	öffentliche Standard-SSH-Schlüssel, die Clients verwenden, um sich mit diesem Konto auf diesem SSH-Server zu verbinden
~/.ssh/identity	geheimer SSH-1 RSA-Schlüssel des Benutzers
~/.ssh/id_rsa	geheimer SSH-2 RSA-Schlüssel des Benutzers
~/.ssh/id_dsa	geheimer SSH-2 DSA-Schlüssel des Benutzers

Tabelle 6.15: Liste von SSH-Konfigurationsdateien

Tipp
Informationen, wie Sie öffentliche und geheime SSH-Schlüssel verwenden, finden Sie in `ssh-keygen(1)`, `ssh-add(1)` und `ssh-agent(1)`.

Tipp
Stellen Sie sicher, dass Sie die Einstellungen über einen Verbindungstest verifizieren. Bei jeglichen Problemen verwenden Sie `"ssh -v"`.

Tipp
Sie können die Passphrase später mit `"ssh-keygen -p"` ändern, um lokale geheime SSH-Schlüssel zu verschlüsseln.

Tipp
Sie können weitere Optionen zu den Einträgen in `"~/.ssh/authorized_keys"` hinzufügen, um die erlaubten Rechner einzuschränken und spezifische Befehle ausführen. Lesen Sie dazu `sshd(8)` bezüglich weiterer Details.

Mit folgenden Befehlen starten Sie eine `ssh(1)`-Verbindung von einem Client:

Befehl	Beschreibung
<code>ssh benutzername@rechnername.domaene.ext</code>	im Standardmodus verbinden
<code>ssh -v benutzername@rechnername.domaene.ext</code>	im Standardmodus verbinden mit aktivierten Debugging-Meldungen (zur Fehlersuche)
<code>ssh -1 benutzername@rechnername.domaene.ext</code>	erzwingen einer Verbindung mit SSH Version 1
<code>ssh -1 -o RSAAuthentication=no -l benutzername rechnername.domaene.ext</code>	Verwendung des Passworts erzwingen mit SSH Version 1
<code>ssh -o PreferredAuthentications=password -l benutzername rechnername.domaene.ext</code>	Verwendung des Passworts erzwingen mit SSH Version 2

Tabelle 6.16: Liste von Beispielen zum Start einer SSH-Verbindung auf einem Client

Wenn Sie den gleichen Benutzernamen auf dem lokalen und dem fernen Rechner nutzen, können Sie die Angabe von `"benutzername@"` weglassen. Sogar wenn Sie unterschiedliche Benutzernamen nutzen, können Sie sie weglassen, sofern Sie `"~/.ssh/config"` verwenden. Für den [Debian Salsa-Service](#) mit dem Kontoname `"foo-guest"` muss `"~/.ssh/config"` z.B. folgendes enthalten:

```
Host salsa.debian.org people.debian.org
  User foo-guest
```

Für den Benutzer fungiert `ssh(1)` als clevere und sichere Alternative zu `telnet(1)`. Anders als der `telnet`-Befehl scheitert `ssh` nicht am `telnet`-Maskierungszeichen (escape character; Standardeinstellung STRG-`]`).

6.9.2 Portweiterleitung für SMTP-/POP3-Tunnelung

Um für eine Verbindung von Port 4025 auf `localhost` zu Port 25 auf `remote-server` (einem fernen Server) oder für eine Verbindung von Port 4110 auf `localhost` zu Port 110 auf `remote-server` via `ssh` eine Weiterleitung aufzubauen, führen Sie auf dem lokalen Rechner (`localhost`) folgendes aus:

```
# ssh -q -L 4025:remote-server:25 4110:remote-server:110 benutzername@remote-server
```

Dies ist ein sicherer Weg für Verbindungen zu SMTP-/POP3-Servern über das Internet. Setzen Sie in `/etc/ssh/sshd_config` auf dem fernen Server den Wert für `AllowTcpForwarding` auf `yes`.

6.9.3 Verbindungen ohne Passwörter für die ferne Seite

Man kann es vermeiden, sich Passwörter für ferne Systeme merken zu müssen, indem man `RSAAuthentication` (SSH-1-Protokoll) oder `PubkeyAuthentication` (SSH-2-Protokoll) nutzt.

Setzen Sie dazu auf dem fernen System die entsprechenden Einträge `RSAAuthentication yes` bzw. `PubkeyAuthentication yes` in `/etc/ssh/sshd_config`.

Erzeugen Sie die Authentifizierungs-Schlüssel lokal und installieren Sie sie wie folgt auf dem fernen System:

- `RSAAuthentication`: RSA-Schlüssel für SSH-1 (nicht mehr empfohlen, da veraltet)

```
$ ssh-keygen
$ cat .ssh/identity.pub | ssh benutzer1@ferner-server "cat - >>.ssh/authorized_keys"
```

- `PubkeyAuthentication`: RSA-Schlüssel für SSH-2

```
$ ssh-keygen -t rsa
$ cat .ssh/id_rsa.pub | ssh benutzer1@ferner-server "cat - >>.ssh/authorized_keys"
```

- `PubkeyAuthentication`: DSA-Schlüssel für SSH-2 (nicht mehr empfohlen, da zu langsam)

```
$ ssh-keygen -t dsa
$ cat .ssh/id_dsa.pub | ssh benutzer1@ferner-server "cat - >>.ssh/authorized_keys"
```

Tipp

Die Verwendung von DSA-Schlüsseln für SSH-2 wird nicht mehr empfohlen, da der Schlüssel kürzer ist und langsame Rechenoperationen beschert. Es gibt keine Gründe mehr, auf eine Verschlüsselung nach dem RSA-Verfahren zu verzichten und DSA zu nutzen, da das RSA-Patent abgelaufen ist. DSA steht für [Digital Signature Algorithm](#) und ist langsam. Lesen Sie auch [DSA-1571-1](#).

Anmerkung

Damit `HostbasedAuthentication` in SSH-2 funktioniert, müssen Sie den Wert von `HostbasedAuthentication` in `yes` ändern, um zwar sowohl in `/etc/ssh/sshd_config` auf dem Server wie auch in `/etc/ssh/ssh_config` oder `~/.ssh/config` auf dem Client.

6.9.4 Der Umgang mit fremden SSH-Clients

Es sind einige freie [SSH](#)-Clients für andere Plattformen verfügbar:

Umgebung	freies SSH-Programm
Windows	puTTY (http://www.chiark.greenend.org.uk/~sgtatham/putty/) (GPL)
Windows (Cygwin)	SSH in Cygwin (http://www.cygwin.com/) (GPL)
Macintosh Classic	macSSH (http://www.macssh.com/) (GPL)
Mac OS X	OpenSSH; verwenden Sie SSH in der Terminal-Applikation (GPL)

Tabelle 6.17: Liste freier SSH-Clients für andere Plattformen

6.9.5 Einrichten von ssh-agent

Es ist sicherer, Ihren geheimen SSH-Authentifizierungs-Schlüssel mit einer Passphrase zu schützen. Falls Sie noch keine Passphrase vergeben haben, verwenden Sie dazu `ssh-keygen -p`.

Legen Sie Ihren öffentlichen SSH-Schlüssel (z.B. `~/ .ssh/id_rsa.pub`) wie folgt in `~/ .ssh/authorized_keys` auf dem fernen Server ab, und zwar über eine passwort-basierte Verbindung zum Server, wie oben beschrieben.

```
$ ssh-agent bash
$ ssh-add ~/.ssh/id_rsa
Enter passphrase for /home/<benutzername>/.ssh/id_rsa:
Identity added: /home/<benutzername>/.ssh/id_rsa (/home/<benutzername>/.ssh/id_rsa)
```

Es ist kein Passwort für das ferne System zur Ausführung des nächsten Befehls mehr nötig:

```
$ scp foo <benutzername>@ferner-server.host:foo
```

Drücken Sie Strg-D, um die ssh-agent-Sitzung zu beenden.

Für den X-Server führt das normale Debian-Start-Skript `ssh-agent` als Eltern-Prozess aus. Sie müssen `ssh-add` daher nur einmal ausführen. Für weitere Details lesen Sie `ssh-agent(1)` und `ssh-add(1)`.

6.9.6 Wie Sie das ferne System über SSH herunterfahren

Sie müssen den Prozess, der das `shutdown -h now` ausführt (lesen Sie Abschnitt 1.1.8), davor schützen, durch SSH beendet zu werden. Verwenden Sie dazu wie folgt den `at(1)`-Befehl (weiteres zu `at` in Abschnitt 9.3.13):

```
# echo "shutdown -h now" | at now
```

`shutdown -h now` in einer `screen(1)`-Sitzung auszuführen (weitere Infos in Abschnitt 9.1) ist ein anderer möglicher Weg.

6.9.7 Fehlersuche bei SSH

Falls Sie Probleme haben, kontrollieren Sie die Zugriffsrechte der Konfigurationsdateien und starten Sie SSH mit der Option `-v`.

Nutzen Sie die Option `-p`, wenn Sie root sind und Probleme mit einer Firewall haben; dadurch wird die Verwendung der Ports 1 - 1023 auf dem Server vermieden.

Wenn SSH-Verbindungen zu einem fernen Rechner plötzlich nicht mehr funktionieren, könnte dies auf Spielereien des Systemadministrators zurückzuführen sein, höchstwahrscheinlich Änderungen am `host_key` im Rahmen von Systemwartungsarbeiten. Nachdem Sie sich versichert haben, dass dies der Fall ist und niemand versucht, den fernen Rechner über einen cleveren Hack zu imitieren, können Sie die Verbindung wiedererlangen, indem Sie den `host_key`-Eintrag in der `~/ .ssh/known_hosts`-Datei auf dem lokalen Rechner entfernen.

Paket	Popcon	Größe	Protokoll	Beschreibung
telnetd	V:1, I:3	115	TELNET	TELNET-Server
telnetd-ssl	V:0, I:0	170	”	” (mit SSL-Unterstützung)
nfs-kernel-server	V:38, I:79	342	NFS	Datei-Netzwerkfreigabe unter Unix
samba	V:102, I:159	16629	SMB	Datei- und Drucker-Netzwerkfreigabe unter Windows
netatalk	V:2, I:3	2077	ATP	Datei- und Drucker-Netzwerkfreigabe unter Apple/Mac (AppleTalk)
proftpd-basic	V:24, I:32	488	FTP	Grundlegender Datei-Download
apache2	V:246, I:315	610	HTTP	Grundlegender Web-Server
squid	V:13, I:15	8385	”	Grundlegender Web-Proxy-Server
squid3	V:4, I:10	240	”	”
bind9	V:52, I:65	1063	DNS	IP-Adresse für andere Rechner
isc-dhcp-server	V:18, I:54	1471	DHCP	IP-Adresse des Clients selbst

Tabelle 6.18: Liste von weiteren Netzwerkanwendungs-Servern

6.10 Weitere Netzwerkanwendungs-Server

Hier einige weitere Netzwerkanwendungs-Server:

Das Common Internet File System - Protokoll (CIFS) ist das gleiche Protokoll wie [Server Message Block \(SMB\)](#) und wird von Microsoft Windows ausgiebig genutzt.

Tipp

In Abschnitt [4.5.2](#) finden Sie Informationen zur Integration von Server-Systemen.

Tipp

Die Rechnernamenauflösung wird normalerweise über den [DNS](#)-Server realisiert. Für IP-Adressen, die dynamisch über einen [DHCP](#)-Server zugewiesen werden, kann [Dynamic DNS](#) eingerichtet werden; dazu können [bind9](#) und [isc-dhcp-server](#) genutzt werden, wie auf der [DDNS-Seite im Debian-Wiki](#) beschrieben.

Tipp

Die Verwendung eines Proxy-Servers wie [squid](#) ist viel effizienter, um Bandbreite zu sparen, als ein lokaler Archiv-Spiegel mit dem vollständigen Inhalt des Debian-Archivs.

6.11 Weitere Netzwerkanwendungs-Clients

Hier einige weitere Netzwerkanwendungs-Clients:

6.12 Diagnose von System-Daemons

Das `telnet`-Programm ermöglicht die manuelle Verbindung zu den System-Daemons und ihren Diagnosefunktionen.

Um einen reinen [POP3](#)-Dienst zu testen, probieren Sie folgendes:

```
$ telnet mail.ispname.net pop3
```

Paket	Popcon	Größe	Protokoll	Beschreibung
netcat	I:41	16	TCP/IP	TCP/IP-Alleskönner
openssl	V:794, I:993	1465	SSL	Secure-Socket-Layer-(SSL-)Binärdatei und dazugehörige Kryptografie-Werkzeuge
stunnel4	V:5, I:17	507	"	universeller SSL-Wrapper
telnet	V:65, I:904	163	TELNET	TELNET-Client
telnet-ssl	V:0, I:3	210	"	" (mit SSL-Unterstützung)
nfs-common	V:181, I:343	768	NFS	Datei-Netzwerkfreigabe unter Unix
smbclient	V:16, I:174	2016	SMB	Datei- und Druckerfreigabe-Client für MS Windows
cifs-utils	V:32, I:123	299	"	Befehle zum Einbinden und Trennen von fern abgelegten MS-Windows-Dateien
ftp	V:18, I:282	137	FTP	FTP-Client
lftp	V:6, I:39	2255	"	"
ncftp	V:3, I:22	1339	"	FTP-Client mit Vollbildschirmdarstellung
wget	V:288, I:988	3477	HTTP und FTP	Programm zum Herunterladen von Dateien aus dem Web
curl	V:151, I:548	426	"	"
axel	V:0, I:4	216	"	Download-Beschleuniger
aria2	V:2, I:19	1854	"	Download-Beschleuniger mit Unterstützung für BitTorrent und Metalink
bind9-host	V:382, I:948	365	DNS	host(1) von bind9, "Priorität: standard"
dnsutils	V:64, I:517	256	"	dig(1) von bind, "Priorität: standard"
isc-dhcp-client	V:231, I:979	686	DHCP	IP-Adresse beziehen
ldap-utils	V:14, I:75	718	LDAP	Daten von einem LDAP-Server beziehen

Tabelle 6.19: Liste von Netzerkanwendungs-Clients

Um einen [POP3](#)-Dienst mit aktiviertem [TLS/SSL](#) (wie sie bei manchen Providern vorkommen) zu testen, benötigen Sie einen [telnet](#)-Client mit aktiviertem TLS/SSL, z.B. aus dem [telnet-ssl](#)- oder [openssl](#)-Paket:

```
$ telnet -z ssl pop.gmail.com 995
```

```
$ openssl s_client -connect pop.gmail.com:995
```

Die folgenden [RFCs](#) enthalten das zur Diagnose erforderliche Wissen für jeden System-Daemon.

RFC	Beschreibung
rfc1939 und rfc2449	POP3 -Dienst
rfc3501	IMAP4 -Dienst
rfc2821 (rfc821)	SMTP -Dienst
rfc2822 (rfc822)	Mail-Datei-Format
rfc2045	Multipurpose Internet Mail Extensions (MIME)
rfc819	DNS -Dienst
rfc2616	HTTP -Dienst
rfc2396	URI -Definition

Tabelle 6.20: Liste populärer RFCs

Die Verwendung der Ports ist in `"/etc/services"` beschrieben.

Kapitel 7

Das X-Window-System

**Warnung**

Dieses Kapitel ist überholt, da es auf dem in 2013 veröffentlichten Debian 7.0 (Wheezy) basiert.

Das [X-Window-System](#) innerhalb von Debian basiert auf dem Quellcode von [X.Org](#).

7.1 Wichtige Pakete

Es existieren einige (Meta-)Pakete, um eine einfache Installation zu ermöglichen:

Grundwissen über X finden Sie in [X\(7\)](#) und im [LDP X-Window User-HOWTO](#).

7.2 Eine Arbeitsplatz-Umgebung einrichten

Eine [Arbeitsplatz-Umgebung](#) (desktop environment) ist gewöhnlich eine Kombination aus einem [X-Fenstermanager](#), einem Dateimanager und einer Sammlung verschiedener kompatibler Werkzeuge.

Sie können eine vollständige [Arbeitsplatz-Umgebung](#) wie [GNOME](#), [KDE](#), [Xfce](#) oder [LXDE](#) über das Tasks-Menü von `aptitude` installieren.

Tipp

In Debian `unstable/testing` kann es vorkommen, dass das Tasks-Menü nicht synchron mit dem aktuellen Status von Paketübergängen ist. In solchen Situationen müssen Sie einige (Meta-)Pakete, die im Tasks-Menü von `aptitude(8)` aufgelistet sind, abwählen, um Paketkonflikte zu vermeiden. Dabei müssen allerdings bestimmte Pakete wieder händisch markiert werden, die nötige Abhängigkeiten bereitstellen, um zu vermeiden, dass sie automatisch entfernt werden.

Alternativ ist es auch möglich, eine einfache Umgebung mit lediglich einem [X-Fenstermanager](#) (wie [Fluxbox](#)) manuell einzurichten.

Unter [Window Managers for X](#) finden Sie eine Übersicht über X-Fenstermanager und Arbeitsplatz-Umgebungen.

(Meta-)Paket	Popcon	Größe	Beschreibung
xorg	I:457	52	X-Bibliotheken, ein X-Server, eine Sammlung von Schriftarten sowie eine Gruppe von grundlegenden X-Clients und Hilfsprogrammen (Metapaket)
xserver-xorg	V:66, I:492	238	vollständige Software-Sammlung des X-Servers und seiner Konfiguration
xbase-clients	I:26	46	verschiedene X-Clients (Metapaket)
x11-common	V:372, I:755	308	Dateisystem-Infrastruktur des X-Window-Systems
xorg-docs	I:6	2036	verschiedene Dokumentationen für die X.Org-Software-Sammlung
menu	V:54, I:197	1509	Erzeugen des Debian-Menüs für alle am Menüsystem beteiligten Anwendungen
menu-xdg	V:31, I:109	27	die Debian-Menü-Struktur in die xdg-Menü-Struktur von freedesktop.org konvertieren
xdg-utils	V:229, I:521	327	Werkzeuge, um die von freedesktop.org bereitgestellte Arbeitsplatz-Umgebung in das System zu integrieren
task-gnome-desktop	I:175	9	Standard-Arbeitsplatz-Umgebung GNOME (Metapaket)
task-kde-desktop	I:66	6	Kern der KDE -Arbeitsplatz-Umgebung (Metapaket)
task-xfce-desktop	I:106	9	ressourcen-schonende Xfce -Arbeitsplatz-Umgebung (Metapaket)
task-lxde-desktop	I:35	9	ressourcen-schonende LXDE -Arbeitsplatz-Umgebung (Metapaket)
fluxbox	V:2, I:9	3860	Fluxbox : Paket für den in hohem Maße konfigurierbaren und ressourcen-schonenden X-Fenstermanager

Tabelle 7.1: Liste von (Meta-)Paketen für X-Window

7.2.1 Debian-Menü

Das [Debian-Menüsystem](#) bietet mit `update-menus(1)` aus dem `menu`-Paket eine grundlegende Schnittstelle sowohl für textbasierte wie auch für grafische Programme. Jedes Paket installiert seine Menüdaten in das Verzeichnis `"/usr/share/menu/"`. Lesen Sie dazu `"/usr/share/menu/README"`.

7.2.2 Freedesktop.org-Menü

Jedes Paket, dass mit [Freedesktop.org's](#) xdg-Menüsystem konform ist, installiert seine Menüdaten aus `"*.desktop"` in `"/usr/share."` Moderne Arbeitsplatz-Umgebungen, die mit dem [Freedesktop.org](#)-Standard kompatibel sind, nutzen diese Daten, um mittels dem Paket `xdg-utils` ihre Menüs zu generieren. Weiteres dazu finden Sie in `"/usr/share/doc/xdg-utils/README"`.

7.2.3 Das Debian-Menü erzeugt aus dem Freedesktop.org-Menü

Um aus einer zum [Freedesktop.org-Menü](#) kompatiblen Fenstermanager-Umgebung (wie [GNOME](#) und [KDE](#)) auf das traditionelle Debian-Menü zugreifen zu können, müssen Sie das Paket `menu-xdg` installieren.

7.3 Der Zusammenhang zwischen Server und Client

Das X-Window-System wird als Kombination aus Server- und Client-Programmen aktiviert. Die Bedeutung von **Server** und **Client** vor dem Hintergrund der Begriffe **lokal** und **fern** erfordert hier eine genauere Betrachtung:

Moderne X-Server enthalten die [MIT Shared Memory Extension](#) und kommunizieren über gemeinsam genutzten Speicher (shared memory) mit den lokalen X-Clients. So wird der netzwerk-transparente Xlib-Interprozess-Kommunikationskanal umgangen und die Performance für große Anzeigen erhöht.

Art	Beschreibung
X-Server	ein Programm, das auf dem lokalen Rechner läuft; ist mit Bildschirm und Eingabegeräten des Benutzers verbunden
X-Client	ein Programm, das auf einem fernen Rechner läuft; verarbeitet Daten und kommuniziert mit dem X-Server
Anwendungs-Server	ein Programm, das auf einem fernen Rechner läuft; verarbeitet Daten und kommuniziert mit den Anwendungs-Clients
Anwendungs-Client	ein Programm, das auf dem lokalen Rechner läuft; ist mit Bildschirm und Eingabegeräten des Benutzers verbunden

Tabelle 7.2: Liste der Server-/Client-Terminologie

7.4 Der X-Server

Unter `xorg(1)` finden Sie detaillierte Informationen über den X-Server.

7.4.1 Die (Neu-)Konfiguration des X-Servers

Folgendes konfiguriert den X-Server (neu):

```
# dpkg-reconfigure --priority=low x11-common
```

Anmerkung

Aktuelle Linux-Kernel enthalten gute Unterstützung für Grafikkarten und Eingabegeräte unter Nutzung von [DRM](#), [KMS](#) und [udev](#). Der X-Server wurde angepasst, um diese zu verwenden. Daher ist `/etc/X11/xorg.conf` üblicherweise auf Ihrem System auch nicht mehr vorhanden. Diese Parameter werden durch den Kernel konfiguriert. Lesen Sie dazu `fb/modeedb.txt` in der Linux-Kernel-Dokumentation.

Bei CRT-Monitoren (Röhren-Monitoren) mit hoher Auflösung ist es eine gute Idee, die Bildwiederholfrequenz (refresh rate) so hoch zu setzen, wie der Monitor es erlaubt (85 Hz wäre toll, 75 Hz ist noch in Ordnung), um das Flackern zu reduzieren. Bei Flachbildschirmen ist aufgrund ihrer langsameren Reaktionszeiten eine niedrigere Bildwiederholfrequenz (60 Hz) gewöhnlich kein Problem.

Anmerkung

Achten Sie darauf, keine zu hohe Bildwiederholfrequenz einzustellen, da dies zu fatalen Schäden an der Hardware Ihres Monitors führen kann.

7.4.2 Die Verbindungsmethoden zum X-Server

Es gibt verschiedene Wege für den "X-Server" (die **Display**-Seite), Verbindungen von einem "X-Client" (der **Anwendungs**-Seite) zu akzeptieren:



Warnung

Nutzen Sie für eine X-Verbindung keine [TCP/IP](#)-Verbindung über ein **unsicheres** Netzwerk, außer Sie haben einen sehr guten Grund dafür wie z.B. dass Sie Verschlüsselung verwenden. Eine TCP/IP-Socket-Verbindung ohne Verschlüsselung ist anfällig für die **Eavesdropping-Attacke** und auf Debian-Systemen standardmäßig deaktiviert. Verwenden Sie `ssh -X`.

Paket	Popcon	Größe	Benutzer	Verschlüsselung	Methode	geeignete Verwendung
xbase-clients	I:26	46	nicht geprüft	nein	xhost-Befehl	veraltet
xbase-clients	I:26	46	geprüft	nein	xauth-Befehl	lokale Verbindung via Pipe (Weiterleitung)
openssh-client	V:803, I:996	4298	geprüft	ja	ssh -X-Befehl	Netzwerkverbindung von fern
gdm3	V:165, I:229	5101	geprüft	nein (XDMCP)	GNOME-Displaymanager	lokale Verbindung via Pipe (Weiterleitung)
sddm	V:53, I:95	1742	geprüft	nein (XDMCP)	KDE-Displaymanager	lokale Verbindung via Pipe (Weiterleitung)
xdm	V:3, I:6	686	geprüft	nein (XDMCP)	X-Displaymanager	lokale Verbindung via Pipe (Weiterleitung)
wdm	V:31, I:284	2289	geprüft	nein (XDMCP)	WindowMaker-Displaymanager	lokale Verbindung via Pipe (Weiterleitung)
ldm	V:0, I:0	436	geprüft	ja	LTSP-Displaymanager	SSH-Netzwerkverbindung von fern (Thin-Client)

Tabelle 7.3: Liste der Verbindungsmethoden zum X-Server

**Warnung**

Verwenden Sie keine [XDMCP-Verbindung](#) über ein **unsicheres** Netzwerk. Es werden dabei Daten via [UDP/IP](#) ohne Verschlüsselung übertragen und diese Verbindung ist anfällig für die **Eavesdropping-Attacke**.

Tipp

LTSP steht für [Linux Terminal Server Project](#).

7.5 Starten des X-Window-Systems

Das X-Window-System wird normalerweise als [X-Sitzung](#) gestartet, was eine Kombination aus einem X-Server und angebotenen X-Clients ist. Auf einem normalen Arbeitsplatz-System werden beide auf dem gleichen Rechner ausgeführt.

Die [X-Sitzung](#) wird durch eine der folgenden Bedingungen gestartet:

- über den `startx`-Befehl, von der Befehlszeile gestartet;
- über einen der [X-Displaymanager](#)-Daemons (*dm), gestartet von `systemd` in Abhängigkeit von `"graphical.target"`.

Tipp

Das Start-Skript für die Displaymanager-Daemons prüft den Inhalt der Datei `"/etc/X11/default-display-manager"`, bevor es sich letztendlich selbst ausführt. So ist sichergestellt, dass nur ein [X-Displaymanager](#) aktiviert wird.

Tipp

Lesen Sie Abschnitt [8.4.5](#) für Informationen über anfängliche Werte von Umgebungsvariablen des X-Displaymanagers.

Letztlich führen all diese Programme das Skript `"/etc/X11/Xsession"` aus. Dann werden von diesem Skript run-parts(8)-ähnliche Aktionen durchgeführt, um Skripte im Verzeichnis `"/etc/X11/Xsession.d/"` auszuführen. Im Wesentlichen wird dabei das erste Programm ausgeführt, dass vom integrierten `exec`-Befehl gefunden wird; bei dieser Suche wird in folgender Reihenfolge vorgegangen:

1. das Skript, das als Argument von `"/etc/X11/Xsession"` durch den X-Displaymanager festgelegt wurde, falls definiert;
2. das `"~/ .xsession"`- oder `"~/ .Xsession"`-Skript, falls definiert;
3. der `"/usr/bin/x-session-manager"`-Befehl, falls definiert;
4. der `"/usr/bin/x-window-manager"`-Befehl, falls definiert;
5. der `"/usr/bin/x-terminal-emulator"`-Befehl, falls definiert.

Dieser Prozess wird beeinflusst durch den Inhalt von `"/etc/X11/Xsession.options"`. Die eigentlichen Programme, auf die diese `"/usr/bin/x-"`-Befehle verweisen, werden durch Debians alternatives-System festgelegt und durch Kommandos wie `"update-alternatives --config x-session-manager"` konfiguriert.

Lesen Sie `Xsession(5)` bezüglich weiterer Details.

7.5.1 Starten einer X-Sitzung mit gdm3

`gdm3(1)` gibt Ihnen die Möglichkeit, über das Menü den Sitzungstyp (bzw. die Arbeitsplatz-Umgebung; lesen Sie Abschnitt 7.2) und die Sprache (bzw. das Gebietsschema (Locale); Näheres in Abschnitt 8.4) der X-Sitzung festzulegen. Die ausgewählten Standardwerte werden wie folgt in `"~/ .dmrc"` gespeichert:

```
[Desktop]
Session=default
Language=de_DE.UTF-8
```

7.5.2 Anpassen der X-Sitzung (klassische Methode)

Auf einem System, auf dem `"/etc/X11/Xsession.options"` eine Zeile `"allow-user-xsession"` ohne vorangestelltes `"#"`-Zeichen enthält, kann jeder Benutzer, der `"~/ .xsession"` oder `"~/ .Xsession"` anlegt und entsprechend beschreibt, die Auswirkungen von `"/etc/X11/Xsession"` anpassen und so die Systemvorgabe vollständig überschreiben. Der letzte Befehl in der Datei `"~/ .xsession"` sollte die Form `"exec some-window/session-manager"` verwenden, um den Fenster-/Sitzungsmanager Ihrer Wahl zu starten.

Bei Nutzung dieser Funktionalität wird die Auswahl des Display- (oder Login-) Managers (DM), des Sitzungs- oder Fenstermanagers (WM) durch die Systemwerkzeuge ignoriert.

7.5.3 Anpassen der X-Sitzung (neue Methode)

Hier ein paar neue Möglichkeiten, um die X-Sitzung anzupassen, ohne dabei die Systemvorgabe zu überschreiben:

- Der Displaymanager `gdm3` kann eine spezifische Sitzung vorwählen und dies als Argument von `"/etc/X11/Xsession"` festlegen.
 - Die Dateien `"/etc/profile"`, `"~/ .profile"`, `"/etc/xprofile"` und `"~/ .xprofile"` werden am Ende des `gdm3`-Startprozesses ausgeführt.
- Die Datei `"~/ .xsessionrc"` wird als Teil des Startprozesses ausgeführt (unabhängig von der Arbeitsplatz-Umgebung).
 - `"#allow-user-xsession"` in `"/etc/X11/Xsession.options"` verhindert nicht die Ausführung der Datei `"~/ .xsessionrc"`.
- Die Datei `"~/ .gnomerc"` wird als Teil des Startprozesses ausgeführt (nur GNOME-Umgebung).

Die Auswahl des Display- (oder Login-) Managers (DM), des Sitzungs- oder Fenstermanagers (WM) durch die Systemwerkzeuge wird respektiert.

Diese Konfigurationsdateien sollten weder `"exec ..."` noch `"exit"` enthalten.

7.5.4 Einen X-Client von fern via SSH verbinden

Die Verwendung von `ssh -X` aktiviert eine sichere Verbindung von einem lokalen X-Server zu einem fernen Anwendungs-Server.

Setzen Sie in `/etc/ssh/sshd_config` des fernen Hosts die `X11Forwarding`-Einträge auf `yes`, wenn Sie die Verwendung der Befehlszeilenoption `-X` vermeiden möchten.

Starten Sie den X-Server auf dem lokalen Rechner.

Öffnen Sie auf dem lokalen Rechner ein `xterm`.

Führen Sie `ssh(1)` wie folgt aus, um eine Verbindung mit der fernen Seite aufzubauen:

```
benutzername @ localhost $ ssh -q -X login-name@ferner-rechner.domaene
Password:
```

Starten Sie auf der fernen Seite einen X-Anwendungsbefehl, z.B. `gimp`:

```
login-name @ ferner-rechner $ gimp &
```

Mit dieser Methode kann die Ausgabe eines fernen X-Clients angezeigt werden, als wäre er lokal über einen UNIX-Domain-Socket verbunden.

7.5.5 Ein sicheres X-Terminal über das Internet

Ein sicheres X-Terminal über das Internet, das eine fern laufende vollständige X-Arbeitsplatz-Umgebung anzeigt, kann einfach über die Nutzung spezieller Pakete wie `ldm` realisiert werden. Ihr lokaler Rechner wird dann zu einem sicheren Thin-Client für den fernen, über SSH verbundenen Anwendungs-Server.

7.6 Schriften in X-Window

[Fontconfig 2.0](#) wurde 2002 als distributionsunabhängige Bibliothek zur Konfiguration und Anpassung von Schriften erstellt. Seit [Squeeze](#) nutzt Debian [Fontconfig 2.0](#) für seine Schriftenkonfiguration.

Die Schriftenunterstützung im X-Window-System kann wie folgt zusammengefasst werden:

- Althergebrachtes X-Server-seitiges System zur Schriftenunterstützung
 - Das originale X11-Schriftensystem bietet Rückwärtskompatibilität für ältere Versionen von X-Client-Anwendungen.
 - Die originalen X11-Schriften sind auf dem X-Server installiert.
- Modernes X-Client-seitiges System zur Schriftenunterstützung
 - Das moderne X-System unterstützt alle unten aufgeführten Schriften (Abschnitt [7.6.1](#), Abschnitt [7.6.2](#) und Abschnitt [7.6.3](#)) mit erweiterten Funktionen wie Anti-Aliasing (Kantenglättung).
 - [Xft 2.0](#) verbindet moderne X-Anwendungen wie solche aus [GNOME](#), [KDE](#) und [LibreOffice](#) mit der [FreeType-2.0](#)-Bibliothek.
 - [FreeType 2.0](#) stellt eine Bibliothek zur Rasterung (Rasterization) von Schriften bereit.
 - [Fontconfig](#) erfüllt die Schriften-Spezifikation für [Xft 2.0](#). Sie finden Infos über die Konfiguration in `fonts.conf(5)`.
 - Alle modernen X-Anwendungen, die [Xft 2.0](#) verwenden, können über die [X Rendering Extension](#) mit dem X-Server kommunizieren.
 - Durch die [X Rendering Extension](#) wird der Zugriff auf die Schriften sowie die Erzeugung von Glyph-Images vom X-Server auf den X-Client verschoben.

Informationen über die Schriften-Konfiguration können wie folgt abgefragt werden:

Paket	Popcon	Größe	Beschreibung
xfonts-utils	V:66, I:542	415	Hilfsprogramme für X-Window-System-Schriften
libxft2	V:143, I:662	122	Xft, eine Bibliothek, die X-Anwendungen mit der FreeType-Bibliothek zur Schriften-Rasterung verbindet
libfontconfig	V:426, I:994	896	FreeType -2.0-Bibliothek zur Schriften-Rasterung
fontconfig	V:354, I:776	583	Fontconfig , eine generische Schriften-Konfigurationsbibliothek - Binärdateien
fontconfig-config	V:367, I:871	442	Fontconfig , eine generische Schriften-Konfigurationsbibliothek - Konfigurationsdaten

Tabelle 7.4: Tabelle der Pakete, die die X-Window-Schriftensysteme unterstützen

- "xset q" liefert Informationen über den grundlegenden X11-Schriften-Pfad (font path);
- "fc-match" zeigt, welche fontconfig-Schriften Standardeinstellung sind;
- "fc-list" zeigt verfügbare fontconfig-Schriften.

Tipp

"[The Penguin and Unicode](#)" ist ein guter Überblick über das moderne X-Window-System. Weitere Dokumentation auf <http://unifont.org/> liefert gute Informationen über Unicode-Schriften, Unicode-taugliche Software, Internationalisierung und Unicode-bezogene Bedienbarkeitsprobleme auf [FLOSS \(Free/Libre Open Source\)](#)-Betriebssystemen.

7.6.1 Basis-Schriftarten

Es gibt zwei Haupttypen von [Computer-Schriftarten](#):

- Bitmap-Schriften (gut für Rasterung mit niedriger Auflösung);
- Outline/stroke-Schriften (gut für Rasterung mit hoher Auflösung).

Während das Skalieren von Bitmap-Schriften eine undeutliche Anzeige erzeugt, erhält man beim Skalieren von Outline/stroke-Schriften eine scharfe Darstellung.

Bitmap-Schriften werden im Debian-System gewöhnlich über Schriftdateien im komprimierten [pcf-Format](#) mit der Dateieindung ".pcf.gz" bereitgestellt.

Outline-Schriften liefert das Debian-System wie folgt aus:

- als [PostScript](#)-Typ1-Schriftdatei mit der Dateieindung ".pfb" (Binär-Schriftdatei) oder ".afm" (Font-Metrics-Datei);
- als [TrueType](#)- (oder [OpenType](#)-) Schriftdatei, gewöhnlich mit der Dateieindung ".ttf".

Tipp

[OpenType](#) soll voraussichtlich sowohl [TrueType](#) wie auch [PostScript](#) Typ 1 ablösen.

Tipp

[DejaVu](#)-Schriften basieren auf und sind die Obermenge der [Bitstream Vera](#)-Schriften.

Schrift-Paket	Popcon	Größe	Sans-Serif-Schrift	Serif-Schrift	Monospace-Schrift	Ursprung der Schrift
PostScript	Nicht verfügbar	Nicht verfügbar	Helvetica	Times	Courier	Adobe
gsfonts	I:599	4439	Nimbus Sans L	Nimbus Roman No9 L	Nimbus Mono L	URW (Adobe-kompatible Größe)
gsfonts-x11	I:82	95	Nimbus Sans L	Nimbus Roman No9 L	Nimbus Mono L	Unterstützung für X-Schriften mit PostScript-Typ1-Schriften
t1-cyrillic	I:19	4878	Free Helvetian	Free Times	Free Courier	URW extended (Adobe-kompatible Größe)
lmodern	V:13, I:113	33270	LMSans*	LMRoman*	LMTypewriter*	skalierbare PostScript- und OpenType-Schriften, basierend auf Computer Modern (von TeX)

Tabelle 7.5: Tabelle von sich entsprechenden [PostScript](#)-Typ1-Schriftarten

Schrift-Paket	Popcon	Größe	Sans-Serif-Schrift	Serif-Schrift	Monospace-Schrift	Ursprung der Schrift
ttf-mscorefonts-installer	V:1, I:64	92	Arial	Times New Roman	Courier New	Microsoft (Adobe-kompatible Größe) (hiermit werden unfreie Daten installiert)
fonts-liberation	I:469	2093	Liberation Sans	Liberation Serif	Liberation Mono	Liberation Fonts Project (Microsoft-kompatible Größe)
fonts-freefont-ttf	V:50, I:276	6656	FreeSans	FreeSerif	FreeMono	GNU Freefont (Microsoft-kompatible Größe)
fonts-dejavu	I:478	39	DejaVu Sans	DejaVu Serif	DejaVu Sans Mono	DejaVu, Bitstream Vera mit Unicode-Abdeckung
fonts-dejavu-core	V:220, I:809	2954	DejaVu Sans	DejaVu Serif	DejaVu Sans Mono	DejaVu, Bitstream Vera mit Unicode-Abdeckung (sans, sans-bold, serif, serif-bold, mono, mono-bold)
fonts-dejavu-extra	I:516	7493	Nicht verfügbar	Nicht verfügbar	Nicht verfügbar	DejaVu, Bitstream Vera mit Unicode-Abdeckung (oblique, italic, bold-oblique, bold-italic, condensed)
ttf-unifont	I:21	21	Nicht verfügbar	Nicht verfügbar	Unifont	GNU Unifont , mit allen druckbaren Zeichen-Codes aus Unicode 5.1 Basic Multilingual Plane (BMP)

Tabelle 7.6: Tabelle von sich entsprechenden [TrueType](#)-Schriftarten

7.6.2 Zusätzliche Schriftarten

aptitude(8) hilft Ihnen, zusätzliche Schriften einfach zu finden:

- über die Paketliste unter "Tasks" → "Lokalisierung";
- über eine gefilterte Liste der Schriftarten in der einfachen Paketansicht, mittels regulärem Ausdruck auf den Debtage: `"~Gmade-of: : d`
- über eine gefilterte Liste der BDF-(Bitmap-)Schriftarten-Pakete in der einfachen Paketansicht, mittels regulärem Ausdruck auf den Paketnamen: `"~nxfonts-";`
- über eine gefilterte Liste der TrueType-(Outline-)Schriftarten-Pakete in der einfachen Paketansicht, mittels regulärem Ausdruck auf den Paketnamen: `"~nttf- |~nfonts-".`

Da freie (**Free**-)Schriften manchmal eingeschränkt sind, ist das Installieren oder gemeinsame Verwenden einiger kommerzieller TrueType-Schriften eine Option für Debian-Benutzer. Um diesen Prozess für den Benutzer zu vereinfachen, wurden einige "Bequemlichkeitspakete" erstellt:

- `mathematica-fonts`
- `fonts-mscorefonts-installer`

Sie werden eine wirklich gute Auswahl von TrueType-Schriften bekommen zum Preis der Kontaminierung Ihres **freien** Systems mit nicht-freien Schriftarten.

7.6.3 CJK-Schriftarten

Hier ein paar Schlüsselinformationen zu Schriften mit [CJK-Zeichen](#).

Schrifttyp	Japanischer Schriftname	Chinesischer Schriftname	Koreanischer Schriftname
sans-serif	gothic, ゴシック	hei, gothic	dodum, gulim, gothic
serif	mincho, 明朝	song, ming	batang

Tabelle 7.7: Tabelle von in CJK-Schriftnamen verwendeten Schlüsselwörtern zur Erkennung von Schrifttypen

Ein Schriftname wie "VL PGothic" mit einem "P" weist auf eine proportionale Schriftart hin, die der "VL Gothic"-Schriftart mit fester Breite entspricht.

Die [Shift_JIS](#)-Code-Tabelle umfasst zum Beispiel 7070 Zeichen. Diese können wie folgt gruppiert werden:

- JIS-X-0201-Zeichen, ein Byte groß (Single-Byte-Zeichen) (191 Zeichen, auch Zeichen mit halber Breite genannt);
- JIS-X-0208-Zeichen, zwei Byte groß (Double-Byte-Zeichen) (6879 Zeichen, auch Zeichen mit voller Breite genannt).

Double-Byte-Zeichen nutzen die doppelte Breite in Konsolenterminals, welche CJK-Schriften mit fester Breite verwenden. Um mit dieser Situation umzugehen, sollte für Schriften, die Single-Byte- und Double-Byte-Zeichen enthalten, eventuell eine Schriftart im [Hanzi-Bitmap-Font-\(HBF\)-Format](#) mit der Dateiendung ".hbf" genutzt werden.

Um bei [TrueType](#)-Schriften Speicherplatz zu sparen, kann eine [TrueType](#)-Schriftarten-Sammlung mit der Dateiendung ".ttc" verwendet werden.

Um mit kompliziertem Code-Raum für bestimmte Zeichen umgehen zu können, wird eine CID-kodierte [PostScript](#)-Typ-1-Schrift mit CMap-Dateien verwendet, die selbsttätig mit `"%!PS-Adobe-3.0 Resource-CMap"` starten. Dies wird für eine normale X-Anzeige nur selten verwendet, aber häufiger für PDF-Rendering usw. (lesen Sie dazu Abschnitt [7.7.2](#)).

Tipp

Aufgrund der [Han-Vereinheitlichung](#) ist damit zu rechnen, dass für einzelne [Unicode](#)-Codepunkte mehrfach vorhandene [Glyphen](#) existieren. Einige der ärgerlichsten Vorkommen sind dabei "U+3001 IDEOGRAPHIC COMMA" und "U+3002 IDEOGRAPHIC FULL STOP", deren Zeichenpositionen sich zwischen den CJK-Staaten unterscheiden. Die Priorität von japanisch-zentrierten Schriften über diejenige von chinesischen Schriften zu setzen (über die Datei `"~/ .fonts . conf"`), sollte dem Japaner seinen Seelenfrieden zurückgeben.

7.7 X-Anwendungen

7.7.1 X-Büroanwendungen

Hier eine Liste mit grundlegenden Büroanwendungen (LO ist LibreOffice):

Paket	Popcon	Paketgröße	Art	Beschreibung
libreoffice-writer	V:188, I:441	39333	LO	Schreibprogramm
libreoffice-calc	V:188, I:436	32973	LO	Tabellenkalkulation
libreoffice-impress	V:176, I:433	9934	LO	Präsentation
libreoffice-base	V:145, I:325	7473	LO	Datenbank-Verwaltung
libreoffice-draw	V:177, I:434	14600	LO	Vektorgrafik-Editor
libreoffice-math	V:174, I:437	1963	LO	mathematischer Gleichungs-/Formel-Editor
abiword	V:1, I:12	5141	GNOME	Schreibprogramm
gnumeric	V:6, I:21	9933	GNOME	Tabellenkalkulation
gimp	V:68, I:341	22313	GTK	Bitmapgrafik-Editor
inkscape	V:55, I:209	84823	GNOME	Vektorgrafik-Editor
dia	V:5, I:31	3727	GTK	Ablaufschema- und Diagramm-Editor
planner	V:0, I:5	1146	GNOME	Projektverwaltung
calligrawords	V:0, I:7	5717	KDE	Schreibprogramm
calligrasheets	V:0, I:6	10890	KDE	Tabellenkalkulation
calligrastage	V:0, I:5	5102	KDE	Präsentation
calligraplan	V:0, I:2	15342	KDE	Projektverwaltung
kexi	V:0, I:2	7576	KDE	Datenbank-Verwaltung
karbon	V:0, I:7	3473	KDE	Vektorgrafik-Editor

Tabelle 7.8: Liste grundlegender X-Büroanwendungen

7.7.2 X-Werkzeuge

Hier eine Liste von grundlegenden Werkzeugen, die mir ins Auge gestochen sind:



Achtung

Das poppler-data-Paket (früher nicht-frei, lesen Sie Abschnitt [11.4.1](#)) muss installiert werden, damit evince und okular CJK-PDF-Dokumente mit CMap-Daten anzeigen können (lesen Sie dazu Abschnitt [7.6.3](#)).

Anmerkung

Die Installation von Software-Paketen wie scribus (aus KDE) innerhalb der GNOME-Arbeitsplatz-Umgebung ist absolut akzeptabel, da eine entsprechende Funktionalität in GNOME nicht enthalten ist. Allerdings könnte die Installation von vielen Paketen mit identischer Funktion Ihr Menü aufblähen.

Paket	Popcon	Paketgröße	Art	Beschreibung
evince	V:116, I:329	954	GNOME	Dokumentenbetrachter (pdf)
okular	V:46, I:118	14646	KDE	Dokumentenbetrachter (pdf)
calibre	V:9, I:36	54876	KDE	E-Book-Konvertierer und Bibliotheksverwaltung
fbreader	V:2, I:15	3074	GTK	E-Book-Reader
evolution	V:31, I:229	475	GNOME	persönliches Informationsmanagement (Groupware und E-Mail)
kontakt	V:1, I:16	2152	KDE	persönliches Informationsmanagement (Groupware und E-Mail)
scribus	V:2, I:23	30375	KDE	Seiten-Layout-Editor für den Arbeitsplatz
glabels	V:0, I:4	1326	GNOME	Aufkleber-Editor
gnucash	V:3, I:12	32304	GNOME	Finanzverwaltung/Homebanking-Programm
homebank	V:0, I:3	1044	GTK	Finanzverwaltung/Homebanking-Programm
kmy money	V:0, I:2	12036	KDE	Finanzverwaltung/Homebanking-Programm
shotwell	V:19, I:223	6451	GTK	Digitalfoto-Verwaltung
xsane	V:17, I:173	2346	GTK	Scanner-Frontend

Tabelle 7.9: Liste grundlegender X-Werkzeuge

7.8 X-Kleinigkeiten

7.8.1 Zwischenablage

Die X-Auswahl (mittels der dritten Maustaste) ist die native Zwischenablage-Funktionalität von X (lesen Sie dazu Abschnitt [1.4.4](#)).

Tipp

Umschalt-Eingf kann equivalent zu einem Klick der dritten Maustaste verwendet werden.

Paket	Popcon	Paketgröße	Art	Beschreibung
xsel	V:10, I:44	59	X	Befehlszeilenprogramm für die X-Auswahl
xclip	V:9, I:49	64	X	Befehlszeilenprogramm für die X-Auswahl

Tabelle 7.10: Liste von Basis-Programmen für die X-Auswahl

Die modernen Arbeitsplatz-Umgebungen (wie GNOME, KDE, ...) enthalten unterschiedliche Zwischenablage-Systeme für Ausschneiden, Kopieren und Einfügen mittels linker Maustaste oder Tastenkürzel (STRG-x, STRG-c und STRG-v).

7.8.2 Tastaturbelegungen und Zuweisungen von Zeigegerätetasten in X

`xmodmap(1)` ist ein Werkzeug zur Modifizierung von Tastaturbelegungen und Zuweisungen von Zeigegerätetasten (z.B. Mauspad-Tasten) im X-Window-System. Um den **keycode** (Tastencode) zu bekommen, starten Sie `xev(1)` in einer X-Sitzung und drücken die fragliche Taste. Wenn Sie die Bedeutung des **keysym** (symbolischer Name für den Tastencode; auch bekannt als Tastensymbol) erfahren möchten, schauen Sie sich die MACRO-Definition in der Datei `"/usr/include/X11/keysymdef.h"` (aus dem `x11proto-core-dev`-Paket) an. Alle `"#define"`-Ausdrücke in dieser Datei sind benannt als **keysym**-Name mit vorangestelltem `"XK_"`.

7.8.3 Klassische X-Clients

Die meisten traditionellen X-Client-Programme, wie z.B. `xterm(1)`, können mit einer Reihe von Standard-Befehlszeilen-Optionen gestartet werden, um Geometrie, Schriftart und zu verwendendes X-Display festzulegen.

Sie nutzen auch die X-resource-Datenbank, um ihre Erscheinungsart zu konfigurieren. Die systemweiten Standardwerte der X-Ressourcen sind in `"/etc/X11/Xresources/*"` gespeichert; anwendungsspezifische Werte für diese Ressourcen sind unter `"/etc/X11/app-defaults/*"` abgelegt. Verwenden Sie diese Werte als Ausgangspunkt.

Die Datei `"~/.Xresources"` wird angelegt, um benutzerspezifische Ressourcen-Einstellungen zu speichern. Diese Datei wird beim Login automatisch in die Standard-X-resources migriert. Um Änderungen an diesen Einstellungen durchzuführen und sie sofort aktiv werden zu lassen, müssen Sie sie mit folgendem Befehl in die Datenbank migrieren:

```
$ xrdp -merge ~/.Xresources
```

Lesen Sie `x(7)` und `xrdb(1)`.

7.8.4 Der X-Terminal-Emulator - xterm

Lernen Sie alles über `xterm(1)` unter <http://dickey.his.com/xterm/xterm.faq.html>.

7.8.5 X-Client-Programme als root laufen lassen



Warnung

Starten Sie niemals den X-Display-/Sitzungsmanager als root, indem Sie root im Displaymanager (z.B. gdm3) eingeben, auch nicht, wenn Sie planen, administrative Tätigkeiten durchzuführen; dies muss als unsicher angesehen werden. Die vollständige Architektur von X ist nicht mehr sicher, wenn sie als root ausgeführt wird. Sie müssen immer die niedrigst-mögliche Privilegienstufe verwenden, wie zum Beispiel ein reguläres Benutzerkonto.

Einfache Wege, ein spezielles X-Client-Programm, sagen wir `"foo"`, als root auszuführen, führen über die Verwendung von `sudo(8)` u.a., wie hier:

```
$ sudo foo &
```

```
$ sudo -s  
# foo &
```

```
$ ssh -X root@localhost  
# foo &
```



Achtung

Die Nutzung von `ssh(1)` für lediglich den obigen Zweck ist allerdings eine Verschwendung von Ressourcen.

Um den X-Client mit dem X-Server zu verbinden, beachten Sie bitte folgendes:

- Die Werte der Umgebungsvariablen `"$XAUTHORITY"` und `"$DISPLAY"` des alten Benutzers müssen in diejenigen des neuen Benutzers kopiert werden.
- Die Datei, auf die der Wert der Umgebungsvariable `"$XAUTHORITY"` zeigt, muss für den neuen Benutzer lesbar sein.

Kapitel 8

I18N und L10N

Multilingualisation (M17N) oder native Sprachunterstützung (die Anpassung eines Software-Produkts zur Unterstützung mehrerer Sprachen) wird in zwei Schritten durchgeführt:

- Internationalisierung (I18N): eine Software so einrichten, dass sie grundsätzlich verschiedene Gebietsschemata (Locales) unterstützt.
- Lokalisierung (L10N): eine Software für die Unterstützung eines bestimmten Gebietsschemas anpassen.

Tipp

Bei den englischen Begriffen Multilingualization, Internationalization und Localization sind 17, 18 oder 10 Buchstaben zwischen "m" und "n", "i" und "n" bzw. "l" und "n", was M17N, I18N und L10N entspricht.

Moderne Software wie GNOME oder KDE ist für die Unterstützung verschiedener Sprachen ausgerüstet. Sie ist mittels **UTF-8** internationalisiert und über die Bereitstellung übersetzter Texte über die gettext(1)-Infrastruktur lokalisiert. Übersetzte Texte werden teilweise als separate Lokalisierungspakete bereitgestellt und können ausgewählt werden, indem entsprechende Umgebungsvariablen auf den Wert für die zugehörige Locale gesetzt werden.

Die einfachste Darstellung von Textdaten ist über **ASCII**, was für Englisch passend ist und weniger als 127 Zeichen enthält (abzubilden mit 7 Bits). Um für internationale Unterstützung erheblich mehr Zeichen zur Verfügung zu haben, wurden viele verschiedene Zeichenkodierungssysteme eingeführt. Das modernste und vernünftigste Kodierungssystem ist **UTF-8**, welches mit nahezu allen dem Menschen bekannten Zeichen umgehen kann (lesen Sie dazu Abschnitt [8.4.1](#)).

Details hierzu finden Sie unter [Introduction to i18n](#).

Internationale Hardware-Unterstützung wird über lokalisierte Hardware-Konfigurationsdaten realisiert.



Warnung

Dieses Kapitel ist überholt, da es auf dem in 2013 veröffentlichten Debian 7.0 (wheezy) basiert.

8.1 Die Tastatureingabe

Das Debian-System kann mittels der keyboard-configuration- und console-setup-Pakete so konfiguriert werden, dass es mit vielen internationalen Tastaturkonfigurationen funktioniert:

```
# dpkg-reconfigure keyboard-configuration
# dpkg-reconfigure console-setup
```

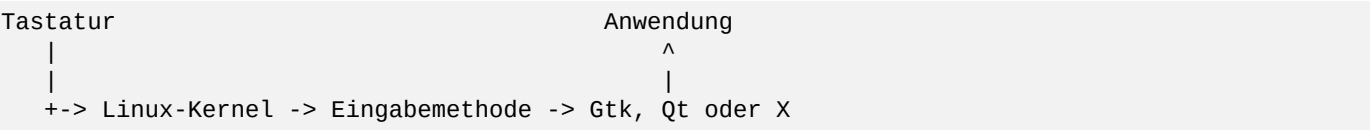
Damit wird die Tastatur für die Linux-Konsole konfiguriert und Parameter für das X-Window-System in `/etc/default/keyboard` und `/etc/default/console-setup` aktualisiert. Es wird auch die Schriftart für die Linux-Konsole festgelegt.

Viele nicht-ASCII-Zeichen, inklusive akzentuierte Zeichen, die in europäischen Sprachen verwendet werden, sind über [Tottaste](#), [AltGr-Taste](#) und [Compose-Taste](#) zugänglich.

Für asiatische Sprachen benötigen Sie Unterstützung für aufwändigere [Eingabemethoden](#) wie [IBus](#), die im folgenden behandelt werden.

8.1.1 Unterstützung für die Eingabemethode mit IBus

Mehrsprachige Eingaben für Anwendungen werden wie hier dargestellt verarbeitet:



Die Einrichtung multilingualer Eingabemethoden wird auf dem Debian-System durch die Verwendung der [IBus](#)-Pakete und des `im-config`-Pakets erheblich vereinfacht. Hier eine Liste der IBus-Pakete:

Paket	Popcon	Größe	unterstütztes Gebietsschema
ibus	V:70, I:87	1581	Eingabemethoden-Rahmenwerk, das dbus verwendet
ibus-mozc	V:1, I:2	999	Japanisch
ibus-anthy	V:0, I:1	8723	”
ibus-kkc	V:0, I:0	214	”
ibus-skk	V:0, I:0	244	”
ibus-pinyin	V:0, I:1	1434	Chinesisch (für zh_CN)
ibus-chewing	V:0, I:0	415	” (für zh_TW)
ibus-hangul	V:0, I:1	288	Koreanisch
ibus-table	V:0, I:1	1801	Tabellen-Engine für IBus
ibus-table-thai	I:0	47	Thailändisch
ibus-unikey	V:0, I:0	318	Vietnamesisch
ibus-m17n	V:0, I:1	187	Multilingual: Indisch, Arabisch und andere

Tabelle 8.1: Liste der unterstützten Eingabemethoden mit IBus

Die `kinput2`-Methode und andere locale-abhängige [Eingabemethoden](#) für klassische asiatische Sprachen existieren zwar noch, werden aber für moderne UTF-8-Umgebungen nicht mehr empfohlen. Die [SCIM](#)- und [uim](#)-Werkzeuge sind ein geringfügig älteres Vorgehen für internationale Eingabemethoden in modernen UTF-8-Umgebungen.

8.1.2 Ein Beispiel für Japanisch

Ich finde diese japanische Eingabemethode, beginnend in einer englischen Umgebung (`"en_US.UTF-8"`), sehr nützlich. Hier ein Beispiel, wie ich dies mit IBus für GNOME3 eingerichtet habe:

1. Installieren Sie das `ibus-anthy`-Paket für das japanische Eingabewerkzeug mit seinen Paketempfehlungen wie z.B. `im-config`.
2. Führen Sie `"im-config"` auf der Shell des Benutzers aus und wählen Sie als Eingabemethode `"ibus"`.
3. Wählen Sie dann `"Einstellungen"` → `"Tastatur"` → `"Eingabequellen"`, klicken Sie bei den `"Eingabequellen"` auf `"+"` → `"Japanisch"` → `"Japanisch (anthy)"` und dann `"Hinzufügen"`.
4. Wählen Sie `"Japanisch"` und klicken Sie `"Hinzufügen"`, um die japanische Tastaturbelegung ohne Zeichenkonvertierung auszuwählen. (Sie möchten vielleicht ebenso viele Eingabequellen auswählen.)

5. Melden Sie sich am Benutzerkonto neu an.
6. Kontrollieren Sie nochmals die Einstellung durch Ausführen von `im-config`.
7. Richten Sie die Eingabequelle durch einen Rechtsklick auf das Icon in der GUI-Werkzeugleiste ein.
8. Wählen Sie zwischen den installierten Eingabequellen mittels SUPERTASTE+LEERTASTE (die SUPERTASTE ist normalerweise die Windows-Taste).

Bitte beachten Sie folgendes:

- `im-config(8)` verhält sich unterschiedlich abhängig davon, ob es von `root` ausgeführt wurde oder nicht.
- `im-config(8)` aktiviert standardmäßig ohne weiteres Agieren des Benutzers die beste Eingabemethode auf dem System.
- Der Eintrag für `im-config(8)` im GUI-Menü ist standardmäßig deaktiviert, um ein Aufblähen des Menüs zu vermeiden.

8.1.3 Die Eingabemethode deaktivieren

Falls Sie bei der Eingabe kein XIM (Mechanismus, der von X genutzt wird) verwenden möchten, setzen Sie während des Programmstarts den Wert von `$XMODIFIERS` auf `none`. Dies könnte sinnvoll sein, wenn Sie die japanische Eingabe-Infrastruktur `egg` in `emacs(1)` nutzen und dazu `ibus` deaktivieren möchten. Führen Sie dazu auf der Shell folgendes aus:

```
$ XMODIFIERS=none emacs
```

Um einen Befehl, der aus dem Debian-Menü ausgeführt wird, anzupassen, legen Sie eine benutzerspezifische Konfiguration in `/etc/menu/` an; folgen Sie dabei der in `/usr/share/doc/menu/html` beschriebenen Methode.

8.2 Die Bildschirmausgabe

Die Linux-Konsole kann nur eine beschränkte Anzahl an Zeichen anzeigen. (Sie benötigen ein spezielles Terminal-Programm wie `jfbterm(1)`, um nicht-europäische Sprachen in der Konsole außerhalb von X anzuzeigen.)

X-Window kann jegliche Zeichen in UTF-8 anzeigen, sofern die benötigten Schriftdateien existieren. (Die Kodierung der Original-Schriftdateien wird durch das X-Window-System und für den Benutzer transparent erledigt.)

8.3 Unbekannte Zeichenbreite bei ostasiatischen Zeichen

Bei Verwendung des ostasiatischen Gebietsschemas können die Zeichnung der Box sowie griechische und kyrillische Schriftzeichen breiter als beabsichtigt dargestellt werden, was zu einer ungleichmäßigen, schlecht ausgerichteten Terminalausgabe führt (näheres unter [Unicode Standard Annex #11](#)).

Sie können dies Problem jedoch umgehen:

- `gnome-terminal`: Bearbeiten → Einstellungen → Profile → Bearbeiten → Kompatibilität → Zeichen mit unbekannter Breite → Schmal
- `ncurses`: Setzen einer Umgebungsvariablen mit `export NCURSES_NO_UTF8_ACS=0`.

8.4 Das Gebietsschema (Locale)

Die folgenden Informationen beziehen sich auf das Gebietsschema für Anwendungen, die in der durch `gdm3(1)` gestarteten X-Window-Umgebung laufen.

8.4.1 Grundlagen der Kodierung

Die Umgebungsvariable `"LANG=xx_YY.ZZZZ"` setzt das Gebietsschema auf den Sprachcode `"xx"`, den Ländercode `"yy"` und die Kodierung `"ZZZZ"` (Näheres in Abschnitt 1.5.2).

Das aktuelle Debian-System setzt das Gebietsschema normalerweise auf `"LANG=xx_YY.UTF-8"`. Dadurch wird die [UTF-8](#)-Kodierung mit dem [Unicode](#)-Zeichensatz verwendet. Diese [UTF-8](#)-Kodierung ist ein Multibyte-Code-System und nutzt intelligent die enthaltenen Codepunkte. [ASCII](#)-Daten, die lediglich aus 7 Bit breiten Codes bestehen, sind immer auch gültige UTF-8-Daten, die aber 1 Byte pro Zeichen belegen.

In früheren Debian-Systemen wurde das Gebietsschema auf `"LANG=C"` oder `"LANG=xx_YY"` (ohne `".UTF-8"`) gesetzt.

- Für `"LANG=C"` oder `"LANG=POSIX"` wird der [ASCII](#)-Zeichensatz benutzt.
- Für `"LANG=xx_YY"` wird in Unix das traditionelle Kodierungssystem benutzt.

Das tatsächliche Kodierungssystem, das für `"LANG=xx_YY"` verwendet wird, kann über `"/usr/share/i18n/SUPPORTED"` identifiziert werden. Zum Beispiel nutzt `"en_US"` die `"ISO-8859-1"`-Kodierung und `"de_DE@euro"` nutzt `"ISO-8859-15"`.

Tipp

Informationen zur Bedeutung der Werte für die verschiedenen Kodierungen finden Sie in Tabelle 11.2.

8.4.2 Argumentation für UTF-8-Gebietsschemata

Der [Unicode](#)-Zeichensatz kann nahezu alle dem Menschen bekannten Zeichen mit Codepunkten aus dem Bereich von 0 bis 10FFFF in hexadezimaler Darstellung abbilden. Seine Speicherung erfordert mindestens 21 Bit.

Das Zeichenkodierungssystem [UTF-8](#) passt Unicode-Codepunkte in einen 8 Bit breiten Datenstrom ein, der zum ASCII-Datenverarbeitung kompatibel ist. **UTF** steht für Unicode Transformation Format.

Ich empfehle, ein [UTF-8](#)-Gebietsschema für das Arbeitsplatzsystem zu verwenden, z.B. `"LANG=de_DE.UTF-8"`. Der erste Teil des Gebietsschemas repräsentiert Texte, die von Anwendungen angezeigt werden. So kann zum Beispiel `gedit(1)` (der Texteditor des GNOME-Desktops) mit `"LANG=de_DE.UTF-8"` chinesische Zeichen anzeigen und editieren (solange die benötigten Schriftarten und Eingabemethoden installiert sind), wobei die Menüs in Deutsch angezeigt werden.

Ich empfehle außerdem, das Gebietsschema nur mittels der Umgebungsvariable `"$LANG"` zu setzen. Ich sehe keinen großen Gewinn darin, bei einer UTF-8-Locale eine komplizierte Kombination aus `"LC_*"`-Variablen zu setzen (lesen Sie dazu `locale(1)`).

Sogar reiner Text in Englisch kann nicht-ASCII-Zeichen enthalten, so sind z.B. die leicht geschweiften rechten und linken Anführungszeichen in ASCII nicht enthalten:

```
b'' "b''Dieser Text mit doppelten Anführungszeichenb''" b'' ist kein reiner ASCII-Text,
"dieser hingegen schon."
b'' 'b''Dieser Text mit einfachen Anführungszeichenb'' ' b'' ist kein reiner ASCII-Text,
'dieser hingegen schon.'
```

Wenn reiner Text in [ASCII](#) nach [UTF-8](#) konvertiert wird, hat er exakt den gleichen Inhalt und die gleiche Größe wie die ASCII-Variante. Daher verlieren Sie nichts beim Einsatz des UTF-8-Gebietsschemas.

Einige Programme verbrauchen mehr Speicher, wenn sie I18N unterstützen. Das kommt daher, weil sie für die Unicode-Unterstützung intern [UTF-32\(UCS4\)](#) verwenden (zwecks Geschwindigkeitsoptimierung; dabei werden 4 Bytes für jedes ASCII-Zeichen genutzt, unabhängig vom gewählten Gebietsschema). Nochmals: Sie verlieren nichts, wenn Sie ein UTF-8-Gebietsschema einsetzen.

Die alten herstellereigenen nicht-UTF-8-Kodierungssysteme neigen dazu, kleine, aber nervige Unterschiede bei einigen Zeichen zu enthalten, wie zum Beispiel grafische Differenzen für viele Länder. Der Einsatz des UTF-8-Systems durch moderne Betriebssysteme löst diese Konflikte.

8.4.3 Die Neukonfiguration des Gebietsschemas (Locale)

Damit das System auf ein bestimmtes Gebietsschema zugreifen kann, müssen die Locale-Daten aus der Locale-Datenbank kompiliert werden. (Auf dem Debian-System sind **nicht** alle verfügbaren Gebietsschemata vorkonfiguriert vorhanden, außer Sie installieren das Paket `locales-all`.) Die vollständige Liste der Gebietsschemata, die zur Kompilierung verfügbar sind, finden Sie in `/usr/share/i18n/SUPPORTED`. Dort sind alle exakten Locale-Namen aufgeführt. Der folgende Befehl listet alle verfügbaren UTF-8-Gebietsschemata auf, die bereits in binäre Form kompiliert vorliegen:

```
$ locale -a | grep utf8
```

Durch Ausführung des folgenden Befehls wird das `locales`-Paket neu konfiguriert:

```
# dpkg-reconfigure locales
```

Dieser Prozess besteht aus drei Schritten:

1. die Liste verfügbarer Gebietsschemata aktualisieren;
2. diese in die binäre Form kompilieren;
3. den Wert des systemweiten Standard-Gebietsschemas in `/etc/default/locale` setzen, damit er durch PAM genutzt werden kann (lesen Sie dazu auch Abschnitt [4.5](#)).

Die Liste verfügbarer Gebietsschemata sollte `en_US.UTF-8` sowie alle für Sie interessanten Sprachen mit `UTF-8` enthalten.

Das empfohlene Standard-Gebietsschema ist `en_US.UTF-8` für US-amerikanisches Englisch. Für andere Sprachen stellen Sie bitte sicher, dass Sie ein Gebietsschema mit `UTF-8` auswählen, für Deutsch z.B. `de_DE.UTF-8`. Bei all diesen Locales-Einstellungen werden jegliche internationale Zeichen unterstützt.

Anmerkung

Obwohl das Setzen des Gebietsschemas auf `C` zur Verwendung von Texten in US-amerikanischem Englisch führt, können so nur ASCII-Zeichen genutzt werden.

8.4.4 Der Wert der Umgebungsvariable `$LANG`

Der Wert der Umgebungsvariable `$LANG` wird durch viele Anwendungen gesetzt und verändert:

- wird anfänglich gesetzt durch den PAM-Mechanismus von `login(1)` für die lokalen Linux-Konsolenprogramme;
- wird anfänglich gesetzt durch den PAM-Mechanismus des Displaymanagers für alle X-Programme;
- wird anfänglich gesetzt durch den PAM-Mechanismus von `ssh(1)` für die fernen Konsolenprogramme;
- wird verändert durch einige Displaymanager wie `gdm3(1)` für alle X-Programme;
- wird verändert durch den Startcode der X-Sitzung via `~/ .xsessionrc` für alle X-Programme;
- wird verändert durch den Shell-Startcode, z.B. `~/ .bashrc` für alle Konsolenprogramme.

Tipp

Es ist eine gute Idee, zwecks maximaler Kompatibilität als systemweites Gebietsschema `en_US.UTF-8` zu installieren.

8.4.5 Spezifisches Gebietsschema nur für X-Window

Über eine Anpassung von PAM können Sie ein spezifisches Gebietsschema nur für X-Window wählen, unabhängig vom systemweiten Standard-Gebietsschema (lesen Sie auch Abschnitt 4.5).

Diese Einstellung sollte zu besten Ergebnissen und höchster Stabilität führen. Sie haben Zugriff auf ein funktionsfähiges textbasiertes Terminal mit verständlichen Meldungen, selbst wenn das X-Window-System nicht läuft. Dies ist essentiell wichtig für Sprachen, die nicht-romanische Zeichen verwenden, wie z.B. Chinesisch, Japanisch und Koreanisch.

Anmerkung

Es könnte auch noch andere Wege geben, wie die Verbesserung des X-Sitzungsmanager-Pakets, aber bitte lesen Sie hier weiter bezüglich allgemeiner und grundlegender Methoden zum Einrichten des Gebietsschemas. Bei gdm3(1) weiß ich, dass man das Gebietsschema der X-Sitzung über das Menü festlegen kann.

Folgende Zeile in der PAM-Konfigurationsdatei (z.B. `/etc/pam.d/gdm3`) definiert den Speicherort der Datei für die Sprachumgebung:

```
auth    required      pam_env.so read_env=1 envfile=/etc/default/locale
```

Ändern Sie dies wie folgt:

```
auth    required      pam_env.so read_env=1 envfile=/etc/default/locale-x
```

Für Japanisch erzeugen Sie eine `/etc/default/locale-x`-Datei mit den Rechten `-rw-r--r-- 1 root root`, die folgendes enthält:

```
LANG="ja_JP.UTF-8"
```

Behalten Sie die Standarddatei `/etc/default/locale` für andere Programme mit folgendem Inhalt bei:

```
LANG="en_US.UTF-8"
```

Dies ist die grundlegendste Technik zur Anpassung des Gebietsschemas und führt dazu, dass selbst der Menüauswahldialog von gdm3(1) lokalisiert ist.

Alternativ hierzu können Sie das Gebietsschema auch einfach über eine Änderung der Datei `~/.xsessionrc` anpassen.

8.4.6 Dateinamenskodierung

Für den Datenaustausch über verschiedene Plattformen hinweg (Näheres in Abschnitt 10.1.7) müssen Sie unter Umständen beim Einbinden einiger Dateisysteme bestimmte Kodierungen vorwählen. Zum Beispiel wird beim Einbinden eines [vfat-Dateisystems](#) durch mount(8) die [CP437](#)-Kodierung genutzt, wenn nichts anderes angegeben ist. Sie müssen eine explizite mount-Option angeben, wenn [UTF-8](#) oder [CP932](#) als Kodierung für die Dateinamen genutzt werden soll.

Anmerkung

Wenn ein während des Betrieb eingesteckter USB-Stick in einer modernen Arbeitsplatzumgebung wie GNOME automatisch eingebunden wird, können Sie solche mount-Optionen möglicherweise durch einen Rechtsklick auf das Icon auf der Arbeitsfläche vorwählen. Klicken Sie auf die Lasche "Laufwerk", klappen Sie per Mausklick die Liste der "Einstellungen" auf und geben Sie dann "utf8" bei den "Mount-Optionen:" ein. Wenn der USB-Stick das nächste Mal eingesteckt wird, ist das Einbinden mit UTF-8 aktiviert.

Anmerkung

Beim Hochrüsten eines Systems oder wenn Sie Laufwerke von älteren nicht-UTF-8-Systemen verwenden, könnten Dateinamen mit nicht-ASCII-Zeichen in historischen und überholten Kodierungen wie [ISO-8859-1](#) oder [eucJP](#) kodiert sein. Bitte suchen Sie die Hilfe von Textkonvertierungswerkzeugen, um diese in [UTF-8](#) zu konvertieren. Details hierzu finden Sie in Abschnitt 11.1.

[Samba](#) verwendet für neuere Clients (Windows NT, 200x, XP und später) standardmäßig Unicode, aber bei älteren für DOS und Windows 9x/Me wird per Voreinstellung [CP850](#) eingesetzt. Dieser Standard für ältere Clients kann mittels "dos charset" in der Datei "/etc/samba/smb.conf" geändert werden, [CP932](#) zum Beispiel für Japanisch.

8.4.7 Lokalisierte Meldungen und übersetzte Dokumentation

Für viele Textmeldungen und Dokumente, die im Debian-System angezeigt werden, wie z.B. Fehlermeldungen, Standard-Programmausgaben, Menüs und Handbuchseiten, existieren Übersetzungen. Die [GNU-gettext\(1\)-Werkzeuge](#) werden für die meisten Übersetzungsaktivitäten als Backend-Programme im Hintergrund verwendet.

`aptitude(8)` bietet unter "Tasks" → "Lokalisierung" eine ausführliche Liste nützlicher Binärpakete, die lokalisierte Meldungen für Anwendungen und übersetzte Dokumentation enthalten.

So können Sie übersetzte Handbuchseiten (`manpages`) in Deutsch erhalten, indem Sie das `manpages-de`-Paket installieren. Um z.B. die deutsche Handbuchseite für `<programmname>` aus `"/usr/share/man/de/"` zu lesen (falls für dieses Programm eine übersetzte Handbuchseite existiert), führen Sie folgendes aus:

```
LANG=de_DE.UTF-8 man <programmname>
```

8.4.8 Auswirkungen des Gebietsschemas

Durch die Sprachauswahl im Gebietsschema wird die Sortierreihenfolge von Zeichen mit `sort(1)` beeinflusst. In den Gebietsschemata für Spanisch und Englisch beispielsweise wird unterschiedlich sortiert.

Das Gebietsschema wirkt sich auch auf das Datumsformat von `ls(1)` aus. Die Ausgaben des Datumsformats von "LANG=C ls -l" und "LANG=en_US.UTF-8" z.B. sind unterschiedlich (Näheres in Abschnitt [9.2.5](#)).

Die Zahlen-Interpunktion unterscheidet sich ebenfalls abhängig vom Gebietsschema. Zum Beispiel wird ein-tausend-eins-komma-eins im englischen Gebietsschema "1,000.1" geschrieben, im deutschen hingegen "1.000,1". Sie können den Unterschied in einem Tabellenkalkulationsprogramm sehen.

Kapitel 9

Systemtipps

Hier beschreibe ich einige grundlegende Tipps zur Konfiguration und Verwaltung des Systems, überwiegend für die Konsole.

9.1 Das Programm screen

screen(1) ist ein sehr nützliches Werkzeug für Leute, die per Fernzugriff über unzuverlässige oder instabile Verbindungen auf ferne Rechner zugreifen möchten, da es für sporadische Unterbrechungen der Netzwerkverbindung gerüstet ist.

Paket	Popcon	Größe	Beschreibung
screen	V:127, I:281	1013	Terminal-Multiplexer mit VT100-/ANSI-Terminal-Emulation
tmux	V:34, I:136	830	Terminal-Multiplexer-Alternative (verwenden Sie stattdessen "Strg-B")

Tabelle 9.1: Liste von Programmen, die Netzwerkverbindungen mit Unterbrechungen unterstützen:

9.1.1 Anwendungsszenario für screen

screen(1) bietet nicht nur die Möglichkeit, in einem Terminal-Fenster mehrere Prozesse laufen zu lassen, sondern erlaubt auch **einem Remote-Shell-Prozess, Verbindungsunterbrechungen zu überstehen**. Hier ein typisches Anwendungsszenario für screen(1):

1. Sie melden sich auf einer fernen Maschine an.
2. Sie starten `screen` auf einer einfachen Konsole.
3. Sie führen mehrere Programme in `screen`-Fenstern aus, die Sie über `^A c` ("Strg-A" gefolgt von "c") erzeugen.
4. Sie können mittels `^A n` ("Strg-A" gefolgt von "n") zwischen den verschiedenen `screen`-Fenstern hin- und herschalten.
5. Plötzlich müssen Sie Ihr Terminal verlassen, aber Sie möchten Ihre aktuelle Arbeit nicht verlieren und deshalb die Verbindung erhalten.
6. Sie können die Verbindung zur `screen`-Sitzung durch eine der folgenden Methoden **lösen**:
 - Ganz brutal: die Netzwerkverbindung trennen;
 - Tippen Sie `^A d` ("Strg-A" gefolgt von "d") und melden Sie sich händisch von der Fernverbindung ab;
 - Tippen Sie `^A DD` ("Strg-A" gefolgt von "DD"), damit `screen` die Verbindung löst und Sie abmeldet.
7. Sie melden Sich am gleichen fernen Rechner erneut an (funktioniert sogar bei Anmeldung über ein anderes Terminal).

8. Sie starten screen über "screen -r".

9. screen **verbindet** ganz von selbst alle vorherigen screen-Fenster mit allen aktuell laufenden Programmen.

Tipp

Bei Verbindungen, die über Zeit oder Volumen abgerechnet werden, können Sie mit screen Kosten sparen, indem Sie einen Prozess aktiv laufen lassen, während die Verbindung aber unterbrochen ist, und die Sitzung später erneut verbinden, wenn Sie sich wieder einwählt haben.

9.1.2 Tastaturkürzel für den screen-Befehl

In einer screen-Sitzung werden alle Tastatureingaben zu Ihrem aktuell laufenden screen-Fenster gesendet, außer Befehlseingaben. Alle screen-Befehlseingaben werden mittels ^A ("Strg-A") plus einer einzelnen Taste [plus eventuellen Parametern] eingegeben. Hier einige wichtige zur Erinnerung:

Tastaturbefehl	Bedeutung
^A ?	eine Hilfe anzeigen (Tastaturkürzel anzeigen)
^A c	ein neues Fenster erstellen und dorthin wechseln
^A n	zu nächstem Fenster wechseln
^A p	zu vorherigem Fenster wechseln
^A 0	zu Fenster Nr. 0 wechseln
^A 1	zu Fenster Nr. 1 wechseln
^A w	eine Liste vorhandener Fenster anzeigen
^A a	ein Strg-A als Tastatureingabe an das aktuelle Fenster senden
^A h	Hardcopy des aktuellen Fensters in eine Datei schreiben
^A H	Starten/stoppen des Aufzeichnens vom aktuellen Fenster in eine Datei
^A ^X	Terminal sperren (Passwort-geschützt)
^A d	Verbindung der screen-Sitzung zum Terminal lösen
^A DD	Verbindung der screen-Sitzung lösen und abmelden

Tabelle 9.2: Liste von Tastaturkürzeln für screen

Lesen Sie screen(1) bezüglich weiterer Details.

9.2 Datenaufzeichnung und -darstellung

9.2.1 Der log-Daemon

Viele Programme zeichnen ihre Aktivitäten im "/var/log/"-Verzeichnis auf:

- der System-Log-Daemon: rsyslogd(8).

Lesen Sie dazu Abschnitt 3.2.5 und Abschnitt 3.2.4.

9.2.2 Analyseprogramme für Logdateien

Hier eine Liste erwähnenswerter Analyseprogramme für Logdateien ("~Gsecurity::log-analyzer" in aptitude(8)):

Anmerkung

CRM114 bietet eine Sprachinfrastruktur, um **Unschärfe**-Filter (fuzzy filters) mittels der [TRE-regex-Bibliothek](#) zu erstellen. Ein populärer Anwendungsfall ist der Spam-Mail-Filter, aber es kann auch als Log-Analysator verwendet werden.

Paket	Popcon	Größe	Beschreibung
logwatch	V:16, I:18	2265	Log-Analysator mit netter Ausgabe, geschrieben in Perl
fail2ban	V:112, I:123	2092	IP-Adressen sperren, die vielfache Authentifizierungsfehler verursachen
analog	V:4, I:109	3534	Webserver-Log-Analysator
awstats	V:9, I:15	6910	leistungsfähiger und mit vielen Funktionen ausgestatteter Webserver-Log-Analysator
sarg	V:3, I:3	843	Analysereportgenerator für Squid
pflogsumm	V:1, I:4	111	Programm zur Zusammenfassung von Postfix-Logeinträgen
syslog-summary	V:0, I:2	30	Programm zum Zusammenfassen des Inhalts einer Syslog-Datei
fwlogwatch	V:0, I:0	479	Firewall-Log-Analysator
squidview	V:0, I:1	189	Programm zum Beobachten und Analysieren von Squid-access.log-Dateien
swatch	V:0, I:0	101	Logdatei-Betrachter mit Unterstützung für reguläre Ausdrücke, Hervorhebung und Einsprungstellen
crm114	V:0, I:0	1119	Programm zur flexiblen Einordnung von E-Mails und anderen Daten (CRM114)
icmpinfo	V:0, I:0	44	Programm zur Interpretation von ICMP-Nachrichten

Tabelle 9.3: Liste von System-Log-Analysatoren

9.2.3 Shell-Aktivitäten sauber aufzeichnen

Die schlichte Verwendung von `script(1)` wie in Abschnitt 1.4.9 erzeugt eine Datei mit Steuerzeichen. Sie können dies durch Nutzung von `col(1)` wie folgt beheben:

```
$ script
Script started, file is typescript (zu deutsch: Script gestartet, die Datei heißt ←
    typescript)
```

Machen Sie irgendetwas ... und drücken Sie `Strg-D`, um `script` zu beenden.

```
$ col -bx <typescript >bereinigte-datei
$ vim bereinigte-datei
```

Falls Sie kein `script` haben (zum Beispiel während des Boot-Vorgangs in der `initramfs`), können Sie stattdessen auch Folgendes verwenden:

```
$ sh -i 2>&1 | tee typescript
```

Tipp

Einige `x-terminal-emulator`-Programme wie z.B. `gnome-terminal` können auch aufzeichnen. Sie sollten vielleicht den Zeilenpuffer erweitern, um zurückscrollen zu können.

Tipp

Sie können ebenso `screen(1)` über "`^A H`" nutzen (lesen Sie dazu auch Abschnitt 9.1.2), um eine Konsolensitzung aufzuzeichnen.

Tipp

Auch `emacs(1)` kann über "`M-x shell`", "`M-x eshell`" oder "`M-x term`" genutzt werden, um eine Konsolensitzung aufzuzeichnen. Um den Puffer später in eine Datei zu schreiben, verwenden Sie "`C-x C-w`".

9.2.4 Angepasste Anzeige von Textdaten

Obwohl Pager wie `more(1)` und `less(1)` (lesen Sie dazu Abschnitt 1.4.5) sowie spezielle Werkzeuge für Hervorhebung und Formatierung (Näheres dazu in Abschnitt 11.1.8) Text gut anzeigen können, sind normale Texteditoren (Details in Abschnitt 1.4.6) vielseitiger und besser anzupassen.

Tipp

Bei `vim(1)` und seinem Pager-Modus alias `view(1)` wird über `":set hls"` die Suche mit Hervorhebung aktiviert.

9.2.5 Angepasste Anzeige von Datum und Zeit

Das Standard-Anzeigeformat von Datum und Zeit bei dem Befehl `"ls -l"` hängt vom **Gebietsschema** (locale) ab (lesen Sie dazu Abschnitt 1.2.6). Die Variable `"$LANG"` wird als erstes abgefragt, der Wert kann jedoch noch mittels der `"$LC_TIME"`-Variablen überschrieben werden.

Das letztlich verwendete Anzeigeformat für das jeweilige Gebietsschema hängt von der verwendeten Standard-C-Bibliothek (aus dem `libc6`-Paket) ab. Aufgrunddessen hatten verschiedene Debian-Veröffentlichungen unterschiedliche Standardwerte.

Wenn Sie das Anzeigeformat von Datum und Zeit wirklich über die **locale**-Einstellung hinaus anpassen möchten, sollten Sie die Option **time style value** entweder über das `"--time-style"`-Argument oder über den Wert der Variable `"$TIME_STYLE"` setzen (Näheres unter `ls(1)`, `date(1)` und `"info coreutils 'ls invocation'"`).

time style value	Locale	Anzeige von Datum und Zeit
iso	alle	01-19 00:15
long-iso	alle	2009-01-19 00:15
full-iso	alle	2009-01-19 00:15:16.000000000 +0900
locale	C	Jan 19 00:15
locale	en_US.UTF-8	Jan 19 00:15
locale	es_ES.UTF-8	ene 19 00:15
+%d.%m.%y %H:%M	alle	19.01.09 00:15
+%d.%b.%y %H:%M	C oder en_US.UTF-8	19.Jan.09 00:15
+%d.%b.%y %H:%M	es_ES.UTF-8	19.ene.09 00:15

Tabelle 9.4: Anzeigebeispiele von Datum und Zeit für den Befehl `"ls -l"` unter Wheezy

Tipp

Sie können die Eingabe langer Optionsparameter auf der Befehlszeile vermeiden, indem sie einen Befehls-Alias verwenden, z.B. `"alias ls='ls --time-style=+%d.%m.%y\ %H:%M'"` (lesen Sie dazu Abschnitt 1.5.9).

Tipp

Obige iso-Formate sind konform mit [ISO 8601](#).

9.2.6 Farbige Shell-Ausgabe

Die Shell-Ausgabe der meisten modernen Terminals kann über [ANSI-Escape-Codes](#) farbig gestaltet werden (lesen Sie dazu auch `"/usr/share/doc/xterm/ctlseqs.txt.gz"`).

Probieren Sie zum Beispiel folgendes:

```
$ RED=$(printf "\x1b[31m")
$ NORMAL=$(printf "\x1b[0m")
$ REVERSE=$(printf "\x1b[7m")
$ echo "${RED}RED-TEXT${NORMAL} ${REVERSE}REVERSE-TEXT${NORMAL}"
```

9.2.7 Farbige Befehle

Farbige Befehle sind praktisch, um deren Ausgabe in einer interaktiven Umgebung zu kontrollieren. Ich habe Folgendes in meiner `~/ .bashrc`:

```
if [ "$TERM" != "dumb" ]; then
    eval "`dircolors -b`"
    alias ls='ls --color=always'
    alias ll='ls --color=always -l'
    alias la='ls --color=always -A'
    alias less='less -R'
    alias ls='ls --color=always'
    alias grep='grep --color=always'
    alias egrep='egrep --color=always'
    alias fgrep='fgrep --color=always'
    alias zgrep='zgrep --color=always'
else
    alias ll='ls -l'
    alias la='ls -A'
fi
```

Die Verwendung von Befehls-Alias beschränkt die Farbeffekte auf interaktive Befehlseingaben. Dies ist ein Vorteil gegenüber dem Exportieren der Umgebungsvariablen (`export GREP_OPTIONS='--color=auto'`), da die Farben auch in Pager-Programmen wie `less(1)` sichtbar sind. Falls Sie die Farbeffekte bei Weiterleitung zu anderen Programmen deaktivieren möchten, nutzen Sie stattdessen `--color=auto` in dem obigen Beispiel für `~/ .bashrc`.

Tipp

Sie können die Farb-Alias in der interaktiven Umgebung ausschalten, indem Sie die Shell mit `TERM=dumb bash` starten.

9.2.8 Aufzeichnen von Editor-Aktivitäten für komplexe Wiederholungen

Sie können Editor-Aktivitäten aufzeichnen, um sie später für komplexe Wiederholungen wiederzuverwenden.

Bei [Vim](#) wie folgt:

- `"qa`: Aufnahme der eingegebenen Zeichen in das Register `"a` starten;
- `...` Editor-Aktivitäten;
- `"q`: Aufnahme der eingegebenen Zeichen beenden;
- `"@a`: Ausführen des Inhalts von Register `"a`.

Bei [Emacs](#) wie folgt:

- `"C-x (`: Definition eines Tastaturmakros starten;
 - `...` Editor-Aktivitäten;
 - `"C-x)`: Definition des Tastaturmakros beenden;
 - `"C-x e`: Ausführen des Tastaturmakros.
-

9.2.9 Die Bildschirmanzeige einer X-Anwendung aufzeichnen

Es gibt mehrere Wege, die Anzeige einer X-Anwendung aufzuzeichnen, inklusive eines `xterm`-Displays:

Paket	Popcon	Größe	Befehl
xbase-clients	I:26	46	<code>xwd(1)</code>
gimp	V:68, I:341	22313	GUI-Menü
imagemagick	I:400	218	<code>import(1)</code>
scrot	V:8, I:80	70	<code>scrot(1)</code>

Tabelle 9.5: Liste von Werkzeugen zur Bildbearbeitung

9.2.10 Aufzeichnen von Änderungen in Konfigurationsdateien

Es gibt spezielle Werkzeuge, um Änderungen in Konfigurationsdateien mit Hilfe von DVCS-Systemen aufzuzeichnen.

Paket	Popcon	Größe	Beschreibung
etckeeper	V:27, I:32	162	Konfigurationsdateien und deren Metadaten mit Git (Standard), Mercurial oder Bazaar (neu) abspeichern
changetrack	V:0, I:0	71	Konfigurationsdateien mit RCS abspeichern (alt)

Tabelle 9.6: Liste von Paketen zur Aufzeichnung der Konfigurations-Historie via VCS

Ich empfehle, das `etckeeper`-Paket mit `git(1)` zu verwenden, wodurch das komplette `/etc`-Verzeichnis unter VCS-Kontrolle gestellt wird. Das Anwenderhandbuch und eine Anleitung finden Sie in `/usr/share/doc/etckeeper/README.gz`.

Grundsätzlich wird durch das Ausführen von `"sudo etckeeper init"` ein `git`-Depot für `/etc` initialisiert (genau wie in Abschnitt 10.6.5, jedoch mit speziellen hook-Skripten für genauere Setups).

Wenn Sie etwas an Ihrer Konfiguration ändern, können Sie `git(1)` nutzen, um dies aufzuzeichnen. Änderungen werden auch automatisch aufgezeichnet, sobald Sie Paketverwaltungsprogramme laufen lassen.

Tip

Sie können die Änderungs-Historie von `/etc` durchsuchen, indem Sie `"sudo GIT_DIR=/etc/.git gitk"` ausführen; so haben Sie eine tolle Sicht auf neu installierte und entfernte Pakete sowie Paketversions-Änderungen.

9.3 Überwachen, Steuern und Starten von Programmaktivitäten

Programmaktivitäten können mittels spezieller Werkzeuge überwacht und kontrolliert werden:

Tip

Das `procs`-Paket enthält sehr grundlegende Werkzeuge zum Überwachen, Steuern und Starten von Programmaktivitäten. Sie sollten sich mit diesen vertraut machen.

9.3.1 Zeitmessung für einen Prozess

Zeit anzeigen, die der von dem Befehl angestossene Prozess benötigt hat:

Paket	Popcon	Größe	Beschreibung
coreutils	V:891, I:999	17478	nice(1): ein Programm mit veränderter Ablaufpriorität ausführen
bsdutils	V:673, I:999	393	renice(1): die Ablaufpriorität eines laufenden Prozess verändern
procps	V:739, I:999	792	Werkzeuge für das <code>"/proc"</code> -Dateisystem: ps(1), top(1), kill(1), watch(1), ...
psmisc	V:427, I:845	679	Werkzeuge für das <code>"/proc"</code> -Dateisystem: killall(1), fuser(1), peekfd(1), pstree(1)
time	V:15, I:279	82	time(1): ein Programm ausführen und die Zeit ausgeben, während der Ressourcen verbraucht wurden
sysstat	V:161, I:183	1918	sar(1), iostat(1), mpstat(1), ...: Werkzeuge zur Messung der System-Performance unter Linux
isag	V:0, I:3	116	Interaktive grafische Darstellung der Systemaktivität, basierend auf sysstat
lsof	V:391, I:946	451	lsof(8): mit der Option <code>"-p"</code> diejenigen Dateien auflisten, die von einem gerade laufenden Prozess geöffnet wurden
strace	V:16, I:153	2367	strace(1): Systemaufrufe und -signale verfolgen
ltrace	V:1, I:21	363	ltrace(1): Bibliotheksaufrufe verfolgen
xtrace	V:0, I:0	353	xtrace(1): Kommunikation zwischen X11-Client und -Server verfolgen
powertop	V:9, I:217	662	powertop(1): Informationen über verbrauchte Systemleistung
cron	V:805, I:997	263	Prozesse laut dem Ablaufplan des cron(8)-Daemons im Hintergrund ausführen
anacron	V:409, I:482	99	cron-ähnlicher Programmablauf-Planer für Systeme, die nicht 24 Stunden am Tag laufen
at	V:162, I:310	161	at(1) oder batch(1): einen Prozess zu einer bestimmten Zeit oder unterhalb einer bestimmten Systembelastung ausführen

Tabelle 9.7: Liste von Werkzeugen zur Überwachung und Steuerung von Programmaktivitäten

```
# time irgendein_befehl >/dev/null
real    0m0.035s    # Zeit auf der Wanduhr (echte vergangene Zeit)
user    0m0.000s    # Zeit im User-Modus
sys     0m0.020s    # Zeit im Kernel-Modus
```

9.3.2 Die Ablaufpriorität

Der nice-Wert wird verwendet, um die Ablaufpriorität von Prozessen zu steuern.

nice-Wert	Ablaufpriorität
19	niedrigste Prozesspriorität (sparsam)
0	sehr hohe Prozesspriorität für Benutzer
-20	sehr hohe Prozesspriorität für root (nicht sparsam)

Tabelle 9.8: Liste der nice-Werte für die Ablaufpriorität

```
# nice -19 top # sehr sparsam
# nice --20 wodim -v -eject speed=2 dev=0,0 disk.img # sehr schnell
```

Manchmal schadet ein extremer nice-Wert dem System mehr als er ihm nützt. Nutzen Sie diesen Befehl mit Vorsicht.

9.3.3 Der Befehl ps

Der ps(1)-Befehl im Debian-System unterstützt sowohl BSD- wie auch SystemV-Funktionalitäten und hilft dabei, die Prozessaktivitäten (in statischem Zustand) zu beurteilen.

Art	typischer Befehl	Funktionalität
BSD	ps aux	anzeigen von %CPU und %MEM
System V	ps -efH	anzeigen der PPID

Tabelle 9.9: Liste von ps-Befehlen

Nicht mehr vorhandene Zombie-Kindprozesse können Sie über die Prozess-ID des Eltern-Prozesses in dem "PPID"-Feld beenden.

Der Befehl pstree(1) zeigt alle Prozesse in Form eines Hierarchie-Baums an.

9.3.4 Der Befehl top

top(1) auf einem Debian-System hat reichhaltige Funktionalitäten und hilft dabei, Prozesse zu identifizieren, die sich sehr auffällig dynamisch verhalten.

Es ist ein interaktives Programm mit Vollbildschirmmodus. Sie erhalten eine Hilfe zur Nutzung, indem Sie die Taste "h" drücken; mit "q" beenden Sie das Programm.

9.3.5 Dateien auflisten, die von einem Prozess geöffnet wurden

Sie können wie folgt alle Dateien auflisten, die von einem Prozess mit einer bestimmten Prozess-ID (PID), z.B. 1, geöffnet wurden:

```
$ sudo lsof -p 1
```

PID 1 ist normalerweise das init-Programm.

9.3.6 Programmaktivitäten verfolgen

Sie können Programmaktivitäten mit `strace(1)` (für Systemaufrufe und -signale), `ltrace(1)` (für Bibliotheksaufrufe) oder `xtrace(1)` (für die Kommunikation zwischen X11-Client und -Server) verfolgen.

Systemaufrufe des `ls`-Befehls verfolgen Sie z.B. wie folgt:

```
$ sudo strace ls
```

9.3.7 Identifikation von Prozessen, die Dateien oder Sockets verwenden

Auch können Sie Prozesse, die bestimmte Dateien nutzen, mit `fuser(1)` identifizieren, hier z.B. `"/var/log/mail.log"`:

```
$ sudo fuser -v /var/log/mail.log
                USER      PID ACCESS COMMAND
/var/log/mail.log: root      2946 F.... rsyslogd
```

Sie sehen, dass die Datei `"/var/log/mail.log"` von dem Befehl `rsyslogd(8)` zum Schreiben geöffnet wurde.

Prozesse, die bestimmte Sockets verwenden, können Sie ebenfalls mittels `fuser(1)` identifizieren, für `"smtp/tcp"` z.B. wie hier:

```
$ sudo fuser -v smtp/tcp
                USER      PID ACCESS COMMAND
smtp/tcp:       Debian-exim 3379 F.... exim4
```

Jetzt wissen Sie, dass auf Ihrem System `exim4(8)` läuft, um [TCP](#)-Verbindungen zum [SMTP](#)-Port (25) zu bedienen.

9.3.8 Einen Befehl mit festem Intervall wiederholt ausführen

`watch(1)` führt ein Programm wiederholt mit einem festen Intervall aus und zeigt dessen Ausgabe im Vollbildschirmmodus an.

```
$ watch w
```

Hiermit wird alle 2 Sekunden neu angezeigt, wer auf dem System angemeldet ist.

9.3.9 Einen Befehl wiederholt mit einer Schleife über verschiedene Dateien ausführen

Es gibt verschiedene Wege, um einen Befehl zu wiederholen und jedes Mal eine andere Datei aus einer bestimmten Menge von Dateien (hier z.B. ausgewählt über ein glob-Suchmuster auf `"*.ext"`) als Argument zu verwenden.

- Methode mit einer `for`-Schleife in der Shell (Näheres in Abschnitt [12.1.4](#)):

```
for x in *.ext; do if [ -f "$x" ]; then command "$x" ; fi; done
```

- Kombination aus `find(1)` und `xargs(1)`:

```
find . -type f -maxdepth 1 -name '*.ext' -print0 | xargs -0 -n 1 command
```

- `find(1)` mit `"-exec"`-Option und einem Befehl:

```
find . -type f -maxdepth 1 -name '*.ext' -exec command '{}' \;
```

- `find(1)` mit `"-exec"`-Option und einem kurzen Shell-Skript:

```
find . -type f -maxdepth 1 -name '*.ext' -exec sh -c "command '{}'" && echo 'successful' \;
```

Die obigen Beispiele wurden geschrieben, um auch eine korrekte Behandlung von komischen Dateinamen, wie z.B. solche mit Leerzeichen, sicherzustellen. Lesen Sie dazu auch Abschnitt [10.1.5](#); dort finden Sie außerdem weitergehende Verwendungen des Befehls `find(1)`.

9.3.10 Ein Programm aus der grafischen Oberfläche heraus starten

Auf der [Befehlszeile \(command-line interface, CLI\)](#) wird das erste Programm mit dem passenden Namen ausgeführt, das in den durch die Umgebungsvariable `$PATH` definierten Verzeichnissen gefunden wird. Lesen Sie dazu auch Abschnitt [1.5.3](#).

Bei einer zum freedesktop.org-Standard konformen [grafischen Oberfläche \(graphical user interface, GUI\)](#) enthalten die `*.desktop`-Dateien in dem Verzeichnis `/usr/share/applications/` alle nötigen Attribute für die Anzeige der Programmeinträge im grafischen Menü. Näheres finden Sie in Abschnitt [7.2.2](#).

Im folgenden Beispiel definiert die Datei `chromium.desktop` die Attribute für den "Chromium-Webbrowser", z.B. "Name" für den Programmnamen, "Exec" für den Pfad zum auszuführenden Programm mit zugehörigen Argumenten, "Icon" für das verwendete Icon-Symbol usw. (Details finden Sie unter [Desktop Entry Specification](#)):

```
[Desktop Entry]
Version=1.0
Name=Chromium Web Browser
Name[de]=Chromium-Webbrowser
GenericName=Web Browser
Comment=Access the Internet
Comment[de]=Internetzugriff
Exec=/usr/bin/chromium %U
Terminal=false
X-MultipleArgs=false
Type=Application
Icon=chromium
Categories=Network;WebBrowser;
MimeType=text/html;text/xml;application/xhtml_xml;x-scheme-handler/http;x-scheme-handler/https;
StartupWMClass=Chromium
StartupNotify=true
```

Dies ist eine vereinfachte Beschreibung. Die `*.desktop`-Dateien werden wie folgt eingelesen:

Die Desktop-Umgebung setzt die Umgebungsvariablen `$XDG_DATA_HOME` und `$XDG_DATA_DIR`. Unter GNOME3 z.B.:

- `$XDG_DATA_HOME` wird nicht genutzt (der Standardwert aus `$HOME/.local/share` wird verwendet).
- `$XDG_DATA_DIRS` wird auf `/usr/share/ gnome: /usr/local/share: /usr/share/` gesetzt.

Daher werden die Basisverzeichnisse (lesen Sie [XDG Base Directory Specification](#)) und `applications`-Verzeichnisse wie folgt gesetzt:

- `$HOME/.local/share/` → `$HOME/.local/share/applications/`
- `/usr/share/gnome/` → `/usr/share/gnome/applications/`
- `/usr/local/share/` → `/usr/local/share/applications/`
- `/usr/share/` → `/usr/share/applications/`

Die `*.desktop`-Dateien aus diesen `applications`-Verzeichnissen werden in obiger Reihenfolge eingelesen.

Tipp

Ein benutzerdefinierter Menüeintrag für die grafische Oberfläche kann erstellt werden, indem eine `*.desktop`-Datei zum Verzeichnis `$HOME/.local/share/applications/` hinzugefügt wird.

Tipp

Ähnlich dazu verhält es sich, wenn eine `*.desktop`-Datei zum `autostart`-Verzeichnis in diesen Basisverzeichnissen hinzugefügt wird: das in der `*.desktop`-Datei definierte Programm wird beim Start der grafischen Benutzeroberfläche automatisch ausgeführt (z.B. nach dem Anmelden). Lesen Sie dazu [Desktop Application Autostart Specification](#).

Tipp

Ähnlich ist es auch, wenn eine *.desktop-Datei im \$HOME/Desktop-Verzeichnis erstellt wird und die grafische Umgebung die Funktionalität der Programmstarter-Icons (desktop icon launcher) unterstützt: das definierte Programm wird bei Anklicken des Icons ausgeführt. Bitte beachten Sie, dass der eigentliche Name des \$HOME/Desktop-Verzeichnisses letztlich vom Gebietsschema (Locale) abhängt; Näheres dazu finden Sie in xdg-user-dirs-update(1).

9.3.11 Anpassen des zu startenden Programms

Einige Programme starten automatisch weitere Programme. Hier einige Punkte bezüglich der Anpassung dieses Prozesses.

- Menü zur Konfiguration von bevorzugten Anwendungen:
 - GNOME3-Umgebung: "Einstellungen" → "Systemeinstellungen" → "Details" → "Vorgabe-Anwendungen";
 - KDE-Umgebung: "K" → "Control-Center" → "KDE-Komponenten" → "Komponentenauswahl";
 - Iceweasel-Browser: "Bearbeiten" → "Einstellungen" → "Anwendungen";
 - mc(1): "/etc/mc/mc.ext";
- Umgebungsvariablen wie "\$BROWSER", "\$EDITOR", "\$VISUAL" und "\$PAGER" (Näheres in environ(7));
- das update-alternatives(1)-System für Programme wie "editor", "view", "x-www-browser", "gnome-www-browser" und "www-browser" (Näheres in Abschnitt 1.4.7);
- die Inhalte der Dateien "~/.mailcap" und "/etc/mailcap", über die die MIME-Types bestimmten Programmen zugeordnet werden (Näheres in mailcap(5));
- die Inhalte der Dateien "~/.mime.types" und "/etc/mime.types", über die Dateinamenerweiterungen entsprechenden MIME-Types zugeordnet werden (Näheres in run-mailcap(1)).

Tipp

update-mime(8) aktualisiert die "/etc/mailcap"-Datei unter Verwendung von "/etc/mailcap.order" (Näheres in mailcap.order(5)).

Tipp

Das debianutils-Paket stellt sensible-browser(1), sensible-editor(1) und sensible-pager(1) bereit; diese treffen eine sinnvolle Auswahl, welcher Editor, Pager bzw. Webbrowser aufgerufen wird. Ich empfehle, dass Sie diese Shellskripte lesen.

Tipp

Um eine Konsolen-Applikation (hier mutt) unter X als bevorzugte Anwendung aufzurufen, sollten Sie wie folgt eine X-Anwendung erzeugen (in diesem Fall z.B. "/usr/local/bin/mutt-term") und diese als bevorzugt zu startende Anwendung konfigurieren:

```
# cat /usr/local/bin/mutt-term <<EOF
#!/bin/sh
gnome-terminal -e "mutt \${@}"
EOF
chmod 755 /usr/local/bin/mutt-term
```

Signalwert	Signalname	Funktion
1	HUP	Daemon neustarten
15	TERM	normal beenden
9	KILL	Beenden erzwingen

Tabelle 9.10: Liste von häufig verwendeten Signalen für den kill-Befehl

9.3.12 Einen Prozess beenden (kill)

Verwenden Sie `kill(1)`, um über die Prozess-ID einen Prozess zu beenden (oder diesem ein Signal zu senden).

Nutzen Sie `killall(1)` oder `pkill(1)`, um das gleiche über den Befehlsnamen und andere Attribute des Prozesses zu erledigen.

9.3.13 Einmalige Aufgaben planen

Führen Sie den `at(1)`-Befehl wie folgt aus, um eine einmalige Aufgabe zu planen:

```
$ echo 'command -args' | at 3:40 monday
```

9.3.14 Regelmäßige Aufgaben planen

Verwenden Sie `cron(8)`, um regelmäßig wiederkehrende Aufgaben zu planen. Lesen Sie dazu `crontab(1)` and `crontab(5)`.

Sie können Prozesse zur Ausführung durch einen normalen Benutzer, z.B. den Benutzer `foo`, einplanen, indem Sie mit dem Befehl `"crontab -e"` eine `crontab(5)`-Datei namens `"/var/spool/cron/crontabs/foo"` erstellen.

Hier ein Beispiel einer `crontab(5)`-Datei:

```
# /bin/sh verwenden, um Befehle auszuführen, egal was in /etc/passwd steht:
SHELL=/bin/sh
# Jegliche Ausgabe an Paul senden, egal wessen crontab dies ist:
MAILTO=paul
# Minute Stunde Tag-des-Monats Monat Tag-der-Woche befehl (Tag-...-Optionen sind oder- ↔
  verknüpft).
# Jeden Tag um 00:05 ausführen:
5 0 * * * $HOME/bin/daily.job >> $HOME/tmp/out 2>&1
# Am Ersten jedes Monats um 14:15 ausführen -- Ausgabe per Mail an Paul:
15 14 1 * * $HOME/bin/monthly
# An allen Werktagen (1-5) um 22:00 Joe nerven. % für eine neue Zeile, das letzte % für CC:
0 22 * * 1-5 mail -s "Es ist 22 Uhr." joe%Joe,%%Wo sind deine Kinder?%.%%
23 */2 1 2 * echo "Am 01. Febr. um 23 Minuten nach 0 Uhr, 2 Uhr, 4 Uhr ... ausführen"
5 4 * * sun echo "Jeden Sonntag um 04:05 ausführen"
# An jedem ersten Montag des Monats um 03:40 ausführen:
40 3 1-7 * * [ "$(date +%a)" == "Mon" ] && command -args
```

Tipp

Auf Systemen, die nicht ständig laufen, installieren Sie das `anacron`-Paket, um regelmäßige Aufgaben auszuführen, sobald nach entsprechender Maschinen-Einschaltdauer ein festgelegter Intervall abgelaufen ist. Näheres unter `anacron(8)` und `anacrontab(5)`.

Tipp

Geplante Skripte zur Systemwartung können Sie vom `root`-Konto aus wiederholt ausführen, indem Sie solche Skripte in `"/etc/cron.hourly/"`, `"/etc/cron.daily/"`, `"/etc/cron.weekly/"` oder `"/etc/cron.monthly/"` einfügen. Der Ausführungszeitpunkt dieser Skripte kann über `"/etc/crontab"` und `"/etc/anacrontab"` angepasst werden.

9.3.15 Die Alt-SysRq-Tastenkombination

Die Kernel-Option "Magic SysRq key" ([SAK key](#)) bietet Ihnen eine Absicherung gegen Systemfehlfunktionen und ist im Debian-Kernel jetzt standardmäßig aktiviert. Durch Drücken von Alt-SysRq gefolgt von einer der folgenden Tasten können Sie in kritischen Situationen wie durch Zauberei die Kontrolle über das System zurückerlangen:

Taste nach Alt-SysRq	Beschreibung der Aktion
r	Tastatur aus dem Roh-Modus (raw) wiederherstellen nach X-Abstürzen
0	das Konsolen-Loglevel auf 0 ändern, um Fehlermeldungen zu reduzieren
k	alle Prozesse auf der aktuellen virtuellen Konsole durch ein kill zum Beenden zwingen
e	alle Prozesse außer init(8) durch ein SIGTERM versuchen zu beenden
i	alle Prozesse außer init(8) durch ein SIGKILL zum Beenden zwingen
s	alle eingebundenen Dateisysteme synchronisieren (Dateisystempuffer auf Platte schreiben), um Datenverluste zu vermeiden
u	alle eingebundenen Dateisysteme neu einbinden mit Nur-Lese-Berechtigung (umount)
b	das System ohne sync oder umount neu starten (reboot)

Tabelle 9.11: Liste der SAK-Befehlstasten

Tipp

Zum Verständnis obiger Aktionen lesen Sie die Handbuchseiten von `signal(7)`, `kill(1)` und `sync(1)`.

Die Kombination von "Alt-SysRq s", "Alt-SysRq u" und "Alt-SysRq r" funktioniert gut, um aus wirklich bösen Situationen herauszukommen und wieder Zugriff auf eine nutzbare Tastatur zu erlangen, ohne das System herunterfahren zu müssen.

Details finden Sie in `"/usr/share/doc/linux-doc-3.* /Documentation/sysrq.txt.gz"`.



Achtung

Die Alt-SysRq-Funktionalität könnte als Sicherheitsrisiko angesehen werden, da sie normalen Benutzern Zugriff auf Funktionen erlaubt, die eigentlich root-Privilegien verlangen. Durch das Einfügen von `"echo 0 >/proc/sys/kernel/sysrq"` in `"/etc/rc.local"` oder `"kernel.sysrq = 0"` in `"/etc/sysctl.conf"` wird die Alt-SysRq-Funktionalität deaktiviert.

Tipp

Von einem SSH-Terminal u.ä. können Sie die Alt-SysRq-Funktionalität nutzen, indem Sie nach `"/proc/sysrq-trigger"` schreiben. Zum Beispiel werden durch `"echo s > /proc/sysrq-trigger"`; `"echo u > /proc/sysrq-trigger"` alle eingebundenen Dateisysteme **synchronisiert** und mittels **umount** aus der Einbindung gelöst.

9.4 Tipps zur Systempflege

9.4.1 Wer ist/war im System aktiv?

Sie können wie folgt überprüfen, wer derzeit im System aktiv ist:

- `who(1)` zeigt, wer angemeldet ist.

- `w(1)` zeigt, wer angemeldet ist und was derjenige jeweils tut.
- `last(1)` zeigt eine Auflistung der zuletzt angemeldeten Benutzer.
- `lastb(1)` zeigt eine Auflistung der letzten Benutzer mit fehlgeschlagenen Anmeldeversuchen.

Tipp

Diese Informationen werden in `/var/run/utmp` und `/var/log/wtmp` zur Verfügung gehalten. Lesen Sie dazu `login(1)` und `utmp(5)`.

9.4.2 Allen eine Warnung schicken

Mit `wall(1)` können Sie wie folgt eine Nachricht an alle am System angemeldeten Benutzer schicken:

```
$ echo "Wir fahren in 1 Stunde das System herunter." | wall
```

9.4.3 Hardware-Identifikation

Bei [PCI](#)-basierten Geräten ([AGP](#), [PCI-Express](#), [CardBus](#), [ExpressCard](#) usw.) ist `lspci(8)` (eventuell mit der Option `-nn`) ein guter Anfang zur Hardware-Identifikation.

Alternativ können Sie die Hardware auch über das Auslesen der Inhalte von `/proc/bus/pci/devices` identifizieren oder indem Sie die Verzeichnisse unterhalb von `/sys/bus/pci` durchsuchen (Näheres in Abschnitt [1.2.12](#)).

Paket	Popcon	Größe	Beschreibung
pciutils	V:195, I:992	196	Hilfsprogramme für PCI unter Linux: <code>lspci(8)</code>
usbutils	V:84, I:862	324	Hilfsprogramme für USB unter Linux: <code>lsusb(8)</code>
pcmciautils	V:13, I:21	97	Hilfsprogramme für PCMCIA unter Linux: <code>pccardctl(8)</code>
scsitools	V:0, I:3	390	Sammlung von Werkzeugen für das SCSI-Hardware-Management: <code>lsscsi(8)</code>
procinfo	V:0, I:13	135	Systeminformationen, gewonnen aus <code>/proc</code> : <code>lsdev(8)</code>
lshw	V:12, I:94	842	Informationen über die Hardware-Konfiguration: <code>lshw(1)</code>
discover	V:41, I:947	90	System zur Hardware-Identifikation: <code>discover(8)</code>

Tabelle 9.12: Liste von Werkzeugen zur Hardware-Identifikation

9.4.4 Hardware-Konfiguration

Obwohl der größte Teil der Hardware-Konfiguration auf modernen GUI-Arbeitsplatz-Systemen wie GNOME und KDE über entsprechende GUI-Werkzeuge verwaltet werden kann, ist es eine gute Idee, zumindest einige grundlegende Methoden für deren Konfiguration zu kennen.

[ACPI](#) ist ein neueres Rahmenwerk für das Power Management und der Nachfolger für das ältere [APM](#).

Tipp

Die Skalierung der CPU-Frequenz wird auf modernen Systemen durch Kernel-Module wie `acpi_cpufreq` abgewickelt.

Paket	Popcon	Größe	Beschreibung
console-setup	V:137, I:959	411	Werkzeuge für Schriftarten und Tastaturbelegungen auf der Linux-Konsole
x11-xserver-utils	V:282, I:534	511	X-Server-Hilfsprogramme: xset(1), xmodmap(1)
acpid	V:145, I:318	176	Daemon zur Verwaltung von Ereignissen der Advanced-Configuration-and-Power-Interface-(ACPI-)Schnittstelle
acpi	V:17, I:302	45	Werkzeug zur Anzeige von Informationen über ACPI-Geräte
sleepd	V:0, I:0	86	Daemon, mit dem man einen Laptop bei Inaktivität in Ruhemodus schicken kann
hdparm	V:408, I:718	256	Optimierung des Festplattenzugriffs (lesen Sie dazu Abschnitt 9.5.9)
smartmontools	V:134, I:197	2117	Speichersysteme mittels S.M.A.R.T. kontrollieren und überwachen
setserial	V:5, I:9	117	Werkzeugsammlung zur Verwaltung von seriellen Ports
memtest86+	V:1, I:29	2391	Werkzeugsammlung zur Speicher-Hardware-Verwaltung
scsitools	V:0, I:3	390	Werkzeugsammlung zur SCSI -Hardware-Verwaltung
setcd	V:0, I:1	35	Optimierung des Zugriffs auf CD-Laufwerke
big-cursor	I:1	27	größere Maus-Cursor für X

Tabelle 9.13: Liste von Werkzeugen zur Hardware-Konfiguration

9.4.5 System- und Hardware-Zeit

Über folgende Befehle wird die System- und Hardware-Zeit auf MM/DD hh:mm, CCYY (MM - Monat, DD - Tag, hh - Stunde, mm - Minute, CCYY - Jahr) gesetzt:

```
# date MMDDhhmmCCYY
# hwclock --utc --systohc
# hwclock --show
```

Zeiten werden auf einem Debian-System normalerweise in der Lokalzeit angezeigt, aber die Hardware- und Systemzeit verwenden üblicherweise [UTC \(GMT\)](#).

Wenn die Hardware-Zeit (auch als BIOS- oder CMOS-Uhr bezeichnet) auf UTC eingestellt ist, ändern Sie die Einstellung in `"/etc/default/rcS"` auf `"UTC=yes"`.

Mit folgendem Befehl konfigurieren Sie die vom Debian-System verwendete Zeitzone neu:

```
# dpkg-reconfigure tzdata
```

Falls Sie die Systemzeit über das Netzwerk aktualisieren möchten, sollten Sie die Verwendung des [NTP](#)-Dienstes in Erwägung ziehen; dazugehörige Pakete sind `ntp`, `ntpd` und `chrony`.

Tipp

Unter [systemd](#) sollten Sie stattdessen `systemd-timesyncd` für die Netzwerk-Zeitsynchronisation verwenden. Lesen Sie dazu `systemd-timesyncd(8)`.

Hier finden Sie weitere Informationen:

- [Managing Accurate Date and Time HOWTO](#);
- [NTP Public Services Project](#);
- im `ntp-doc`-Paket.

Tipp

`ntptrace(8)` aus dem `ntp`-Paket kann eine Hintereinanderschaltung mehrerer NTP-Server zu deren primärer Quelle zurückverfolgen.

9.4.6 Die Terminal-Konfiguration

Es existieren mehrere Komponenten zur Konfiguration von Systemfunktionalitäten für textbasierte Konsolen und ncurses(3):

- die Datei `"/etc/terminfo/*/*"` (`terminfo(5)`);
- die Umgebungsvariable `"$TERM"` (`term(7)`);
- `setterm(1)`, `stty(1)`, `tic(1)` und `toe(1)`.

Falls bei Anmeldung von fern auf einem Debian-System mit einem Debian-fremden `xterm` der `terminfo`-Eintrag für `xterm` nicht funktioniert, ändern Sie den Terminaltyp (`"$TERM"`) von `"xterm"` auf eine der funktionseingeschränkten Versionen wie `"xterm-r6"`. Näheres dazu finden Sie in `"/usr/share/doc/libncurses5/FAQ"`. `"dumb"` ist der kleinste gemeinsame Nenner für `"$TERM"`.

9.4.7 Die Audio-Infrastruktur

Gerätetreiber für Soundkarten werden in aktuellen Linux-Systemen von [Advanced Linux Sound Architecture \(ALSA\)](#) bereitgestellt. ALSA enthält zwecks Kompatibilität auch einen Emulationsmodus für das ältere [Open Sound System \(OSS\)](#).

Tipp

Verwenden Sie `"cat /dev/urandom > /dev/audio"` oder `speaker-test(1)`, um die Lautsprecher zu testen (^C zum stoppen).

Tipp

Falls Sie keinen Ton hören, sind Ihre Lautsprecher möglicherweise an einem stummgeschalteten Ausgang angeschlossen. Moderne Sound-Systeme haben oft mehrere Ausgänge. `alsamixer(1)` aus dem `alsa-utils`-Paket ist nützlich für die Einstellung von Lautstärke und Stummschaltung.

Anwendungssoftware kann konfiguriert sein, nicht die Soundkarte direkt anzusprechen, sondern über standardisierte Sound-Server-Systeme darauf zuzugreifen.

Paket	Popcon	Größe	Beschreibung
alsa-utils	V:341, I:476	2283	Werkzeuge zur Konfiguration und Nutzung von ALSA
oss-compat	V:2, I:29	20	Paket für OSS-Kompatibilität, verhindert unter ALSA das Auftreten von <code>"/dev/dsp not found"</code> -Fehlern
jackd	V:4, I:27	9	JACK Audio Connection Kit (JACK) -Server (niedrige Latenzzeiten)
libjack0	V:1, I:13	338	JACK Audio Connection Kit (JACK) -Bibliothek (niedrige Latenzzeiten)
nas	V:0, I:0	243	Network Audio System (NAS) -Server
libaudio2	V:60, I:488	165	Network Audio System (NAS) -Bibliothek
pulseaudio	V:350, I:471	6398	PulseAudio -Server, ersetzt ESD
libpulse0	V:289, I:604	969	PulseAudio -Client-Bibliothek, ersetzt ESD
libgstreamer1.0-0	V:372, I:574	5280	GStreamer : GNOME-Sound-Engine
libphonon4	I:121	680	Phonon : KDE-Sound-Engine

Tabelle 9.14: Liste von Audio-Paketen

Es gibt normalerweise für jede populäre Arbeitsplatz-Umgebung eine Sound-Engine. Für jede Sound-Engine, die von einer Anwendung verwendet wird, kann ausgewählt werden, mit welchem der verschiedenen Sound-Server sie sich verbindet.

Umgebung	Befehl
Linux-Konsole	<code>setterm -powersave off</code>
X-Window (Bildschirmschoner ausschalten)	<code>xset s off</code>
X-Window (dpms deaktivieren)	<code>xset -dpms</code>
X-Window (grafische Oberfläche zur Konfiguration des Bildschirmschoners)	<code>xscreensaver-command -prefs</code>

Tabelle 9.15: Liste von Befehlen zur Deaktivierung des Bildschirmschoners

9.4.8 Deaktivieren des Bildschirmschoners

Verwenden Sie folgende Befehle, um den Bildschirmschoner zu deaktivieren:

9.4.9 Ausschalten von Pieptönen

Man kann immer den Stecker des internen PC-Lautsprechers abziehen, um jegliche Pieptöne loszuwerden, aber auch über das Entfernen des `pcspkr`-Kernel-Moduls kann dies erreicht werden.

Folgender Befehl verhindert, dass das `readline(3)`-Programm, das von `bash(1)` genutzt wird, einen Piepton ausgibt, wenn es ein Alarm-Zeichen (ASCII=7) empfängt:

```
$ echo "set bell-style none">> ~/.inputrc
```

9.4.10 Arbeitsspeichernutzung

Es gibt zwei Ressourcen, über die Sie die aktuelle Situation zur Arbeitsspeichernutzung abfragen können:

- Die Boot-Meldungen des Kernels in `/var/log/dmesg` enthalten die exakte Größe des verfügbaren Arbeitsspeichers.
- `free(1)` und `top(1)` zeigen Informationen über Arbeitsspeicher-Ressourcen auf dem laufenden System an.

Hier ein Beispiel:

```
# grep '\] Memory' /var/log/dmesg
[  0.004000] Memory: 990528k/1016784k available (1975k kernel code, 25868k reserved, 931k ↵
data, 296k init)
$ free -k
              total             used             free             shared        buffers         cached
Mem:          997184          976928           20256                0           129592          171932
-/+ buffers/cache:          675404          321780
Swap:        4545576              4          4545572
```

Sie fragen sich vielleicht: "Warum sagt `dmesg` mir, dass 990 MB frei sind, während `free -k` sagt, 320 MB seien frei. Da fehlen mehr als 600 MB ..."

Sorgen Sie sich nicht über den hohen Wert von "used" und die kleine Größe von "free" in der "Mem:"-Zeile; schauen Sie sich stattdessen die Werte darunter an (inklusive Puffer/Cache, 675404 und 321780 in obigem Beispiel) und entspannen Sie sich.

Auf meinem MacBook mit 1GB=1048576k DRAM (das Grafik-System stiehlt einiges davon) sehe ich Folgendes:

9.4.11 Systemsicherheits- und Integritätsüberprüfung

Schlechte Systemwartung könnte Ihr System für Angriffe von extern anfällig machen.

Um eine Systemsicherheits- und Integritätsüberprüfung durchzuführen, sollten Sie mit folgendem beginnen:

gemeldet	Größe
absolute Größe in dmesg	1016784k = 1GB - 31792k
frei in dmesg	990528k
absolut auf der Shell	997184k
frei auf der Shell	20256k (aber effektiv 321780k)

Tabelle 9.16: Liste der gemeldeten Arbeitsspeichergößen

- dem `debsums`-Paket, lesen Sie dazu `debsums(1)` und Abschnitt [2.5.2](#);
- dem `chkrootkit`-Paket, lesen Sie dazu `chkrootkit(1)`;
- der `clamav`-Paketfamilie, lesen Sie dazu `clamscan(1)` und `freshclam(1)`;
- der [Debian Sicherheits-FAQ](#);
- der [Anleitung zum Absichern von Debian](#).

Paket	Popcon	Größe	Beschreibung
logcheck	V:8, I:10	102	Daemon, um Anomalien in den System-Logdateien per Mail an den Administrator zu melden
debsums	V:5, I:42	107	Hilfsprogramm, um installierte Paketdateien anhand von MD5-Prüfsummen zu verifizieren
chkrootkit	V:5, I:24	970	rootkit -Erkennungsprogramm
clamav	V:13, I:58	774	Antiviren-Programm für Unix - Befehlszeilenschnittstelle
tiger	V:2, I:3	7822	Sicherheitslücken im System melden
tripwire	V:2, I:3	11521	Integritätsüberprüfung für Dateien und Verzeichnisse
john	V:2, I:12	452	Programm zum aktiven Knacken von Passwörtern
aide	V:1, I:2	2063	Fortschrittliche Intrusion-Detection-Umgebung - statische Binärdatei
integrit	V:0, I:0	329	Dateiintegritäts-Verifizierungsprogramm
crack	V:0, I:1	149	Programm zum Erraten von Passwörtern

Tabelle 9.17: Liste von Werkzeugen für eine Systemsicherheits- und Integritätsüberprüfung

Hier ein einfaches Skript, um nach Dateien mit typischerweise falschen Dateiberechtigungen (schreibbar für alle) zu suchen:

```
# find / -perm 777 -a \! -type s -a \! -type l -a \! \! -type d -a -perm 1777 \)
```



Achtung

Da das `debsums`-Paket [MD5](#)-Prüfsummen verwendet, die lokal gespeichert sind, kann ihm als Sicherheits-Audit-Werkzeug gegen bösartige Angriffe nicht vollständig vertraut werden.

9.5 Tipps zur Speicherung von Daten

Indem Sie Ihr System mit einer Linux [Live-CD](#) oder einer [Debian-Installer-CD](#) im Rettungsmodus starten, können Sie auf einfache Weise den Speicher auf Ihrem Boot-Laufwerk neu konfigurieren.

9.5.1 Verwendung des Plattenplatzes

Der verwendete Plattenplatz kann mit Programmen aus den Paketen `mount`, `coreutils` und `xdu` beurteilt werden:

- `mount(8)` gibt Informationen über alle eingebundenen Dateisysteme (= "Laufwerke") aus.
- `df(1)` gibt Informationen über den verwendeten Plattenplatz für das Dateisystem aus.
- `du(1)` gibt Informationen über den verwendeten Plattenplatz für den Verzeichnisbaum aus.

Tipp

Mit "`du -k . |xdu`", "`sudo du -k -x / |xdu`" usw. können Sie die Ausgabe von `du(8)` an `xdu(1x)` weiterleiten, um eine grafische und interaktive Präsentation zu erhalten.

9.5.2 Konfiguration der Plattenpartitionen

Bei der Konfiguration von [Laufwerkspartitionen](#) können Sie erwägen, `parted(8)` zu verwenden (obwohl `fdisk(8)` lange als Standard angesehen wurde). Die Begriffe "Disk partitioning data", "Partitionstabelle", "Partition map" und "Disk label" werden alle gleichbedeutend verwendet.

Die meisten PCs verwenden das klassische Schema des [Master Boot Record \(MBR\)](#) und legen die [Partitionstabelle](#) im ersten Sektor, also im [LBA](#)-Sektor 0 (512 Byte) ab.

Anmerkung

Neuere PCs mit [Extensible Firmware Interface \(EFI\)](#), inklusive Intel-basierten Macs, verwenden teilweise das [GUID Partition Table \(GPT\)](#)-Schema, bei dem die [Partitionstabelle](#) nicht im ersten Sektor liegt.

Obwohl `fdisk(8)` lange Zeit das Standardwerkzeug zur Laufwerkspartitionierung war, ist `parted(8)` dabei, ihm den Rang abzulaufen.

Paket	Popcon	Größe	GPT	Beschreibung
util-linux	V:891, I:999	4598	Nicht unterstützt	verschiedene System-Werkzeuge inklusive <code>fdisk(8)</code> und <code>cdisk(8)</code>
parted	V:363, I:561	304	Unterstützt	GNU Parted - Programm zum Anlegen / Ändern von Partitionen
gparted	V:19, I:132	2046	Unterstützt	GNOME Partitionseditor, basierend auf <code>libparted</code>
gdisk	V:278, I:513	852	Unterstützt	Partitionseditor für GPT-Laufwerke
kpartx	V:16, I:29	87	Unterstützt	Programm zur Erzeugung von Gerätezuordnungen (device mappings) für Partitionen

Tabelle 9.18: Liste von Paketen zum Partitions-Management



Achtung

Obwohl `parted(8)` vorgibt, Dateisysteme erzeugen und auch nachträglich in der Größe verändern zu können, ist es sicherer, für solche Dinge gut betreute und spezialisierte Werkzeuge wie `mkfs(8)` (`mkfs.msdos(8)`, `mkfs.ext2(8)`, `mkfs.ext3(8)`, `mkfs.ext4(8)`, ...) und `resize2fs(8)` zu verwenden.

Anmerkung

Um zwischen [GPT](#) und [MBR](#) zu wechseln, müssen Sie die ersten paar Blöcke der Platte löschen (lesen Sie dazu Abschnitt [9.7.6](#)) und "`parted /dev/sdx mklabel gpt`" oder "`parted /dev/sdx mklabel msdos`" nutzen, um die gewünschte Einstellung neu zu setzen. Bitte beachten Sie, dass hierbei "msdos" für [MBR](#) verwendet wird.

9.5.3 Zugriff auf Partitionen über die UUID-Kennung

Obwohl aufgrund der Neukonfiguration einer Partition oder der Aktivierungsreihenfolge von Wechseldatenträgern Laufwerke von einem zum anderen Mal unterschiedliche Laufwerksnamen haben können, gibt es auch eine Möglichkeit, konsistente Bezeichnungen für den Laufwerkszugriff zu verwenden. Dies ist ebenfalls hilfreich, wenn Sie mehrere Festplatten haben und Ihr BIOS die Gerätenamen nicht jedes Mal identisch zuweist.

- `mount(8)` kann über die Option `-U` ein blockorientiertes Gerät mittels seiner [UUID](#)-Kennung einbinden, statt dessen Geräte-dateinamen (wie z.B. `"/dev/sda3"`) zu nutzen.
- `"/etc/fstab"` (lesen Sie hierzu auch `fstab(5)`) kann [UUID](#) verwenden.
- Auch Bootloader (Abschnitt [3.1.2](#)) können unter Umständen die [UUID](#) benutzen.

Tipp

Sie können die [UUID](#) eines blockorientierten Gerätes mit `blkid(8)` abfragen.

Tipp

Geräteknotten von bestimmten Geräten wie Wechseldatenträgern können, falls nötig, auch über [udev-Regeln](#) statisch festgelegt werden, so dass sie sich nicht mehr ändern. Lesen Sie dazu Abschnitt [3.3](#).

9.5.4 LVM2

LVM2 ist ein [Logical Volume Manager](#) für den Linux-Kernel. Mit LVM2 können Partitionen auf logischen Volumes erzeugt werden statt auf physikalischen Festplatten.

LVM erfordert folgendes:

- device-mapper-Unterstützung im Linux-Kernel (Standardeinstellung für Debian-Kernel);
- die Userspace-Bibliothek zur Unterstützung von device-mapper (`libdevmapper*`-Paket);
- die Userspace-LVM2-Werkzeuge (`lvm2`-Paket).

Lehrreiche Informationen über LVM2 finden Sie mittels folgender Handbuchseiten:

- `lvm(8)`: Grundlagen des LVM2-Mechanismus' (Liste aller LVM2-Befehle);
- `lvm.conf(5)`: Konfigurationsdatei für LVM2;
- `lvs(8)`: Informationen über logische Volumes ausgeben;
- `vgs(8)`: Informationen über Volume-Gruppen ausgeben;
- `pvs(8)`: Informationen über physikalische Volumes ausgeben.

9.5.5 Konfiguration von Dateisystemen

Für das [ext4](#)-Dateisystem enthält das `e2fsprogs`-Paket folgendes:

- `mkfs.ext4(8)` zur Erzeugung neuer [ext4](#)-Dateisysteme;
 - `fsck.ext4(8)` zur Überprüfung und Reparatur vorhandener [ext4](#)-Dateisysteme;
 - `tune2fs(8)` zur Konfiguration des Superblocks von [ext4](#)-Dateisystemen;
-

- `debugfs(8)` für interaktive Fehlersuche in [ext4](#)-Dateisystemen (es enthält den `unde l`-Befehl, um gelöschte Dateien wiederherzustellen).

Die Befehle `mkfs(8)` und `fsck(8)` werden durch das `e2fsprogs`-Paket bereitgestellt und sind Frontends für Dateisystem-abhängige Programme (`mkfs.<dateisystemtyp>` und `fsck.<dateisystemtyp>`). Für [ext4](#) sind das `mkfs.ext4(8)` und `fsck.ext4(8)` (dies sind symbolischer Link auf `mke2fs(8)` und `e2fsck(8)`).

Ähnliche Befehle sind für jedes von Linux unterstützte Dateisystem verfügbar.

Paket	Popcon	Größe	Beschreibung
e2fsprogs	V:576, I:999	1449	Hilfsprogramme für ext2/ext3/ext4 -Dateisysteme
reiserfsprogs	V:11, I:29	1132	Hilfsprogramme für das Reiserfs -Dateisystem
dosfstools	V:128, I:524	235	Hilfsprogramme für das FAT -Dateisystem (Microsoft: MS-DOS, Windows)
xfsprogs	V:21, I:98	3191	Hilfsprogramme für das XFS -Dateisystem (SGI: IRIX)
ntfs-3g	V:186, I:512	1479	Hilfsprogramme für das NTFS -Dateisystem (Microsoft: Windows NT, Windows 7 ...)
jfsutils	V:1, I:12	1577	Hilfsprogramme für das JFS -Dateisystem (IBM: AIX, OS/2)
reiser4progs	V:0, I:4	1373	Hilfsprogramme für das Reiser4 -Dateisystem
hfsprogs	V:0, I:8	356	Hilfsprogramme für HFS - und HFS Plus -Dateisysteme (Apple: Mac OS)
btrfs-progs	V:38, I:64	4027	Hilfsprogramme für das Btrfs -Dateisystem
zerofree	V:3, I:94	25	Programm, um freie Blöcke auf ext2/3/4 -Dateisystemen mit Nullen zu überschreiben

Tabelle 9.19: Liste von Paketen für das Dateisystem-Management

Tipp

[Ext4](#) ist das Standard-Dateisystem für Linux-Systeme und Sie sollten dies verwenden, außer Sie haben einen bestimmten Grund, ein anderes zu nutzen.

Tipp

Das [Btrfs](#)-Dateisystem ist ab dem Linux-Kernel 3.2 (Debian wheezy) verfügbar. Man geht davon aus, dass dies nach [ext4](#) das neue Standard-Dateisystem wird.



Warnung

Sie sollten das [Btrfs](#)-Dateisystem nicht für kritische Daten verwenden, solange es noch keine Live-Kernelspace-`fsck(8)`-Funktionalität und Bootloader-Unterstützung enthält.

Tipp

Einige Werkzeuge erlauben den Zugriff auf Dateisysteme ohne entsprechende Unterstützung im Linux-Kernel (lesen Sie dazu Abschnitt [9.7.2](#)).

9.5.6 Dateisystemerzeugung und Integritätsüberprüfung

Der `mkfs(8)`-Befehl erzeugt unter Linux ein Dateisystem. Mit `fsck(8)` führen Sie eine Integritätsüberprüfung oder Reparatur des Dateisystems durch.

Standardmäßig gibt es jetzt auf Debian-Systemen nach der Erzeugung des Dateisystems keinen periodischen Dateisystem-Check (`fsck`) mehr.

**Achtung**

Es ist grundsätzlich nicht sicher, fsck auf **eingebundenen Dateisystemen** laufen zu lassen.

Tipp

Sie können den fsck(8)-Befehl auf allen Dateisystemen inklusive dem root-Dateisystem (/) gefahrlos beim Reboot ausführen, indem Sie "enable_periodic_fsck" in "/etc/mke2fs.conf" setzen und den Max-Wert für die mount-Vorgänge mittels "tune2fs -c0 /dev/<partitionsname>" auf 0 setzen. Details finden Sie in mke2fs.conf(5) und tune2fs(8).

Tipp

In "/var/log/fsck/" finden Sie Ergebnisse von dem fsck(8)-Lauf, der bei jedem Systemstart durch das Boot-Skript durchgeführt wird.

9.5.7 Optimierung von Dateisystemen über mount-Optionen

Die grundlegende statische Dateisystem-Konfiguration wird in "/etc/fstab" festgelegt. Zum Beispiel:

```
# <Dateisystem> <Einbindungspunkt> <Typ> <Optionen> <dump> <pass>
proc /proc proc defaults 0 0
UUID=709cbe4c-80c1-56db-8ab1-dbce3146d2f7 / ext4 noatime,errors=remount-ro 0 1
UUID=817bae6b-45d2-5aca-4d2a-1267ab46ac23 none swap sw 0 0
/dev/scd0 /media/cdrom0 udf,iso9660 user,noauto 0 0
```

Tipp

Die **UUID** (Näheres in Abschnitt 9.5.3) kann statt der normalen Namen für blockorientierte Geräte wie "/dev/sda1", "/dev/sda2" ...verwendet werden, um das Gerät zu identifizieren.

Die Performance und Charakteristik eines Dateisystems kann über mount-Optionen optimiert werden (lesen Sie dazu fstab(5) und mount(8)). Erwähnenswert sind dabei die folgenden:

- "defaults": impliziert die Standardoptionen "rw, suid, dev, exec, auto, nouser, async". (allgemein)
- "noatime" oder "relatime": sehr effektiv, um den Lesezugriff zu beschleunigen. (allgemein)
- "user": erlaubt einem unprivilegierten Benutzer, das Dateisystem einzubinden. Diese Option impliziert die Kombination "noexec, nosuid, nodev". (allgemein, verwendet für CDs oder USB-Speichermedien)
- "noexec, nodev, nosuid": eine Kombination zur Erhöhung der Sicherheit. (allgemein)
- "noauto": beschränkt das Einbinden auf händisch ausgeführte Operationen (kein automatisches Einbinden). (allgemein)
- "data=journal": Option für ext3fs; kann die Datenintegrität bei Stromausfällen erhöhen, allerdings zum Preis einer leichten Verringerung der Schreibgeschwindigkeit.

Tipp

Sie müssen entsprechende Kernel-Bootparameter angeben (lesen Sie dazu Abschnitt 3.1.2), wie z.B. "rootflags=data=journal", um für das root-Dateisystem den Journal-Modus zu aktivieren (standardmäßig nicht aktiv). Für Lenny ist der Standard-Journal-Modus "rootflags=data=ordered". Bei Squeeze ist es "rootflags=data=writeback".

9.5.8 Optimierung von Dateisystemen über den Superblock

Die Charakteristik eines Dateisystems kann über seinen Superblock optimiert werden; verwenden Sie dazu den Befehl `tune2fs(8)`:

- `"sudo tune2fs -l /dev/hda1"` zeigt den Inhalt des Dateisystem-Superblocks auf `"/dev/hda1"` an.
- `"sudo tune2fs -c 50 /dev/hda1"` ändert den Intervall zur Überprüfung des Dateisystems (Ausführung von `fsck` während des Systemstarts) für `"/dev/hda1"` auf jeden 50. Start.
- `"sudo tune2fs -j /dev/hda1"` fügt Journalfunktionalität zum Dateisystem hinzu; das bedeutet, dass ein [ext2](#)-Dateisystem auf `"/dev/hda1"` nach [ext3](#) konvertiert wird. (Führen Sie dies nur bei einem nicht eingebundenen Dateisystem durch.)
- `"sudo tune2fs -O extents,uninit_bg,dir_index /dev/hda1 && fsck -pf /dev/hda1"` konvertiert ein [ext3](#)-Dateisystem auf `"/dev/hda1"` nach [ext4](#). (Führen Sie dies nur bei einem nicht eingebundenen Dateisystem durch.)

Tipp

Ungeachtet seines Namens funktioniert `tune2fs(8)` nicht nur für [ext2](#)-Dateisysteme, sondern auch für [ext3](#) und [ext4](#).

9.5.9 Optimierung der Festplatte



Warnung

Bitte überprüfen Sie Ihre Hardware und lesen Sie die Handbuchseite von `hdparm(8)`, bevor Sie mit der Festplattenkonfiguration herumspielen, da dies ziemlich gefährlich für die Datenintegrität sein kann.

Sie können die Zugriffsgeschwindigkeit einer Festplatte testen, für `"/dev/hda"` z.B. mit `"hdparm -tT /dev/hda"`. Bei einigen Festplatten, die über (E)IDE angeschlossen sind, kann diese über `"hdparm -q -c3 -d1 -u1 -m16 /dev/hda"` erhöht werden; dabei wird "(E)IDE 32-Bit I/O-Unterstützung" aktiviert, außerdem das `"using_dma-Flag"` und das `"interrupt-unmask-Flag"` gesetzt sowie die (gefährliche!) `"Multiple 16 Sector I/O"-Einstellung` aktiviert.

Sie können die Cache-Funktionalität für das Schreiben auf eine Festplatte testen, für `"/dev/sda"` z.B. mit `"hdparm -W /dev/sda"`. Mit `"hdparm -W 0 /dev/sda"` deaktivieren Sie diese Funktion.

Im Falle von Problemen beim Lesen von schlecht gepressten CD-ROMs in modernen Hochgeschwindigkeits-CD-ROM-Laufwerken können Sie diese möglicherweise trotzdem lesen, indem Sie mit `"setcd -x 2"` die Geschwindigkeit herabsetzen.

9.5.10 Optimierung von Solid State Disks

Die Performance und Lebensdauer einer [Solid State Disk \(SSD\)](#) kann wie folgt optimiert werden:

- Verwenden Sie den aktuellsten Linux-Kernel (≥ 3.2).
- Reduzieren Sie Schreibvorgänge für Lesezugriffe.
 - Setzen Sie in `/etc/fstab` die mount-Optionen `"noatime"` oder `"relatime"`.
- Aktivieren Sie den [TRIM](#)-Befehl.
 - Setzen Sie in `/etc/fstab` die mount-Option `"discard"` für [ext4](#)-Dateisysteme, Swap-Partitionen, [Btrfs](#) usw. Lesen Sie dazu `fstab(5)`.
 - Setzen Sie in `/etc/lvm/lvm.conf` die Option `"discard"` für [LVM](#). Lesen Sie dazu `lvm.conf(5)`.
 - Setzen Sie in `/etc/crypttab` die Option `"discard"` für [dm-crypt](#). Lesen Sie dazu `crypttab(5)`.

- Aktivieren Sie das für SSD optimierte Festplatten-Zuordnungsschema.
 - Setzen Sie bei Btrfs-Dateisystemen die mount-Option "ssd" in `/etc/fstab`.
- Konfigurieren Sie Ihren Laptop so, dass die Daten nur alle 10 Minuten aus dem Cache auf die Platte geschrieben werden.
 - Setzen Sie die mount-Option "commit=600" in `/etc/fstab`. Lesen Sie dazu `fstab(5)`.
 - Konfigurieren Sie `pm-utils` so, dass auch bei Betrieb mit Netzteil (an der Steckdose) der Laptop-Modus aktiviert wird. Näheres im [Debian-Fehlerbericht #659260](#).

**Warnung**

Das Ändern des Intervalls zur Übertragung der Daten auf die Disk vom normalen Standardwert 5 Sekunden auf 10 Minuten macht Ihre Daten anfälliger für Datenverluste im Falle eines Stromausfalls.

9.5.11 SMART verwenden, um Festplattenausfälle vorherzusehen

Sie können Ihre Festplatte mit dem `smartd(8)`-Daemon überwachen und protokollieren, sofern diese mit dem [SMART](#)-Standard kompatibel ist:

1. Aktivieren Sie die [SMART](#)-Funktionalität im [BIOS](#).
2. Installieren Sie das Paket `smartmontools`.
3. Identifizieren Sie Ihre Festplatten, indem Sie sie mit `df(1)` auflisten.
 - Wir gehen hier davon aus, dass die zu überwachende Festplatte als `/dev/hda` auftaucht.
4. Überprüfen Sie die Ausgabe von `smartctl -a /dev/hda`, um festzustellen, ob die [SMART](#)-Funktionalität derzeit wirklich aktiv ist.
 - Falls nicht, aktivieren Sie sie mit `smartctl -s on -a /dev/hda`.
5. Aktivieren Sie den `smartd(8)`-Daemon wie folgt:
 - Entfernen Sie das Kommentarzeichen vor `start_smartd=yes` in der Datei `/etc/default/smartmontools`.
 - Führen Sie einen Neustart des `smartd(8)`-Daemons über `sudo /etc/init.d/smartmontools restart` durch.

Tipp

Die Einstellungen des `smartd(8)`-Daemons können über die `/etc/smartd.conf`-Datei angepasst werden; dazu gehört auch die Einstellung, auf welchem Wege Sie über Warnungen informiert werden möchten.

9.5.12 Angeben eines Verzeichnisses für temporäre Dateien über \$TMPDIR

Anwendungen erzeugen temporäre Dateien normalerweise unterhalb des temporären Verzeichnisses `/tmp`. Falls `/tmp` nicht genug freien Speicherplatz bietet, können Sie für Anwendungen, die sich diesbezüglich korrekt verhalten, auch mittels der `$TMPDIR`-Variable festlegen, welches Verzeichnis für solche temporären Daten genutzt werden soll.

9.5.13 Vergrößerung des nutzbaren Speicherplatzes mittels LVM

Partitionen, die bei der Installation über den [Logical Volume Manager \(LVM\)](#) (Linux-Funktionalität) erzeugt wurden, können einfach und ohne größere System-Neukonfiguration in der Größe verändert werden, indem Speicherplatz hinzugefügt oder entfernt wird, und zwar über die Grenzen einzelner Laufwerke hinweg.

9.5.14 Vergrößerung des nutzbaren Speicherplatzes über das Einbinden anderer Partitionen

Wenn Sie eine leere Partition (z.B. `/dev/sdx`) haben, können Sie sie mit `mkfs.ext4(1)` formatieren und dann mit `mount(8)` in ein Verzeichnis einbinden, in dem Sie mehr Platz benötigen (Sie müssen die originalen Daten kopieren):

```
$ sudo mv arbeit alt
$ sudo mkfs.ext4 /dev/sdx
$ sudo mount -t ext4 /dev/sdx arbeit
$ sudo cp -a alt/* arbeit
$ sudo rm -rf alt
```

Tipp

Alternativ können Sie eine leere Festplatten-Image-Datei (lesen Sie dazu Abschnitt [9.6.5](#)) als loop-device einbinden (Näheres dazu in Abschnitt [9.6.3](#)). Die reelle Größe des Festplatten-Images wächst mit den wirklich darin abgelegten Daten.

9.5.15 Vergrößerung des nutzbaren Speicherplatzes, indem ein anderes Verzeichnis mit `mount --bind` eingebunden wird

Wenn Sie ein leeres Verzeichnis (z.B. `/pfad/zu/leer`) auf einer anderen Partition haben, auf der noch Platz frei ist, können Sie dieses Verzeichnis mit der `mount`-Option `--bind` in ein anderes Verzeichnis (z.B. `arbeit`) einbinden, in dem Sie mehr Speicherplatz benötigen:

```
$ sudo mount --bind /pfad/zu/leer arbeit
```

9.5.16 Vergrößerung des nutzbaren Speicherplatzes, indem ein anderes Verzeichnis mit `Overlay-mounting` eingebunden wird

Wenn Sie freien Platz auf einer anderen Partition haben (mit 2 Verzeichnissen wie `/pfad/zu/leer` und `/pfad/zu/arbeit`), können Sie dort ein Verzeichnis erstellen und dieses einem anderen alten Verzeichnis (z.B. `/pfad/zu/alt`) "überstülpen", in dem Sie mehr Platz benötigen. Dies wird ermöglicht durch die [OverlayFS](#)-Funktionalität im Linux-Kernel 3.18 oder neuer (ab Debian Stretch 9.0).

```
$ sudo mount -t overlay overlay \
  -olowerdir=/pfad/zu/alt,upperdir=/pfad/zu/leer,workdir=/pfad/zu/arbeit
```

Hierbei sollten `/pfad/zu/leer` und `/pfad/zu/nutzverzeichnis` auf einer schreibbaren (RW) Partition liegen, um `/pfad/zu/alt` zu überlagern.

9.5.17 Vergrößerung des nutzbaren Speicherplatzes über einen symbolischen Link



Achtung

Diese Methode ist überholt. Manche Software könnte nicht korrekt funktionieren mit "symbolischen Links auf ein Verzeichnis". Verwenden Sie stattdessen einen der oben beschriebenen "mounting"-basierten Ansätze.

Wenn Sie ein leeres Verzeichnis (z.B. `/pfad/zu/leer`) auf einer anderen Partition haben, auf der noch Platz frei ist, können Sie mit `ln(8)` einen symbolischen Link zu einem anderen Verzeichnis erstellen:

```
$ sudo mv arbeit alt
$ sudo mkdir -p /pfad/zu/leer
$ sudo ln -sf /pfad/zu/leer arbeit
$ sudo cp -a alt/* arbeit
$ sudo rm -rf alt
```

**Warnung**

Verwenden Sie solch einen symbolischen Link auf ein Verzeichnis nicht für Verzeichnisse, die von dem System verwaltet werden, wie z.B. `/opt`. Solch ein Link könnte bei einer Systemhochrüstung überschrieben werden.

9.6 Das Festplatten-Abbild

Hier wird die Veränderung eines Festplatten-Abbilds behandelt.

9.6.1 Erzeugung der Festplatten-Abbild-Datei

Die Festplatten-Abbild-Datei (`"disk.img"`) eines nicht eingebundenen Gerätes, z.B. der zweiten SCSI- oder Serial-ATA-Festplatte `"/dev/sdb"`, kann mit `cp(1)` oder `dd(1)` wie folgt erzeugt werden:

```
# cp /dev/sdb disk.img
# dd if=/dev/sdb of=disk.img
```

Ein Abbild des [Master Boot Record \(MBR\)](#), wie er in traditionellen PCs verwendet wird (lesen Sie auch Abschnitt 9.5.2) und der im ersten Sektor der primären IDE-Festplatte abgelegt ist, kann mit `dd(1)` erstellt werden, wie hier:

```
# dd if=/dev/hda of=mbr.img bs=512 count=1
# dd if=/dev/hda of=mbr-nopart.img bs=446 count=1
# dd if=/dev/hda of=mbr-part.img skip=446 bs=1 count=66
```

- `"mbr.img"`: der MBR mit der Partitionstabelle
- `"mbr-nopart.img"`: der MBR ohne Partitionstabelle
- `"mbr-part.img"`: die Partitionstabelle nur vom MBR

Falls Sie eine SCSI- oder Serial-ATA-Festplatte als Boot-Festplatte haben, ersetzen Sie `"/dev/hda"` durch `"/dev/sda"`.

Falls Sie ein Abbild einer einzelnen Partition der Festplatte erstellen möchten, ersetzen Sie `"/dev/hda"` z.B. durch `"/dev/hda1"`.

9.6.2 Direkt auf eine Festplatte schreiben

Die Festplatten-Abbild-Datei `"disk.img"` kann wie folgt auf ein nicht eingebundenes Laufwerk passender Größe (hier z.B. auf die zweite SCSI- oder Serial-ATA-Festplatte `"/dev/sdb"`) geschrieben werden:

```
# dd if=disk.img of=/dev/sdb
```

Ähnlich dazu kann mit folgendem Befehl die Abbild-Datei `"partition.img"` einer einzelnen Partition auf eine nicht eingebundene Partition passender Größe (hier z.B. die erste Partition der zweiten SCSI- oder Serial-ATA-Festplatte `"/dev/sdb1"`) geschrieben werden:

```
# dd if=partition.img of=/dev/sdb1
```

9.6.3 Einbinden der Festplatten-Abbild-Datei

Das Festplatten-Abbild "partition.img", welches ein einfaches Partitions-Abbild enthält, kann mittels einem [loop device](#) wie folgt eingebunden und anschließend wieder gelöst werden:

```
# losetup -v -f partition.img
Loop device is /dev/loop0
# mkdir -p /mnt/loop0
# mount -t auto /dev/loop0 /mnt/loop0
...hack...hack...hack
# umount /dev/loop0
# losetup -d /dev/loop0
```

Das kann noch weiter vereinfacht werden:

```
# mkdir -p /mnt/loop0
# mount -t auto -o loop partition.img /mnt/loop0
...hack...hack...hack
# umount partition.img
```

Jede Partition des Festplatten-Abbilds "disk.img", das mehrere Partitionen enthält, kann mittels [loop device](#) eingebunden werden. Da das loop device standardmäßig keine Partitionen verwaltet, muss es wie folgt zurückgesetzt werden:

```
# modinfo -p loop # Funktionalität des Kernels verifizieren
max_part:Maximum number of partitions per loop device
(auf deutsch: max. Anzahl von Partitionen pro loop device)
max_loop:Maximum number of loop devices
(auf deutsch: max. Anzahl von loop devices)
# losetup -a # Verifizieren, dass das loop device nicht verwendet wird
# rmmod loop
# modprobe loop max_part=16
```

Jetzt kann das loop device bis zu 16 Partitionen verwalten.

```
# losetup -v -f disk.img
Loop device is /dev/loop0
# fdisk -l /dev/loop0
```

```
Disk /dev/loop0: 5368 MB, 5368709120 bytes
255 heads, 63 sectors/track, 652 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x452b6464
```

Device	Boot	Start	End	Blocks	Id	System
/dev/loop0p1		1	600	4819468+	83	Linux
/dev/loop0p2		601	652	417690	83	Linux

```
# mkdir -p /mnt/loop0p1
# mount -t ext4 /dev/loop0p1 /mnt/loop0p1
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/loop0p2 /mnt/loop0p2
...hack...hack...hack
# umount /dev/loop0p1
# umount /dev/loop0p2
# losetup -d /dev/loop0
```

Alternativ können Sie ähnliche Funktionalitäten auch erreichen, indem Sie wie hier die [device-mapper](#)-Geräte nutzen, die von kpartx(8) aus dem kpartx-Paket erzeugt werden:

```
# kpartx -a -v disk.img
...
# mkdir -p /mnt/loop0p2
# mount -t ext4 /dev/mapper/loop0p2 /mnt/loop0p2
```



```
...
...hack...hack...hack
# umount /dev/mapper/loop0p2
...
# kpartx -d /mnt/loop0
```

Anmerkung

Sie können auch eine einzelne Partition eines solchen Festplatten-Abbilds mit einem [loop device](#) einbinden, indem Sie ein Offset verwenden, um den [MBR](#) usw. zu überspringen. Dies ist allerdings fehleranfälliger.

9.6.4 Eine Festplatten-Abbild-Datei bereinigen

Eine Festplatten-Abbild-Datei (hier "disk.img") kann wie folgt von allen gelöschten Dateien bereinigt und in eine gesäuberte kompakte Abbild-Datei "new.img" geschrieben werden:

```
# mkdir old; mkdir new
# mount -t auto -o loop disk.img old
# dd bs=1 count=0 if=/dev/zero of=new.img seek=5G
# mount -t auto -o loop new.img new
# cd old
# cp -a --sparse=always ./ ../new/
# cd ..
# umount new.img
# umount disk.img
```

Wenn "disk.img" auf einem ext2-, ext3- oder ext4-Dateisystem liegt, können Sie auch zerofree(8) aus dem zerofree-Paket verwenden, wie hier gezeigt:

```
# losetup -f -v disk.img
Loop device is /dev/loop3
# zerofree /dev/loop3
# cp --sparse=always disk.img new.img
```

9.6.5 Eine leere Abbild-Datei erstellen

Ein leeres Festplatten-Abbild "disk.img", das bis zu einer Größe von 5 GiB anwachsen kann, erzeugen Sie mit dd(1) wie folgt:

```
$ dd bs=1 count=0 if=/dev/zero of=disk.img seek=5G
```

Mittels eines [loop device](#) erstellen Sie auf diesem Festplatten-Abbild "disk.img" wie folgt ein ext4-Dateisystem:

```
# losetup -f -v disk.img
Loop device is /dev/loop1
# mkfs.ext4 /dev/loop1
...hack...hack...hack
# losetup -d /dev/loop1
$ du --apparent-size -h disk.img
5.0G disk.img
$ du -h disk.img
83M disk.img
```

Die Dateigröße von "disk.img" ist 5.0 GiB, aber der aktuell von ihm verwendete Speicherplatz ist lediglich 83 MiB. Diese Diskrepanz ist möglich, da [ext4](#) die Verwendung von [Sparse-Dateien](#) unterstützt.

Tipp

Der wirklich genutzte Speicherplatz von [Sparse-Dateien](#) wächst mit den Daten, die in diese hineingeschrieben werden.

Die Nutzung von Befehlen ähnlich denen aus Abschnitt 9.6.3 auf Geräten, die von [loop device](#) oder [device-mapper](#)-Geräten erzeugt wurden, erlaubt Ihnen ebenfalls, das Abbild "disk.img" über parted(8) oder fdisk(8) zu erzeugen, sowie Dateisysteme mit mkfs.ext4(8), mkswap(8) usw. zu erstellen.

9.6.6 Erstellen einer ISO9660-Abbild-Datei

Eine [ISO9660](#)-Abbild-Datei "cd.iso" aus einem Quellverzeichnisbaum (hier in "quell_verzeichnis") kann mit genisoimage(1) aus dem [cdrkit](#)-Paket erstellt werden:

```
# genisoimage -r -J -T -V volume_id -o cd.iso quell_verzeichnis
```

Ähnlich dazu kann ein boot-fähiges ISO9660-Abbild "cdboot.iso" aus einem Verzeichnisbaum ("quell_verzeichnis") erstellt werden, der dem des debian-installer ähnelt:

```
# genisoimage -r -o cdboot.iso -V volume_id \  
-b isolinux/isolinux.bin -c isolinux/boot.cat \  
-no-emul-boot -boot-load-size 4 -boot-info-table quell_verzeichnis
```

Hier wird der [Isolinux-Bootloader](#) zum Booten verwendet (Näheres in Abschnitt 3.1.2).

Sie können die Berechnung der md5sum-Prüfsumme und die Erstellung eines ISO9660-Abbilds direkt von der CD-ROM durchführen, wie hier gezeigt:

```
$ isoinfo -d -i /dev/cdrom  
CD-ROM is in ISO 9660 format  
...  
Logical block size is: 2048  
Volume size is: 23150592  
...  
# dd if=/dev/cdrom bs=2048 count=23150592 conv=notrunc,noerror | md5sum  
# dd if=/dev/cdrom bs=2048 count=23150592 conv=notrunc,noerror > cd.iso
```

**Warnung**

Sie müssen wie oben gezeigt dafür sorgen, dass Sie nicht von einem Fehler im Linux-Kernel ("ISO9660 filesystem read ahead bug") betroffen werden; nur so können Sie korrekte Ergebnisse erhalten.

9.6.7 Direkt auf die CD/DVD-R/RW schreiben

Tipp

Eine DVD ist für wodim(1) aus dem [cdrkit](#)-Paket lediglich eine große CD.

Sie finden nutzbare Laufwerke wie folgt:

```
# wodim --devices
```

Dann wird die leere CD-R in das Laufwerk eingelegt und die ISO9660-Abbild-Datei "cd.iso" wird wie folgt mit wodim(1) auf das Laufwerk (hier "/dev/hda") geschrieben:

```
# wodim -v -eject dev=/dev/hda cd.iso
```

Falls statt der CD-R eine CD-RW genutzt wird, verwenden Sie dies:

```
# wodim -v -eject blank=fast dev=/dev/hda cd.iso
```

Tipp

Wenn Ihre Arbeitsplatz-Umgebung CDs automatisch einbindet, lösen Sie die Einbindung mit "sudo umount /dev/hda" über eine Konsole, bevor Sie den wodim(1)-Befehl ausführen.

9.6.8 Einbinden einer ISO9660-Abbild-Datei

Wenn "cd.iso" ein ISO9660-Abbild enthält, können Sie es wie folgt in "/cdrom" einbinden:

```
# mount -t iso9660 -o ro,loop cd.iso /cdrom
```

Tipp

Moderne Arbeitsplatz-Systeme binden Wechseldatenträger (wie mit ISO9660 formatierte CDs) automatisch ein (lesen Sie dazu Abschnitt [10.1.7](#)).

9.7 Binärdaten

Hier behandeln wir das Bearbeiten von Binärdaten auf einem Speichermedium.

9.7.1 Betrachten und Bearbeiten von Binärdaten

Die grundlegendste Methode zum Betrachten von Binärdaten ist die Verwendung des Befehls "od -t x1".

Paket	Popcon	Größe	Beschreibung
coreutils	V:891, I:999	17478	grundlegendes Paket, das od(1) für die Ausgabe von Dateien (HEX, ASCII, OCTAL, ...) enthält
bsdmainutils	V:60, I:996	26	Hilfspaket, das hd(1) für die Ausgabe von Dateien (HEX, ASCII, OCTAL, ...) enthält
hexedit	V:1, I:12	72	Editor und Betrachter für Binärdateien (HEX, ASCII)
bless	V:0, I:4	1028	voll ausgestatteter Hexadezimal-Editor (GNOME)
okteta	V:1, I:15	1508	voll ausgestatteter Hexadezimal-Editor (KDE4)
ncurses-hexedit	V:0, I:2	132	Editor und Betrachter für Binärdateien (HEX, ASCII, EBCDIC)
beav	V:0, I:0	133	Editor und Betrachter für Binärdateien (HEX, ASCII, EBCDIC, OKTAL, ...)

Tabelle 9.20: Liste von Paketen zum Betrachten und Bearbeiten von Binärdaten

Tipp

HEX wird als Acronym für das [Hexadezimal](#)-Format mit einer Basis ([Radix](#)) von 16 verwendet. OKTAL steht für das [Oktal](#)-Format mit einer Basis von 8. ASCII ist der [American Standard Code for Information Interchange](#), also für normalen englischsprachigen Text-Code. EBCDIC steht für den [Extended Binary Coded Decimal Interchange Code](#), der auf [IBM Mainframe](#)-Betriebssystemen verwendet wird.

9.7.2 Manipulieren von Dateien ohne Einbinden der Festplatte

Diese Werkzeuge können Dateien lesen und schreiben, ohne dass dazu die Festplatte eingebunden werden muss:

Paket	Popcon	Größe	Beschreibung
mtools	V:10, I:83	389	Hilfsprogramme zur Bearbeitung von MSDOS-Dateien
hfsutils	V:0, I:7	1884	Hilfsprogramme zur Bearbeitung von HFS- und HFS+-Dateien

Tabelle 9.21: Liste von Paketen zur Manipulation von Dateien ohne Einbinden der Festplatte

9.7.3 Datenredundanz

Software [RAID](#)-Systeme, bereitgestellt durch den Linux-Kernel, bieten Datenredundanz auf Ebene des Kernel-Dateisystems und erreichen so eine sehr hohe Zuverlässigkeit der Datenspeicherung.

Es gibt auch Werkzeuge, die Datenredundanz für Dateien auf Ebene der Anwendungen ermöglichen und so ebenfalls eine sehr zuverlässige Datenspeicherung erlauben:

Paket	Popcon	Größe	Beschreibung
par2	V:4, I:15	271	Parity Archive Volume Set, für die Dateiüberprüfung und -reparatur
dvdaster	V:0, I:2	1741	Schutz vor Datenverlust/Kratzern/Alterung von CD-/DVD-Medien
dvbackup	V:0, I:0	413	Backup-Werkzeug, das MiniDV-Camcorder verwendet (enthält rsbep(1))
vdmfec	V:0, I:0	97	Wiederherstellung verlorener Blöcke mittels Vorwärtsfehlerkorrektur (Forward Error Correction)

Tabelle 9.22: Liste von Werkzeugen, um Redundanz für Dateien hinzuzufügen

9.7.4 Datenwiederherstellung und forensische Analyse

Hier einige Programme für Datenwiederherstellung und forensische Analysen:

Tipp

Sie können gelöschte Dateien auf einem ext2-Dateisystem wiederherstellen, indem Sie `list_deleted_inodes` und den `unde1`-Befehl von `debugfs(8)` (aus dem `e2fsprogs`-Paket) verwenden.

9.7.5 Aufsplitten einer großen in mehrere kleine Dateien

Wenn eine Datei zu groß ist, um sie als einzelne Datei zu sichern, können Sie trotzdem ein Backup davon erstellen, nachdem Sie sie in kleinere, z.B. 2000 MiB große Stücke aufgeteilt haben; später können diese Stücke wieder zur Originaldatei zusammengesetzt werden.

```
$ split -b 2000m grosse_datei
$ cat x* >grosse_datei
```



Achtung

Stellen Sie sicher, dass Sie dort keine anderen Dateien haben, die mit einem "x" beginnen, um Kollisionen bei den Dateinamen zu vermeiden.

Paket	Popcon	Größe	Beschreibung
testdisk	V:3, I:38	1426	Hilfsprogramm zum Scannen von Partitionen und Wiederherstellen von Daten
magicrescue	V:0, I:3	259	Hilfsprogramm zur Dateiwiederherstellung mittels Suche nach "magischen Bytes"
scalpel	V:0, I:4	87	ein sparsamer, sehr leistungsfähiger Datei-Carver
myrescue	V:0, I:3	83	Daten von beschädigten Festplatten retten
extundelete	V:1, I:11	148	Hilfsprogramm, um gelöschte Dateien von ext3-/ext4-Dateisystemen wiederherzustellen
ext4magic	V:0, I:4	233	Hilfsprogramm, um gelöschte Dateien von ext3-/ext4-Dateisystemen wiederherzustellen
ext3grep	V:0, I:3	281	Werkzeug, das bei der Wiederherstellung von gelöschten Dateien auf ext3-Dateisystemen hilft
scrounge-ntfs	V:0, I:3	50	Datenwiederherstellungsprogramm für NTFS-Dateisysteme
gzrt	V:0, I:0	33	Werkzeugsatz für die Wiederherstellung von gzip-Daten
sleuthkit	V:2, I:24	1511	Werkzeuge für forensische Analysen (SleuthKit)
autopsy	V:0, I:2	1027	grafische Oberfläche für SleuthKit
foremost	V:0, I:7	104	forensische Anwendung zur Datenwiederherstellung
guymager	V:0, I:1	1030	forensisches Imaging-Werkzeug, basierend auf Qt
dcfldd	V:0, I:5	106	erweitertes dd für Forensik und Sicherheit

Tabelle 9.23: Liste von Paketen für Datenwiederherstellung und forensische Analysen

9.7.6 Leeren von Dateiinhalten

Zum Leeren des Inhalt einer Datei, z.B. einer Logdatei, verwenden Sie nicht `rm(1)`, um die Datei zu löschen und anschließend eine neue leere Datei zu erstellen, da in der Zeit zwischen den Befehlen möglicherweise Zugriffsversuche auf die Datei erfolgen könnten. Folgender Weg ist der sicherste, um Dateiinhalte zu leeren:

```
$ :>zu_leerende_datei
```

9.7.7 Dummy-Dateien

Folgende Befehle erzeugen Dummy- oder leere Dateien:

```
$ dd if=/dev/zero of=5kb.file bs=1k count=5
$ dd if=/dev/urandom of=7mb.file bs=1M count=7
$ touch zero.file
$ : > alwayszero.file
```

Sie sollten folgende Dateien vorfinden:

- `"5kb.file"` ist eine 5KB große Datei, die Nullen enthält.
- `"7mb.file"` ist eine 7MB große Datei mit zufälligem Inhalt.
- `"zero.file"` könnte eine 0 Byte große Datei sein. Falls sie bereits existierte, wurde ihr `mtime`-Attribut aktualisiert, aber der Inhalt wurde beibehalten.
- `"alwayszero.file"` ist immer eine 0 Byte große Datei. Falls sie bereits existierte, wurde ihr `mtime`-Attribut aktualisiert und der Inhalt entfernt.

9.7.8 Eine vollständige Festplatte löschen

Es gibt mehrere Wege, um alle Daten von einer Festplatte oder einem ähnlichen Gerät zu entfernen, z.B. einem USB-Speicherstick unter `"/dev/sda"`.

**Achtung**

Überprüfen Sie zuerst den Speicherort des USB-Sticks mit `mount(8)`, bevor Sie irgendwelche dieser Befehle ausführen. Das Gerät, auf das `"/dev/sda"` verweist, könnte auch die SCSI- oder Serial-ATA-Festplatte sein, die Ihr komplettes Betriebssystem enthält.

Löschen Sie den kompletten Inhalt der Platte, indem Sie wie folgt die Daten auf 0 setzen:

```
# dd if=/dev/zero of=/dev/sda
```

Löschen Sie alles, indem Sie es wie folgt mit zufälligen Daten überschreiben:

```
# dd if=/dev/urandom of=/dev/sda
```

Löschen Sie alles, indem Sie es wie hier auf sehr effiziente Art mit zufälligen Daten überschreiben:

```
# shred -v -n 1 /dev/sda
```

Da `dd(1)` auf der Shell vieler boot-fähiger Linux-CDs (wie einer Debian-Installer-CD) verfügbar ist, können Sie Ihr installiertes Betriebssystem vollständig entfernen, indem Sie von solch einer CD einen Löschbefehl auf die System-Festplatte (z.B. `"/dev/hda"`, `"/dev/sda"` o.ä.) ausführen.

9.7.9 Einen ungenutzten Bereich einer Festplatte löschen

In einem nicht genutzten Bereich auf einer Festplatte (oder einem USB-Speicherstick), z.B. `"/dev/sdb1"`, könnten gelöschte Daten noch immer vorhanden sein, da beim Löschen lediglich die Verbindung zum Dateisystem entfernt wurde. Diese Daten können durch Überschreiben entfernt werden:

```
# mount -t auto /dev/sdb1 /mnt/foo
# cd /mnt/foo
# dd if=/dev/zero of=junk
dd: writing to 'junk': No space left on device
...
# sync
# umount /dev/sdb1
```

**Warnung**

Dies ist für Ihren USB-Stick normalerweise ausreichend, aber es ist nicht perfekt. Die meisten Teile von gelöschten Dateinamen und deren Attribute könnten versteckt sein und im Dateisystem verbleiben.

9.7.10 Wiederherstellen von gelöschten, aber noch geöffneten Dateien

Wenn Sie versehentlich eine Datei gelöscht haben, die noch von einem Programm verwendet wird (lesend oder schreibend), ist es möglich, diese Datei wiederherzustellen.

Probieren Sie zum Beispiel folgendes:

```
$ echo foo > bar
$ less bar
$ ps aux | grep ' less[ ]'
bozo  4775  0.0  0.0  92200   884 pts/8    S+   00:18   0:00 less bar
$ rm bar
$ ls -l /proc/4775/fd | grep bar
lr-x----- 1 bozo bozo 64 2008-05-09 00:19 4 -> /home/bozo/bar (deleted)
```

```
$ cat /proc/4775/fd/4 >bar
$ ls -l
-rw-r--r-- 1 bozo bozo 4 2008-05-09 00:25 bar
$ cat bar
foo
```

Führen Sie auf einem anderen Terminal (wenn Sie das `lsuf`-Paket installiert haben) folgendes aus:

```
$ ls -li bar
2228329 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:02 bar
$ lsuf |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar
$ rm bar
$ lsuf |grep bar|grep less
less 4775 bozo 4r REG 8,3 4 2228329 /home/bozo/bar (deleted)
$ cat /proc/4775/fd/4 >bar
$ ls -li bar
2228302 -rw-r--r-- 1 bozo bozo 4 2008-05-11 11:05 bar
$ cat bar
foo
```

9.7.11 Alle harten Links suchen

Dateien mit harten Links können mittels `"ls -li"` identifiziert werden:

```
$ ls -li
total 0
2738405 -rw-r--r-- 1 root root 0 2008-09-15 20:21 bar
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 baz
2738404 -rw-r--r-- 2 root root 0 2008-09-15 20:21 foo
```

Sowohl bei `"baz"` wie auch bei `"foo"` zeigt die Anzahl der Links von `"2"` (`>1`) an, dass für sie harte Links existieren. Sie haben beide die gemeinsame [Inode](#)-Nummer `"2738404"`. Das bedeutet, dass dies beides die gleiche hart verlinkte Datei ist. Falls Sie nicht zufällig alle hart verlinkten Dateien finden, können Sie über die [Inode](#) (z.B. `"2738404"`) danach suchen:

```
# find /pfad/zum/einbindungspunkt -xdev -inum 2738404
```

9.7.12 Unsichtbarer Verbrauch von Festplattenplatz

Alle gelöschten, aber noch geöffneten Dateien verbrauchen Festplattenplatz, obwohl dies über einen normalen `du(1)`-Befehl nicht festgestellt werden kann. Sie können wie folgt über ihre Größe aufgelistet werden:

```
# lsuf -s -X / |grep deleted
```

9.8 Tipps zur Datenverschlüsselung

Mit physikalischem Zugriff auf den Rechner kann jeder ganz einfach uneingeschränkte Rechte und Zugriffe auf alle Dateien auf Ihrem PC erlangen (lesen Sie dazu [Abschnitt 4.7.4](#)). Das passwortgeschützte Anmeldesystem ist nicht in der Lage, Ihre Privatsphäre und sensible Daten vor einem möglichen Diebstahl Ihres PCs zu schützen. Dies kann nur durch Verwendung einer Technologie zur Datenverschlüsselung erreicht werden. Obwohl [GNU Privacy Guard](#) (Näheres in [Abschnitt 10.3](#)) Dateien verschlüsseln kann, bedeutet es für den Benutzer einigen Aufwand.

[dm-crypt](#) und [eCryptfs](#) ermöglichen eine automatische Datenverschlüsselung direkt über Linux-Kernel-Module bei minimalem Aufwand für den Benutzer.

Paket	Popcon	Größe	Beschreibung
cryptsetup	V:29, I:78	402	Werkzeug zur Verschlüsselung von blockorientierten Geräten (dm-crypt / LUKS)
cryptmount	V:4, I:5	228	Werkzeuge zur Verschlüsselung von blockorientierten Geräten (dm-crypt / LUKS) mit Fokus auf das Einbinden/Trennen durch normale Benutzer
ecryptfs-utils	V:3, I:5	460	Werkzeuge zur Verschlüsselung von unterlagerten Dateisystemen (eCryptfs)

Tabelle 9.24: Liste von Werkzeugen zur Datenverschlüsselung

[Dm-crypt](#) ist ein kryptografisches Dateisystem, welches [device-mapper](#) verwendet. [Device-mapper](#) bildet ein Blockgerät auf einem anderen ab.

[eCryptfs](#) ist ein anderes kryptografisches System, das ein überlagertes Dateisystem ("stacked filesystem") nutzt. Ein "stacked filesystem" überlagert ein vorhandenes Verzeichnis eines anderen, eingebundenen Dateisystems.

**Achtung**

Datenverschlüsselung kostet CPU-Zeit usw. Bitte wägen Sie Kosten und Nutzen gegeneinander ab.

Anmerkung

Mit dem [debian-installer](#) (Lenny und später) kann ein vollständiges Debian-System auf einer verschlüsselten Festplatte installiert werden; dabei werden [dm-crypt/LUKS](#) und [initramfs](#) verwendet.

Tipp

In Abschnitt [10.3](#) finden Sie Infos über eine Verschlüsselungslösung, die komplett auf Benutzerebene abläuft: [GNU Privacy Guard](#).

9.8.1 Verschlüsselung von Wechseldatenträgern mit dm-crypt/LUKS

Sie können die Inhalte eines Wechseldatenträgers, wie z.B. eines USB-Sticks auf `/dev/sdx`, mit [dm-crypt/LUKS](#) verschlüsseln. Formatieren Sie ihn einfach wie folgt:

```
# badblocks -c 1024 -s -w -t random -v /dev/sdx
# fdisk /dev/sdx
... "n" "p" "1" "return" "return" "w"
# cryptsetup luksFormat /dev/sdx1
...
# cryptsetup open --type luks /dev/sdx1 sdx1
...
# ls -l /dev/mapper/
total 0
crw-rw---- 1 root root 10, 60 2008-10-04 18:44 control
brw-rw---- 1 root disk 254, 0 2008-10-04 23:55 sdx1
# mkfs.vfat /dev/mapper/sdx1
...
# cryptsetup luksClose sdx1
```

Dann kann er in einer modernen Arbeitsplatsumgebung wie ein normaler USB-Stick unter `/media/<datenträger-name>` eingebunden werden (Näheres hierzu in Abschnitt [10.1.7](#)), nur dass dabei nach dem Passwort gefragt wird; in GNOME nutzen Sie dazu `gnome-mount(1)`. Der Unterschied ist, dass jegliche Daten, die auf den Stick geschrieben werden, verschlüsselt sind. Sie können den Stick alternativ auch mit einem anderen Dateisystem formatieren, z.B. als `ext4` mit `mkfs.ext4 /dev/mapper/sdx1`.

Anmerkung

Wenn Sie wirklich paranoid sind, was die Sicherheit Ihrer Daten anbelangt, sollten Sie unter Umständen die Daten auf dem Stick mehrfach überschreiben (mit dem "badblocks"-Befehl aus dem obigen Beispiel). Diese Operation ist allerdings sehr zeitaufwendig.

9.8.2 Verschlüsselte Swap-Partition mit dm-crypt

Wir gehen davon aus, dass Ihre originale `/etc/fstab` folgendes enthält:

```
/dev/sda7 swap sw 0 0
```

Eine verschlüsselte Swap-Partition aktivieren Sie mit [dm-crypt](#) wie folgt:

```
# aptitude install cryptsetup
# swapoff -a
# echo "cswap /dev/sda7 /dev/urandom swap" >> /etc/crypttab
# perl -i -p -e "s/\/dev\/sda7\/dev\/mapper\/cswap/" /etc/fstab
# /etc/init.d/cryptdisks restart
...
# swapon -a
```

9.8.3 Einbinden verschlüsselter Laufwerke mit dm-crypt/LUKS

Eine verschlüsselte Plattenpartition, die mit dm-crypt/LUKS auf `/dev/sdc5` erstellt wurde, kann wie folgt unter `/mnt` eingebunden werden:

```
$ sudo cryptsetup open /dev/sdc5 ninja --type luks
Enter passphrase for /dev/sdc5: ****
$ sudo lvm
lvm> lvscan
inactive          '/dev/ninja-vg/root' [13.52 GiB] inherit
inactive          '/dev/ninja-vg/swap_1' [640.00 MiB] inherit
ACTIVE            '/dev/goofy/root' [180.00 GiB] inherit
ACTIVE            '/dev/goofy/swap' [9.70 GiB] inherit
lvm> lvchange -a y /dev/ninja-vg/root
lvm> exit
Exiting.
$ sudo mount /dev/ninja-vg/root /mnt
```

9.8.4 Automatische Verschlüsselung von Dateien mit eCryptfs

Dateien, die unterhalb von `~/Private/` abgelegt werden, können mit [eCryptfs](#) und dem `ecryptfs-utils`-Paket automatisch verschlüsselt werden:

- Starten Sie `ecryptfs-setup-private(1)` und geben Sie in den folgenden Dialogen `~/Private/` als zu verschlüsselndes Verzeichnis an.
 - Aktivieren Sie `~/Private/`, indem Sie `ecryptfs-mount-private(1)` ausführen.
 - Verschieben Sie sensible Dateien nach `~/Private/` und erstellen Sie symbolische Links, falls erforderlich.
 - Mögliche Kandidaten: `~/ .fetchmailrc`, `~/ .ssh/identity`, `~/ .ssh/id_rsa`, `~/ .ssh/id_dsa` sowie andere Dateien mit den Berechtigungen `go-rwx`.
 - Verschieben Sie Verzeichnisse mit sensiblem Inhalt nach `~/Private/` und erstellen Sie symbolische Links, falls erforderlich.
-

- Mögliche Kandidaten: `~/ .gnupg` sowie andere Verzeichnisse mit den Berechtigungen `go-rwx`.
- Erzeugen Sie einen symbolischen Link von `~/Desktop/Private/` auf `~/Private/`, um Operationen auf der Arbeitsfläche zu erleichtern.
- Deaktivieren Sie `~/Private/`, indem Sie `ecryptfs-umount-private(1)` ausführen.
- Immer wenn Sie Zugriff auf die verschlüsselten Daten benötigen, aktivieren Sie `~/Private/` mittels `ecryptfs-mount-private`.

Tipp

Da [eCryptfs](#) selektiv nur bestimmte, sensible Dateien verschlüsselt, ist die Systembelastung hier viel kleiner als bei der Verwendung von [dm-crypt](#) für das ganze root- oder `/home`-Dateisystem. Es erfordert keine besonderen Anstrengungen zwecks Einrichtung des Speichers auf dem Laufwerk, allerdings können mit eCryptfs nicht alle Dateisystem-Metadaten geheim gehalten werden.

9.8.5 Automatisches Einbinden von eCryptfs

Wenn Sie Ihr Login-Passwort verwenden, um darüber Schlüssel für die Verschlüsselung gesichert abzulegen, können Sie eCryptfs automatisch mittels [PAM \(Pluggable Authentication Modules\)](#) einbinden.

Fügen Sie folgende Zeile vor `pam_permit.so` in `/etc/pam.d/common-auth` ein:

```
auth required pam_ecryptfs.so unwrap
```

Fügen Sie folgende Zeile als letzte Zeile in `/etc/pam.d/common-session` ein:

```
session optional pam_ecryptfs.so unwrap
```

Fügen Sie folgende Zeile als erste aktive Zeile in `/etc/pam.d/common-password` ein:

```
password required pam_ecryptfs.so
```

Dies ist ziemlich bequem.

**Warnung**

Konfigurationsfehler in [PAM](#) könnten Sie aus Ihrem eigenen System aussperren. Weiteres hierzu in Kapitel 4.

**Achtung**

Wenn Sie Ihr Login-Passwort verwenden, um darüber Schlüssel für die Verschlüsselung gesichert abzulegen, sind Ihre verschlüsselten Daten nur so sicher wie Ihr Login-Passwort. Ihre Daten unterliegen so dem Risiko, dass jemand nach dem Diebstahl Ihres Laptops Software zum [Knacken von Passwörtern](#) darauf los läßt; dieses Risiko kann nur durch die sorgfältige Wahl eines [starken Passworts](#) minimiert werden (lesen Sie dazu Abschnitt 4.7.4).

9.9 Der Kernel

Debian stellt für unterstützte Architekturen modulare [Linux-Kernel](#) als Pakete bereit.

9.9.1 Linux-Kernel 2.6/3.x

Es gibt einige erwähnenswerte Funktionalitäten im Linux-Kernel 2.6/3.x verglichen mit Version 2.4:

- Gerätedateien werden durch das udev-System erzeugt (Näheres in Abschnitt 3.3).
- Für Lese-/Schreibzugriffe auf IDE-CD/DVD-Laufwerke wird nicht mehr das `ide-scsi`-Modul verwendet.
- Netzwerk-Paketfilter-Funktionen nutzen die `iptables`-Kernel-Module.

Der Versionssprung von Linux 2.6.39 auf Linux 3.0 ist nicht durch gravierende technologische Änderungen begründet, sondern durch den 20. Geburtstag von Linux.

9.9.2 Kernel-Parameter

Viele Linux-Funktionalitäten sind wie folgt über Kernel-Parameter konfigurierbar:

- Kernel-Parameter, die durch den Bootloader initialisiert werden (Näheres in Abschnitt 3.1.2);
- Kernel-Parameter, die durch `sysctl(8)` zur Laufzeit geändert werden und die über `sysfs` erreichbar sind (Näheres in Abschnitt 1.2.12);
- Modul-Parameter, die über Argumente von `modprobe(8)` gesetzt werden, wenn ein Modul aktiviert wird (Näheres in Abschnitt 9.6.3).

Lesen Sie `"kernel-parameters.txt(.gz)"` und andere zugehörige Dokumente in der Linux-Kernel-Dokumentation (`"usr/share/doc/linux-kernel-*/Documentation"`) bereitgestellt durch das `linux-doc-3.*`-Paket.

9.9.3 Kernel-Header

Die meisten **normalen** Programme benötigen keine Kernel-Header (Kopfteil) und könnten im Gegenteil sogar gestört werden, wenn Sie sie direkt zum Kompilieren verwenden würden. Sie sollten stattdessen gegen die Header in `"usr/include/linux"` und `"usr/include/asm"` aus dem Paket `libc6-dev` kompiliert werden (diese werden auf einem Debian-System aus dem `glibc`-Quellpaket erzeugt).

Anmerkung

Um einige Kernel-spezifische Programme wie Kernel-Module aus externen Linux-Quellen oder den automounter-Daemon (`amd`) zu kompilieren, müssen Sie den Pfad zu den entsprechenden Kernel-Headern auf der Befehlszeile mit angeben. `module-assistant(8)` (oder seine Kurzform `m-a`) hilft Benutzern, auf einfache Art Modulpakete für einen oder mehrere Kernel zu bauen und zu installieren.

9.9.4 Kompilieren des Kernels und dazugehöriger Module

Debian hat seine eigene Methode zur Kompilierung des Kernels und zugehöriger Module.

Wenn Sie eine `initrd` wie in Abschnitt 3.1.2 verwenden, lesen Sie unbedingt die entsprechenden Informationen in `initramfs-tools(8)`, `update-initramfs(8)`, `mkinitramfs(8)` und `initramfs.conf(5)`.



Warnung

Setzen Sie in den Verzeichnissen Ihres Quellcode-Baums (z.B. `"usr/src/linux"`) keine symbolischen Links auf `"usr/include/linux"` and `"usr/include/asm"` (einige veraltete Dokumentationen empfehlen dies).

Paket	Popcon	Größe	Beschreibung
build-essential	I:499	20	Pakete, die zum Bauen von Debian-Paketen essentiell nötig sind: make, gcc, ...
bzip2	V:157, I:970	122	Werkzeuge zum Komprimieren und Dekomprimieren von bz2-Dateien
libncurses5-dev	I:116	6	Entwickler-Bibliothek und Dokumentation für ncurses
git	V:305, I:478	35040	git: vom Linux-Kernel verwendetes verteiltes Versionsverwaltungssystem
fakeroot	V:35, I:521	228	bietet eine fakeroot-Umgebung, um Pakete als nicht-root-Benutzer zu bauen
initramfs-tools	V:371, I:989	112	Werkzeug zur Erzeugung eines initramfs (Debian-spezifisch)
dkms	V:70, I:219	294	Dynamic Kernel Module Support (DKMS) (generisch)
devscripts	V:9, I:57	2623	Helfer-Skripte für einen Debian-Paketbetreuer (Debian-spezifisch)

Tabelle 9.25: Liste von Schlüsselpaketen für die Neukompilierung des Kernels auf einem Debian-System

Anmerkung

Um den aktuellsten Linux-Kernel auf einem Debian-Stable-System zu kompilieren, könnte die Verwendung von rückportierten aktuellen Werkzeugen aus Unstable nötig sein.

Anmerkung

[Dynamic Kernel Module Support \(DKMS\)](#) ist ein neues distributions-unabhängiges Rahmenwerk, das entwickelt wurde, um die Aktualisierung einzelner Kernel-Module ohne Austausch des kompletten Kernels zu ermöglichen. Dies wird verwendet für die Betreuung von Modulen außerhalb des Linux-Quellcode-Baums. Auch ist es damit sehr einfach, im Zuge der Hochrüstung des Kernels Module neu zu bauen.

9.9.5 Kompilieren des Kernel-Quellcodes: Empfehlung des Debian-Kernel-Teams

Um eigene Kernel-Binärpakete aus den Upstream-Kernel-Quelltexten zu erstellen, sollten Sie das angebotene "deb-pkg"-Target nutzen:

```
$ sudo apt-get build-dep linux
$ cd /usr/src
$ wget http://www.kernel.org/pub/linux/kernel/v3.11/linux-<version>.tar.bz2
$ tar -xjvf linux-<version>.tar.bz2
$ cd linux-<version>
$ cp /boot/config-<version> .config
$ make menuconfig
...
$ make deb-pkg
```

Tipp

Das Paket linux-source-<version> stellt den Linux-Kernel-Quelltext inklusive Debian-Patches als "/usr/src/linux-<version>.tar.bz2" bereit.

Um spezifische Binärpakete aus dem Debian-Kernel-Quellpaket zu bauen, sollten Sie die Targets "binary-arch_<architecture>" in "debian/rules.gen" verwenden:

```
$ sudo apt-get build-dep linux
$ apt-get source linux
$ cd linux-3.*
$ fakeroot make -f debian/rules.gen binary-arch_i386_none_686
```

Hier finden Sie weitere Informationen:

- Debian Wiki: [KernelFAQ](#)
- Debian Wiki: [DebianKernel](#)
- Debian Linux Kernel Handbook: <https://kernel-handbook.debian.net>

9.9.6 Hardware-Treiber und Firmware

Der Hardware-Treiber ist Code, der auf dem Zielsystem läuft. Die meisten Hardware-Treiber sind heutzutage als freie Software verfügbar und in den normalen Debian-Kernel-Paketen im `main`-Bereich des Debian-Archivs enthalten.

- [GPU](#)-Treiber
 - Intel GPU-Treiber (`main`)
 - AMD/ATI GPU-Treiber (`main`)
 - NVIDIA GPU-Treiber (`main` für den [nouveau](#)-Treiber, `non-free` für Treiber, die vom Hersteller als Nur-Binär-Variante bereitgestellt werden)
- [Softmodem](#)-Treiber
 - `martian-modem`- und `sl-modem-dkms`-Pakete (`non-free`)

Die Firmware ist Code, der auf das Gerät geladen wird (z.B. CPU-[Mikrocode](#), Rendering-Code, der auf der GPU läuft, oder [FPGA](#)-/[CPLD](#)-Daten ...). Einige Firmware-Pakete gibt es als freie Software, aber viele davon werden nicht als freie Software bereitgestellt, da sie Binärdaten ohne den zugehörigen Quelltext enthalten.

- `firmware-linux-free` (`main`)
- `firmware-linux-nonfree` (`non-free`)
- `firmware-linux-*` (`non-free`)
- `*-firmware` (`non-free`)
- `intel-microcode` (`non-free`)
- `amd64-microcode` (`non-free`)

Bitte beachten Sie, dass Pakete aus `non-free` und `contrib` nicht Teil des Debian-Systems sind. Die Konfiguration, um den Zugriff auf die Bereiche `non-free` und `contrib` zu aktivieren oder zu deaktivieren, ist in Abschnitt [2.1.4](#) beschrieben. Sie sollten sich etwaiger negativer Auswirkungen bei der Verwendung von Paketen aus `non-free` und `contrib` bewusst sein; lesen Sie hierzu auch Abschnitt [2.1.5](#).

9.10 Virtualisierte Systeme

Die Verwendung eines virtualisierten Systems ermöglicht es uns, mehrere Instanzen eines Systems gleichzeitig auf einer einzigen Hardware laufen zu lassen.

Tipp

Lesen Sie dazu auch <http://wiki.debian.org/SystemVirtualization>.

9.10.1 Virtualisierungswerkzeuge

Neben dem einfachen [chroot](#) gibt es in Debian eine Vielzahl von Paketen betreffend [System-Virtualisierung](#) und [Emulation](#).

Paket	Popcon	Größe	Beschreibung
schroot	V:7, I:10	2708	spezialisiertes Werkzeug, um Debian-Binärpakete in chroot-Umgebungen auszuführen
sbuild	V:1, I:4	286	Werkzeug, um Debian-Binärpakete aus Debian-Quellen zu bauen
pbuilder	V:2, I:16	966	persönlicher Paketersteller für Debian-Pakete
debootstrap	V:6, I:63	298	Programm zum Bootstrap eines grundlegenden Debian-Systems (geschrieben in sh)
cdebootstrap	V:0, I:3	116	Programm zum Bootstrap eines Debian-Systems (geschrieben in C)
virt-manager	V:10, I:42	2298	Virtual Machine Manager : Arbeitsplatz-Anwendung zur Verwaltung von virtuellen Maschinen
libvirt-clients	V:43, I:62	1167	Programme für die libvirt -Bibliothek
bochs	V:0, I:1	7194	Bochs : IA-32 PC-Emulator
qemu	I:34	94	QEMU : schneller generischer Prozessor-Emulator
qemu-system	I:21	95	QEMU : Binärdateien zur Emulation eines vollständigen Systems
qemu-user	V:0, I:13	89671	QEMU : Binärdateien für User-Mode-Emulation
qemu-utils	V:11, I:107	6083	QEMU : Hilfsprogramme
qemu-kvm	V:10, I:61	107	KVM : vollständige Virtualisierungslösung auf x86-Hardware mit hardware-unterstützter Virtualisierung
virtualbox	V:12, I:16	106495	VirtualBox : x86-Virtualisierungslösung auf i386 und amd64
xen-tools	V:0, I:4	727	Werkzeuge zur Verwaltung von virtuellen Debian- XEN -Servern
wine	V:19, I:82	192	Wine : Windows-API-Implementierung (Standard-Programm-Suite)
dosbox	V:2, I:18	2742	DOSBox : x86-Emulator mit Tandy-/Herc-/CGA-/EGA-/VGA-/SVGA-Grafik, Audioausgabe und DOS
dosemu	V:0, I:2	4891	DOSEMU : der Linux-DOS-Emulator
vzctl	V:0, I:1	1112	OpenVZ Server-Virtualisierungslösung - Steuerungs-Werkzeuge
vzquota	V:0, I:1	236	OpenVZ Server-Virtualisierungslösung - Quota-Werkzeuge
lxc	V:10, I:15	18761	Linux-Container verwenden Werkzeuge im User-Space-Bereich

Tabelle 9.26: Liste von Virtualisierungswerkzeugen

Der Wikipedia-Artikel [Comparison of platform virtual machines](#) enthält detaillierte Gegenüberstellungen der verschiedenen Plattform-Virtualisierungslösungen.

9.10.2 Arbeitsablauf bei Virtualisierung

Anmerkung

Einige der hier beschriebenen Funktionalitäten sind nur in Squeeze oder späteren Veröffentlichungen verfügbar.

Anmerkung

Standard-Debian-Kernel unterstützen [KVM](#) seit Lenny.

Ein typischer Arbeitsablauf für eine [Virtualisierung](#) enthält folgende Schritte:

- Erzeugen eines leeren Dateisystems (ein Verzeichnisbaum oder ein Festplatten-Image);
 - Ein Verzeichnisbaum kann über `mkdir -p /path/to/chroot` erzeugt werden.

- Eine rohe (leere) Image-Datei kann mittels `dd(1)` erstellt werden (lesen Sie dazu Abschnitt 9.6.1 und Abschnitt 9.6.5).
- `qemu-img(1)` kann verwendet werden, um zu **QEMU** kompatible Image-Dateien zu erzeugen.
- Rohe Image-Dateien und solche im **VMDK**-Format sind weit verbreitet und können bei verschiedenen Virtualisierungslösungen eingesetzt werden.
- Einbinden des Festplatten-Images in das Dateisystem mit `mount(8)` (optional);
 - Bei einer rohen Image-Datei verwenden Sie zum Einbinden ein **loop device** oder **Device Mapper**-Geräte (Näheres in Abschnitt 9.6.3).
 - Festplatten-Images, die von **QEMU** unterstützt werden, binden Sie als **Network Block Device** ein (lesen Sie dazu Abschnitt 9.10.3).
- Bestücken des Zielsystems mit den benötigten Systemdaten;
 - Die Nutzung von Programmen wie `debootstrap` und `cdebootstrap` hilft Ihnen bei diesem Schritt (Details dazu in Abschnitt 9.10.4).
 - Verwenden Sie die Installationsroutinen anderer Betriebssysteme für das in der Emulation laufende System.
- Ausführen eines Programms in der virtualisierten Umgebung;
 - **chroot** bietet eine grundlegende virtualisierte Umgebung, die zur Kompilierung von Programmen, Ausführung von Konsolenanwendungen sowie Daemons ausreichende Funktionalitäten hat.
 - **QEMU** stellt eine CPU-Emulation quer über verschiedene Plattformen zur Verfügung.
 - **QEMU** mit **KVM** bietet eine vollständige Systememulation mit **hardware-unterstützter Virtualisierung**.
 - **VirtualBox** bietet eine vollständige System-Emulation auf i386 und amd64 mit oder ohne **hardware-unterstützter Virtualisierung**.

9.10.3 Einbinden des virtuellen Festplatten-Images

Wenn Sie eine rohe Image-Datei verwenden, finden Sie die nötigen Informationen in Abschnitt 9.6.

Bei anderen Dateiformaten für virtuelle Festplatten-Images können Sie `qemu-nbd(8)` verwenden, um diese über das **Network Block Device**-Protokoll zu exportieren; dann können sie mittels dem `nbd`-Kernelmodul in das Dateisystem eingebunden werden (`mount`).

`qemu-nbd(8)` unterstützt Festplattenformate, die auch von **QEMU** unterstützt werden. **QEMU** wiederum unterstützt folgende Formate: roh (raw), `qcow2`, `qcow`, `vmdk`, `vdi`, `bochs`, `cow` (User-Mode-Linux Copy-on-Write), `parallels`, `dmg`, `cloop`, `vpc`, `vvfat` (virtuelles VFAT) und `host_device`.

Das **Network Block Device** unterstützt Partitionen auf die gleiche Art wie das **loop device** (Näheres hierzu in Abschnitt 9.6.3). Sie können die erste Partition von `"disk.img"` wie folgt einbinden:

```
# modprobe nbd max_part=16
# qemu-nbd -v -c /dev/nbd0 disk.img
...
# mkdir /mnt/part1
# mount /dev/nbd0p1 /mnt/part1
```

Tipp

Eventuell möchten Sie lediglich die erste Partition von `"disk.img"` exportieren; geben Sie dazu für `qemu-nbd(8)` die Option `"-P 1"` an.

9.10.4 Chroot-System

chroot(8) bietet die grundlegendsten Möglichkeiten, um verschiedene Instanzen einer GNU/Linux-Umgebung simultan und ohne Neustart auf dem gleichen System laufen zu lassen.

**Achtung**

Die hier angegebenen Beispiele gehen davon aus, dass sowohl das Elternsystem wie auch das chroot-System die gleiche CPU-Architektur haben.

Sie können lernen, wie man chroot(8) einrichtet und verwendet, indem Sie das pbuilder(8)-Programm wie folgt in einer script(1)-Sitzung laufen lassen:

```
$ sudo mkdir /sid-root
$ sudo pbuilder --create --no-targz --debug --buildplace /sid-root
```

Sie sehen jetzt, wie debootstrap(8) oder cdebootstrap(1) Systemdaten für eine sid-Umgebung in `"/sid-root"` ablegt.

Tipp

Diese debootstrap(8)- oder cdebootstrap(1)-Befehle werden verwendet, um [Debian mit dem Debian-Installer zu installieren](#). Sie können ebenfalls genutzt werden, um Debian auf einem System zu installieren, ohne dabei eine Debian-Installations-CD zu verwenden, sondern stattdessen den Installationsvorgang von einer anderen, bereits auf dem System laufenden GNU/Linux-Distribution zu starten.

```
$ sudo pbuilder --login --no-targz --debug --buildplace /sid-root
```

Hier sehen Sie nun, wie eine System-Shell in einer sid-Umgebung erstellt wird:

1. Lokale Konfiguration kopieren (`"/etc/hosts"`, `"/etc/hostname"`, `"/etc/resolv.conf"`);
2. `"/proc"`-Dateisystem einbinden;
3. `"/dev/pts"`-Dateisystem einbinden;
4. `"/usr/sbin/policy-rc.d"` erzeugen; wird immer mit einem Rückgabewert von 101 beendet;
5. Ausführen von `"chroot /sid-root bin/bash -c 'exec -a -bash bin/bash'"`.

Anmerkung

Einige Programme könnten unter chroot Zugriff auf mehr Dateien des Elternsystems erfordern, als pbuilder bereitstellt. Zum Beispiel könnte es nötig sein, `"/sys"`, `"/etc/passwd"`, `"/etc/group"`, `"/var/run/utmp"`, `"/var/log/wtmp"` usw. über bind-mount einzubinden oder zu kopieren.

Anmerkung

Die Datei `"/usr/sbin/policy-rc.d"` verhindert, dass Daemon-Programme automatisch auf dem Debian-System gestartet werden. Lesen Sie hierzu `"/usr/share/doc/sysv-rc/README.policy-rc.d.gz"`.

Tipp

Der ursprüngliche Ansatz des spezialisierten chroot-Pakets pbuilder ist, ein chroot-System anzulegen und ein Paket innerhalb des chroot zu bauen. Es ist ein ideales System, um zu überprüfen, ob die Bauabhängigkeiten eines Pakets korrekt sind und sicherzustellen, dass in dem so erstellten Paket keine unnötigen oder falschen Bauabhängigkeiten existieren.

Tipp

Ähnlich dazu gibt Ihnen das `schroot`-Paket eine Idee davon, ein i386-chroot-System auf einem amd64-Elternsystem laufen zu lassen.

9.10.5 System mit mehrfachen Arbeitsplatzumgebungen

Ich empfehle Ihnen, [QEMU](#) oder [VirtualBox](#) auf einem Debian-Stable-System zu verwenden, um mittels [Virtualisierung](#) mehrere Arbeitsplatzumgebungen (Desktops) sicher auf einem System zu betreiben. So ist es möglich, Desktop-Anwendungen aus Debian Unstable und Testing ohne die üblichen damit verbundenen Risiken laufen zu lassen.

Da reines [QEMU](#) sehr langsam ist, wird empfohlen, es mit [KVM](#) zu beschleunigen, falls das Host-System dies unterstützt.

Ein Image einer virtuellen Festplatte "virtdisk.qcow2", das ein Debian-System für [QEMU](#) enthält, kann mittels [debian-installer: Kleine CDs](#) wie folgt erzeugt werden:

```
$ wget http://cdimage.debian.org/debian-cd/5.0.3/amd64/iso-cd/debian-503-amd64-netinst.iso
$ qemu-img create -f qcow2 virtdisk.qcow2 5G
$ qemu -hda virtdisk.qcow2 -cdrom debian-503-amd64-netinst.iso -boot d -m 256
...
```

Weitere Tipps finden Sie unter [Debian Wiki: QEMU](#).

[VirtualBox](#) enthält grafische, Qt-basierte Werkzeuge und ist ziemlich intuitiv. Seine Benutzeroberfläche und die Befehlszeilenwerkzeuge sind im [VirtualBox User Manual \(html\)](#) bzw. [VirtualBox User Manual \(PDF\)](#) beschrieben.

Tipp

Andere GNU/Linux-Distributionen wie [Ubuntu](#) und [Fedora](#) mittels [Virtualisierung](#) laufen zu lassen ist eine tolle Möglichkeit, Tipps zur Konfiguration zu bekommen. Auch andere proprietäre Betriebssysteme können über diese GNU/Linux-[Virtualisierung](#) bequem zum Laufen gebracht werden.

Kapitel 10

Datenmanagement

Hier werden Werkzeuge und Tipps zur Verwaltung von Binär- und Textdateien auf einem Debian-System beschrieben.

10.1 Austauschen, kopieren und archivieren von Dateien

**Warnung**

Nicht-koordinierte zeitgleiche Schreibzugriffe auf Geräte und Dateien durch mehrere Prozesse sind nicht erlaubt, da dadurch eine [Race Condition](#) (Wettlaufsituation um konkurrierende Zugriffe auf die Ressource) entstehen könnte. Dies kann über einen [File locking](#)-Mechanismus mittels `flock(1)` umgangen werden.

Die Sicherheit der Daten und deren kontrolliertes Austauschen mit anderen hat verschiedene Aspekte:

- Erzeugung von Datenarchiven;
- Fern-Speicherzugriff;
- Vervielfältigung;
- Nachverfolgung der Änderungshistorie;
- Erleichterung des Tauschens von Daten;
- Verhinderung von unerlaubten Dateizugriffen;
- Erkennung von unerlaubten Dateiveränderungen.

Diese können über die Verwendung einer Kombination von Werkzeugen realisiert werden:

- Archivierungs- und Kompressionswerkzeuge
 - Kopier- und Synchronisationswerkzeuge
 - Netzwerk-Dateisysteme;
 - Wechseldatenträger;
 - Secure Shell;
 - Authentifizierungssysteme;
 - Versionskontrollsysteme;
 - Hash- und kryptographische Verschlüsselungswerkzeuge.
-

10.1.1 Archivierungs- und Kompressionswerkzeuge

Hier eine Zusammenfassung von Archivierungs- und Kompressionswerkzeugen im Debian-System:

**Warnung**

Setzen Sie nicht die "\$TAPE"-Variable, außer Sie sind sich über die Folgen im klaren. Sie verändern dadurch das Verhalten von tar(1).

Anmerkung

Ein gezipptes tar(1)-Archiv verwendet die Dateierweiterung ".tgz" oder ".tar.gz".

Anmerkung

Ein xz-komprimiertes tar(1)-Archiv verwendet die Dateierweiterung ".txz" oder ".tar.xz".

Anmerkung

Die Popularität der Kompressionsmethoden in FOSS-Programmen wie tar(1) hat sich mit der Zeit wie folgt verschoben: gzip → bzip2 → xz.

Anmerkung

cp(1), scp(1) und tar(1) könnten Einschränkungen bei speziellen Dateien haben. cpio(1) ist dabei erheblich vielseitiger.

Anmerkung

cpio(1) wurde entwickelt, um zusammen mit find(1) und anderen Befehlen verwendet zu werden und ist geeignet, Backup-Skripte zu erstellen, da der Teil des Skriptes zur Dateiauswahl eigenständig getestet werden kann.

Anmerkung

Die interne Struktur von Libreoffice-Dateien entspricht der von ".jar"-Dateien, die auch mit unzip geöffnet werden können.

Anmerkung

Das de-facto plattform-übergreifende Archivwerkzeug ist zip. Benutzen Sie es in der Art "zip -rX", um maximal mögliche Kompatibilität zu erreichen. Verwenden Sie auch die Option "-s", wenn die maximale Dateigröße bei Ihnen relevant ist.

10.1.2 Kopier- und Synchronisationswerkzeuge

Hier eine Zusammenfassung von einfachen Kopier- und Backup-Werkzeugen im Debian-System:

Das Kopieren von Dateien mit rsync(8) bietet mehr Funktionalitäten als mit anderen Werkzeugen:

- Delta-Transfer-Algorithmus, der nur die Unterschiede zwischen Quelle und Ziel überträgt;
 - schneller Prüf-Algorithmus, der (standardmäßig) nach Dateien sucht, die sich in Größe oder Änderungs-Zeitstempel verändert haben;
-

Paket	Popcon	Größe	Erweiterung	Befehl	Erläuterung
tar	V:905, I:999	3098	.tar	tar(1)	Standard-Archivierungs-Programm (De-Facto-Standard)
cpio	V:412, I:998	1136	.cpio	cpio(1)	Unix-Archivier-Programm im System-V-Stil, zu verwenden mit find(1)
binutils	V:164, I:678	97	.ar	ar(1)	Archivier-Programm zur Erstellung von statischen Bibliotheken
fastjar	V:2, I:29	183	.jar	fastjar(1)	Archivier-Programm für Java (zip-artig)
pax	V:13, I:26	170	.pax	pax(1)	neues POSIX-Archivier-Programm, Kompromiss zwischen tar und cpio
gzip	V:883, I:999	245	.gz	gzip(1) , zcat(1) , ...	GNU LZ77 -Kompressionswerkzeug (De-Facto-Standard)
bzip2	V:157, I:970	122	.bz2	bzip2(1) , bzip2cat(1) , ...	Burrows-Wheeler Block-Sorting Kompressions -Werkzeug mit höherem Kompressionsverhältnis als gzip(1) (langsamer als gzip mit ähnlicher Syntax)
lzma	V:2, I:29	149	.lzma	lzma(1)	LZMA -Kompressionswerkzeug mit höherem Kompressionsverhältnis als gzip(1) (überholt)
xz-utils	V:454, I:977	612	.xz	xz(1) , xzdec(1) , ...	XZ -Kompressionswerkzeug mit höherem Kompressionsverhältnis als bzip2(1) (langsamer als gzip , aber schneller als bzip2 ; ersetzt das LZMA -Kompressionswerkzeug)
p7zip	V:89, I:464	987	.7z	7zr(1) , p7zip(1)	7-Zip -Dateiarchivier-Programm mit hohem Kompressionsverhältnis (LZMA -Kompression)
p7zip-full	V:113, I:486	4664	.7z	7z(1) , 7za(1)	7-Zip -Dateiarchivier-Programm mit hohem Kompressionsverhältnis (LZMA - und andere Kompressionsalgorithmen)
lzop	V:9, I:76	164	.lzo	lzop(1)	LZO -Kompressionswerkzeug mit höherer Kompressions- und Dekompressionsgeschwindigkeit als gzip(1) (niedrigeres Kompressionsverhältnis als gzip mit ähnlicher Syntax)
zip	V:51, I:432	608	.zip	zip(1)	InfoZIP : DOS-Archivierungs- und Kompressionswerkzeug
unzip	V:154, I:798	566	.zip	unzip(1)	InfoZIP : DOS-Dearchivierungs- und Dekompressionswerkzeug

Tabelle 10.1: Liste von Archivierungs- und Kompressionswerkzeugen

Paket	Popcon	Größe	Werkzeug	Funktion
coreutils	V:891, I:999	17478	GNU cp	Dateien und Verzeichnisse lokal kopieren (verwenden Sie "-a" für rekursive Operationen)
openssh-client	V:803, I:996	4298	scp	Dateien und Verzeichnisse von fern kopieren (Client, verwenden Sie "-r" für rekursive Operationen)
openssh-server	V:690, I:834	1567	sshd	Dateien und Verzeichnisse von fern kopieren (ferner Server)
rsync	V:281, I:560	677	-	1-Weg-Synchronisation und -Backup von fern
unison	V:4, I:17	14	-	2-Wege-Synchronisation und -Backup von fern

Tabelle 10.2: Liste von Kopier- und Synchronisationswerkzeugen

- die Optionen `--exclude` und `--exclude-from`, vergleichbar mit denen von `tar(1)`;
- die Syntax des Schrägstrichs hinter dem Quellverzeichnis, die verhindert, dass eine zusätzliche Verzeichnisebene auf dem Zielort erstellt wird.

Tipp

Die Nutzung des in Abschnitt 10.2.3 erwähnten Skripts für statische Datenarchive sollte mit der Option `-gl` über `cron(8)` eine ähnliche Funktionalität bieten wie `Plan9s dumpfs`.

Tipp

Versionskontrollsysteme (VCS) wie die in Tabelle 10.11 können ebenfalls als Mehrwege-Kopier- und Synchronisationswerkzeuge dienen.

10.1.3 Aufrufe für Archivierungsoperationen

Hier finden Sie mehrere Wege, um den kompletten Inhalt des Verzeichnisses `./source` mit verschiedenen Werkzeugen zu archivieren oder dearchivieren.

GNU `tar(1)`:

```
$ tar -cvJf archive.tar.xz ./source
$ tar -xvJf archive.tar.xz
```

Alternativ auch:

```
$ find ./source -xdev -print0 | tar -cvJf archive.tar.xz --null -F -
```

`cpio(1)`:

```
$ find ./source -xdev -print0 | cpio -ov --null > archive.cpio; xz archive.cpio
$ zcat archive.cpio.xz | cpio -i
```

10.1.4 Aufrufe für Kopieroperationen

Hier finden Sie mehrere Wege, um den kompletten Inhalt des Verzeichnisses `./source` mit verschiedenen Werkzeugen zu kopieren.

- Lokale Kopie: `./source`-Verzeichnis → `/dest`-Verzeichnis
- Kopie auf einen fernen Rechner: `./source`-Verzeichnis auf dem lokalen Rechner → `/dest`-Verzeichnis auf dem Rechner `user@host.dom`

`rsync(8)`:

```
# cd ./source; rsync -aHAXSv . /dest
# cd ./source; rsync -aHAXSv . user@host.dom:/dest
```

Sie können auch die Syntax mit Schrägstrich am Ende des Zielverzeichnisses nutzen:

```
# rsync -aHAXSv ./source/ /dest
# rsync -aHAXSv ./source/ user@host.dom:/dest
```

Alternativ auch:

```
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . /dest
# cd ./source; find . -print0 | rsync -aHAXSv0 --files-from=- . user@host.dom:/dest
```

GNU cp(1) und openSSH scp(1):

```
# cd ./source; cp -a . /dest
# cd ./source; scp -pr . user@host.dom:/dest
```

GNU tar(1):

```
# (cd ./source && tar cf - . ) | (cd /dest && tar xvpf - )
# (cd ./source && tar cf - . ) | ssh user@host.dom '(cd /dest && tar xvpf - )'
```

cpio(1):

```
# cd ./source; find . -print0 | cpio -pvdm --null --sparse /dest
```

Sie können in allen Beispielen, die einen "." enthalten, diesen "." durch "foo" ersetzen, um Dateien aus dem Verzeichnis "./source/foo" in das Verzeichnis "/dest/foo" zu kopieren.

Auch kann in allen Beispielen, die einen "." enthalten, dieser "." durch einen absoluten Pfad wie "/pfad/zur/source/foo" ersetzt werden, damit kann dann auf "cd ./source;" verzichtet werden. Dadurch werden, abhängig von den verwendeten Werkzeugen, die Dateien an unterschiedliche Orte kopiert:

- bei rsync(8), GNU cp(1) und scp(1): nach "/dest/foo";
- bei GNU tar(1) und cpio(1): nach "/dest/pfad/zur/source/foo".

Tipp

rsync(8) und GNU cp(1) unterstützen die Option "-u", um Dateien zu überspringen, die am Zielort neuer sind als im Quellverzeichnis.

10.1.5 Aufrufe für die Auswahl von Dateien

find(1) wird verwendet, um Dateien für Archivierungs- und Kopierbefehle auszuwählen (lesen Sie Abschnitt 10.1.3 und Abschnitt 10.1.4) oder xargs(1) (Näheres in Abschnitt 9.3.9). Dies kann mit deren Befehlsargumenten noch erweitert werden.

Die grundsätzliche Syntax von find(1) kann wie folgt zusammengefasst werden:

- Seine bedingten Argumente werden von links nach rechts ausgewertet.
 - Die Auswertung wird beendet, sobald ihr Resultat ermittelt wurde.
 - Ein "logisches **ODER**" (definiert über "-o" zwischen den Bedingungen) hat eine niedrigere Priorität als ein "logisches **UND**" (das über "-a" oder nichts zwischen den Bedingungen definiert wird).
 - Ein "logisches **NICHT**" (definiert über "!" vor der Bedingung) hat eine höhere Priorität als ein "logisches **UND**".
 - "-prune" liefert immer ein logisches **WAHR** zurück und, falls es ein Verzeichnis ist, wird die Suche nach Dateien an diesem Punkt beendet.
 - "-name" findet Dateien über den Anfang des Dateinamens mittels Shell-Glob-Suchmuster (lesen Sie dazu Abschnitt 1.5.6), findet sie aber über Metazeichen wie "*" und "?" auch bei einem führenden "." (neue [POSIX](#)-Funktionalität).
 - "-regex" findet Dateien mit vollständigem Pfad standardmäßig über Emacs-artige reguläre Ausdrücke (**BRE**, nähere Infos finden Sie in Abschnitt 1.6.2).
 - "-size" findet Dateien basierend auf der Dateigröße (mit einem "+" vor dem Wert für größer, mit einem "-" für kleiner).
-

- `"-newer"` findet Dateien, die neuer sind als in dem Argument angegeben.
- `"-print0"` liefert immer ein logisches **WAHR** zurück und gibt den kompletten Dateinamen ([abgeschlossen durch ein Nullzeichen](#)) auf der Standardausgabe aus.

`find(1)` wird oft in einem idiomatischen Stil verwendet, wie hier:

```
# find /pfad/zur \
  -xdev -regextype posix-extended \
  -type f -regex ".*\.cpio|.*~" -prune -o \
  -type d -regex ".*\/\.git" -prune -o \
  -type f -size +99M -prune -o \
  -type f -newer /pfad/zur/zeitstempel -print0
```

Das bedeutet folgendes:

1. nach allen Dateien suchen, beginnend in `"/pfad/zur"`;
2. die Suche global auf das Dateisystem beschränken, in dem sie begonnen wurde, und stattdessen reguläre Ausdrücke (**ERE**, lesen Sie dazu Abschnitt [1.6.2](#)) verwenden;
3. Dateien ausschließen, auf die die regulären Ausdrücke `.*\.cpio` oder `.*~` zutreffen, indem die Suche abgebrochen wird;
4. Verzeichnisse ausschließen, auf die der reguläre Ausdruck `.*\/\.git` zutrifft, indem die Suche abgebrochen wird;
5. Dateien ausschließen, die größer als 99 Megabyte (1048576 Byte) sind, indem die Suche abgebrochen wird;
6. Dateinamen ausgeben, die obige Suchkriterien erfüllen und neuer als `"/path/to/timestamp"` sind.

Bitte beachten Sie die idiomatische Verwendung von `"-prune -o"`, um in obigen Beispielen Dateien auszuschließen.

Anmerkung

Auf [Unix-artigen](#) Nicht-Debian-Systemen werden einige Optionen von `find(1)` unter Umständen nicht unterstützt. Versuchen Sie in diesem Fall, die Suchmethoden anzupassen und `"-print0"` durch `"-print"` zu ersetzen. Unter Umständen müssen Sie auch zugehörige Befehle anpassen.

10.1.6 Archivierungsmedien

Wenn Sie [Speichermedien](#) für die Sicherung wichtiger Daten suchen, sollten Sie sorgfältig deren Einschränkungen abwägen. Für eine kleine persönliche Datensicherung verwende ich CD-R und DVD-R von einem Markenhersteller und lagere die Disks in einer kühlen, dunklen, trockenen und sauberen Umgebung. (Für die professionelle Nutzung scheinen Tapes (Speicherbänder) zur Archivierung sehr beliebt zu sein.)

Anmerkung

[Ein feuerbeständiger Tresor](#) ist gedacht für Dokumente in Papierform. Jedoch haben die meisten Speichermedien eine niedrigere Temperatortoleranz als Papier. Ich baue auf mehrere sicher verschlüsselte Kopien, die an verschiedenen sicheren Orten aufbewahrt werden.

Optimistische Angabe der Lebensdauer von Speichermedien, gefunden im Internet (überwiegend Herstellerangaben):

- 100 Jahre und mehr: säurefreies Papier mit Tinte;
 - 100 Jahre: Optische Speichermedien (CD/DVD, CD-R/DVD-R);
 - 30 Jahre: Magnetische Speichermedien (Bänder, Disketten);
-

- 20 Jahre: Optische Medien basierend auf Phasenänderung (CD-RW).

Hierbei sind keine mechanischen Ausfälle durch Handhabung usw. berücksichtigt.

Optimistische Angabe von Schreibzyklen der Speichermedien, gefunden im Internet (überwiegend Herstellerangaben):

- 250000 Zyklen und mehr: Festplattenlaufwerk;
- 10000 Zyklen und mehr: Flash-Speicher;
- 1000 Zyklen: CD-RW/DVD-RW;
- 1 Zyklus: CD-R/DVD-R, Papier.

**Achtung**

Die obigen Angaben zu Lebensdauer und Schreibzyklen sollten nicht für Entscheidungen bezüglich kritischer Datenspeicherung herangezogen werden. Bitte konsultieren Sie in diesem Fall die spezifischen Produktinformationen des Herstellers.

Tipp

Da CD-R/DVD-R und Papier nur einmal beschrieben werden können, schützen sie von Natur aus vor dem versehentlichen Datenverlust durch Überschreiben. Dies ist ein Vorteil!

Tipp

Wenn Sie eine schnelle und wiederholte Sicherung großer Datenmengen benötigen, könnte eine Festplatte in einem fernen Rechner, verbunden über eine schnelle Internetverbindung, die einzige realistische Option sein.

10.1.7 Wechseldatenträger

Wechseldatenträger können folgende Geräte sein:

- [USB-Flash-Speicher](#) (USB-Stick);
- [Festplattenlaufwerk](#);
- [Optisches Laufwerk](#) (CD/DVD/BD);
- Digitalkamera;
- Digitaler Mediaplayer.

Sie können über eine dieser Möglichkeiten verbunden sein:

- [USB](#);
- [IEEE 1394 / FireWire](#);
- [PC Card](#).

Moderne Arbeitsplatzumgebungen wie GNOME und KDE können diese Wechseldatenträger auch ohne einen entsprechenden `"/etc/fstab"`-Eintrag automatisiert einbinden.

- Das `udisks`-Paket enthält einen Daemon und dazugehörige Hilfsprogramme, um diese Datenträger automatisch einzubinden und zu trennen.
-

- [D-bus](#) erzeugt Ereignisse, um automatische Prozesse anzustoßen.
- [PolicyKit](#) stellt die erforderlichen Berechtigungen bereit.

Tipp

Automatisch eingebundene Geräte haben eventuell die mount-Option "uhelper=", die von umount(8) genutzt wird.

Tipp

In modernen Arbeitsplatzumgebungen funktioniert das automatische Einbinden von Laufwerken nur, wenn diese Geräte nicht in "/etc/fstab" aufgelistet sind.

Der Einbindungspunkt wird in modernen Umgebungen als "/media/<laufwerksbezeichnung>" abgebildet; die Laufwerksbezeichnung (manchmal auch als Disk-Label bezeichnet) kann wie folgt angepasst werden:

- mlabel(1) bei FAT-Dateisystemen;
- genisoimage(1) mit der Option "-V" bei ISO9660-Dateisystemen;
- tune2fs(1) mit der Option "-L" bei ext2-/ext3-/ext4-Dateisystemen.

Tipp

Die verwendete Zeichenkodierung muss unter Umständen als mount-Option angegeben werden (lesen Sie dazu Abschnitt [8.4.6](#)).

Tipp

Die Verwendung des grafischen GUI-Menüs zur Trennung eines eingebundenen Dateisystems könnte auch dessen dynamisch erzeugte Gerätedatei (z.B. "/dev/sdc" entfernen. Falls Sie diese Gerätedatei erhalten möchten, trennen Sie die Einbindung mit dem umount(8)-Befehl von einem Shell-Prompt.

10.1.8 Dateisystemauswahl für den Datenaustausch

Um Daten mit anderen Systemen über Wechseldatenträger auszutauschen, sollten Sie diese mit einem [Dateisystem](#) formatieren, das von beiden Systemen unterstützt wird.

Tipp

Details zum plattformübergreifenden Datenaustausch mit Verschlüsselung auf Geräteebe­ne finden Sie in Abschnitt [9.8.1](#).

Das FAT-Dateisystem wird von nahezu allen modernen Dateisystemen unterstützt und ist für den Datenaustausch über Wechseldatenträger sehr nützlich.

Wenn Sie Geräte wie externe Festplatten für den plattformübergreifenden Datenaustausch mit dem FAT-Dateisystem formatieren, sollten die folgenden Varianten eine sichere Wahl sein:

- Partitionieren Sie das Medium mit fdisk(8), cfdisk(8) oder parted(8) (lesen Sie dazu Abschnitt [9.5.2](#)) mit einer einzigen primären Partition und markieren Sie sie wie folgt:
 - Typ "6" (FAT16) für Medien kleiner als 2 GB;
 - Typ "c" (FAT32, LBA) für größere Medien.
-

Dateisystem	Beschreibung des typischen Anwendungsszenarios
FAT12	Plattformübergreifender Datenaustausch mittels Diskette (<32 MiB)
FAT16	Plattformübergreifender Datenaustausch mittels kleiner festplatten-ähnlicher Geräte (<2 GiB)
FAT32	Plattformübergreifender Datenaustausch mittels großer festplatten-ähnlicher Geräte (<8 TiB, unterstützt von Systemen neuer als MS Windows95 OSR2)
NTFS	Plattformübergreifender Datenaustausch mittels großer festplatten-ähnlicher Geräte (nativ unterstützt von MS Windows NT und späteren Versionen, außerdem unterstützt durch NTFS-3G via FUSE unter Linux)
ISO9660	Plattformübergreifender Austausch statischer Daten mittels CD-R und DVD+/-R
UDF	inkrementelles Schreiben von Daten auf CD-R und DVD+/-R (neu)
MINIX-Dateisystem	speicherplatz-effizientes Ablegen von Unix-Daten auf Diskette
ext2-Dateisystem	Datenaustausch mit älteren Linux-Systemen auf festplatten-ähnlichen Geräten
ext3-Dateisystem	Datenaustausch mit älteren Linux-Systemen auf festplatten-ähnlichen Geräten
ext4-Dateisystem	Datenaustausch mit aktuellen Linux-Systemen auf festplatten-ähnlichen Geräten

Tabelle 10.3: Liste von Dateisystemen für Wechseldatenträger mit typischen Anwendungsszenarien

- Formatieren der primären Partition mit `mkfs.vfat(8)` wie folgt:
 - einfach über den Gerätenamen, z.B. `"/dev/sda1"` für FAT16;
 - über die explizite Option und den Gerätenamen, z.B. `"-F 32 /dev/sda1"` für FAT32.

Wenn das FAT- oder ISO9660-Dateisystem für den Dateiaustausch verwendet wird, sollte folgendes eine sichere Variante sein:

- archivieren der Dateien in eine Archivdatei mittels `tar(1)` oder `cpio(1)`, um die langen Dateinamen, symbolischen Links, originalen Unix-Dateiberechtigungen und Benutzerinformationen zu erhalten;
- splitten der Archivdatei in Stücke kleiner als 2 GiB mittels `split(1)`, um so die Beschränkung der Dateigröße zu umgehen;
- verschlüsseln der Archivdatei, um den Inhalt vor unberechtigtem Zugriff zu schützen.

Anmerkung

Bei dem FAT-Dateisystem liegt, begründet durch sein Design, die maximale Dateigröße bei $(2^{32} - 1)$ Byte = $(4 \text{ GiB} - 1 \text{ Byte})$. Bei einigen Anwendungen auf älteren 32-Bit-Betriebssystemen war die maximale Dateigröße sogar noch kleiner: $(2^{31} - 1)$ Byte = $(2 \text{ GiB} - 1 \text{ Byte})$. Debian ist von letzterem nicht betroffen.

Anmerkung

Microsoft selbst empfiehlt FAT nicht für Laufwerke oder Partitionen über 200 MB Größe. Microsoft hebt die Nachteile wie ineffiziente Speicherplatznutzung in seiner "[Übersicht über die Dateisysteme FAT, HPFS und NTFS](#)" hervor. Natürlich sollten wir für Linux normalerweise das ext4-Dateisystem nutzen.

Tipp

Mehr Informationen über Dateisysteme und Dateisystemzugriffe finden Sie im "[Filesystems HOWTO](#)".

10.1.9 Datenaustausch über das Netzwerk

Wenn Sie Daten mit anderen Systemen über das Netzwerk austauschen, sollten Sie allgemein gängige Dienste verwenden. Hier einige Hinweise:

Obwohl Dateisysteme, die über solche Netzwerk-gestützten Transfermethoden eingebunden sind, für den Datenaustausch sehr praktisch sind, könnten Sie unsicher sein. Die genutzte Netzwerkverbindung muss wie folgt abgesichert sein:

Netzwerkdienst	Beschreibung des typischen Anwendungsszenarios
SMB/CIFS - über Netzwerk mit Samba eingebundenes Dateisystem	Datenaustausch über "Microsoft Windows Network", Näheres in smb.conf(5) und The Official Samba 3.x.x HOWTO and Reference Guide oder im samba-doc -Paket
NFS - über Netzwerk mittels Linux-Kernel eingebundenes Dateisystem	Datenaustausch über "Unix/Linux Network", Näheres in exports(5) und im Linux NFS-HOWTO
HTTP -Dienst	Datenaustausch zwischen Web-Server/-Client
HTTPS -Dienst	Datenaustausch zwischen Web-Server/-Client mit verschlüsseltem Secure Sockets Layer (SSL) oder Transport Layer Security (TLS)
FTP -Dienst	Datenaustausch zwischen FTP-Server/-Client

Tabelle 10.4: Liste von Netzwerkdiensten mit typischen Anwendungsszenarien

- verschlüsseln Sie sie mit [SSL/TLS](#);
- tunneln Sie sie via [SSH](#);
- tunneln Sie sie via [VPN](#);
- schränken Sie den Zugriff über eine Firewall ein.

Lesen Sie auch Abschnitt [6.10](#) und Abschnitt [6.11](#).

10.2 Datensicherung und -wiederherstellung

Wir alle wissen, dass Computer manchmal defekt sein können oder menschliche Fehler System- und Datenausfälle verursachen. Aktionen zur Datensicherung und -wiederherstellung sind unverzichtbarer Teil einer erfolgreichen Systemadministration. Alle möglichen Fehlervarianten werden Sie eines Tages ereilen.

Tipp

Halten Sie Ihr Datensicherungssystem einfach und führen Sie häufig Sicherungen durch. Aktuelle Sicherungen von seinen Daten zu haben ist wichtiger als die technische Qualität der Sicherungsmethodik.

Es gibt drei Schlüsselfaktoren, die letztendlich die Sicherungs- und Wiederherstellungsstrategie bestimmen:

1. Zu wissen, was man sichern und wiederherstellen muss:
 - Daten, die direkt von Ihnen selbst erstellt wurden: Daten in `"~/`";
 - Daten, die von Anwendungen, die Sie verwenden, erstellt wurden: Daten in `"var/"` (außer `"var/cache/"`, `"var/run/"` und `"var/tmp/"`);
 - Systemkonfigurationsdateien: Daten in `"etc/"`;
 - Lokale Software: Daten in `"usr/local/"` oder `"opt/"`;
 - Informationen zur Systeminstallation: Aufzeichnungen über wichtige Schritte (Partitionierung, ...) in einfacher Textform;
 - Daten, von denen Sie wissen, dass sie wichtig sind, bestätigt durch im Vornherein versuchsweise durchgeführte Wiederherstellungsoperationen.
2. Wissen, wie Sie Daten sichern und wiederherstellen:
 - Sicheres Speichern von Daten: geschützt vor Überschreiben und Systemausfällen;
 - Häufige Datensicherungen: Sicherungen planen;
 - Redundante Datensicherungen: Spiegeln der Sicherungsdateien;

- Idiotensicheres Vorgehen: Sicherung durch einen einfachen Befehl.

3. Bewertung der entstehenden Risiken und Kosten:

- Wert der Daten, falls sie verloren gehen;
- Für die Sicherung benötigte Ressourcen: Personen, Hardware, Software, ...;
- Mögliche Ausfälle und deren Wahrscheinlichkeit.

Anmerkung

Machen Sie keine Sicherungen von den Pseudo-Dateisystemen in `/proc`, `/sys`, `/tmp` und `/run` (Näheres dazu in Abschnitt 1.2.12 und Abschnitt 1.2.13). Dies sind absolut nutzlose Daten, außer Sie genau wissen, was Sie tun.

Aus Gründen der sicheren Speicherung sollten die Daten zumindest auf verschiedenen Festplattenpartitionen, bevorzugt sogar auf separaten Festplatten und Rechnern abgelegt werden, um Beschädigungen von Dateisystemen standzuhalten. Wichtige Daten werden am besten auf Medien gespeichert, die nur einmal beschrieben werden können (wie CD-R/DVD-R), um versehentliches Überschreiben auszuschließen. (In Abschnitt 9.7 lesen Sie, wie Sie von der Befehlszeile aus auf die Speichermedien schreiben können. Die grafische Oberfläche der GNOME-Arbeitsplatzumgebung bietet Ihnen auch einen einfachen Weg über das Menü: "Orte → CD/DVD-Ersteller".)

Anmerkung

Sie sollten eventuell einige Anwendungs-Daemons wie den MTA (lesen Sie dazu Abschnitt 6.3) beenden, während Sie die Datensicherung durchführen.

Anmerkung

Besondere Sorgfalt bei Sicherung und Wiederherstellung sollte Identitäts-bezogenen Dateien gelten, wie `/etc/ssh/ssh_host_dsa_key`, `/etc/ssh/ssh_host_rsa_key`, `~/.gnupg/*`, `~/.ssh/*`, `/etc/passwd`, `/etc/shadow`, `/etc/fetchmailrc`, `popularity-contest.conf`, `/etc/ppp/pap-secrets` und `/etc/exim4/passwd.client`. Einige dieser Daten können nicht wiederhergestellt werden, selbst wenn die exakt gleichen Eingaben erneut getätigt werden.

Anmerkung

Wenn Sie einen Cron-Job als Benutzerprozess ausführen, müssen Sie die Dateien im Verzeichnis `/var/spool/cron/crontabs` wiederherstellen und `cron(8)` neu starten. Informationen über `cron(8)` und `crontab(1)` finden Sie in Abschnitt 9.3.14.

10.2.1 Programmsammlungen für Datensicherungsaufgaben

Hier eine Auswahl erwähnenswerter, im Debian-System verfügbarer Datensicherungsprogramme:

Datensicherungs-Werkzeuge haben alle ihren speziellen Fokus:

- [Mondo Rescue](#) ist ein Backup-System, das die schnelle Wiederherstellung eines vollständigen Systems von CD/DVD ermöglicht, ohne dass dabei die normalen Systeminstallations-Prozesse durchlaufen werden müssen.
 - Regelmäßige Sicherungen von Nutzerdaten können über ein einfaches Skript (siehe Abschnitt 10.2.2) und `cron(8)` realisiert werden.
 - [Bacula](#), [Amanda](#) und [BackupPC](#) sind voll ausgestattete Backup-Lösungen, die auf die regelmäßige Datensicherung über Netzwerk fokussiert sind.
-

Paket	Popcon	Größe	Beschreibung
dump	V:1, I:6	352	4.4bsd dump(8)- und restore(8)-Programm für ext2/ext3/ext4-Dateisysteme
xfsdump	V:0, I:9	854	dump (Abbilderstellung) und restore (Wiederherstellung) mit xfsdump(8) und xfsrestore(8) für das XFS -Dateisystem unter GNU/Linux und IRIX
backupninja	V:4, I:5	355	ressourcenschonendes, erweiterbares Meta-Backup -System
bacula-common	V:10, I:15	2158	Bacula : Datensicherung, -wiederherstellung und -verifizierung über das Netzwerk - gemeinsame Hilfsdateien
bacula-client	I:3	183	Bacula : Datensicherung, -wiederherstellung und -verifizierung über das Netzwerk - Client-Metapaket
bacula-console	V:1, I:5	107	Bacula : Datensicherung, -wiederherstellung und -verifizierung über das Netzwerk - Textkonsole
bacula-server	I:1	183	Bacula : Datensicherung, -wiederherstellung und -verifizierung über das Netzwerk - Server-Metapaket
amanda-common	V:0, I:2	10031	Amanda : Advanced Maryland Automatic Network Disk Archiver (Netzwerk-Backup-System - Bibliotheken)
amanda-client	V:0, I:2	1089	Amanda : Advanced Maryland Automatic Network Disk Archiver (Netzwerk-Backup-System - Client)
amanda-server	V:0, I:0	1076	Amanda : Advanced Maryland Automatic Network Disk Archiver (Netzwerk-Backup-System - Server)
backup-manager	V:1, I:2	572	befehlszeilen-basiertes Datensicherungs-Werkzeug
backup2l	V:0, I:1	114	wartungsarmes Sicherungs-/Wiederherstellungsprogramm für mount-fähige Medien (festplatten-basiert)
backuppc	V:3, I:3	3182	BackupPC ist ein hochleistungsfähiges System der Enterprise-Klasse zur Datensicherung von PCs (festplatten-basiert)
duplicity	V:7, I:15	1761	Inkrementelles Backup (auch von fern)
flexbackup	V:0, I:0	243	Inkrementelles Backup (auch von fern)
rdiff-backup	V:7, I:15	733	Inkrementelles Backup (auch von fern)
restic	V:1, I:3	20595	Inkrementelles Backup (auch von fern)
rsnapshot	V:5, I:11	462	Inkrementelles Backup (auch von fern)
slbackup	V:0, I:0	151	Inkrementelles Backup (auch von fern)

Tabelle 10.5: Liste von Datensicherungsprogrammen

Grundlegende Werkzeuge (in Abschnitt 10.1.1 und Abschnitt 10.1.2 beschrieben) können verwendet werden, um Systemdatensicherungen über eigene Skripte durchzuführen. Solche Skripte können wie folgt erweitert werden:

- Das `restic`-Paket ermöglicht inkrementelle Datensicherungen (auch auf ferne Rechner).
- Auch das `rdiff-backup`-Paket ermöglicht inkrementelle Datensicherungen (ebenfalls auf ferne Rechner).
- Das `dump`-Paket hilft bei der inkrementellen und effizienten Archivierung und Wiederherstellung eines kompletten Dateisystems.

Tipp

Lesen Sie `"/usr/share/doc/dump/"` und `"ls dump really deprecated?"`, um mehr über das `dump`-Paket zu lernen.

10.2.2 Ein Beispielskript für die Systemsicherung

Bei einem privatem Debian-Arbeitsplatzsystem mit der `Unstable`-Suite muss ich lediglich persönliche und kritische Daten sichern. Ich installiere das System eh einmal im Jahr neu. Daher sehe ich keinen Grund, das komplette System zu sichern oder ein voll ausgestattetes Backup-Werkzeug zu installieren.

Ich verwende ein einfaches Skript, um ein Backup-Archiv zu erstellen und es über eine grafische GUI-Oberfläche auf CD/DVD zu brennen. Hier ein Beispiel für dieses Skript:

```
#!/bin/sh -e
# Copyright (C) 2007-2008 Osamu Aoki <osamu@debian.org>, Public Domain
BUUID=1000; USER=osamu # UID and name of a user who accesses backup files
BUDIR="/var/backups"
XDIR0=".+ /Mail |.+ /Desktop"
XDIR1=".+ /\..thumbnails |.+ /\..?Trash |.+ /\..?[cC]ache |.+ /\..gvfs |.+ /sessions"
XDIR2=".+ /CVS |.+ /\..git |.+ /\..svn |.+ /Downloads |.+ /Archive |.+ /Checkout |.+ /tmp"
XSFX=".+ \.iso |.+ \.tgz |.+ \.tar \.gz |.+ \.tar \.bz2 |.+ \.cpio |.+ \.tmp |.+ \.swp |.+ ~"
SIZE="+99M"
DATE=$(date --utc +"%Y%m%d-%H%M")
[ -d "$BUDIR" ] || mkdir -p "$BUDIR"
umask 077
dpkg --get-selections \* > /var/lib/dpkg/dpkg-selections.list
debconf-get-selections > /var/cache/debconf/debconf-selections

{
find /etc /usr/local /opt /var/lib/dpkg/dpkg-selections.list \
    /var/cache/debconf/debconf-selections -xdev -print0
find /home/$USER /root -xdev -regextype posix-extended \
    -type d -regex "$XDIR0|$XDIR1" -prune -o -type f -regex "$XSFX" -prune -o \
    -type f -size "$SIZE" -prune -o -print0
find /home/$USER/Mail/Inbox /home/$USER/Mail/Outbox -print0
find /home/$USER/Desktop -xdev -regextype posix-extended \
    -type d -regex "$XDIR2" -prune -o -type f -regex "$XSFX" -prune -o \
    -type f -size "$SIZE" -prune -o -print0
} | cpio -ov --null -O $BUDIR/BU$DATE.cpio
chown $BUUID $BUDIR/BU$DATE.cpio
touch $BUDIR/backup.stamp
```

Dieses Beispielskript ist dazu gedacht, als `root` ausgeführt zu werden.

Ich gehe davon aus, dass Sie es anpassen und wie folgt ausführen:

- Bearbeiten Sie das Skript, so dass all Ihre wichtigen Daten enthalten sind (lesen Sie dazu Abschnitt 10.1.5 und Abschnitt 10.2).
 - Ersetzen Sie `"find ...-print0"` durch `"find ...-newer $BUDIR/backup.stamp -print0"`, um eine inkrementelle Sicherung zu erstellen.
-

- Transferieren Sie die Sicherungsdateien mittels `scp(1)` oder `rsync(1)` auf einen fernen Rechner oder brennen Sie sie für zusätzliche Sicherheit auf CD/DVD. (Ich verwende ein grafisches GNOME-Programm zum Brennen auf CD/DVD. Lesen Sie Abschnitt [12.1.8](#) für zusätzliche Redundanz.

Halten Sie es einfach!

Tipp

Sie können die `debconf`-Konfigurationsdaten mit `"debconf-set-selections debconf-selections"` wiederherstellen und die `dpkg`-Paketauswahl mit `"dpkg --set-selection <dpkg-selections.list"`.

10.2.3 Ein Kopierskript für die Datensicherung

Für eine Datensammlung in einem Verzeichnisbaum ermöglicht das Kopieren mit `"cp -a"` eine normale Datensicherung.

Bei vielen großen, nicht zu überschreibenden statischen Daten in einem Verzeichnisbaum wie der unterhalb des Verzeichnisses `"/var/cache/apt/packages/"` bieten harte Links mit `"cp -al"` eine Alternative zum normalen Backup bei effizienter Nutzung des Festplattenplatzes.

Hier ein Kopierskript für die Datensicherung, das ich `bkup` genannt habe. Dieses Skript kopiert alle Dateien im aktuellen Verzeichnis, die nicht unter Versionskontrolle stehen (nicht-VCS), in das Elternverzeichnis oder auf einen fernen Rechner.

```
#!/bin/sh -e
# Copyright (C) 2007-2008 Osamu Aoki <osamu@debian.org>, Public Domain
fdot(){ find . -type d \( -iname ".?*" -o -iname "CVS" \) -prune -o -print0;}
fall(){ find . -print0;}
mkdircd(){ mkdir -p "$1";chmod 700 "$1";cd "$1">/dev/null;}
FIND="fdot";OPT="-a";MODE="CPIO";HOST="localhost";EXTP="$(hostname -f)"
BKUP="$(basename $(pwd)).bkup";TIME="$(date +%Y%m%d-%H%M%S)";BU="$BKUP/$TIME"
while getopts gcCsStrlLaAxe:h:T f; do case $f in
g)  MODE="GNUCP";; # cp (GNU)
c)  MODE="CPIO";; # cpio -p
C)  MODE="CPIOI";; # cpio -i
s)  MODE="CPIOSSH";; # cpio/ssh
t)  MODE="TARSSH";; # tar/ssh
r)  MODE="RSYNCSH";; # rsync/ssh
l)  OPT="-alv";; # hardlink (GNU cp)
L)  OPT="-av";; # copy (GNU cp)
a)  FIND="fall";; # find all
A)  FIND="fdot";; # find non CVS/ .???/
x)  set -x;; # trace
e)  EXTP="${OPTARG}";; # hostname -f
h)  HOST="${OPTARG}";; # user@remotehost.example.com
T)  MODE="TEST";; # test find mode
\?) echo "use -x for trace."
esac; done
shift $(expr $OPTIND - 1)
if [ $# -gt 0 ]; then
    for x in $@; do cp $OPT $x $x.$TIME; done
elif [ $MODE = GNUCP ]; then
    mkdir -p "../$BU";chmod 700 "../$BU";cp $OPT . "../$BU/"
elif [ $MODE = CPIO ]; then
    mkdir -p "../$BU";chmod 700 "../$BU"
    $FIND|cpio --null --sparse -pvd ../$BU
elif [ $MODE = CPIOI ]; then
    $FIND|cpio -ov --null | ( mkdircd "../$BU"&&cpio -i )
elif [ $MODE = CPIOSSH ]; then
    $FIND|cpio -ov --null|ssh -C $HOST "( mkdircd \"\$EXTP/$BU\"&&cpio -i )"
elif [ $MODE = TARSSH ]; then
    (tar cvf - . )|ssh -C $HOST "( mkdircd \"\$EXTP/$BU\"&& tar xvfp - )"
fi
```

```

elif [ $MODE = RSYNCSSH ]; then
    rsync -aHAXsv ./ "${HOST}:${EXTP}-${BKUP}-${TIME}"
else
    echo "Any other idea to backup?"
    $FIND |xargs -0 -n 1 echo
fi

```

Dies sollen Befehlsbeispiele sein. Bitte lesen Sie das Skript und passen Sie es für sich selbst an, bevor Sie es nutzen.

Tipp

Ich habe dieses bkup in meinem "/usr/local/bin/"-Verzeichnis abgelegt. Ich führe es ohne jegliche Optionen im jeweiligen Arbeitsverzeichnis aus, wann immer ich ein temporäres Abbild des Verzeichnisses als Sicherung benötige.

Tipp

Um Abbilder inklusive der Historie von einem Quelldateibaum oder einem Konfigurationsdateibaum zu erstellen, ist es einfacher und bezüglich des Speicherplatzes effizienter, git(7) zu verwenden (lesen Sie dazu Abschnitt 10.6.5).

10.3 Datensicherheits-Infrastruktur

Die Sicherheitsinfrastruktur für Ihre Daten wird durch eine Kombination verschiedener Programme gewährleistet: Verschlüsselungswerkzeug, Message-Digest-Werkzeug und Signaturwerkzeug.

Paket	Popcon	Größe	Befehl	Beschreibung
gnupg	V:531, I:950	787	gpg(1)	GNU Privacy Guard - OpenPGP Verschlüsselungs- und Signaturwerkzeug
gpgv	V:880, I:999	859	gpgv(1)	GNU Privacy Guard - Werkzeug zur Verifizierung von Signaturen
paperkey	V:1, I:13	58	paperkey(1)	lediglich die geheimen Informationen aus geheimen OpenPGP-Schlüsseln extrahieren
cryptsetup	V:29, I:78	402	cryptsetup(8), ...	Hilfsprogramme für die dm-crypto -Blockgeräte-Verschlüsselung (enthält Unterstützung für LUKS)
ecryptfs-utils	V:3, I:5	460	ecryptfs(7), ...	Hilfsprogramme für die auf "stacked filesystems" (überlagerten Dateisystemen) aufbauende ecryptfs -Verschlüsselung
coreutils	V:891, I:999	17478	md5sum(1)	MD5-Message-Digest (Prüfsummen) berechnen und überprüfen
coreutils	V:891, I:999	17478	sha1sum(1)	SHA1-Message-Digest (Prüfsummen) berechnen und überprüfen
openssl	V:794, I:993	1465	openssl(1ssl)	Message-Digest mit "openssl dgst" berechnen (OpenSSL)

Tabelle 10.6: Liste von Werkzeugen für die Datensicherheits-Infrastruktur

In Abschnitt 9.8 finden Sie Infos über [dm-crypto](#) und [ecryptfs](#), die automatische Datenverschlüsselungs-Infrastrukturen über Linux-Kernelmodule implementieren.

Befehl	Beschreibung
<code>gpg --gen-key</code>	einen neuen Schlüssel generieren
<code>gpg --gen-revoke meine_Nutzer_ID</code>	einen Widerrufs Schlüssel (revoke key) für meine_Nutzer_ID generieren
<code>gpg --edit-key Nutzer_ID</code>	Schlüssel interaktiv editieren, "help" eingeben für weitere Hilfe
<code>gpg -o datei --export</code>	alle Schlüssel in datei exportieren
<code>gpg --import datei</code>	alle Schlüssel aus datei importieren
<code>gpg --send-keys Nutzer_ID</code>	Schlüssel von Nutzer_ID zum Schlüsselserver übertragen
<code>gpg --recv-keys Nutzer_ID</code>	Schlüssel von Nutzer_ID vom Schlüsselserver empfangen
<code>gpg --list-keys Nutzer_ID</code>	Schlüssel von Nutzer_ID auflisten
<code>gpg --list-sigs user_ID</code>	Signaturen von Nutzer_ID auflisten
<code>gpg --check-sigs Nutzer_ID</code>	Signaturen von Nutzer_ID überprüfen
<code>gpg --fingerprint Nutzer_ID</code>	Fingerabdruck von Nutzer_ID überprüfen
<code>gpg --refresh-keys</code>	lokalen Schlüsselring aktualisieren

Tabelle 10.7: Liste von GNU Privacy Guard-Befehlen für die Schlüsselverwaltung

10.3.1 Schlüsselverwaltung für GnuPG

Hier einige Befehle für die grundlegende Schlüsselverwaltung mit [GNU Privacy Guard](#):

Hier die Bedeutung des Vertrauenscodes:

Code	Beschreibung des Vertrauenscodes
-	Kein Besitzervertrauen zugewiesen / noch nicht berechnet
e	Vertrauensberechnung fehlgeschlagen
q	nicht genügend Informationen für Berechnung
n	diesem Schlüssel niemals vertrauen
m	gerade noch vertrauenswert
f	voll vertrauenswert
u	absolut vertrauenswert

Tabelle 10.8: Liste der Bedeutungen des Vertrauenscodes

Folgendes lädt meinen Schlüssel "1DD8D791" auf den populären Schlüsselserver "hkp://keys.gnupg.net" hoch:

```
$ gpg --keyserver hkp://keys.gnupg.net --send-keys 1DD8D791
```

Ein gutes Standard-Schlüsselserver-Setup in "~/.gnupg/gpg.conf" (oder dem alten Ort "~/.gnupg/options") enthält folgendes:

```
keyserver hkp://keys.gnupg.net
```

Mit dem folgenden Befehl beziehen Sie unbekannte Schlüssel vom Schlüsselserver:

```
$ gpg --list-sigs --with-colons | grep '^sig.*\[User ID not found\]' |\
  cut -d ':' -f 5 | sort | uniq | xargs gpg --recv-keys
```

Es gab einen Fehler in [OpenPGP Public Key Server](#) (Versionen vor 0.9.6), durch den Schlüssel mit mehr als zwei Unterschlüsseln korruptiert wurden. Das neue gnupg-Paket (>1.2.1-2) kann mit diesen korruptierten Unterschlüsseln umgehen. Lesen Sie in `gpg(1)` den Abschnitt zur Option "--repair-pks-subkey-bug".

10.3.2 Verwendung von GnuPG mit Dateien

Hier einige Beispiele für die Verwendung von [GNU Privacy Guard](#) mit Dateien:

Befehl	Beschreibung
<code>gpg -a -s datei</code>	Signieren von datei in die ASCII -bewehrte datei.asc
<code>gpg --armor --sign datei</code>	„
<code>gpg --clearsign datei</code>	Signieren einer Nachricht mit Klartextsignatur
<code>gpg --clearsign datei mail foo@example.org</code>	Versenden einer Nachricht mit Klartextsignatur an foo@example.org
<code>gpg --clearsign --not-dash-escaped patch-datei</code>	Klartext-signieren von patch-datei
<code>gpg --verify datei</code>	eine klartextsignierte datei verifizieren
<code>gpg -o datei.sig -b datei</code>	Erzeugen einer losgelösten Signatur
<code>gpg -o datei.sig --detach-sig datei</code>	„
<code>gpg --verify datei.sig datei</code>	Verifizieren von datei mit datei.sig
<code>gpg -o crypt_datei.gpg -r name -e datei</code>	Verschlüsselung mit öffentlichem Schlüssel für name von datei in Binärdatei crypt_datei.gpg
<code>gpg -o crypt_datei.gpg --recipient name --encrypt datei</code>	„
<code>gpg -o crypt_datei.asc -a -r name -e datei</code>	Verschlüsselung mit öffentlichem Schlüssel für name von datei in ASCII -bewehrte Datei crypt_datei.gpg
<code>gpg -o crypt_datei.gpg -c datei</code>	symmetrische Verschlüsselung von "datei" in crypt_datei.gpg
<code>gpg -o crypt_datei.gpg --symmetric datei</code>	„
<code>gpg -o crypt_datei.asc -a -c datei</code>	symmetrische Verschlüsselung für name von datei in ASCII -bewehrte Datei crypt_datei.asc
<code>gpg -o datei -d crypt_datei.gpg -r name</code>	Entschlüsselung
<code>gpg -o datei --decrypt crypt_datei.gpg</code>	„

Tabelle 10.9: Liste von GNU Privacy Guard-Befehlen mit Dateien

10.3.3 Verwendung von GnuPG mit Mutt

Fügen Sie Folgendes zu `~/ .muttrc` hinzu, um zu verhindern, dass das langsame GnuPG automatisch gestartet wird, während es durch Drücken von `"S"` im Indexmenü händisch gestartet werden kann:

```
macro index S ":toggle pgp_verify_sig\n"
set pgp_verify_sig=no
```

10.3.4 Verwendung von GnuPG mit Vim

Das `gnupg`-Plugin ermöglicht Ihnen, GnuPG für Dateien mit den Endungen `".gpg"`, `".asc"` und `".pgp"` transparent laufen zu lassen:

```
# aptitude install vim-scripts vim-addon-manager
$ vim-addons install gnupg
```

10.3.5 Die MD5-Prüfsumme

`md5sum(1)` enthält ein Werkzeug, um über die in [RFC1321](#) beschriebene Methode eine Digest-Datei zu erzeugen und jede Datei darüber zu verifizieren:

```
$ md5sum foo bar >baz.md5
$ cat baz.md5
d3b07384d113edec49eaa6238ad5ff00  foo
c157a79031e1c40f85931829bc5fc552  bar
$ md5sum -c baz.md5
foo: OK
bar: OK
```

Anmerkung

Die Berechnung der [MD5](#)-Prüfsumme ist weniger CPU-intensiv als die für die kryptographische Signatur durch [GNU Privacy Guard \(GnuPG\)](#). Üblicherweise ist nur die Digest-Datei der obersten Ebene (z.B. das Wurzelverzeichnis eines Verzeichnisbaums) kryptographisch signiert, um die Datenintegrität sicherzustellen.

10.4 Werkzeuge zur Quellcode-Zusammenführung

Es gibt viele Werkzeuge für die Zusammenführung von Quellcode. Folgende Befehle haben meine Aufmerksamkeit erregt:

10.4.1 Unterschiede für Quelldateien extrahieren

Die folgenden Prozeduren extrahieren die Unterschiede zwischen zwei Quelldateien und erzeugen vereinheitlichte Diff-Dateien `"file.patch0"` bzw. `"file.patch1"`, abhängig vom Speicherort der Dateien:

```
$ diff -u datei.alt datei.neu > datei.patch0
$ diff -u alt/datei neu/datei > datei.patch1
```

10.4.2 Aktualisierungen für Quelldateien zusammenführen

Die Diff-Datei (alternativ auch Patch-Datei genannt) wird genutzt, um eine Programmaktualisierung zu senden. Die Seite, die die Änderungen empfängt, wendet die Aktualisierung wie folgt an:

```
$ patch -p0 datei < datei.patch0
$ patch -p1 datei < datei.patch1
```

Paket	Popcon	Größe	Befehl	Beschreibung
diffutils	V:871, I:991	1598	diff(1)	Dateien Zeile für Zeile vergleichen
diffutils	V:871, I:991	1598	diff3(1)	drei Dateien Zeile für Zeile vergleichen und zusammenführen
vim	V:106, I:398	3231	vimdiff(1)	zwei Dateien nebeneinander in vim vergleichen
patch	V:99, I:725	248	patch(1)	eine diff-Datei auf ein Original anwenden
dpatch	V:0, I:11	191	dpatch(1)	Serien von Patches für Debian-Pakete verwalten
diffstat	V:16, I:154	73	diffstat(1)	eine Grafik der Veränderungen für eine diff-Datei erstellen
patchutils	V:18, I:150	232	combinediff(1)	einen kumulativen Patch aus zwei inkrementellen Patches erzeugen
patchutils	V:18, I:150	232	dehtmldiff(1)	ein Diff aus einer HTML-Seite extrahieren
patchutils	V:18, I:150	232	filterdiff(1)	Diffs aus einer Diff-Datei extrahieren oder entfernen
patchutils	V:18, I:150	232	fixcvsdiff(1)	von CVS erstellte Diff-Dateien, die patch(1) falsch interpretiert, reparieren
patchutils	V:18, I:150	232	flipdiff(1)	die Reihenfolge zweier Patches ändern
patchutils	V:18, I:150	232	grepdiff(1)	anzeigen, welche Dateien von einem Patch entsprechend einem regulären Ausdruck modifiziert werden
patchutils	V:18, I:150	232	interdiff(1)	Unterschiede zwischen zwei Unified-Diff-Dateien anzeigen
patchutils	V:18, I:150	232	lsdiff(1)	zeigen, welche Dateien von einem Patch verändert werden
patchutils	V:18, I:150	232	recountdiff(1)	Zähler und Offsets in vereinheitlichten Context-Diffs neu berechnen
patchutils	V:18, I:150	232	rediff(1)	Offsets und Zähler eines hand-editierten Diffs bereinigen
patchutils	V:18, I:150	232	splitdiff(1)	inkrementelle Patches aussortieren
patchutils	V:18, I:150	232	unwrapdiff(1)	Patches von fehlerhaften Zeilenumbrüchen befreien
wiggle	V:0, I:0	174	wiggle(1)	zurückgewiesene Patches anwenden
quilt	V:3, I:33	788	quilt(1)	Serien von Patches verwalten
meld	V:14, I:39	2972	meld(1)	vergleichen und zusammenführen von Dateien (GTK)
dirdiff	V:0, I:2	166	dirdiff(1)	Unterschiede zwischen Verzeichnissen anzeigen und zusammenführen
docdiff	V:0, I:0	555	docdiff(1)	zwei Dateien Wort-für-Wort / Buchstabe-für-Buchstabe vergleichen
imediff	V:0, I:0	157	imediff(1)	interaktives Zwei-/Dreiwege-Zusammenführungs-Werkzeug mit Vollbildschirmmodus
makepatch	V:0, I:0	102	makepatch(1)	erweiterte Patch-Dateien erzeugen
makepatch	V:0, I:0	102	applypatch(1)	erweiterte Patch-Dateien anwenden
wdiff	V:9, I:72	644	wdiff(1)	Wort-für-Wort-Unterschiede zwischen Textdateien anzeigen

Tabelle 10.10: Liste von Werkzeugen zur Quellcode-Zusammenführung

10.4.3 Aktualisierung via 3-Wege-Zusammenführung

Wenn Sie drei Varianten eines Quelltextes haben, können Sie mittels `diff3(1)` eine effektive 3-Wege-Zusammenführung vollführen:

```
$ diff3 -m meine.datei alte.datei deine.datei > datei
```

10.5 Versionskontrollsysteme

Hier eine Zusammenfassung der [Versionskontrollsysteme \(VCS\)](#) im Debian-System.

Anmerkung

Wenn Sie neu sind auf dem Gebiet der VCS-Systeme, sollten Sie damit beginnen, **Git** zu lernen, das auf der Beliebtheitsskala schnell steigt.

Paket	Popcon	Größe	Werkzeug	VCS-Typ	Erläuterung
cssc	V:0, I:2	2044	CSSC	lokal	Klon von Unix SCCS (veraltet)
rcs	V:3, I:19	562	RCS	lokal	" Unix SCCS done right" (Unix SCCS richtig gemacht)
cvs	V:5, I:41	4609	CVS	fern	früherer Standard für fernes VCS
subversion	V:20, I:109	4858	Subversion	fern	"CVS done right" (CVS richtig gemacht), der neue De-Facto-Standard für fernes VCS
git	V:305, I:478	35040	Git	verteilt	schnelles DVCS in C (wird vom Linux-Kernel und anderen verwendet)
mercurial	V:8, I:48	1053	Mercurial	verteilt	DVCS in Python und ein bisschen C
bazaar	V:2, I:16	28	Bazaar	verteilt	von t la beeinflusstes DVCS, geschrieben in Python (wird verwendet von Ubuntu)
darcs	V:0, I:7	23159	Darcs	verteilt	DVCS mit schlauer Algebra von Patches (langsam)
tla	V:0, I:2	1011	GNU arch	verteilt	DVCS hauptsächlich von Tom Lord (historisch)
monotone	V:0, I:0	5815	Monotone	verteilt	DVCS in C++
tkcvs	V:0, I:1	1498	CVS, ...	fern	GUI-Oberfläche von VCS-Depots (CVS, Subversion, RCS)
gitk	V:6, I:42	1723	Git	verteilt	GUI-Oberfläche von VCS-Depots (Git)

Tabelle 10.11: Liste von Werkzeugen für die Versionskontrolle

VCS ist teilweise auch bekannt als Revision Control System (RCS) oder Software Configuration Management (SCM).

Verteilte Versionskontrollsysteme (DVCS) wie Git sind dieser Tage das System der Wahl. CVS und Subversion könnten nach wie vor nützlich sein, um die Aktivitäten einiger vorhandener quelloffener Programme zu vereinen.

Debian stellt freie Git-Dienste über [Debian Salsa](#) bereit. Dokumentation finden Sie unter <https://wiki.debian.org/Salsa>.



Achtung

Debian's alte Alioth-Dienste wurden beendet; alte Daten der Dienste sind noch als [Tarball-Archive](#) verfügbar.

Es gibt einige Grundlagen für die Erstellung eines VCS-Archivs mit verteiltem Zugriff:

- verwenden Sie "umask 002" (lesen Sie dazu Abschnitt 1.2.4);
- alle Dateien im VCS-Archiv müssen der entsprechenden Gruppe angehören;
- bei allen Verzeichnissen im VCS-Archiv sollte das Setzen der Gruppen-ID aktiviert sein (BSD-ähnliches Dateierzeugungsschema, lesen Sie Abschnitt 1.2.3);
- alle Benutzer, die das VCS-Archiv verwenden, müssen dieser Gruppe angehören.

10.5.1 Gegenüberstellung von VCS-Befehlen

Hier eine sehr vereinfachte Gegenüberstellung nativer VCS-Befehle, um einen groben Überblick zu ermöglichen. Eine typische Befehlssequenz könnte weitere Optionen und Argumente erfordern.

Git	CVS	Subversion	Funktion
git init	cvsv init	svn create	ein (lokales) Depot erstellen
-	cvsv login	-	am fernen Depot anmelden
git clone	cvsv co	svn co	Herunterladen des fernen Depots als Arbeitsverzeichnis (Checkout)
git pull	cvsv up	svn up	Aktualisierung des Arbeitsverzeichnisses durch Abgleich mit dem fernen Depot
git add .	cvsv add	svn add	Datei(en) im Verzeichnisbaum zum VCS hinzufügen
git rm	cvsv rm	svn rm	Datei(en) im Verzeichnisbaum aus dem VCS entfernen
-	cvsv ci	svn ci	Änderungen zum fernen Depot hochladen (Commit)
git commit -a	-	-	Änderungen in das lokale Depot übernehmen (Commit)
git push	-	-	das ferne mit dem lokalen Depot aktualisieren (durch Hochladen der Änderungen)
git status	cvsv status	svn status	den Status des Verzeichnisbaums aus dem VCS anzeigen
git diff	cvsv diff	svn diff	diff <referenzdepot> <verzeichnisbaum>: Unterschiede ermitteln zwischen <referenzdepot> und <verzeichnisbaum>
git repack -a -d; git prune	-	-	das lokale Depot neu in ein einzelnes Pack verpacken
gitk	tkcvs	tkcvs	grafische GUI-Anzeige eines VCS-Depots

Tabelle 10.12: Gegenüberstellung nativer VCS-Befehle



Achtung

Der direkte Aufruf eines git-Unterbefehls mittels "git-xyz" über die Befehlszeile ist seit dem Frühjahr 2006 veraltet.

Tipp

Wenn eine ausführbare Datei namens git-foo in dem durch \$PATH definierten Pfad existiert, wird dieser git-foo-Befehl ausgeführt, indem man "git foo" (ohne Bindestrich!) auf der Befehlszeile aufruft. Dies ist eine Funktionalität des git-Befehls.

Tipp

Grafische GUI-Werkzeuge wie tkcvs(1) und gitk(1) helfen Ihnen wirklich bei der Verfolgung der Vergangenheit von Dateien. Die Web-Oberfläche, die von vielen öffentlichen Archiven bereitgestellt wird, um die Depots mit einem Webbrowser zu durchsuchen, ist ebenfalls sehr nützlich.

Tipp

Git kann direkt mit verschiedenen VCS-Depots wie denen von CVS und Subversion umgehen; es bietet über die git-cvs- und git-svn-Pakete lokale Depots für lokale Änderungen. Lesen Sie dazu die [gitcvs-migration-Handbuchseite](#) sowie Abschnitt 10.6.4.

Tipp

Git enthält Befehle, für die es in CVS und Subversion keine äquivalenten Entsprechungen gibt: "fetch", "rebase", "cherry-pick", ...

10.6 Git

Git kann alles sowohl für die lokale wie auch für die ferne Quellcode-Verwaltung erledigen. Das bedeutet, dass Sie lokal alle Änderungen einpflegen können, ohne dazu eine Netzwerkverbindung zum fernen Depot zu benötigen.

10.6.1 Konfiguration eines Git-Clients

Sie möchten vielleicht wie folgt verschiedene globale Konfigurationsoptionen in "~/.gitconfig" zur Verwendung durch Git setzen, z.B. Name und E-Mail-Adresse:

```
$ git config --global user.name "Vorname Nachname"
$ git config --global user.email ihr-name@example.com
```

Wenn Sie an CVS- oder Subversion-Befehle gewohnt sind, können Sie wie hier auch folgende Befehls-Alias setzen:

```
$ git config --global alias.ci "commit -a"
$ git config --global alias.co checkout
```

Sie können wie folgt Ihre globale Konfiguration kontrollieren:

```
$ git config --global --list
```

10.6.2 Weitere Referenzen zu Git

Hier finden Sie weitere Informationen:

- [Handbuchseite: git\(1\)](#) (/usr/share/doc/git-doc/git.html)
 - [git-Benutzerhandbuch \(engl.\)](#) (/usr/share/doc/git-doc/user-manual.html)
 - [gittutorial: Einführung in git \(engl.\)](#) (/usr/share/doc/git-doc/gittutorial.html)
 - [gittutorial-2: Einführung in git - Teil 2 \(engl.\)](#) (/usr/share/doc/git-doc/gittutorial-2.html)
 - [Anleitung für die alltägliche git-Nutzung mit ungefähr 20 Befehlen \(engl.\)](#) (/usr/share/doc/git-doc/everyday.html)
 - [git für CVS-Benutzer \(engl.\)](#) (/usr/share/doc/git-doc/gitcvs-migration.html)
-

- Hier wird auch beschrieben, wie Sie einen Server identisch zu CVS einrichten und alte Daten aus CVS in Git extrahieren.
- [weitere im Web verfügbare Git-Ressourcen](#)
 - [Git-SVN-Crashkurs \(engl.\)](#)
 - [Git Magic \(/usr/share/doc/gitmagic/html/index.html\)](#)

Die git-gui(1)- und gitk(1)-Befehle machen die Git-Nutzung sehr einfach.



Warnung

Nutzen Sie in der Tag-Zeichenkette keine Leerzeichen, auch wenn Werkzeuge wie gitk(1) dies ermöglichen. Andere git-Befehle könnten dadurch ins Stocken geraten.

10.6.3 Git-Befehle

Selbst wenn bei Ihnen Upstream ein anderes VCS verwendet, könnte es eine gute Idee sein, für lokale Aktivitäten git(1) zu nutzen, da Sie Ihre lokale Kopie eines Quellcodebaums ohne Netzwerkverbindung zum Upstream-Depot verwalten können. Hier einige Pakete und Programme, die mit git(1) verwendet werden.

Paket	Popcon	Größe	Befehl	Beschreibung
git-doc	I:15	11762	Nicht verfügbar	offizielle Dokumentation für Git
gitmagic	I:1	721	Nicht verfügbar	”Git Magic”, eine einfacher verständliche Anleitung für Git
git	V:305, I:478	35040	git(7)	Git, das schnelle, skalierbare, verteilte Versionsverwaltungssystem
gitk	V:6, I:42	1723	gitk(1)	grafische Browser-Oberfläche mit Historie für Git-Depots
git-gui	V:2, I:24	2317	git-gui(1)	grafische Oberfläche (ohne Historie) für Git
git-svn	V:1, I:22	1144	git-svnimport(1)	Daten aus Subversion in Git importieren
git-svn	V:1, I:22	1144	git-svn(1)	bidirektionaler Einsatz von Subversion und Git
git-cvs	V:0, I:10	1279	git-cvsexportcommit(1)	Daten aus CVS in Git importieren
git-cvs	V:0, I:10	1279	git-cvsexportcommit(1)	einen Commit aus Git in einen CVS-Checkout importieren
git-cvs	V:0, I:10	1279	git-cvsserver(1)	CVS-Server-Emulator für Git
git-email	V:0, I:11	966	git-send-email(1)	eine Patch-Sammlung aus Git als E-Mail versenden
stgit	V:0, I:0	603	stg(1)	Aufsatz für Git (Python)
git-buildpackage	V:2, I:12	4193	git-buildpackage(1)	das Debian-Paketieren mit Git automatisieren
guilt	V:0, I:0	146	guilt(7)	Aufsatz für Git (SH/AWK/SED/...)

Tabelle 10.13: Liste von zu Git gehörigen Paketen und Befehlen

Tipp

Bei git(1) führen Sie viele Commits (Übernahme der Änderungen ins Depot) in einem lokalen Zweig (Branch) durch und nutzen später etwas wie `git rebase -i master`, um die Änderungshistorie im Nachhinein neu zu organisieren. Dies erlaubt es Ihnen, eine saubere Änderungshistorie zu erstellen. Lesen Sie dazu `git-rebase(1)` und `git-cherry-pick(1)`.

Tipp

Falls Sie zu einem sauberen Arbeitsverzeichnis zurückkehren möchten, ohne den aktuellen Status des Arbeitsverzeichnisses zu verlieren, können Sie "git stash" verwenden. Details finden Sie unter [git-stash\(1\)](#).

10.6.4 Git mit einem Subversion-Depot

Sie können einen Checkout eines Subversion-Depots unter "svn+ssh://svn.example.org/project/modul/trunk" in ein lokales Git-Depot in ". /dest" ausführen, Änderungen durchführen und diese dann zurück zum Subversion-Depot übergeben, z.B. wie hier:

```
$ git svn clone -s -rHEAD svn+ssh://svn.example.org/project dest
$ cd dest
... führen Sie Ihre Änderungen durch ...
$ git commit -a
... weitere lokale Arbeiten mit git ...
$ git svn dcommit
```

Tipp

Dank der Verwendung von "-rHEAD" müssen wir nicht die vollständigen historischen Inhalte aus dem Subversion-Depot klonen.

10.6.5 Git zur Aufzeichnung der Historie von Konfigurationsdateien

Mit [Git](#)-Werkzeugen können Sie die Historie von Konfigurationsdateien chronologisch aufzeichnen. Hier zur Übung ein einfaches Beispiel zur Aufzeichnung der Inhalte von "/etc/apt/":

```
$ cd /etc/apt/
$ sudo git init
$ sudo chmod 700 .git
$ sudo git add .
$ sudo git commit -a
```

Die Konfiguration mit einer Beschreibung einpflegen.

Machen Sie einige Änderungen an den Konfigurationsdateien.

```
$ cd /etc/apt/
$ sudo git commit -a
```

Übernehmen Sie dann die Konfiguration mit einer Beschreibung der Änderungen in das Git-Depot und fahren Sie mit Ihrer normalen Arbeit fort.

```
$ cd /etc/apt/
$ sudo gitk --all
```

Sie haben jetzt die vollständige Historie der Konfigurationsdateien vorliegen.

Anmerkung

sudo(8) wird benötigt, um jegliche Dateiberechtigungen von Konfigurationsdateien zu bearbeiten. Bei Konfigurationsdaten von Benutzern können Sie das sudo unter Umständen weglassen.

Anmerkung

Der Befehl "chmod 700 .git" im obigen Beispiel ist erforderlich, um die Archivdaten vor nicht autorisierten Lesezugriffen zu schützen.

Tipp

Falls Sie ein leistungsfähigeres Setup zur Aufzeichnung der Historie von Konfigurationsdateien benötigen, schauen Sie sich das Paket `etckeeper` an: Abschnitt [9.2.10](#).

10.7 CVS

CVS ist ein **älteres** Versionskontrollsystem aus der Zeit vor Subversion und Git.

**Achtung**

Viele URLs in unten stehenden Beispielen für CVS funktionieren nicht mehr.

Hier finden Sie weitere Informationen:

- `cvs(1)`
- `"/usr/share/doc/cvs/html-cvscsclient"`
- `"/usr/share/doc/cvs/html-info"`
- `"/usr/share/doc/cvsbook"`
- `"info cvs"`

10.7.1 Konfiguration eines CVS-Depots

Folgende Konfiguration erlaubt Commits (Übernehmen von Änderungen in das CVS-Depot) nur durch Mitglieder der Gruppe "src", und die Administration von CVS nur durch Mitglieder der Gruppe "staff"; dies reduziert die Möglichkeit, den Server versehentlich zu misskonfigurieren.

```
# cd /var/lib; umask 002; mkdir cvs
# export CVSR00T=/srv/cvs/project
# cd $CVSR00T
# chown root:src .
# chmod 2775 .
# cvs -d $CVSR00T init
# cd CVSR00T
# chown -R root:staff .
# chmod 2775 .
# touch val-tags
# chmod 664 history val-tags
# chown root:src history val-tags
```

Tipp

Sie sollten vielleicht das Erstellen von neuen Projekten einschränken, indem Sie den Eigentümer des "\$CVSR00T"-Verzeichnisses in "root:staff" ändern sowie dessen Berechtigung in "3775".

10.7.2 Lokaler Zugriff auf CVS

Das Standard-CVS-Depot verweist auf "\$CVSR00T". Folgendes richtet "\$CVSR00T" für lokalen Zugriff ein:

```
$ export CVSROOT=/srv/cvs/project
```

10.7.3 Fernzugriff auf CVS mit pserver

Viele öffentliche CVS-Server gestatten über den pserver-Dienst einen Nur-Lese-Zugriff für den Benutzer "anonymous". Zum Beispiel wurden die Inhalte der Debian-Website durch das [webwml-Projekt](#) via CVS über den Debian-Alioth-Service betreut. Mit folgendem Beispiel hat man "\$CVSR00T" für den Fernzugriff auf dieses (alte) CVS-Depot eingerichtet:

```
$ export CVSROOT=:pserver:anonymous@anonscm.debian.org:/cvs/webwml
$ cvs login
```

Anmerkung

Da pserver für Abhörattacken und Unsicherheiten bekannt ist, wird der entsprechende Schreibzugriff gewöhnlich durch die Systemadministratoren deaktiviert.

10.7.4 Fernzugriff auf CVS mit ssh

Folgendes richtete "\$CVS_RSH" und "\$CVSR00T" für den Fernzugriff über SSH auf das (alte) CVS-Depot des [webwml-Projekts](#) ein:

```
$ export CVS_RSH=ssh
$ export CVSROOT=:ext:account@cvs.alioth.debian.org:/cvs/webwml
```

Sie können auch die Authentifizierung für SSH über einen öffentlichen Schlüssel aktivieren und so die Abfrage des Passworts unterbinden.

10.7.5 Eine neue Quelle in CVS importieren

Erzeugen Sie einen neuen lokalen Quellcodebaum unter "~/pfad/zu/modul1", indem Sie dies eingeben:

```
$ mkdir -p ~/pfad/zu/modul1; cd ~/pfad/zu/modul1
```

Legen Sie Dateien in dem neuen Quellcodebaum unter "~/pfad/zu/modul1" an.

Importieren Sie es mit folgenden Parametern in CVS:

- Modulname: "modul1"
- Vendor tag: "Haupt - Zweig" (Markierung für den kompletten Zweig)
- Release tag: "initiale-Veröffentlichung" (Markierung für eine spezifische Version)

```
$ cd ~/pfad/zu/modul1
$ cvs import -m "Start von modul1" modul1 Haupt-Zweig initiale-Veröffentlichung
$ rm -Rf . # optional
```

10.7.6 Dateiberechtigungen im CVS-Depot

CVS überschreibt nicht die aktuelle Verzeichnisdatei, sondern ersetzt sie durch eine andere. Daher sind Schreibrechte für ein Verzeichnis im Depot kritisch. Führen Sie für jedes neue Modul von "modul1" im Depot unter "/srv/cvs/project" folgendes aus, um den korrekten Zustand falls nötig sicherzustellen:

```
# cd /srv/cvs/project
# chown -R root:src modul1
# chmod -R ug+rwX modul1
# chmod 2775 modul1
```

10.7.7 Arbeitsablauf bei CVS

Hier ein Beispiel eines typischen Arbeitsablaufs bei CVS:

Überprüfen Sie alle verfügbaren Module aus dem CVS-Projekt, auf das "\$CVSR00T" zeigt:

```
$ cvs rls
CVSR00T
modul1
modul2
...
```

Führen Sie einen Checkout von "modul1" in das Standard-Verzeichnis "./modul1" durch:

```
$ cd ~/pfad/zu
$ cvs co modul1
$ cd modul1
```

Führen Sie nötige Änderungen am Inhalt durch ...

Überprüfen Sie Ihre Änderungen über einen Befehl ähnlich zu "diff -u [depot] [lokal]":

```
$ cvs diff -u
```

Sie stellen fest, dass Sie eine Datei "file_to_undo" ernsthaft beschädigt haben, aber andere Dateien sind in Ordnung.

Überschreiben Sie die Datei "file_to_undo" mit der sauberen Kopie aus dem CVS, indem Sie folgendes ausführen:

```
$ cvs up -C file_to_undo
```

Sichern Sie den aktualisierten Quellcodebaum in das CVS-Depot:

```
$ cvs ci -m "Beschreibung der Änderungen"
```

Erzeugen Sie eine Datei "file_to_add" und fügen Sie sie zum CVS hinzu, wie folgt:

```
$ vi file_to_add
$ cvs add file_to_add
$ cvs ci -m "Datei file_to_add hinzugefügt"
```

Führen Sie die aktuellste Version aus dem CVS-Depot mit Ihrem lokalen Quellcode zusammen:

```
$ cvs up -d
```

Achten Sie auf Zeilen, die mit "C dateiname" beginnen; dies zeigt sich widersprechende Änderungen an.

Schauen Sie nach unverändertem Code in ".#dateiname.version".

Suchen Sie nach "<<<<<<" und ">>>>>>" in den Dateien, bei denen sich widersprechende Änderungen festgestellt wurden ("C dateiname").

Bearbeiten Sie die Dateien falls nötig, um Konflikte zu bereinigen.

Fügen Sie wie folgt eine Markierung (Tag) "Veröffentlichung-1" für die Version hinzu:

```
$ cvs ci -m "Letzte Änderungen für Veröffentlichung-1"
$ cvs tag Veröffentlichung-1
```

Weitere Änderungen

Entfernen Sie die "Veröffentlichung-1"-Markierung wie folgt:

```
$ cvs tag -d Veröffentlichung-1
```

Fügen Sie die Änderungen wie folgt zum CVS hinzu:

```
$ cvs ci -m "Wirklich letzter Commit für Veröffentlichung-1"
```

Fügen Sie die "Veröffentlichung-1"-Markierung erneut zum aktualisierten CVS HEAD hinzu:

```
$ cvs tag Veröffentlichung-1
```

Erzeugen Sie einen Zweig mit einer fixen Zweig-Markierung "initiale-Veröffentlichung-Fehlerkorrekturen" aus der originalen Version, auf die die Markierung "initiale-Veröffentlichung" zeigt, und führen Sie einen Checkout in das Verzeichnis "~/pfad/zu/alt" durch:

```
$ cvs rtag -b -r initiale-Veröffentlichung initiale-Veröffentlichung-Fehlerkorrekturen ↔
modul1
$ cd ~/pfad/zu
$ cvs co -r initiale-Veröffentlichung-Fehlerkorrekturen -d alt modul1
$ cd alt
```

Tipp

Verwenden Sie "-D 2005-12-20" (Datumsformat gemäß [ISO 8601](#)) statt "-r initiale-Veröffentlichung", um ein bestimmtes Datum für den Zeitpunkt des Zweigs festzulegen.

Arbeiten Sie mit diesem lokalen Quellcodebaum, der die Markierung "initiale-Veröffentlichung-Fehlerkorrekturen" hat und der auf der originalen Version beruht.

Arbeiten Sie selbst an diesem Zweig ...bis jemand weiteres zu diesem "initiale-Veröffentlichung-Fehlerkorrekturen" Zweig hinzustößt.

Synchronisieren Sie die Dateien mit von anderen veränderten Dateien dieses Zweigs und erzeugen Sie dabei falls nötig auch neue Verzeichnisse, wie folgt:

```
$ cvs up -d
```

Bearbeiten Sie die Dateien falls nötig, um Konflikte zu bereinigen.

Fügen Sie die Änderungen wie folgt zum CVS hinzu:

```
$ cvs ci -m "in diesen Zweig hinzugefügt"
```

Aktualisieren Sie den lokalen Quellcodebaum bei HEAD von main; dabei wird die fixe Zweig-Markierung entfernt ("-A") und es wird keine Tastatur-Expansion verwendet ("-kk"):

```
$ cvs up -d -kk -A
```

Aktualisieren Sie den lokalen Quellcodebaum (Inhalt = HEAD von main), indem Sie die Änderungen aus dem "initiale-Veröffentlichung-Fehlerkorrekturen" Zweig (ohne Verwendung von Tastatur-Expansion) zusammenführen:

```
$ cvs up -d -kk -j initiale-Veröffentlichung-Fehlerkorrekturen
```

Beheben Sie Konflikte mit einem Editor.

Fügen Sie die Änderungen wie folgt zum CVS hinzu:

```
$ cvs ci -m "initiale-Veröffentlichung-Fehlerkorrekturen zusammengeführt"
```

Erstellen Sie ein Archiv, wie folgt:

```
$ cd ..  
$ mv alt alt-modul1-korrekturen  
$ tar -cvzf alt-modul1-korrekturen.tar.gz alt-modul1-korrekturen  
$ rm -rf alt-modul1-korrekturen
```

Tipp

Dem Befehl "cvs up" kann die Option "-d" zur Erstellung neuer Verzeichnisse angehängt werden sowie "-P", um leere Verzeichnisse zu entfernen.

Tipp

Sie können einen Checkout von nur einem Unterverzeichnis von "modul1" durchführen, indem Sie seinen Namen angeben, z.B. "cvs co modul1/unterverzeichnis".

Option	Bedeutung
-n	Probelauf, ohne Wirkung
-t	Meldungen ausgeben, die die einzelnen Schritte der CVS-Aktivitäten anzeigen

Tabelle 10.14: Erwähnenswerte Optionen für CVS-Befehle (als erste Argumente bei cvs(1) zu verwenden):

10.7.8 Aktuellste Dateien von CVS

Um die aktuellsten Dateien vom CVS zu bekommen, verwenden Sie "tomorrow", wie hier:

```
$ cvs ex -D tomorrow modulname
```

10.7.9 Administrierung von CVS

Fügen Sie wie folgt einen Modul-Alias "mx" zum CVS-Projekt (lokaler Server) hinzu:

```
$ export CVSR00T=/srv/cvs/project  
$ cvs co CVSR00T/modules  
$ cd CVSR00T  
$ echo "mx -a modul1" >>modules  
$ cvs ci -m "mx ist ein Alias für modul1"  
$ cvs release -d .
```

Jetzt können Sie einen Checkout von "modul1" (Alias "mx") vom CVS in das "neu"-Verzeichnis durchführen:

```
$ cvs co -d neu mx  
$ cd neu
```

Anmerkung

Um obiges Prozedere durchzuführen, sollten Sie die entsprechenden Dateirechte haben.

10.7.10 Ausführungs-Bit für CVS-Checkout

Bei dem Checkout von Dateien vom CVS wird deren Ausführungs-Berechtigungs-Bit beibehalten.

Wenn Sie Berechtigungsprobleme bei der Ausführung von Dateien aus dem CVS bemerken, z.B. bei der Datei "dateiname", ändern Sie die Berechtigungen im entsprechenden CVS-Depot wie hier, um die Probleme zu beheben:

```
# chmod ugo-x dateiname
```

10.8 Subversion

Subversion ist ein **älteres** Versionskontrollsystem; es ist älter als Git, aber jünger als CVS. Ihm fehlen jedoch Funktionalitäten wie Markierungen (Tags) und Zweige (Branches), die CVS und Git unterstützen.

Sie müssen die Pakete `subversion`, `libapache2-mod-svn` und `subversion-tools` installieren, um einen Subversion-Server einzurichten.

10.8.1 Konfiguration eines Subversion-Depots

Derzeit richtet das `subversion`-Paket kein Depot ein, Sie müssen es manuell erzeugen. Ein möglicher Ort für ein Depot ist `/srv/svn/project`.

Erzeugen Sie ein Verzeichnis wie folgt:

```
# mkdir -p /srv/svn/project
```

Erzeugen Sie die Datenbank des Depots wie hier:

```
# svnadmin create /srv/svn/project
```

10.8.2 Zugriff auf Subversion über einen Apache2-Server

Wenn Sie nur über einen Apache2-Server auf das Subversion-Depot zugreifen wollen, muss das Depot lediglich für den WWW-Server schreibbar sein; führen Sie dazu dies aus:

```
# chown -R www-data:www-data /srv/svn/project
```

Fügen Sie folgendes zu `/etc/apache2/mods-available/dav_svn.conf` hinzu (oder entfernen Sie die Kommentarteichen, falls dies bereits auskommentiert existiert), um den Zugriff auf das Depot über Benutzerauthentifizierung zu erlauben:

```
<Location /project>
  DAV svn
  SVNPath /srv/svn/project
  AuthType Basic
  AuthName "Subversion repository"
  AuthUserFile /etc/subversion/passwd
<LimitExcept GET PROPFIND OPTIONS REPORT>
  Require valid-user
</LimitExcept>
</Location>
```

Erzeugen Sie eine Benutzerauthentifizierungsdatei mit dem Kommando:

```
# htpasswd2 -c /etc/subversion/passwd irgendein-benutzername
```

Starten Sie Apache2 neu.

Ihr neues Subversion-Depot ist jetzt mit `svn(1)` über `"http://localhost/project"` und `"http://example.com/project"` erreichbar (angenommen die URL des Web-Servers ist `"http://example.com/"`).

10.8.3 Lokaler Zugriff auf Subversion durch die Gruppe

Folgendes setzt für das Subversion-Depot den lokalen Zugriff durch eine Gruppe, hier z.B. die Gruppe `project`:

```
# chmod 2775 /srv/svn/project
# chown -R root:src /srv/svn/project
# chmod -R ug+rwX /srv/svn/project
```

Ihr neues Subversion-Depot ist mit `svn(1)` durch die Gruppe über `"file:///localhost/srv/svn/project"` oder `"file:///s"` zugänglich, wenn die lokalen Benutzer Mitglieder der Gruppe `project` sind. Sie müssen Befehle wie `svn`, `svnserve`, `svnlook` und `svnadmin` unter `"umask 002"` ausführen, um den Gruppenzugriff sicherzustellen.

10.8.4 Fernzugriff auf Subversion über SSH

Ein Subversion-Depot mit Gruppen-Zugriffsrechten ist via SSH über `"example.com:/srv/svn/project"` erreichbar, Sie können mit `svn(1)` über `"svn+ssh://example.com:/srv/svn/project"` daraufzugreifen.

10.8.5 Subversion-Verzeichnisstruktur

Viele Projekte verwenden für Subversion Verzeichnisbäume ähnlich dem folgenden, um das Fehlen der Funktionalitäten `Zweig` (Branch) und `Markierung` (Tag) zu kompensieren:

```
----- modul1
| |-- zweige
| |-- markierungen
| | |-- veröffentlichung-1.0
| | |-- veröffentlichung-2.0
| | |
| | |-- trunk
| | |-- datei1
| | |-- datei2
| | |-- datei3
| |
|-- modul2
```

Tipp

Sie müssen den Befehl `"svn copy ..."` benutzen, um Zweige und Markierungen zu kennzeichnen. Dies stellt sicher, dass Subversion die Änderungshistorie der Dateien korrekt aufzeichnet und dabei Speicherplatz spart.

10.8.6 Eine neue Quelle in Subversion importieren

Erzeugen Sie einen neuen lokalen Quellcodebaum unter `"~/pfad/zu/modul1"`, indem Sie dies eingeben:

```
$ mkdir -p ~/pfad/zu/modul1; cd ~/pfad/zu/modul1
```

Legen Sie Dateien in dem neuen Quellcodebaum unter `"~/pfad/zu/modul1"` an.

Importieren Sie sie in Subversion mit folgenden Parametern:

- Modulname: `"modul1"`
 - Subversion Site-URL: `"file:///srv/svn/project"`
 - Subversion-Verzeichnis: `"modul1/trunk"`
-

- Subversion-Tag: "modul1/tags/initiale-Veröffentlichung"

```
$ cd ~/pfad/zu/modul1
$ svn import file:///srv/svn/project/modul1/trunk -m "Start von modul1"
$ svn cp file:///srv/svn/project/modul1/trunk file:///srv/svn/project/modul1/tags/initiale- ←
  Veröffentlichung
```

Alternativ auch:

```
$ svn import ~/pfad/zu/modul1 file:///srv/svn/project/modul1/trunk -m "Start von modul1"
$ svn cp file:///srv/svn/project/modul1/trunk file:///srv/svn/project/modul1/tags/initiale- ←
  Veröffentlichung
```

Tipp

Sie können URLs wie "file:///..." durch jegliches andere URL-Format wie "http:///..." oder "svn+ssh:///..." ersetzen.

10.8.7 Arbeitsablauf bei Subversion

Hier ein Beispiel eines typischen Arbeitsablaufs mit Subversion und seinem nativen Client.

Tipp

Client-Befehle, die durch das `git-svn`-Paket bereitgestellt werden, könnten durch die Nutzung von `git` einen alternativen Arbeitsablauf ermöglichen. Lesen Sie dazu Abschnitt [10.6.4](#).

Überprüfen Sie wie folgt alle verfügbaren Module aus dem Subversion-Projekt, auf das die URL "file:///srv/svn/project" verweist:

```
$ svn list file:///srv/svn/project
modul1
modul2
...
```

Führen Sie einen Checkout von "modul1/trunk" in ein Verzeichnis "modul1" durch:

```
$ cd ~/pfad/zu
$ svn co file:///srv/svn/project/modul1/trunk modul1
$ cd modul1
```

Führen Sie nötige Änderungen am Inhalt durch ...

Überprüfen Sie Ihre Änderungen über einen Befehl ähnlich zu "diff -u [depot] [lokal]":

```
$ svn diff
```

Sie stellen fest, dass Sie eine Datei "file_to_undo" ernsthaft beschädigt haben, aber andere Dateien sind in Ordnung.

Überschreiben Sie die Datei "file_to_undo" mit einer sauberen Kopie aus Subversion:

```
$ svn revert file_to_undo
```

Sichern Sie den aktualisierten lokalen Quellcodebaum nach Subversion durch:

```
$ svn ci -m "Beschreibung der Änderungen"
```

Erzeugen Sie eine Datei "file_to_add" und fügen Sie sie zu Subversion hinzu:

```
$ vi file_to_add
$ svn add file_to_add
$ svn ci -m "Datei file_to_add hinzugefügt"
```

Führen Sie die aktuellste Version aus dem Subversion-Depot mit Ihrer lokalen Kopie zusammen:

```
$ svn up
```

Achten Sie auf Zeilen, die mit "C dateiname" beginnen; dies zeigt sich widersprechende Änderungen an.

Suchen Sie nach unveränderten Code-Teilen, z.B. in "filename.r6", "filename.r9" und "filename.mine".

Suchen Sie nach "<<<<<<" und ">>>>>>" in den Dateien, bei denen sich widersprechende Änderungen festgestellt wurden ("C dateiname").

Bearbeiten Sie die Dateien falls nötig, um Konflikte zu bereinigen.

Fügen Sie wie folgt eine Markierung (Tag) "Veröffentlichung-1" für die Version hinzu:

```
$ svn ci -m "Letzter Commit für Veröffentlichung-1"
$ svn cp file:///srv/svn/project/modul1/trunk file:///srv/svn/project/modul1/tags/ ↵
  Veröffentlichung-1
```

Weitere Änderungen

Entfernen Sie die "Veröffentlichung-1"-Markierung wie folgt:

```
$ svn rm file:///srv/svn/project/modul1/tags/Veröffentlichung-1
```

Sichern Sie Ihre Änderungen in das Subversion-Depot:

```
$ svn ci -m "Wirklich letzter Commit für Veröffentlichung-1"
```

Fügen Sie die "Veröffentlichung-1"-Markierung erneut aus dem aktualisierten Subversion-HEAD von trunk hinzu:

```
$ svn cp file:///srv/svn/project/modul1/trunk file:///srv/svn/project/modul1/tags/ ↵
  Veröffentlichung-1
```

Erzeugen Sie einen neuen Zweig mit dem Pfad "modul1/branches/initiale-Veröffentlichung-Fehlerkorrekturen" aus der originalen Version, auf die der Pfad "modul1/tags/initiale-Veröffentlichung" zeigt, und sichern Sie ihn in das "~/pfad/zu/alt"-Verzeichnis, wie hier:

```
$ svn cp file:///srv/svn/project/modul1/tags/initiale-Veröffentlichung file:///srv/svn/ ↵
  project/modul1/branches/initiale-Veröffentlichung-Fehlerkorrekturen
$ cd ~/pfad/zu
$ svn co file:///srv/svn/project/modul1/branches/initiale-Veröffentlichung- ↵
  Fehlerkorrekturen alt
$ cd alt
```

Tipp

Verwenden Sie "modul1/trunk@{2005-12-20}" (Datumsformat gemäß [ISO 8601](#)) statt "modul1/tags/initiale-Veröffentlichung", um ein spezielles Datum für den Zeitpunkt des Zweigs festzulegen.

Arbeiten Sie weiter an dem lokalen Quellcodebaum, der auf "initiale-Veröffentlichung-Fehlerkorrekturen" verweist und der auf der originalen Version beruht.

Arbeiten Sie selbst an diesem Zweig ...bis jemand weiteres zu diesem "initiale-Veröffentlichung-Fehlerkorrekturen" Zweig hinstößt.

Synchronisieren Sie die Dateien in diesem Zweig, die von anderen modifiziert wurden, wie hier:

```
$ svn up
```

Bearbeiten Sie die Dateien falls nötig, um Konflikte zu bereinigen.

Sichern Sie Ihre Änderungen in das Subversion-Depot:

```
$ svn ci -m "In diesen Zweig gesichert"
```

Aktualisieren Sie den lokalen Baum mit HEAD von trunk wie folgt:

```
$ svn switch file:///srv/svn/project/modul1/trunk
```

Aktualisieren Sie den lokalen Baum (Kontext = HEAD von trunk), indem Sie ihn mit dem Zweig "initiale-Veröffentlichung-" zusammenführen:

```
$ svn merge file:///srv/svn/project/modul1/branches/initiale-Veröffentlichung-   
Fehlerkorrekturen
```

Beheben Sie Konflikte mit einem Editor.

Sichern Sie Ihre Änderungen in das Subversion-Depot:

```
$ svn ci -m "initiale-Veröffentlichung-Fehlerkorrekturen zusammengeführt"
```

Erstellen Sie ein Archiv, wie folgt:

```
$ cd ..  
$ mv alt alt-modul1-korrekturen  
$ tar -cvzf alt-modul1-korrekturen.tar.gz alt-modul1-korrekturen  
$ rm -rf alt-modul1-korrekturen
```

Tipp

Sie können URLs wie "file:///..." durch jegliches andere URL-Format wie "http:///..." oder "svn+ssh:///..." ersetzen.

Tipp

Sie können einen Checkout von nur einem Unterverzeichnis von "modul1" durchführen, indem Sie seinen Namen angeben, z.B. "svn co file:///srv/svn/project/modul1/trunk/unterverzeichnis modul1/unterverzeichnis".

Option	Bedeutung
--dry-run	Probelauf, ohne Wirkung
-v	detaillierte Meldungen der svn-Aktivitäten anzeigen

Tabelle 10.15: Erwähnenswerte Optionen für Subversion-Befehle (als erste Argumente bei svn(1)) zu verwenden:

Kapitel 11

Datenkonvertierung

Hier werden Werkzeuge und Tipps für die Umwandlung von Dateien in andere Formate beschrieben.

Bei Werkzeugen für auf Standards basierende Formate ist die Situation sehr gut, aber bei proprietären Formaten sind die Möglichkeiten eingeschränkt.

11.1 Werkzeuge für Textkonvertierung

Folgende Pakete zur Textkonvertierung sind mir aufgefallen:

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
libc6	V:935, I:999	12771	charset	Konvertierung der Textkodierung zwischen verschiedenen Gebietsschemata (Locales) mit iconv(1) (elementar)
recode	V:3, I:25	603	charset+eol	Konvertierung der Textkodierung zwischen verschiedenen Gebietsschemata (Locales) (vielfältig, mehr Alias-Befehle und Funktionalitäten)
konwert	V:1, I:54	134	charset	Konvertierung der Textkodierung zwischen verschiedenen Gebietsschemata (Locales) (extravagant)
nkf	V:0, I:11	358	charset	Zeichensatzkonvertierer für Japanisch
tcs	V:0, I:0	518	charset	Zeichensatzkonvertierer
unaccent	V:0, I:0	29	charset	akzentuierte Buchstaben durch ihre nicht akzentuierten Pendants ersetzen
tofrodos	V:1, I:25	55	eol	Konvertiert Textformate zwischen DOS und Unix: fromdos(1) und todos(1)
macutils	V:0, I:1	298	eol	Konvertiert Textformate zwischen Macintosh und Unix: frommac(1) und tomac(1)

Tabelle 11.1: Liste von Textkonvertierungs-Werkzeugen

11.1.1 Konvertieren einer Textdatei mit iconv

Tipp

iconv(1) ist Teil des libc6-Pakets und immer auf nahezu allen Unix-artigen Systemen für die Änderung der Zeichenkodierung verfügbar.

Sie können die Kodierung einer Textdatei wie folgt mit `iconv(1)` ändern:

```
$ iconv -f kodierung1 -t kodierung2 input.txt >output.txt
```

Bei den Werten für die Kodierung ist die Groß-/Kleinschreibung nicht relevant, „-“ und „_“ werden ignoriert. Mit `iconv -l` können Sie überprüfen, welche Kodierungen unterstützt werden.

Wert für Zeichenkodierung	Verwendung
ASCII	American Standard Code for Information Interchange , 7-Bit-Code ohne akzentuierte Zeichen
UTF-8	aktueller multilingualer Standard für alle modernen Betriebssysteme
ISO-8859-1	alter Standard für westeuropäische Sprachen; ASCII + akzentuierte Zeichen
ISO-8859-2	alter Standard für osteuropäische Sprachen; ASCII + akzentuierte Zeichen
ISO-8859-15	alter Standard für westeuropäische Sprachen; ISO-8859-1 + Euro-Zeichen
CP850	Codepage 850, Microsoft-DOS-Zeichen mit Grafiken für westeuropäische Sprachen, Variante von ISO-8859-1
CP932	Codepage 932, Shift-JIS -Variante für Japanisch (angelehnt an Microsoft Windows)
CP936	Codepage 936, GB2312 -, GBK - oder GB18030 -Variante für vereinfachtes Chinesisch (angelehnt an Microsoft Windows)
CP949	Codepage 949, EUC-KR - oder Unified-Hangul-Code-Variante für Koreanisch (angelehnt an Microsoft Windows)
CP950	Codepage 950, Big5 -Variante für traditionelles Chinesisch (angelehnt an Microsoft Windows)
CP1251	Codepage 1251, Kodierung für das kyrillische Alphabet (angelehnt an Microsoft Windows)
CP1252	Codepage 1252, ISO-8859-15 -Variante für westeuropäische Sprachen (angelehnt an Microsoft Windows)
KOI8-R	alter russischer UNIX-Standard für das kyrillische Alphabet
ISO-2022-JP	Standard-Kodierung für japanische E-Mails, die nur 7-Bit-Codes verwenden
eucJP	alter japanischer UNIX-Standard-Code (8-Bit), völlig verschieden von Shift-JIS
Shift-JIS	Standard für Japanisch gemäß JIS X 0208 Anhang 1 (siehe auch CP932)

Tabelle 11.2: Liste von Werten für die Zeichenkodierung und deren Verwendung

Anmerkung

Einige der obigen Kodierungen werden nur für die Konvertierung unterstützt und nicht als Wert für das Gebietschema (Locale) (Abschnitt [8.4.1](#)).

Bei Zeichensätzen, die nur ein einziges Byte benötigen (wie [ASCII](#) und [ISO-8859](#)), entspricht die [Zeichenkodierung](#) nahezu dem Zeichensatz.

Bei Zeichensätzen mit vielen Zeichen (wie [JIS X 0213](#) für Japanisch oder [Universal Character Set \(UCS, Unicode, ISO-10646-1\)](#) für praktisch alle Sprachen) gibt es viele Kodierungsschemata, die in die Sequenz der Byte-Daten eingepasst werden:

- [EUC](#) und [ISO/IEC 2022](#) (auch bekannt als [JIS X 0202](#)) für Japanisch;
- [UTF-8](#), [UTF-16/UCS-2](#) und [UTF-32/UCS-4](#) für Unicode.

Bei diesen gibt es klare Differenzierungen zwischen Zeichensatz und Zeichenkodierung.

Die [Codepage](#) wird als Synonym für einige hersteller-spezifische Zeichenkodierungstabellen verwendet.

Anmerkung

Bitte beachten Sie, dass die meisten Kodierungssysteme sich bei den 7-Bit-Zeichen identischen Code mit ASCII teilen, aber es gibt einige Ausnahmen. Wenn Sie alte japanische C-Programme und URLs aus dem Shift-JIS genannten Kodierungsformat nach UTF-8 konvertieren, müssen Sie "CP932" als Kodierungsname statt "shift-JIS" verwenden, um die erwarteten Resultate zu bekommen: 0x5C → "\"" und 0x7E → "~". Andernfalls werden diese falsch konvertiert.

Tipp

recode(1) kann ebenfalls verwendet werden und bietet mehr als die kombinierte Funktionalität von iconv(1), fromdos(1), todos(1), frommac(1) und tomac(1). Weitere Informationen finden Sie unter "info recode".

11.1.2 Prüfen mit iconv, ob eine Datei UTF-8-kodiert ist

Sie können mit iconv(1) wie folgt überprüfen, ob eine Textdatei in UTF-8 kodiert ist:

```
$ iconv -f utf8 -t utf8 input.txt >/dev/null || echo "non-UTF-8 found"
```

Tipp

Verwenden Sie die Option "--verbose" in obigem Beispiel, um das erste nicht in UTF-8 kodierte Zeichen zu finden.

11.1.3 Dateinamen konvertieren mit iconv

Hier ein Beispielskript, um die Kodierung für alle Dateinamen in einem Verzeichnis von einer auf einem älteren Betriebssystem erzeugten Form in das moderne UTF-8 zu konvertieren:

```
#!/bin/sh
ENCDN=iso-8859-1
for x in *;
do
  mv "$x" "$(echo "$x" | iconv -f $ENCDN -t utf-8)"
done
```

Die "\$ENCDN"-Variable gibt dabei die Original-Kodierung (gemäß Tabelle 11.2) an, die in dem älteren Betriebssystem für die Dateinamen verwendet wurde.

In komplizierteren Fällen binden Sie bitte ein Dateisystem, das solche Dateinamen enthält (z.B. eine Festplattenpartition), mit korrekter Angabe der Dateinamenkodierung als Option zum mount(8)-Befehl ein (lesen Sie dazu Abschnitt 8.4.6), und kopieren Sie mit "cp -a" den vollständigen Inhalt der Partition in ein anderes Dateisystem, das als UTF-8 eingebunden ist.

11.1.4 EOL-Konvertierung

Das Format einer Textdatei, speziell der EOL-Code (end-of-line, Zeilenende), ist abhängig von der Systemplattform.

Die Programme fromdos(1), todos(1), frommac(1) und tomac(1) zur Konvertierung des EOL-Formats sind ziemlich praktisch. recode(1) ist ebenfalls sehr nützlich.

Plattform	EOL-Code	Steuerung	dezimal	hexadezimal
Debian (Unix)	LF	^J	10	0A
MSDOS und Windows	CR-LF	^M^J	13 10	0D 0A
Apples Macintosh	CR	^M	13	0D

Tabelle 11.3: Liste der EOL-Codes für verschiedene Plattformen

Anmerkung

Einige Daten im Debian-System, wie z.B. die Daten zur Wiki-Seite für das `python-moinmoin`-Paket, nutzen CR-LF gemäß MSDOS-Art als EOL-Code. Daher sind obige Aussagen nur als allgemeiner Grundsatz zu verstehen.

Anmerkung

Die meisten Editoren (wie `vim`, `emacs`, `gedit`, ...) können mit Dateien mit EOL-Code im MSDOS-Stil transparent umgehen, ohne dass Sie es merken.

Tipp

Um eine Mischung aus MSDOS- und Unix-artigem EOL-Stil (z.B. nach der Zusammenführung von zwei Dateien im MSDOS-Stil mit `diff3(1)`) in einen einheitlichen MSDOS-Stil umzuwandeln, ist die Verwendung von `"sed -e '\r$/!s/$/\r/'"` der von `todos(1)` vorzuziehen. Der Grund hierfür ist, dass `todos` jeder Zeile ein CR hinzufügt.

11.1.5 TAB-Konvertierung

Es gibt ein paar bekannte spezialisierte Programme für die Konvertierung der TAB-Codes:

Funktion	<code>bsdmainutils</code>	<code>coreutils</code>
TAB in Leerzeichen wandeln	<code>"col -x"</code>	<code>expand</code>
Leerzeichen in TAB wandeln	<code>"col -h"</code>	<code>unexpand</code>

Tabelle 11.4: Liste der Befehle zur TAB-Konvertierung aus den Paketen `bsdmainutils` und `coreutils`

`indent(1)` aus dem `indent`-Paket formatiert alle Whitespaces (Leerraumzeichen) in einem C-Programm neu.

Auch Editoren wie `vim` und `emacs` können zur TAB-Konvertierung genutzt werden. Bei `vim` z.B. verwenden Sie die Befehlssequenz `":set expandtab" + ":%retab"`, um ein TAB zum Leerzeichen zu expandieren. Den umgekehrten Fall erreichen Sie mit `":set noexpandtab" + ":%retab!"`.

11.1.6 Editoren mit automatischer Konvertierung

Moderne Editoren wie `vim` sind sehr clever und können mit jeglichen Kodierungssystemen und Dateiformaten umgehen. Für beste Kompatibilität sollten Sie diese Editoren mit einem UTF-8-Gebietsschema in einer UTF-8-tauglichen Konsole verwenden.

Eine alte westeuropäische Unix-Textdatei `"u-datei.txt"`, gespeichert in der alten latin1-Kodierung (ISO-8859-1), kann mit `vim` einfach wie folgt bearbeitet werden:

```
$ vim u-datei.txt
```

Dies ist möglich, da die automatische Erkennung der Dateikodierung in `vim` zunächst von einer UTF-8-Kodierung ausgeht und, falls dies fehlschlägt, latin1 verwendet.

Eine alte polnische Unix-Textdatei `"pu-datei.txt"`, gespeichert in der alten latin2-Kodierung (ISO-8859-2), kann mit `vim` wie folgt bearbeitet werden:

```
$ vim '+e ++enc=latin2 pu-datei.txt'
```

Eine alte japanische Unix-Textdatei "ju-datei.txt", gespeichert in der eucJP-Kodierung, kann mit `vim` wie folgt bearbeitet werden:

```
$ vim '+e ++enc=eucJP ju-datei.txt'
```

Eine alte japanische MS-Windows-Textdatei "jw-datei.txt", gespeichert in der sogenannten Shift-JIS-Kodierung (präziser: CP932), kann mit `vim` wie folgt bearbeitet werden:

```
$ vim '+e ++enc=CP932 ++ff=dos jw-datei.txt'
```

Wenn eine Datei mit den Optionen "++enc" und "++ff" geöffnet wird, speichert ":w" in der Vim-Befehlszeile sie im Originalformat ab und überschreibt die Originaldatei. Sie können auch das zum Speichern zu nutzende Format und den Dateinamen mit angeben, z.B. ":w ++enc=utf8 neu.txt".

Bitte lesen Sie den Abschnitt zum "Multi-byte text support" (mbyte.txt) in der `vim`-Online-Hilfe sowie Tabelle 11.2 für Infos, welche Locale-Werte mit der "++enc"-Option genutzt werden können.

Die `emacs`-Programmfamilie bietet ähnliche Funktionalitäten.

11.1.7 Extrahieren von reinem Text

Folgender Befehl liest eine Webseite in eine Textdatei ein. Das ist sehr nützlich, wenn Sie Konfigurationsbeispiele aus dem Web kopieren oder grundlegende Unix-Textbearbeitungswerkzeuge wie `grep(1)` auf den Inhalt der Webseite anwenden möchten.

```
$ w3m -dump http://www.remote-site.com/help-info.html >textdatei
```

Ähnlich dazu können Sie reine Textdaten wie folgt aus anderen Formaten extrahieren:

Paket	Popcon	Größe	Schlüsselwort	Funktion
w3m	V:31, I:284	2289	html → text	HTML-zu-Text-Konvertierung mit dem Befehl "w3m -dump"
html2text	V:3, I:33	274	html → text	fortgeschrittener HTML-zu-Text-Konvertierer (ISO 8859-1)
lynx	V:13, I:98	1948	html → text	HTML-zu-Text-Konvertierung mit dem Befehl "lynx -dump"
elinks	V:6, I:28	1767	html → text	HTML-zu-Text-Konvertierung mit dem Befehl "elinks -dump"
links	V:6, I:39	2249	html → text	HTML-zu-Text-Konvertierung mit dem Befehl "links -dump"
links2	V:1, I:15	5417	html → text	HTML-zu-Text-Konvertierung mit dem Befehl "links2 -dump"
antiword	V:2, I:10	589	MSWord → text,ps	Konvertierung von MSWord-Dateien in reinen Text oder PostScript
catdoc	V:27, I:127	675	MSWord → text,TeX	Konvertierung von MSWord-Dateien in reinen Text oder TeX
pstotext	V:1, I:3	126	ps/pdf → text	Extrahierung von reinem Text aus PostScript- oder PDF-Dateien
unhtml	V:0, I:0	43	html → text	Entfernen der Markups (Markierungen) aus einer HTML-Datei
odt2txt	V:1, I:7	60	odt → text	Konvertierung von OpenDocument-Text in reinen Text

Tabelle 11.5: Liste von Werkzeugen zum Extrahieren von reinen Textdaten

11.1.8 Hervorheben und Formatieren von reinen Textdaten

Reine Textdaten können mit folgenden Befehlen hervorgehoben und formatiert werden:

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
vim-runtime	V:19, I:435	31723	Hervorheben	Vim-Macro zur Konvertierung von Quellcode nach HTML (mit <code>":source \$VIMRUNTIME/syntax/html.vim"</code>)
cxref	V:0, I:0	1193	c → html	Konvertierung eines C-Programms nach Latex und HTML (C-Sprache)
src2tex	V:0, I:0	622	Hervorheben	Konvertierung von vielen Quellcode-Formaten nach TeX (C-Sprache)
source-highlight	V:0, I:7	1992	Hervorheben	Konvertierung von vielen Quellcode-Formaten nach HTML, XHTML, LaTeX, Texinfo, ANSI-Color-Escape-Sequenzen und DocBook-Dateien mit Hervorhebung (C++)
highlight	V:1, I:12	1083	Hervorheben	Konvertierung von vielen Quellcode-Formaten nach HTML, XHTML, RTF, LaTeX, TeX oder XSL-FO-Dateien mit Hervorhebung (C++)
grc	V:0, I:3	190	text → color	grundlegender Einfärber für alles (Python)
txt2html	V:0, I:3	259	text → html	Text-nach-HTML-Konvertierer (Perl)
markdown	V:0, I:8	57	text → html	markdown: Textdokument-Konvertierer in (X)HTML (Perl)
asciidoc	I:13	81	text → any	AsciiDoc: Textdokument-Konvertierer in XML/HTML (Python)
pandoc	V:8, I:47	151714	text → any	grundlegender Markup-Konvertierer (Haskell)
python-docutils	V:12, I:133	1771	text → any	reStructuredText-Dokument-Formatierer nach XML (Python)
txt2tags	V:0, I:1	342	text → any	Dokumentenkonvertierer von Text nach HTML, SGML, LaTeX, manpage (Handbuchseite), MoinMoin, Magic Point und PageMaker (Python)
udo	V:0, I:0	583	text → any	universelles Dokumentenverarbeitungs-Werkzeug (C-Sprache)
stx2any	V:0, I:0	264	text → any	Dokumentenkonvertierer von Structured Text in andere Formate (m4)
rest2web	V:0, I:0	527	text → html	Dokumentenkonvertierer von ReStructured Text nach HTML (Python)
aft	V:0, I:0	235	text → any	"Freiform"-Dokumenten-Vorbereitungssystem (Perl)
yodl	V:0, I:0	610	text → any	Vordokument-Sprache und Werkzeuge für deren Verarbeitung (C-Sprache)
sdf	V:0, I:0	1445	text → any	einfacher Dokumentenanalysator (Perl)
sisu	V:0, I:0	5344	text → any	Rahmenwerk für Dokumentenstrukturierung, -veröffentlichung und -suche (Ruby)

Tabelle 11.6: Liste von Werkzeugen für Hervorhebung/Formatierung von Textdaten

11.2 XML-Daten

Die [Extensible Markup Language \(XML\)](#) ist eine Markup-Sprache für Dokumente mit Strukturinformationen.

Einführende Informationen finden Sie unter [XML.COM](#):

- ["Was ist XML? \(englisch\)"](#)
- ["Was ist XSLT? \(englisch\)"](#)

- ["Was ist XSL-FO? \(englisch\)"](#)
- ["Was ist XLink? \(englisch\)"](#)

11.2.1 Grundlegende Hinweise für XML

Ein XML-Text sieht ein wenig wie [HTML](#) aus. XML ermöglicht es uns, verschiedene Ausgabeformate für ein und dasselbe Dokument zu verwalten. Ein einfaches XML-System ist `docbook-xsl`, das auch für dieses Dokument verwendet wird.

Jede XML-Datei beginnt mit einer Standard-XML-Deklaration wie der folgenden:

```
<?xml version="1.0" encoding="UTF-8"?>
```

Die grundlegende Syntax für ein XML-Element ist wie folgt gekennzeichnet:

```
<name attribute="Wert">Inhalt</name>
```

Die Kurzform für ein XML-Element mit leerem Inhalt ist wie folgt:

```
<name attribute="Wert"/>
```

Das `attribute="Wert"` in obigen Beispielen ist optional.

Ein Kommentar-Abschnitt wird in XML wie folgt gekennzeichnet:

```
<!-- Kommentar -->
```

Anders als beim Hinzufügen von Markierungen (Markups) erfordert XML minimale Konvertierungen am Inhalt, um vordefinierte Entitäten für die folgenden Zeichen zu nutzen:

vordefinierte Entität	Zeichen, in das konvertiert werden soll
<code>&quot;</code>	" (Anführungszeichen)
<code>&apos;</code>	' (Apostroph)
<code>&lt;</code>	< (kleiner-als)
<code>&gt;</code>	> (größer-als)
<code>&amp;</code>	& (kaufmännisches Und)

Tabelle 11.7: Liste von vordefinierten Entitäten für XML



Achtung

"<" oder "&" können nicht in Attributen oder Elementen verwendet werden.

Anmerkung

Wenn anwenderdefinierte Entitäten im SGML-Stil verwendet werden, wie z.B. `"&irgendein-begriff:"`, wird die erste Definition gegenüber darauffolgenden bevorzugt. Die Entität wird in Form von `"<!ENTITY irgendein-begriff "Wert der Entität">"` definiert.

Anmerkung

Solange die XML-Markierungen konsistent mit einer bestimmten Art von Namen für diese Markierungen (entweder Daten als Inhalt oder Attributwert) ausgeführt sind, ist die Konvertierung in eine andere XML-Form mittels [Extensible Stylesheet Language Transformations \(XSLT\)](#) eine banale Aufgabe.

11.2.2 XML-Verarbeitung

Es gibt viele Werkzeuge zur Verarbeitung von XML-Dateien, wie z.B. die [Extensible Stylesheet Language \(XSL\)](#).

Grundsätzlich ist es so, dass Sie eine korrekt formatierte XML-Datei mittels [Extensible Stylesheet Language Transformation \(XSLT\)](#) in jegliches Format umwandeln können.

[Extensible Stylesheet Language for Formatting Objects \(XSL-FO\)](#) ist eine Lösung zur Formatierung. Das `fop`-Paket ist (aufgrund seiner Abhängigkeit zur [Java-Programmiersprache](#)) neu in Debians `main`-Archiv. Daher wird im Allgemeinen LaTeX-Code mittels XSLT aus XML erstellt und das LaTeX-System dann verwendet, um druckfähige Dateien wie DVI, PostScript oder PDF zu erzeugen.

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
docbook-xml	V:15, I:280	2133	xml	XML-Dokumententyp-Definition (DTD) für DocBook
xsltproc	V:15, I:104	160	xslt	XSLT-Befehlszeilen-Prozessor (XML → XML, HTML, reinen Text usw.)
docbook-xsl	V:13, I:165	14870	xml/xslt	XSL-Stylesheets (Stilvorlagen), um DocBook-XML mittels XSLT in verschiedene Ausgabeformate umzuwandeln
xmlto	V:1, I:23	130	xml/xslt	XML-nach-alles-Konvertierer mit XSLT
dbtoepub	V:0, I:0	37	xml/xslt	DocBook-XML-nach-epub-Konvertierer
dblatex	V:3, I:16	4643	xml/xslt	DocBook-Dateien mittels XSLT nach DVI, PostScript und PDF konvertieren
fop	V:1, I:24	291	xml/xsl-fo	Docbook-XML-Dateien nach PDF konvertieren

Tabelle 11.8: Liste von XML-Werkzeugen

Da XML eine Untermenge der [Standard Generalized Markup Language \(SGML\)](#) ist, kann es mit den umfangreichen Werkzeugen, die für SGML verfügbar sind (wie der [Document Style Semantics and Specification Language \(DSSSL\)](#)) verarbeitet werden.

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
openjade	V:2, I:38	1019	dsssl	ISO/IEC 10179:1996 - standardkonformer DSSSL-Prozessor (aktuell)
docbook-dsssl	V:1, I:23	2604	xml/dsssl	DSSSL-Stylesheets (Stilvorlagen), um DocBook-XML mittels DSSSL in verschiedene Ausgabeformate umzuwandeln
docbook-utils	V:0, I:16	281	xml/dsssl	Werkzeuge für DocBook-Dateien (inklusive Konvertierung in andere Formate (HTML, RTF, PS, man, PDF) mit DSSSL mittels <code>docbook2*</code> -Befehlen)
sgml2x	V:0, I:0	90	SGML/dsssl	Konvertierer von SGML und XML mittels DSSSL-Stylesheets (Stilvorlagen)

Tabelle 11.9: Liste von DSSSL-Werkzeugen

Tipp

[GNOME](#)s `ye1p` ist manchmal praktisch, da es [DocBook](#)-XML-Dateien ohne Konvertierung direkt vernünftig darstellen kann.

11.2.3 Extrahierung von XML-Daten

Sie können HTML- oder XML-Daten mit folgenden Programmen aus anderen Formaten extrahieren:

HTML-Dateien (nicht-XML) können Sie nach XHTML konvertieren, was eine Instanz von korrekt formatiertem XML ist. XHTML kann von XML-Werkzeugen verarbeitet werden.

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
wv	V:0, I:7	717	MSWord → alle	Dokumentenkonvertierer von Microsoft Word nach HTML, LaTeX usw.
texi2html	V:0, I:8	1833	texi → html	Konvertierer von Texinfo nach HTML
man2html	V:0, I:2	138	manpage → html	Konvertierer von Handbuchseite (manpage) nach HTML (CGI-Unterstützung)
unrtf	V:0, I:4	148	rtf → html	Dokumentenkonvertierer von RTF nach HTML usw.
info2www	V:1, I:3	76	info → html	Konvertierer von GNU info nach HTML (CGI-Unterstützung)
ooo2dbk	V:0, I:0	217	sxw → xml	Konvertierer von OpenOffice.org-SXW-Dokumenten nach DocBook XML
wp2x	V:0, I:0	202	WordPerfect → any	WordPerfect-5.0- und -5.1-Dateien nach TeX, LaTeX, troff, GML und HTML konvertieren
doclifter	V:0, I:0	451	troff → xml	Konvertierer von troff nach DocBook XML

Tabelle 11.10: Liste von Werkzeugen zur Extrahierung von XML-Daten

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
libxml2-utils	V:22, I:246	182	xml ↔ html ↔ xhtml	Befehlszeilen-XML-Werkzeug mit xmllint(1) (Syntaxüberprüfung, Neuformatierung, ...)
tidy	V:2, I:14	84	xml ↔ html ↔ xhtml	HTML-Syntaxüberprüfung und Neuformatierung

Tabelle 11.11: Liste von XML-Druck-Werkzeugen

Sobald eine saubere XML-Basis generiert wurde, können Sie die XSLT-Technologie nutzen, um Daten basierend auf dem Markup-Kontext zu extrahieren usw.

11.3 Textsatz

Das Unix-Programm [troff](#), ursprünglich von AT&T entwickelt, kann für einfachen Textsatz verwendet werden. Es wird normalerweise genutzt, um Handbuchseiten (manpages) zu erzeugen.

[TeX](#), entwickelt von Donald Knuth, ist ein sehr leistungsfähiges Textsatz-Werkzeug und der De-Facto-Standard. [LaTeX](#) (ursprünglich geschrieben von Leslie Lamport) ermöglicht einen sehr hochentwickelten Zugriff auf die Fähigkeiten von TeX.

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
texlive	V:3, I:50	71	(La)TeX	TeX-System für Textsatz, Vorschau und Druck
groff	V:3, I:64	11838	troff	GNU troff Textformatierungs-System

Tabelle 11.12: Liste von Textsatz-Werkzeugen

11.3.1 roff-Textsatz

Traditionell ist [roff](#) das Haupt-Unix-System zur Textverarbeitung. Lesen Sie [roff\(7\)](#), [groff\(7\)](#), [groff\(1\)](#), [grotty\(1\)](#), [troff\(1\)](#), [groff_mdoc\(7\)](#), [groff_man\(7\)](#), [groff_ms\(7\)](#), [groff_me\(7\)](#), [groff_mm\(7\)](#) und "info groff".

Sie bekommen eine gute Einführung und Referenz zum "-me"-[Makro](#) in "/usr/share/doc/groff/", wenn Sie das [groff](#)-Paket installiert haben.

Tipp

"groff -Tascii -me -" erzeugt reinen Text mit [ANSI-Escape-Sequenzen](#). Wenn Sie eine Ausgabe ähnlich zu den Handbuchseiten mit vielen "^H" und "_" möchten, verwenden Sie stattdessen "GROFF_NO_SGR=1 groff -Tascii -me -".

Tipp

Um "^H" und "_" aus einer mit groff erzeugten Textdatei zu entfernen, filtern Sie diese mit "col -b -x".

11.3.2 TeX/LaTeX

Die [TeX Live](#)-Software-Distribution stellt ein vollständiges TeX-System bereit. Das `texlive`-Metapaket enthält eine sinnvolle Auswahl von [TeX Live](#)-Paketen, die für die meisten Aufgaben ausreichend sein sollten.

Es gibt viele Ressourcen für [TeX](#) und [LaTeX](#):

- [The teTeX HOWTO: The Linux-teTeX Local Guide](#);
- `tex(1)`;
- `latex(1)`;
- `texdoc(1)`;
- `texdoctk(1)`;
- "The TeXbook", von Donald E. Knuth (Addison-Wesley);
- "LaTeX - A Document Preparation System", von Leslie Lamport (Addison-Wesley);
- "The LaTeX Companion", von Goossens, Mittelbach, Samarin (Addison-Wesley).

TeX/LaTeX ist die leistungsfähigste Textsatz-Umgebung. Viele [SGML](#)-Prozessoren nutzen es im Hintergrund zur Textverarbeitung. [Lyx](#) aus dem `lyx`-Paket sowie [GNU TeXmacs](#) aus dem `texmacs`-Paket bieten eine nette [WYSIWYG](#)-Umgebung zum Editieren von [LaTeX](#)-Dokumenten, zu der viele Leute [Emacs](#) oder [Vim](#) als Quelltext-Editor wählen.

Es sind viele Online-Ressourcen verfügbar:

- The TEX Live Guide - TEX Live 2007 ("`/usr/share/doc/texlive-doc-base/english/texlive-en/live.html`" aus dem `texlive-doc-base`-Paket);
- [A Simple Guide to Latex/Lyx](#);
- [Word Processing Using LaTeX](#);
- [Local User Guide to teTeX/LaTeX](#).

Wenn die Dokumente größer werden, kann TeX eventuell Fehler verursachen. Sie müssen dann die Pool-Größe in "`/etc/texmf/texmf.d/95NonPath`" erhöhen (oder editieren Sie besser "`/etc/texmf/texmf.d/95NonPath`" und führen `update-texmf(8)` aus), um dieses Problem zu beheben.

Anmerkung

Der TeX-Quelltext von "The TeXbook" ist unter <http://tug.ctan.org/tex-archive/systems/knuth/dist/tex/texbook.tex> verfügbar. Diese Datei enthält die meisten der benötigten Makros. Ich habe gehört, dass Sie dieses Dokument mit `tex(1)` verarbeiten können, wenn Sie die Zeilen 7 - 10 auskommentieren und "`\input manmac\proofmodefalse`" hinzufügen. Es wird dringend empfohlen, dass Sie dieses Buch (wie auch alle anderen Bücher von Donald E. Knuth) kaufen, statt die Online-Version zu lesen, aber der Quelltext ist ein tolles Beispiel für TeX-Eingaben!

11.3.3 Schöner Ausdruck einer Handbuchseite

Sie können eine Handbuchseite (manpage) in PostScript mit einem der folgenden Befehle schön ausdrucken:

```
$ man -Tps irgendeine_manpage | lpr
```

11.3.4 Erstellen einer Handbuchseite

Obwohl es möglich ist, eine Handbuchseite (manpage) in reinem [troff](#)-Format zu schreiben, gibt es auch einige Pakete mit Hilfsprogrammen zur Erstellung einer Handbuchseite:

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
docbook-to-man	V:0, I:13	191	SGML → manpage	Konvertierer vom DocBook-SGML-Format in roff-man-Makros
help2man	V:0, I:10	498	text → manpage	Automatisch Handbuchseiten aus --help-Ausgabe generieren
info2man	V:0, I:0	134	info → manpage	Konvertierer von GNU-info-Dateien nach POD oder Manpage
txt2man	V:0, I:1	114	text → manpage	Einfachen ASCII-Text in das Manpage-Format umwandeln

Tabelle 11.13: Liste von Paketen, die bei der Erstellung einer Handbuchseite helfen

11.4 Druckfähige Daten

Druckfähige Daten werden im Debian-System im [PostScript](#)-Format dargestellt. [CUPS \(Common Unix Printing System\)](#) verwendet Ghostscript als Rasterizer-Backend-Programm für Drucker, die selbst kein PostScript interpretieren können.

11.4.1 Ghostscript

Der Kern der Verarbeitung von druckfähigen Daten ist ein [Ghostscript-PostScript \(PS\)](#)-Interpreter, der ein Raster-Image erzeugt.

Die letzte Ghostscript-Upstream-Version von Artifex wurde von AFPL nach GPL um-lizenziert und vereinigt in der 8.60-Veröffentlichung alle Änderungen der letzten ESP-Version (wie z.B. solche, die CUPS betreffen).

Paket	Popcon	Größe	Beschreibung
ghostscript	V:252, I:598	231	der GPL Ghostscript-PostScript/PDF-Interpreter
ghostscript-x	V:15, I:65	223	GPL Ghostscript-PostScript/PDF-Interpreter - Unterstützung für Anzeige unter X
libpoppler95	I:3	4172	Bibliothek zur PDF-Darstellung, Abspaltung von dem PDF-Anzeigeprogramm xpdf
libpoppler-glib8	V:217, I:481	449	Bibliothek zur PDF-Darstellung (Laufzeitbibliothek auf Basis von GLib)
poppler-data	V:111, I:637	13090	CMaps für Bibliothek zur PDF-Darstellung (für CJK -Unterstützung: Adobe-*)

Tabelle 11.14: Liste von Ghostscript-PostScript-Interpretern

Tipp

"gs -h" zeigt die Konfiguration von Ghostscript an.

11.4.2 Zwei PS- oder PDF-Dateien zusammenführen

Sie können zwei [PostScript \(PS\)](#)- oder [Portable Document Format \(PDF\)](#)-Dateien mit dem `gs(1)`-Befehl von Ghostscript zusammenführen:

```
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pswrite -sOutputFile=bla.ps -f datei1.ps datei2.ps
$ gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -sOutputFile=bla.pdf -f datei1.pdf datei2.pdf
```

Anmerkung

[PDF](#), ein plattform-unabhängiges Datenformat für druckfähige Daten, ist im Grunde ein komprimiertes [PS](#)-Format mit einigen zusätzlichen Funktionalitäten und Erweiterungen.

Tipp

Auf der Befehlszeile sind `psmerge(1)` und andere Befehle aus dem `psutils`-Paket sehr nützlich zur Bearbeitung von PostScript-Dokumenten. `pdftk(1)` aus dem `pdftk`-Paket ist ebenfalls praktisch, um PDF-Dokumente zu bearbeiten.

11.4.3 Werkzeuge für druckfähige Daten

Folgende Pakete mit Werkzeugen für druckfähige Daten sind mir ins Auge gestochen:

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
poppler-utils	V:241, I:434	689	pdf → ps, text, ...	PDF-Hilfsprogramme: <code>pdftops</code> , <code>pdfinfo</code> , <code>pdfimages</code> , <code>pdftotext</code> , <code>pdf fonts</code>
psutils	V:6, I:105	219	ps → ps	Werkzeuge zur Konvertierung von PostScript-Dokumenten
poster	V:0, I:5	58	ps → ps	Erzeugen großer Poster aus PostScript-Seiten
enscript	V:1, I:20	2132	text → ps, html, rtf	Konvertieren von ASCII-Text nach PostScript, HTML, RTF oder Pretty-Print
a2ps	V:1, I:15	3651	text → ps	'Anything to PostScript'-Konvertierer (jegliches nach PostScript) und Pretty-Printer
pdftk	I:51	28	pdf → pdf	Werkzeug zur Konvertierung von PDF-Dokumenten: <code>pdftk</code>
html2ps	V:0, I:3	249	html → ps	Konvertierer von HTML nach PostScript
gnuhtml2latex	V:0, I:1	27	html → latex	Konvertierer von HTML nach LaTeX
latex2rtf	V:0, I:6	480	latex → rtf	Konvertieren von Dokumenten von LaTeX nach RTF, die dann von MS Word gelesen werden können
ps2eps	V:3, I:68	98	ps → eps	Konvertierer von PostScript nach EPS (Encapsulated PostScript)
e2ps	V:0, I:0	109	text → ps	Text-nach-PostScript-Konvertierer mit Unterstützung für japanische Zeichenkodierung
impose+	V:0, I:0	119	ps → ps	PostScript-Hilfsprogramme
trueprint	V:0, I:0	146	text → ps	Viele Quellformate (C, C++, Java, Pascal, Perl, Pike, Sh und Verilog) nach PostScript konvertieren mittels Pretty-Print (C-Sprache)
pdf2svg	V:0, I:4	30	ps → svg	Konvertierer von PDF in das Scalable Vector Graphics -Format
pdftoipe	V:0, I:0	71	ps → ipe	Konvertierer von PDF in IPE's XML-Format

Tabelle 11.15: Liste von Werkzeugen für druckfähige Daten

11.4.4 Drucken mit CUPS

Die beiden vom [Common Unix Printing System \(CUPS\)](#) angebotenen Befehle `lp(1)` und `lpr(1)` bieten Optionen, um das Drucken von druckfähigen Daten spezifisch anzupassen.

Mit einem der folgenden Befehle können Sie drei Kopien einer Datei auf einmal ausdrucken:

```
$ lp -n 3 -o Collate=True dateiname
```

```
$ lpr -#3 -o Collate=True dateiname
```

Druckoperationen können über Optionen für den Drucker noch weitgehender angepasst werden, z.B. mit `”-o number-up=2”`, `”-o page-set=even”`, `”-o page-set=odd”`, `”-o scaling=200”`, `”-o natural-scaling=200”` usw. Diese Optionen sind dokumentiert unter [Command-Line Printing and Options](#) (wenn Sie CUPS installiert haben).

11.5 Konvertierung von Mail-Daten

Folgende Pakete für die Konvertierung von Mail-Daten sind mir aufgefallen:

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
sharutils	V:4, I:55	1421	mail	<code>shar(1)</code> , <code>unshar(1)</code> , <code>uuencode(1)</code> , <code>uudecode(1)</code>
mpack	V:1, I:18	106	MIME	Kodieren und Dekodieren von MIME -Nachrichten: <code>mpack(1)</code> und <code>munpack(1)</code>
tnef	V:1, I:10	110	ms-tnef	Auspacken von MIME -Anhängen des Typs <code>”application/ms-tnef”</code> (ein nur von Microsoft verwendetes Format)
uudeview	V:0, I:5	109	mail	Kodieren und Dekodieren folgender Formate: uuencode , xxencode , BASE64 , quoted printable und BinHex

Tabelle 11.16: Liste von Paketen zur Konvertierung von Mail-Daten

Tipp

Ein [Internet Message Access Protocol](#)-Server der Version 4 (IMAP4) (Näheres in Abschnitt [6.7](#)) kann verwendet werden, um Mails von proprietären Mail-Systemen zu exportieren; dazu muss der Mail-Client (z.B. unter Windows) so konfiguriert werden, dass er den auf dem Debian-System laufenden IMAP4-Server nutzt.

11.5.1 Grundlagen zu Mail-Daten

Mail-Daten ([SMTP](#)) sollten auf eine Abfolge von 7-Bit-Zeichen beschränkt werden. Daher werden Binärdaten und 8-Bit-Textdaten über [Multipurpose Internet Mail Extensions \(MIME\)](#) und die Auswahl des Zeichensatzes (lesen Sie dazu Abschnitt [8.4.1](#)) im 7-Bit-Format kodiert.

Das Standardformat zum Speichern von Mails ist mbox gemäß [RFC2822 \(aktualisiertes RFC822\)](#). Näheres dazu in `mbox(5)` (aus dem `mutt`-Paket).

Für europäische Sprachen wird normalerweise `”Content-Transfer-Encoding: quoted-printable”` mit dem ISO-8859-1-Zeichensatz bei Mails verwendet, da es dabei nicht viele 8-Bit-Zeichen gibt. Wenn europäischer Text in UTF-8 kodiert ist, wird dabei voraussichtlich `”Content-Transfer-Encoding: quoted-printable”` benutzt, da es überwiegend 7-Bit-Zeichen sind.

Für Japanisch wird gewöhnlich `”Content-Type: text/plain; charset=ISO-2022-JP”` bei Mails verwendet, um den Text im 7-Bit-Format zu halten. Ältere Microsoft-Systeme könnten jedoch Mail-Daten in Shift-JIS versenden, ohne dies korrekt zu deklarieren. Wenn japanischer Text in UTF-8 kodiert ist, wird dabei voraussichtlich [Base64](#) benutzt, da dabei viele 8-Bit-Zeichen enthalten sind. Die Situation bei anderen asiatischen Sprachen ist ähnlich.

Anmerkung

Falls Sie mit einer Debian-fremden Client-Software auf Ihre nicht-Unix-Mail-Daten zugreifen können und dieser Client auch mit einem IMAP4-Server kommunizieren kann, können Sie die Mails von dem Fremdsystem herunterladen, indem Sie einen eigenen IMAP4-Server laufen lassen (lesen Sie dazu Abschnitt [6.7](#)).

Anmerkung

Nutzen Sie andere Formate zur Speicherung Ihrer Mails, ist die Umstellung auf das mbox-Format ein guter erster Schritt. Ein vielseitiges Client-Programm wie z.B. `mutt(1)` kann dabei nützlich sein.

Sie können den Inhalt einer Mailbox auf einzelne Nachrichten aufsplitten, indem Sie `procmail(1)` und `formail(1)` verwenden.

Jede Mail kann mittels `munpack(1)` aus dem `mpack`-Paket (oder mit anderen spezialisierten Werkzeugen) entpackt werden, um die MIME-kodierten Inhalte zu erhalten.

11.6 Werkzeuge für Grafikdaten

Folgende Pakete zur Konvertierung, Bearbeitung und Organisation von grafischen Daten sind mir aufgefallen:

Tipp

Weitere Werkzeuge zur Bildbearbeitung finden Sie mit dem regulären Ausdruck `"~Gworks-with::image"` in `aptitude(8)` (lesen Sie dazu Abschnitt [2.2.6](#)).

Obwohl grafische GUI-Programme wie `gimp(1)` sehr leistungsfähig sind, können auch Befehlszeilenwerkzeuge wie `imagemagick(1)` nützlich sein, um automatisierte Bildbearbeitung über Skripte durchzuführen.

Der De-Facto-Standard für Bilddateien von Digitalkameras ist das [Exchangeable Image File Format \(EXIF\)](#); dies entspricht dem [JPEG](#)-Dateiformat mit zusätzlichen Metainformationen. Es kann auch Informationen wie Datum, Zeit und Kameraeinstellungen speichern.

Das Patent zur [verlustlosen Datenkompression mit dem Lempel-Ziv-Welch-\(LZW-\)Algorithmus](#) ist abgelaufen. [Graphics Interchange Format \(GIF\)](#)-Werkzeuge, die die LZW-Kompressionsmethode nutzen, sind jetzt frei im Debian-System verfügbar.

Tipp

Alle Digitalkameras oder Scanner mit Wechseldatenträgern als Speichermedium funktionieren unter Linux über [USB-Speicher](#)-Lesegeräte, da sie die [Design-Regeln für Kamera-Dateisysteme](#) befolgen und [FAT](#) als Dateisystem verwenden. Näheres finden Sie in Abschnitt [10.1.7](#).

11.7 Verschiedene Datenkonvertierungen

Es gibt viele andere Programme zum Konvertieren von Daten. Folgende Pakete habe ich über den regulären Ausdruck `"~Guse::convert"` in `aptitude(8)` gefunden (Näheres dazu in Abschnitt [2.2.6](#)):

Sie können die Daten aus einem RPM-Archiv auch mit folgendem Befehl extrahieren:

```
$ rpm2cpio datei.src.rpm | cpio --extract
```

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
gimp	V:68, I:341	22313	image(bitmap)	GNU Image Manipulation Program (GIMP)
imagemagick	I:400	218	image(bitmap)	Programme zur Bildbearbeitung
graphicsmagick	V:3, I:17	5224	image(bitmap)	Programme zur Bildbearbeitung (Abspaltung von imagemagick)
xsane	V:17, I:173	2346	image(bitmap)	GTK+-basierte X11-Oberfläche für das Scanner-Programm SANE (Scanner Access Now Easy)
netpbm	V:32, I:409	4302	image(bitmap)	Werkzeuge zur Grafikkonvertierung
icoutils	V:21, I:127	221	png ↔ ico(bitmap)	Konvertieren von MS-Windows-Icons und -Cursor in das und vom PNG-Format (Favicons)
scribus	V:2, I:23	30375	ps/pdf/SVG/ ...	Scribus DTP-Editor
libreoffice-draw	V:177, I:434	14600	image(vector)	LibreOffice Office-Programm - Zeichnen
inkscape	V:55, I:209	84823	image(vector)	SVG (Scalable Vector Graphics) -Editor
dia	V:5, I:31	3727	image(vector)	Diagramm-Editor (Gtk)
xfig	V:2, I:15	1793	image(vector)	Programm zur interaktiven Erzeugung von Objekten in X11
pstoedit	V:4, I:98	988	ps/pdf → image(vector)	PostScript- und PDF-Dateien in editierbare Vector-Grafiken (SVG) konvertieren
libwmf-bin	V:10, I:211	113	Windows/image (vector)	Konvertierungswerkzeuge für Windows-Metadaten (Vector-Grafiken)
fig2sxd	V:0, I:0	149	fig → sxd(vector)	XFig-Dateien in das OpenOffice.org-Draw-Format konvertieren
unpaper	V:2, I:19	460	image → image	Werkzeug zum Nachbearbeiten von eingescannten Seiten für OCR
tesseract-ocr	V:8, I:37	1500	image → text	freie OCR -Software basierend auf HPs kommerzieller OCR-Engine
tesseract-ocr-eng	V:7, I:37	4032	image → text	OCR-Engine-Daten: tesseract-ocr-Sprachdateien für englischen Text
gocr	V:1, I:13	531	image → text	freie OCR-Software
ocrad	V:0, I:5	303	image → text	freie OCR-Software
eog	V:71, I:264	10189	image(Exif)	Bildbetrachter Eye of GNOME
gthumb	V:5, I:22	5475	image(Exif)	Bildbetrachter und -browser (GNOME)
geeqie	V:6, I:21	14643	image(Exif)	Bildbetrachter, der GTK+ verwendet
shotwell	V:19, I:223	6451	image(Exif)	Digitalfoto-Organizer (GNOME)
gtkam	V:0, I:6	1154	image(Exif)	Anwendung zum Herunterladen der Bilder/Videos von Digitalkameras (GTK+)
gphoto2	V:1, I:12	955	image(Exif)	gphoto2-Befehlszeilen-Client für Digitalkameras
gwenview	V:28, I:97	10570	image(Exif)	Bildbetrachter (KDE)
kamera	I:97	798	image(Exif)	Unterstützung von Digitalkameras für KDE-Anwendungen
digikam	V:2, I:13	2646	image(Exif)	Digitalfoto-Verwaltung für KDE
exiv2	V:3, I:44	321	image(Exif)	Werkzeug zur Bearbeitung von Exif-/IPTC-Metadaten
exiftran	V:1, I:21	70	image(Exif)	Programm für die Umwandlung der JPEG-Bilder von Digitalkameras
jhead	V:1, I:11	109	image(Exif)	Bearbeitung des Nicht-Grafik-Teils von Exif-konformen JPEG-Dateien
exif	V:1, I:12	339	image(Exif)	Befehlszeilen-Werkzeug, um EXIF-Informationen von JPEG-Dateien anzuzeigen
exiftags	V:0, I:5	292	image(Exif)	Werkzeug, um Exif-Informationen aus Digitalkamera-JPEG-Dateien auszulesen
exifprobe	V:0, I:4	499	image(Exif)	Metadaten aus Digitalbildern auslesen
dcraw	V:2, I:19	535	image(Raw) → ppm	Dekodierer für Digitalkamerabilder im RAW-Format
findimagedupes	V:0, I:1	79	image → fingerprint	Visuell ähnliche oder doppelte Bilder finden
ale	V:0, I:0	839	image → image	Bilder zusammenfügen, um die Wiedergabetreue zu erhöhen oder Mosaik zu erzeugen
imageindex	V:0, I:2	145	image(Exif) → html	Erzeugen von statischen HTML-Galerien aus Bildern

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
alien	V:2, I:34	161	rpm/tgz → deb	Programm zur Konvertierung von fremden Software-Paketen in das Debian-Paketformat
freepwing	V:0, I:0	421	EB → EPWING	Konvertierer von "Electric Book" (beliebt in Japan) in ein separates JIS X 4081 -Format (eine Untermenge von EPWING V1)
calibre	V:9, I:36	54876	alle → EPUB	E-Book-Konvertierer und Bibliotheksverwaltung

Tabelle 11.18: Liste verschiedener Werkzeuge zur Datenkonvertierung

Kapitel 12

Programmierung

Ich gebe hier einige Startimpulse, um Programmierung im Debian-System zumindest so weit zu erlernen, dass der Programmcode aus Debian-Quellpaketen zwecks Fehlersuche nachverfolgt werden kann. Hier einige erwähnenswerte Pakete und dazugehörige Dokumentation:

Paket	Popcon	Größe	Dokumentation
autoconf	V:41, I:282	1846	"info autoconf" aus dem autoconf-doc-Paket
automake	V:42, I:279	1830	"info automake" aus dem automake1.10-doc-Paket
bash	V:791, I:999	6469	"info bash" aus dem bash-doc-Paket
bison	V:9, I:103	2815	"info bison" aus dem bison-doc-Paket
cpp	V:319, I:770	42	"info cpp" aus dem cpp-doc-Paket
ddd	V:0, I:10	4184	"info ddd" aus dem ddd-doc-Paket
exuberant-ctags	V:5, I:37	341	exuberant-ctags(1)
flex	V:9, I:93	1279	"info flex" aus dem flex-doc-Paket
gawk	V:368, I:454	2558	"info gawk" aus dem gawk-doc-Paket
gcc	V:165, I:604	45	"info gcc" aus dem gcc-doc-Paket
gdb	V:13, I:114	9789	"info gdb" aus dem gdb-doc-Paket
gettext	V:48, I:312	5843	"info gettext" aus dem gettext-doc-Paket
gfortran	V:11, I:98	16	"info gfortran" aus dem gfortran-doc-Paket (Fortran 95)
fpc	I:3	121	fpc(1) bzw. die HTML-Variante aus dem fp-docs-Paket (Pascal)
glade	V:0, I:8	1730	Hilfe verfügbar über das Menü (UI Builder)
libc6	V:935, I:999	12771	"info libc" aus dem glibc-doc- und glibc-doc-reference-Paket
make	V:157, I:609	1592	"info make" aus dem make-doc-Paket
xutils-dev	V:1, I:12	1466	imake(1), xmkmf(1) usw.
mawk	V:372, I:997	242	mawk(1)
perl	V:610, I:992	705	perl(1) bzw. die HTML-Seiten aus dem perl-doc- oder perl-doc-html-Paket
python	V:293, I:923	68	python(1) bzw. die HTML-Seiten aus dem python-doc-Paket
tcl	V:31, I:414	22	tcl(3) und detaillierte Handbuchseiten (manpages) aus dem tcl-doc-Paket
tk	V:30, I:406	22	tk(3) und detaillierte Handbuchseiten (manpages) aus dem tk-doc-Paket
ruby	V:137, I:318	35	ruby(1) und die interaktive Referenz aus dem ri-Paket
vim	V:106, I:398	3231	Hilfe-Menü (F1) aus dem vim-doc-Paket
susv2	I:0	16	die "Single UNIX Specifications v2"
susv3	I:0	16	die "Single UNIX Specifications v3"

Tabelle 12.1: Liste von Paketen für die Programmierung

Handbuchseiten (manpages) sind verfügbar, indem Sie nach Installation der `manpages-` und `manpages-dev`-Pakete `"man programmname"` eingeben. Informationen über GNU-Werkzeuge erhalten Sie über `"info programmname"` nach Installation der entsprechenden Dokumentationspakete. Sie müssen unter Umständen die Bereiche `contrib` und `non-free` des Archivs zusätzlich zu `main` freigeben, da einige GFDL-Dokumentation als nicht DFSG-frei angesehen wird.

**Warnung**

Verwenden Sie nicht `"test"` als Namen für eine ausführbare Testdatei. `"test"` ist ein fest integrierter Shell-Builtin.

**Achtung**

Sie sollten Software, die direkt aus den Quellen kompiliert wurden, in `"/usr/local"` oder `"/opt"` installieren, um Kollisionen mit Systemprogrammen zu vermeiden.

Tipp

Die Code-Beispiele zum Erzeugen des Songs "99 Bottles of Beer" sollten Ihnen gute Ideen zu nahezu allen Programmiersprachen liefern.

12.1 Das Shellskript

Das [Shellskript](#) ist eine Textdatei mit gesetztem Ausführungsbit und enthält Befehle in der folgenden Form:

```
#!/bin/sh
... Befehle
```

Die erste Zeile spezifiziert den Shell-Interpreter, der den Inhalt dieser Datei liest und ausführt.

Das Lesen von Shellskripten ist der **beste** Weg, um zu verstehen, wie ein Unix-artiges System arbeitet. Ich gebe hier einige Hinweise zur Shellprogrammierung. Lesen Sie "Shell Mistakes" (<http://www.greenend.org.uk/rjk/2001/04/shell.html>), um aus Fehlern zu lernen.

Anders als der interaktive Shell-Modus (lesen Sie Abschnitt 1.5 und Abschnitt 1.6) verwenden Shellskripte häufig Parameter, bedingte Ausdrücke und Schleifen.

12.1.1 POSIX-Shell-Kompatibilität

Viele Systemskripte können von jeder [POSIX](#)-konformen Shell (lesen Sie Tabelle 1.13) interpretiert werden. Die Standard-Shell für das System ist `"/bin/sh"`, was ein symbolischer Link auf das entsprechende Programm ist:

- `bash(1)` für Lenny oder älter;
- `dash(1)` für Squeeze oder neuer.

Vermeiden Sie, in Shellskripten **Bashisms** oder **Zshisms** (speziell für **Bash** oder **Zsh** angepasste Shell-Befehle) zu verwenden, damit sie auf alle POSIX-Shells portierbar sind. Sie können dies mittels `checkbashisms(1)` überprüfen.

Der Befehl `"echo"` muss mit Vorsicht verwendet werden, da seine Implementierung sich zwischen Shell-Builtins und externen Befehlen unterscheidet:

- vermeiden Sie die Verwendung jeglicher Befehlsoptionen außer `"-n"`;
-

Gut: POSIX	Vermeiden: Bashism
if ["\$foo" = "\$bar"] ; then ...	if ["\$foo" == "\$bar"] ; then ...
diff -u datei.c.orig datei.c	diff -u datei.c{.orig,}
mkdir /foobar /foobaz	mkdir /foo{bar,baz}
funktionsname() { ...}	function funktionsname() { ...}
Oktalformat: "\377"	Hexadezimalformat: "\xff"

Tabelle 12.2: Liste typischer Bashisms

- vermeiden Sie die Verwendung von Escape-Sequenzen in Zeichenketten, da deren Handhabung unterschiedlich ist.

Anmerkung

Obwohl die Option "-n" **nicht** wirklich der POSIX-Syntax entspricht, ist sie grundsätzlich akzeptiert.

Tipp

Nutzen Sie den "printf"-Befehl statt "echo", wenn Sie Escape-Sequenzen in auszugebenen Zeichenketten einbetten möchten.

12.1.2 Shellparameter

Einige spezielle Shellparameter werden oft in Shellskripten verwendet:

Shellparameter	Wert
\$0	Name der Shell oder des Shellskripts
\$1	erstes (1.) Shellargument
\$9	neuntes (9.) Shellargument
\$#	Anzahl der Parameter
"\$*"	"\$1 \$2 \$3 \$4 ..."
"\$@"	"\$1" "\$2" "\$3" "\$4" ...
\$?	Beendigungs-Status des zuletzt ausgeführten Befehls
\$\$	Prozessnummer (PID) des Shellskripts
\$_	PID des zuletzt ausgeführten Hintergrundbefehls

Tabelle 12.3: Liste von Shellparametern

Grundlegende **Parameterauswertungen**, die Sie sich einprägen sollten:

Parameterauswertung	Wert, falls var gesetzt ist	Wert, falls var nicht gesetzt ist
\${var:-string}	"\$var"	"string"
\${var:+string}	"string"	"null"
\${var:=string}	"\$var"	"string" (und ausführen von "var=string")
\${var:?string}	"\$var"	Ausgabe von "string" auf stderr (und mit Fehlerstatus beenden)

Tabelle 12.4: Liste von Parameterauswertungen

Hierbei ist der Doppelpunkt ":" in allen Operatoren optional:

- **mit** ":" = Operortest für **existiert** und **nicht Null**;
- **ohne** ":" = Operortest nur für **existiert**.

Parameterersetzung	Ergebnis
<code>\${var%suffix}</code>	entferne kleinstes Suffix-Muster
<code>\${var%%suffix}</code>	entferne größtes Suffix-Muster
<code>\${var#prefix}</code>	entferne kleinstes Prefix-Muster
<code>\${var##prefix}</code>	entferne größtes Prefix-Muster

Tabelle 12.5: Liste von Shellparameterersetzungen

12.1.3 Bedingte Ausdrücke in der Shell

Jeder Befehl gibt einen **Beendigungs-Status** (Exit-Status) zurück, der für einen bedingten Ausdruck verwendet werden kann:

- Erfolg: 0 ("Wahr/True")
- Fehler: nicht 0 ("Falsch/False")

Anmerkung

"0" im Kontext eines bedingten Ausdrucks für die Shell bedeutet "Wahr", während "0" im Kontext eines bedingten Ausdrucks für ein C-Programm "Falsch" bedeutet.

Anmerkung

"[" ist das Äquivalent des `test`-Befehls; Argumente bis zum "]" werden als bedingter Ausdruck gewertet.

Grundlegende **Ausdrucksformen für bedingte Ausdrücke**, die Sie sich einprägen sollten:

- `"<befehl> && <bei_erfolg_auch_diesen_befehl_ausführen> || true"`
- `"<befehl> || <falls_kein_erfolg_auch_diesen_befehl_ausführen> || true"`
- ein mehrzeiliger Skriptschnipsel wie dieser:

```
if [ <bedingter_ausdruck> ]; then
  <bei_erfolg_diesen_befehl_ausführen>
else
  <falls_kein_erfolg_diesen_befehl_ausführen>
fi
```

Hierbei ist das `"|| true"` am Ende erforderlich, um sicherzustellen, dass das Shellskript sich bei dieser Zeile nicht fälschlicherweise beendet, wenn die Shell mit der `"-e"`-Option aufgerufen wird.

Gleichung	Wahr wenn ...
<code>-e <datei></code>	<code><datei></code> existiert
<code>-d <datei></code>	<code><datei></code> existiert und ein Verzeichnis ist
<code>-f <datei></code>	<code><datei></code> existiert und eine reguläre Datei ist
<code>-w <datei></code>	<code><datei></code> existiert und schreibbar ist
<code>-x <datei></code>	<code><datei></code> existiert und ausführbar ist
<code><datei1> -nt <datei2></code>	<code><datei1></code> neuer als <code><datei2></code> ist (Änderungszeitpunkt)
<code><datei1> -ot <datei2></code>	<code><datei1></code> älter als <code><datei2></code> ist (Änderungszeitpunkt)
<code><datei1> -ef <datei2></code>	<code><datei1></code> und <code><datei2></code> die gleiche Device- und Inode-Nummer haben

Tabelle 12.6: Liste von Dateivergleichsoperatoren in bedingten Ausdrücken

Arithmetische Ganzzahlvergleicher in bedingten Ausdrücken sind `"-eq"`, `"-ne"`, `"-lt"`, `"-le"`, `"-gt"` und `"-ge"`.

Gleichung	Wahr wenn ...
-z <str>	die Länge von <str> gleich Null ist
-n <str>	die Länge von <str> nicht Null ist
<str1> = <str2>	<str1> und <str2> gleich sind
<str1> != <str2>	<str1> und <str2> ungleich sind
<str1> < <str2>	<str1> in der Sortierung vor <str2> erscheint (Locale-abhängig)
<str1> > <str2>	<str1> in der Sortierung hinter <str2> erscheint (Locale-abhängig)

Tabelle 12.7: Liste von String-Vergleichsoperatoren im bedingten Ausdruck

12.1.4 Shellschleifen

Es gibt mehrere Ausdrucksweisen für Schleifen, die in POSIX-Shells genutzt werden können:

- "for x in foo1 foo2 ...; do befehl ; done" führt Schleifen aus, indem Einträge aus der Liste "foo1 foo2 ..." der Variable "x" zugewiesen werden und dann "befehl" ausgeführt wird.
- "while bedingung ; do befehl ; done" wiederholt "befehl", solange "bedingung" wahr ist.
- "until bedingung ; do befehl ; done" wiederholt "befehl", solange "bedingung" nicht wahr ist.
- "break" ermöglicht, die Bearbeitung der Schleife zu beenden.
- "continue" ermöglicht, den nächsten Umlauf der Schleife fortzusetzen.

Tipp

Die der C-Sprache ähnliche numerische Wiederholung (Iteration) kann realisiert werden, indem seq(1) für die Erzeugung der "foo1 foo2 ..." -Liste genutzt wird.

Tipp

Lesen Sie auch Abschnitt [9.3.9](#).

12.1.5 Befehlsabfolge auf der Shell

Die Shell verarbeitet ein Skript im Prinzip in der folgenden Abfolge:

- Die Shell liest eine Zeile.
- Die Shell gruppiert Teile der Zeile zu **zusammengehörigen Ausdrücken (Token)** zusammen, wenn diese sich innerhalb von "..." oder '...' befinden.
- Die Shell splittet andere Teile der Zeile in **einzelne Ausdrücke (Token)** auf, wenn diese wie folgt von einander getrennt sind:
 - Whitespace-Zeichen: <Leerzeichen> <Tabulator> <newline>
 - Metazeichen: < > | ; & ()
- Die Shell prüft jeden Ausdruck auf **Schlüsselworte** (wenn nicht innerhalb von "..." oder '...'), um deren Verhalten anzupassen.
 - **Schlüsselwörter** sind: if then elif else fi for in while unless do done case esac
- Die Shell expandiert **Alias**-Befehle (wenn nicht innerhalb von "..." oder '...').
- Die Shell expandiert eine **Tilde** (wenn nicht innerhalb von "..." oder '...'):

- “~” → Heimatverzeichnis des aktuellen Benutzers
- “~<benutzer>” → Heimatverzeichnis von <benutzer>
- Die Shell expandiert **Parameter** in deren Wert (wenn nicht innerhalb von ‘...’):
 - **Parameter**: “\$PARAMETER” oder “\${PARAMETER}”
- Die Shell expandiert **Befehlsersetzungen** / **command substitutions** (wenn nicht innerhalb von ‘...’):
 - “\$(befehl)” → die Ausgabe von “befehl”
 - “` befehl `” → die Ausgabe von “befehl”
- Die Shell expandiert **Pfadnamenmuster** in die passenden Dateinamen (wenn nicht innerhalb von “...” oder ‘...’):
 - * → jegliche Zeichen (eins oder mehrere)
 - ? → irgendein (nur ein) Zeichen
 - [...] → jegliche Zeichen von denen in “...”
- Die Shell sucht **befehl** in folgenden Definitionen und führt ihn aus:
 - **Funktions**-Definition
 - **Builtin** (integrierter Befehl)
 - **ausführbare Datei** in “\$PATH”
- Die Shell geht zur nächsten Zeile und wiederholt diesen kompletten Ablauf vom Anfang.

Einfache Anführungszeichen innerhalb von doppelten Anführungszeichen haben keine Wirkung.

Das Ausführen von “set -x” in der Shell oder das Aufrufen einer Shell mit der Option “-x” veranlasst die Shell, alle ausgeführten Befehle auch auf dem Bildschirm auszugeben. Dies ist sehr nützlich zur Fehlersuche.

12.1.6 Hilfsprogramme für Shellskripte

Um Ihr Shellprogramm innerhalb des Debian-Systems möglichst weit portierbar zu machen, ist es eine gute Idee, die zu nutzenden Hilfsprogramme auf diejenigen einzuschränken, welche durch die **essential**-Pakete bereitgestellt werden:

- “aptitude search ~E” listet alle **essential**-Pakete auf;
- “dpkg -L <paketname> |grep '/man/man.*/'” listet Handbuchseiten (manpages) derjenigen Befehle auf, die von dem Paket <paketname> bereitgestellt werden.

Paket	Popcon	Größe	Beschreibung
coreutils	V:891, I:999	17478	Grundlegende GNU-Werkzeuge
debianutils	V:925, I:999	230	verschiedene Hilfsprogramme speziell für Debian
bsdmainutils	V:60, I:996	26	Sammlung weiterer FreeBSD-Hilfsprogramme
bsdutils	V:673, I:999	393	grundlegende Kommandos aus 4.4BSD-Lite
moreutils	V:11, I:35	237	zusätzliche Unix-Hilfsprogramme

Tabelle 12.8: Liste der Pakete, die kleine Hilfsprogramme für Shellskripte enthalten

Tipp

Obwohl `moreutils` außerhalb von Debian unter Umständen nicht verfügbar ist, bietet es interessante kleine Programme. Das erwähnenswerteste ist `sponge(8)`; es ist sehr nützlich, wenn Sie eine Originaldatei überschreiben möchten.

12.1.7 Shellskript-Dialog

Die Schnittstelle zwischen einem Shellprogramm und dem Benutzer ("Bedienoberfläche") kann über die plumpe Verwendung von `echo` und `read` hinaus mit einer interaktiven Oberfläche aufgewertet werden, indem eines der sogenannten Dialog-Programme genutzt wird:

Paket	Popcon	Größe	Beschreibung
x11-utils	V:180, I:599	712	xmessage(1): eine Nachricht oder Abfrage in einem Fenster anzeigen (X)
whiptail	V:87, I:995	71	nutzerfreundliche Dialogboxen von Shellskripten anzeigen (newt)
dialog	V:15, I:123	1222	nutzerfreundliche Dialogboxen von Shellskripten anzeigen (ncurses)
zenity	V:87, I:409	384	grafische Dialogboxen von Shellskripten anzeigen (gtk2.0)
ssft	V:0, I:0	75	Werkzeug für Shellskript-Oberflächen (Aufrufprogramm für zenity, kdialog und dialog mit gettext)
gettext	V:48, I:312	5843	"/usr/bin/gettext.sh": Nachricht übersetzen

Tabelle 12.9: Liste von Programmen für die Benutzerschnittstelle

12.1.8 Shellskript-Beispiel mit zenity

Hier ist ein einfaches Skript, das ein ISO-Image aus RS02-Daten erzeugt, welche von `dvdaster(1)` bereitgestellt werden.

```
#!/bin/sh -e
# gmkr02 : Copyright (C) 2007 Osamu Aoki <osamu@debian.org>, Public Domain
#set -x
error_exit()
{
    echo "$1" >&2
    exit 1
}
# Variablen initialisieren
DATA_ISO="$HOME/Desktop/iso-$$img"
LABEL=$(date +%Y%m%d-%H%M%S-%Z)
if [ $# != 0 ] && [ -d "$1" ]; then
    DATA_SRC="$1"
else
    # Auswahl eines Verzeichnisses zur Erzeugung des ISO-Images
    DATA_SRC=$(zenity --file-selection --directory \
        --title="Select the directory tree root to create ISO image") \
        || error_exit "Exit on directory selection"
fi
# Größe des Archivs überprüfen
xterm -T "Check size $DATA_SRC" -e du -s $DATA_SRC/*
SIZE=$((du -s $DATA_SRC | awk '{print $1}')/1024)
if [ $SIZE -le 520 ]; then
    zenity --info --title="Dvdaster RS02" --width 640 --height 400 \
        --text="The data size is good for CD backup:\n $SIZE MB"
elif [ $SIZE -le 3500 ]; then
    zenity --info --title="Dvdaster RS02" --width 640 --height 400 \
        --text="The data size is good for DVD backup :\n $SIZE MB"
else
    zenity --info --title="Dvdaster RS02" --width 640 --height 400 \
        --text="The data size is too big to backup : $SIZE MB"
    error_exit "The data size is too big to backup :\n $SIZE MB"
fi
# Nur xterm hat eine sicher funktionierende -e-Option
# Rohes ISO-Image erzeugen
rm -f "$DATA_ISO" || true
```

```
xterm -T "genisoimage $DATA_ISO" \
-e genisoimage -r -J -v "$LABEL" -o "$DATA_ISO" "$DATA_SRC"
# Eine zusätzliche RS02-Redundanz erzeugen
xterm -T "dvdaster $DATA_ISO" -e dvdaster -i "$DATA_ISO" -mRS02 -c
zenity --info --title="Dvdaster RS02" --width 640 --height 400 \
--text="ISO/RS02 data ($SIZE MB) \n created at: $DATA_ISO"
# EOF (Dateiende)
```

Sie möchten vielleicht für solch ein Skript einen Starter auf der Arbeitsfläche erstellen; nutzen Sie etwas wie `"/usr/local/bin/gmkr%d"`.

12.2 Make

Make ist ein Werkzeug, um Gruppen von Programmen zu betreuen. Bei Ausführung des Befehls `make(1)` liest `make` die Regeldatei `"Makefile"` und aktualisiert ein Ziel (Target), falls sich Dateien, von denen das Makefile abhängt, seit der letzten Modifizierung des Targets verändert haben oder falls das Target nicht existiert. Die Ausführungen dieser Aktualisierungen können zeitgleich erfolgen.

Die Syntax der Regeldatei ist folgende:

```
target: [ Voraussetzungen ... ]
[TAB] befehl1
[TAB] -befehl2 # Fehler ignorieren
[TAB] @befehl3 # Ausgabe unterdrücken
```

Hierbei ist `"[TAB]"` ein TAB-Code. Jede Zeile wird nach Ersetzung der Variablen durch die Shell interpretiert. Verwenden Sie `"\"` am Ende einer Zeile, um den Befehl in der nächsten Zeile fortzusetzen. Zur Angabe von Umgebungsvariablen müssen Sie statt `"$"` hier `"$$"` schreiben.

Implizite Regeln für das Target und Voraussetzungen können z.B. wie folgt angegeben werden:

```
%.o: %.c header.h
```

Hier enthält das Target das Zeichen `"%"` (exakt eines davon). Das `"%"` passt auf jeden nicht leeren Teil-String in den eigentlichen Dateinamen des Targets. Auch die Voraussetzungen nutzen auf ähnliche Art ein `"%"`, um den Bezug zum Namen des Targets anzuzeigen.

automatische Variable	Wert
<code>\$@</code>	Target
<code>\$<</code>	erste Voraussetzung
<code>\$?</code>	alle neueren Voraussetzungen
<code>\$^</code>	alle Voraussetzungen
<code>\$*</code>	<code>"%"</code> trifft auf den Stamm im Target-Muster zu

Tabelle 12.10: Liste von automatischen make-Variablen

Variablenexpandierung	Beschreibung
<code>foo1 := bar</code>	einmalige Expandierung
<code>foo2 = bar</code>	rekursive Expandierung
<code>foo3 += bar</code>	anhängen

Tabelle 12.11: Liste von make-Variablenexpandierungen

Führen Sie `"make -p -f/dev/null"` aus, um alle internen automatischen Regeln zu sehen.

12.3 C

Sie können wie folgt eine korrekte Umgebung zum Kompilieren von in der [C-Programmiersprache](#) geschriebenen Programmen einrichten:

```
# apt-get install glibc-doc manpages-dev libc6-dev gcc build-essential
```

Das Paket `libc6-dev` (d.h. die GNU-C-Bibliothek) bietet als [C-Standard-Bibliothek](#) eine Sammlung von Header-Dateien und Bibliotheksroutinen, die von der C-Sprache genutzt werden.

Referenzen für C finden Sie über:

- `"info libc"` (Referenz für Funktionen der C-Bibliothek)
- `gcc(1)` und `"info gcc"`
- `jeglicher_funktionsname_aus_der_c_bibliothek(3)`
- Kernighan & Ritchie, "The C Programming Language", 2. Ausgabe (Prentice Hall)

12.3.1 Ein einfaches C-Programm (gcc)

Hier ein einfaches Beispiel zum Kompilieren von `"example.c"` mit der Bibliothek `"libm"` in eine ausführbare Datei `"run_example"`:

```
$ cat > example.c << EOF
#include <stdio.h>
#include <math.h>
#include <string.h>

int main(int argc, char **argv, char **envp){
    double x;
    char y[11];
    x=sqrt(argc+7.5);
    strncpy(y, argv[0], 10); /* prevent buffer overflow */
    y[10] = '\0'; /* fill to make sure string ends with '\0' */
    printf("%5i, %5.3f, %10s, %10s\n", argc, x, y, argv[1]);
    return 0;
}
EOF
$ gcc -Wall -g -o run_example example.c -lm
$ ./run_example
    1, 2.915, ./run_exam,      (null)
$ ./run_example 1234567890qwerty
    2, 3.082, ./run_exam, 1234567890qwerty
```

Hierbei wird `"-lm"` benötigt, um die Bibliothek `"/usr/lib/libm.so"` aus dem `libc6`-Paket für `sqrt(3)` zu verlinken. Die eigentliche Bibliothek liegt in `"/lib/"` und hat den Dateinamen `"libm.so.6"`, was ein symbolischer Link auf `"libm-2.7.so"` ist.

Schauen Sie sich den letzten Parameter im ausgegebenen Text an. Dort werden mehr als 10 Zeichen ausgegeben, obwohl `"%10s"` angegeben wurde.

Die Verwendung von Funktionen, die Zeiger auf Speicherbereiche ohne Bereichsüberprüfung nutzen (wie `sprintf(3)` und `strcpy(3)`), wird missbilligt, um das Ausnutzen von Pufferüberläufen zu verhindern, die obige Überlaufteffekte in Gang bringen. Verwenden Sie stattdessen `snprintf(3)` und `strncpy(3)`.

12.4 Fehlersuche (Debugging)

Debugging ist ein wichtiger Teil der Programmieraktivitäten. Das Wissen darüber, wie man in Programmen einen Fehler sucht, macht Sie zu einem guten Debian-Nutzer, der aussagekräftige Fehlerberichte erstellen kann.

12.4.1 Grundlegende Ausführung von gdb

Das primäre [Programm zur Fehlersuche \(Debugger\)](#) im Debian-System ist `gdb(1)`, welches Ihnen erlaubt, ein Programm zu inspizieren, während es läuft.

Wir installieren `gdb` und zugehörige Programme wie folgt:

```
# apt-get install gdb gdb-doc build-essential devscripts
```

Eine gute Einführung zu `gdb` erhalten Sie mit `"info gdb"` oder [im Netz](#). Hier ein einfaches Beispiel zur Verwendung von `gdb(1)` bei einem Programm namens `"program"`, kompiliert mit der Option `"-g"`, um Debugging-Informationen auszugeben.

```
$ gdb program
(gdb) b 1                # Haltepunkt in Zeile 1 setzen
(gdb) run args           # Programm mit Argumenten ausführen
(gdb) next               # nächste Zeile
...
(gdb) step               # einen Schritt vorwärts
...
(gdb) p parm             # parm ausgeben
...
(gdb) p parm=12          # Wert auf 12 setzen
...
(gdb) quit
```

Tipp

Viele `gdb(1)`-Befehle können abgekürzt werden. Vervollständigungen funktionieren wie in der Shell mit der Tabulator-Taste.

12.4.2 Fehlersuche in einem Debian-Paket (Debugging)

Da alle installierten Binärdateien im Debian-System standardmäßig auf das Nötige reduziert sein sollten, sind in normalen Paketen die meisten Debugging-Symbole entfernt. Zur Fehlersuche in Debian-Paketen mittels `gdb(1)` müssen entsprechende `*-dbg-` oder `*-dbgsym-`Pakete installiert werden (z.B. `libc6-dbg` im Fall von `libc6`, oder `coreutils-dbgsym` für `coreutils`).

Pakete im alten Stil stellen ein zugehöriges `*-dbg-`Paket bereit. Es liegt direkt im Debian-main-Archiv, bei dem Originalpaket selbst. Neuere Pakete können automatisch beim Bau `*-dbgsym-`Pakete generieren und diese Debug-Pakete werden in dem separaten Archiv [debian-debug](#) abgelegt. Bitte lesen Sie die entsprechenden [Artikel im Debian Wiki](#), wenn Sie weitere Informationen benötigen.

Falls ein Paket, bei dem eine Fehlersuche durchgeführt werden soll, kein `*-dbg-` oder `*-dbgsym-`Paket anbietet, müssen Sie es händisch neu bauen und dann installieren, wie hier:

```
$ mkdir /path/new ; cd /path/new
$ sudo apt-get update
$ sudo apt-get dist-upgrade
$ sudo apt-get install fakeroot devscripts build-essential
$ apt-get source package_name
$ cd package_name*
$ sudo apt-get build-dep ./
```

Beheben Sie die Fehler, falls erforderlich.

Erhöhen Sie die Paketversion auf eine Versionsbezeichnung, die nicht mit offiziellen Debian-Versionen kollidiert (Sie können z.B. ein `"~+debug1"` anhängen, wenn Sie eine existierende Paketversion neu kompilieren, oder Sie hängen `"~pre1"` an, wenn Sie eine noch nicht veröffentlichte Paketversion selbst kompilieren). Verwenden Sie dazu:

```
$ dch -i
```

Kompilieren und installieren Sie ein Paket mit Debugging-Symbolen wie folgt:

```
$ export DEB_BUILD_OPTIONS=nostrip noopt
$ debuild
$ cd ..
$ sudo debi <paketname>*.changes
```

Sie müssen die Build-Skripte des Pakets überprüfen und sicherstellen, dass "CFLAGS=-g -Wall" zum Kompilieren der Binärdateien verwendet wird.

12.4.3 Gewinnen von Backtrace-Informationen

Wenn Sie einen Programmabsturz erlitten haben, ist es eine gute Idee, einen Fehlerbericht einzureichen und an diesen per Kopieren-und-Einfügen Backtrace-Informationen (Daten zur Rückverfolgung von Vorgängen in Programmen zwecks Fehleranalyse) anzuhängen.

Solche Backtrace-Informationen lassen sich wie folgt gewinnen:

- Lassen Sie das Programm unter gdb(1) laufen.
- Reproduzieren Sie den Absturz.
 - Dies bringt Sie zurück zum gdb-Prompt.
- Geben Sie "bt" am gdb-Prompt ein.

Falls sich das Programm aufgehängt hat, können Sie es mittels Strg-C in dem Terminal, in dem gdb läuft, beenden und zum gdb-Prompt zurückgelangen.

Tipp

Oft stellt man fest, dass in den Backtrace-Informationen eine oder mehrere der ersten Zeilen "malloc()" oder "g_malloc()" enthalten. Wenn dies passiert, besteht eine hohe Wahrscheinlichkeit, dass Ihr Backtrace nicht sehr nützlich sein wird. Der einfachste Weg nützliche Informationen zu bekommen ist, die Umgebungsvariable "\$MALLOCCHECK_" auf einen Wert von 2 zu setzen (Näheres in malloc(3)). Sie können dies wie folgt erledigen, während gdb läuft:

```
$ MALLOCCHECK_=2 gdb hello
```

12.4.4 Erweiterte gdb-Befehle

Befehl	Beschreibung
(gdb) thread apply all bt	Backtrace für alle Threads eines Multi-Thread-Programms auslesen
(gdb) bt full	Parameter auslesen, die auf dem Aufruf-Stack der Funktionsaufrufe aufgelaufen sind
(gdb) thread apply all bt full	Backtrace und Parameter als Kombination der vorangegangenen Optionen auslesen
(gdb) thread apply all bt full 10	Backtrace und Parameter für die ersten 10 Aufrufe auslesen, um irrelevante Ausgaben abzuschneiden
(gdb) set logging on	Protokoll der Ausgabe von gdb in eine Datei schreiben (Standard ist "gdb.txt")

Tabelle 12.12: Liste erweiterter gdb-Befehle

12.4.5 Fehleranalyse bei X-Fehlern

Wenn ein GNOME-Programm `preview1` einen X-Fehler empfangen hat, sollten Sie eine Meldung wie diese sehen:

```
The program 'preview1' received an X Window System error.
```

Sollte dies der Fall sein, können Sie versuchen, das Programm mit der Option `"- -sync"` zu starten und einen Haltepunkt für die Funktion `"gdk_x_error"` zu setzen, um einen Backtrace zu bekommen.

12.4.6 Überprüfen der Abhängigkeiten von Bibliotheken

Verwenden Sie `ldd(1)` wie hier, um zu ermitteln, von welchen Bibliotheken ein Programm abhängt:

```
$ ldd /bin/ls
    librt.so.1 => /lib/librt.so.1 (0x4001e000)
    libc.so.6 => /lib/libc.so.6 (0x40030000)
    libpthread.so.0 => /lib/libpthread.so.0 (0x40153000)
    /lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Damit `ls(1)` in einer `"chroot"`-Umgebung funktioniert, müssen die obigen Bibliotheken in der `"chroot"`-Umgebung vorhanden sein.

Lesen Sie dazu Abschnitt [9.3.6](#).

12.4.7 Werkzeuge zur Erkennung von Speicherlecks

Es gibt verschiedene Werkzeuge zur Erkennung von Speicherlecks in Debian:

Paket	Popcon	Größe	Beschreibung
libc6-dev	V:249, I:620	14357	mtrace(1): malloc-Debugging-Funktionalität in glibc
valgrind	V:6, I:46	80378	Speicher-Debugging- und Analyse-Programm
electric-fence	V:0, I:5	70	malloc(3)-Debugging-Programm
leaktracer	V:0, I:3	57	Werkzeug zur Verfolgung von Speicherlecks in C++-Programmen
libdmalloc5	V:0, I:3	393	Bibliothek zur Fehlersuche bei Speicherzuweisungen

Tabelle 12.13: Liste von Werkzeugen zur Erkennung von Speicherlecks

12.4.8 Werkzeuge zur statischen Code-Analyse

Es gibt [lint](#)-ähnliche Werkzeuge für die [statische Code-Analyse](#):

12.4.9 Disassemblieren von Binärdateien

Sie können Binär-Code wie folgt mit `objdump(1)` disassemblieren:

```
$ objdump -m i386 -b binary -D /usr/lib/grub/x86_64-pc/stage1
```

Anmerkung

`gdb(1)` kann verwendet werden, um Code interaktiv zu disassemblieren.

Paket	Popcon	Größe	Beschreibung
splint	V:0, I:4	2315	Werkzeug zur statischen Überprüfung von C-Programmen auf Programmfehler
flawfinder	V:0, I:0	181	Werkzeug zur Durchsuchung von Quellcode auf Sicherheitsschwächen
perl	V:610, I:992	705	Interpreter mit internem statischen Code-Prüfmechanismus: B::Lint(3perl)
pylint	V:2, I:13	1371	Statisches Prüfprogramm für Python-Code
weblint-perl	V:0, I:1	32	Ein Syntax- und (Minimal-)Stil-Prüfprogramm für HTML
linklint	V:0, I:0	344	schneller Linkchecker und Werkzeug für die Webseitenpflege
libxml2-utils	V:22, I:246	182	Hilfsprogramme zur Validierung von XML-Dateien mit xmllint(1)

Tabelle 12.14: Liste von Werkzeugen für die statische Code-Analyse

12.5 Flex - ein besseres Lex

[Flex](#) ist ein [Lex](#)-kompatibler schneller [lexikalischer Analysegenerator](#).

Eine Einführung zu `flex(1)` finden Sie in `"info flex"`.

Sie müssen Ihre eigenen `"main()"`- und `"yywrap()"`-Funktionen bereitstellen oder Ihr Programm sollte wie folgt aussehen, um ohne eine Bibliothek zu kompilieren (dies ist so, weil `"yywrap"` ein Makro ist und `"%option main"` implizit `"%option noyywrap"` aktiviert):

```
%option main
%%
.|\\n    ECHO ;
%%
```

Alternativ könnten Sie mit der `"-lfl"`-Linker-Option am Ende Ihrer `cc(1)`-Kommandozeile kompilieren (wie AT&T-Lex mit `"-ll"`). Es wird in diesem Fall kein `"%option"` benötigt.

12.6 Bison - ein besseres Yacc

Einige Pakete stellen [Yacc](#)-kompatible [LR-Parser](#)- oder [LALR-Parser](#)-Generatoren in Debian bereit:

Paket	Popcon	Größe	Beschreibung
bison	V:9, I:103	2815	GNU LALR-Parser-Generator
byacc	V:0, I:6	160	Berkeley LALR-Parser-Generator
btyacc	V:0, I:0	243	rückverfolgender Parser-Generator basierend auf <code>byacc</code>

Tabelle 12.15: Liste Yacc-kompatibler LALR-Parser-Generatoren

Eine Einführung zu `bison(1)` finden Sie in `"info bison"`.

Sie müssen Ihre eigenen `"main()"`- und `"yyerror()"`-Funktionen bereitstellen. `"main()"` ruft `"yyparse()"` auf, das wiederum `"yylex()"` aufruft, was gewöhnlich von `Flex` erzeugt wird.

```
%%
%%
```


12.7 Autoconf

[Autoconf](#) ist ein Werkzeug zum Erzeugen von Shellskripten, welche Quellpakete automatisch so konfigurieren, dass sie sich unter Verwendung des vollständigen GNU-Build-Systems an viele UNIX-artige Systeme anpassen.

autoconf(1) erzeugt das Konfigurationsskript "configure". "configure" erzeugt automatisch ein angepasstes "Makefile" aus einer "Makefile.in"-Vorlage.

12.7.1 Kompilieren und Installieren eines Programms



Warnung

Überschreiben Sie keine Systemdateien, wenn Sie Ihre selbst kompilierten Programme installieren.

Debian verändert keine Dateien unter "/usr/local/" oder "/opt". Wenn Sie also ein Programm aus den Quellen kompilieren, sollten Sie es in "/usr/local/" installieren, damit es nicht mit Debian kollidiert.

```
$ cd src
$ ./configure --prefix=/usr/local
$ make
$ make install # hiermit werden die Dateien auf dem System installiert.
```

12.7.2 Deinstallation eines Programms

Wenn Sie noch den Original-Quellcode haben, dieser autoconf(1)/automake(1) nutzt und Sie noch wissen, wie Sie es konfiguriert haben, verfahren Sie wie folgt, um das Programm zu deinstallieren:

```
$ ./configure "alle-optionen-die-sie-damals-angegeben-haben"
# make uninstall
```

Wenn Sie sich absolut sicher sind, dass der Installationsprozess Dateien nur unter "/usr/local/" abgelegt hat und es nichts Wichtiges mehr dort gibt, können Sie alternativ auch alles löschen, wie hier:

```
# find /usr/local -type f -print0 | xargs -0 rm -f
```

Falls Sie nicht sicher sind, wo die Dateien installiert sind, sollten Sie einen Blick auf checkinstall(8) aus dem checkinstall-Paket werfen, das einen leeren Pfad für die Deinstallation liefert. Es unterstützt jetzt auch die Erzeugung eines Debian-Pakets mit der Option "-D".

12.8 Verrücktheiten bei kurzen Perl-Skripten

Obwohl man jedes [AWK](#)-Skript mittels a2p(1) automatisiert nach [Perl](#) umschreiben kann, werden einzeilige AWK-Skripte am besten per Hand nach Perl konvertiert.

Denken wir an folgende AWK-Skriptzeile:

```
awk '($2=="1957") { print $3 }' |
```

Sie ist äquivalent zu jeder der folgenden Zeilen:

```
perl -ne '@f=split; if ($f[1] eq "1957") { print "$f[2]\n"}' |
```

```
perl -ne 'if ((@f=split)[1] eq "1957") { print "$f[2]\n"}' |
```

```
perl -ne '@f=split; print $f[2] if ( $f[1]==1957 )' |
```

```
perl -lane 'print $F[2] if $F[1] eq "1957"' |
```

```
perl -lane 'print$F[2]if$F[1]eq+1957' |
```

Das letzte ist eine Knobelaufgabe. Es nutzt die Vorteile folgender Perl-Funktionalitäten:

- Der Whitespace ist optional.
- Es existiert eine automatische Konvertierung von Zahlen zu Zeichenketten.

Details zu den Befehlszeilenoptionen finden Sie in `perlrun(1)`. Haben Sie Bedarf an noch verrückteren Perl-Skripten, könnte [Perl Golf](#) interessant sein.

12.9 Web

Einfache interaktive dynamische Webseiten können wie folgt erstellt werden:

- Abfragen werden mittels [HTML](#)-Formularen dem Browser-Nutzer präsentiert.
- Das Ausfüllen und Anklicken von Formulareinträgen sendet einen [URL](#)-String mit kodierten Parametern vom Browser zum Webserver:
 - `"http://www.foo.dom/cgi-bin/program.pl?WERT1=WERT1&WERT2=WERT2&WERT3=WERT3"`
 - `"http://www.foo.dom/cgi-bin/program.py?VAR1=WERT1&VAR2=WERT2&VAR3=WERT3"`
 - `"http://www.foo.dom/program.php?VAR1=WERT1&VAR2=WERT2&VAR3=WERT3"`
- `"%nn"` in einer URL wird durch ein Zeichen mit hexadezimalen `nn`-Wert ersetzt.
- Die Umgebungsvariable wird gesetzt als: `"ABFRAGE_STRING="VAR1=WERT1 VAR2=WERT2 VAR3=WERT3"`.
- Ein [CGI](#)-Programm (irgendeines von `"program.*"`) auf dem Webserver führt sich selbst mit der Umgebungsvariable `"$ABFRAGE_STRING"` aus.
- Die Standardausgabe (`stdout`) eines CGI-Programms wird zum Webbrowser gesandt und dort als interaktive dynamische Webseite angezeigt.

Aus Sicherheitsgründen wird empfohlen, keinen eigenen zusammengebastelten Code zum Parsen von CGI-Parametern zu verwenden. Es gibt dafür etablierte Module in Perl und Python. [PHP](#) unterstützt diese Funktionalitäten. Wenn eine Speicherung der Daten auf dem Client nötig ist, werden [HTTP-Cookies](#) verwendet. Ist eine Verarbeitung der Daten auf dem Client erforderlich, wird häufig [Javascript](#) genutzt.

Für weitere Informationen wird auf das [Common Gateway Interface](#), die [Apache Software Foundation](#) und [JavaScript](#) verwiesen.

Die Suche nach "CGI tutorial" auf Google durch Eingabe der kodierten URL <http://www.google.com/search?hl=en&ie=UTF-8&q=CGI+tutorial> direkt in der Adresszeile des Browsers ist eine gute Möglichkeit, das CGI-Skript auf dem Google-Server in Aktion zu beobachten.

12.10 Die Quellcode-Übersetzung

Es gibt verschiedene Programme zur Übersetzung von Quellcode:

Paket	Popcon	Größe	Schlüsselwort	Beschreibung
perl	V:610, I:992	705	AWK → PERL	Quellcode von AWK nach PERL konvertieren: a2p(1)
f2c	V:0, I:6	442	FORTTRAN → C	Quellcode von FORTRAN 77 nach C/C++ konvertieren: f2c(1)
intel2gas	V:0, I:0	178	intel → gas	Konvertierer von NASM (Intel-Format) nach GNU Assembler (GAS)

Tabelle 12.16: Liste von Programmen zur Übersetzung von Quellcode

12.11 Erstellen von Debian-Paketen

Wenn Sie ein Debian-Paket erstellen möchten, lesen Sie folgendes:

- Kapitel 2, um die Grundlagen des Paketsystems zu verstehen;
- Abschnitt 2.7.13, um den grundlegenden Portierungsprozess zu verstehen;
- Abschnitt 9.10.4, um die grundlegenden chroot-Techniken zu verstehen;
- `debuild(1)`, `pbuilder(1)` und `pdebuild(1)`;
- Abschnitt 12.4.2 für Informationen zum Rekompilieren von Quellcode zwecks Fehlersuche (Debugging);
- [Debian-Leitfaden für neue Paketbetreuer](#) als Einführung (aus dem `maint-guide`-Paket);
- [Debian-Entwicklerreferenz](#) (aus dem `developers-reference`-Paket);
- [Debian Policy-Handbuch](#) (aus dem `debian-policy`-Paket);
- [Handbuch für Debian-Paketbetreuer](#) (aus dem `debmake-doc`-Paket).

Es gibt auch Pakete wie `debmake`, `dh-make`, `dh-make-perl` usw., die beim Paketieren helfen.

Anhang A

Anhang

Hier einige Hintergrundinformationen zu diesem Dokument.

A.1 Das Debian-Labyrinth

Das Linux-System ist eine sehr leistungsfähige Computer-Plattform für einen mit dem Netzwerk verbundenen Rechner. Allerdings ist es nicht einfach, zu erlernen, wie man all dessen Fähigkeiten nutzen kann. Die Einrichtung der LPR-Drucker-Warteschlange mit einem nicht PostScript-fähigen Drucker war ein gutes Beispiel für die möglichen Stolpersteine. (Es gibt mittlerweile keine solchen Probleme mehr, da neue Installationen jetzt das CUPS-System nutzen).

Es gibt eine vollständige Quelle für detaillierte Informationen, genannt der "QUELLCODE". Dieser ist sehr exakt, aber auch sehr schwer zu verstehen. Es gibt auch andere Quellen wie HOWTOs oder mini-HOWTOs. Diese sind leichter verständlich, neigen aber dazu, zu viele Details zu liefern und den großen Überblick zu verlieren. Ich habe manchmal ein Problem, den richtigen Abschnitt in einem langen HOWTO zu finden, wenn ich nur ein paar Befehle, die ich aufrufen muss, suche.

Ich hoffe, diese "Debian-Referenz (Version 2.77)" (2021-01-10 06:32:51 UTC) bietet eine gute Starthilfe für Leute im Debian-Labyrinth.

A.2 Copyright-Vergangenheit

Die Debian Reference wurde initiiert von mir, Osamu Aoki <osamu at debian dot org> als persönliches Memo zur Systemadministration. Viele Inhalte entstammen dem Wissen, das ich von der [debian-user-Mailingliste](#) und anderen Debian-Ressourcen bezogen habe.

Einem Vorschlag von Josip Rodin folgend, der sehr aktiv in dem [Debian Documentation Project \(DDP\)](#) war, wurde die "Debian Reference (version 1, 2001-2007)" als Teil der DDP-Dokumente erstellt.

Nach sechs Jahren stellte ich fest, dass die originale "Debian Reference (version 1)" veraltet war und begann, viele Inhalte neu zu schreiben. Die neue "Debian Reference (version 2)" wurde 2008 veröffentlicht.

Die Lehrinhalte haben ihren Ursprung und ihre Inspiration in folgenden Quellen:

- "[Linux User's Guide](#)" von Larry Greenfield (Dezember 1996)
 - abgelöst durch das "Debian Tutorial"
- "[Debian Tutorial](#)" von Havoc Pennington (11. Dezember 1998)
 - teilweise geschrieben von Oliver Elphick, Ole Tetlie, James Treacy, Craig Sawyer und Ivan E. Moore II
 - abgelöst durch "Debian GNU/Linux: Guide to Installation and Usage"

- ["Debian GNU/Linux: Guide to Installation and Usage"](#) von John Goerzen und Ossama Othman (1999)
 - abgelöst durch die "Debian Reference (version 1)"

Die Paket- und Archivbeschreibungen haben ihren Ursprung und ihre Inspiration in folgenden Quellen:

- ["Debian FAQ"](#) (Version von März 2002, zu dieser Zeit betreut durch Josip Rodin)

Andere Inhalte haben ihren Ursprung und ihre Inspiration in folgenden Quellen:

- ["Debian Reference \(version 1\)"](#) von Osamu Aoki (2001-2007)
 - abgelöst durch die neuere "Debian Referenz (Version 2)" in 2008.

Die vorherige "Debian Reference (version 1)" wurde unter Mithilfe von vielen Beteiligten erstellt:

- ein Großteil der Beiträge zur Netzwerkkonfiguration von Thomas Hood;
- erhebliche Beiträge zu X- und VCS-bezogenen Themen von Brian Nelson;
- Hilfe bei den Build-Scripten und viele Korrekturen des Inhaltes durch Jens Seidel;
- ausgiebige Korrekturlesung durch David Sewell;
- viele Beiträge von Übersetzern und Leuten, die Inhalte beigetragen oder Fehler berichtet haben.

Viele Handbuch- und Infoseiten im Debian-System wurden als primäre Referenzen verwendet, um dieses Dokument zu schreiben. Bis zu einem gewissen Maß, das Osamu Aoki im Sinne von [Fair Use](#) als angemessen angesehen hat, wurden viele Teile davon, speziell Befehlsdefinitionen, in die Debian Reference übernommen (nach sorgfältiger Überarbeitung zwecks Anpassung an Stil und Zielvorgabe dieses Dokuments).

Die Beschreibung des Debuggers gdb wurde unter Verwendung von [Inhalten aus dem Debian Wiki über backtrace](#) in Konsens mit Ari Pollak, Loïc Minier und Dafydd Harries erweitert.

Die Inhalte der derzeitigen "Debian Referenz (Version 2.77)" (2021-01-10 06:32:51 UTC) sind überwiegend meine eigene Arbeit, mit Ausnahme der oben genannten. Diese wurden auch durch diejenigen, die damals die Inhalte beigetragen haben, aktualisiert.

Die "Debian Referenz (Version 1)" wurde ins Deutsche übersetzt von Jens Seidel und anderen.

Die "Debian Referenz (Version 2)" wurde ins Deutsche übersetzt von Holger Wansing <linux@wansing-online.de> sowie (teilweise) von Florian Rehnisch <fm-r@gmx.de> und anderen Mitgliedern des debian-l10n-german-Übersetzerteams.

Der Autor Osamu Aoki dankt allen, die geholfen haben, dieses Dokument möglich zu machen.

A.3 Dokumentenformat

Die Quellen des englischen Originaldokuments werden derzeit in [AsciiDoc](#)-Textdateien geschrieben. [AsciiDoc](#) wird nur der Einfachheit halber verwendet, da es weniger Schreiberei ist als reines XML und Tabellen in einem sehr intuitiven Format unterstützt. Sie sollten die XML- und PO-Dateien als wahre Quelldateien ansehen. Über ein Build-Script wird es in das DocBook-XML-Format konvertiert und automatisch generierte Daten werden eingefügt, um eine finale DocBook-XML-Quelle zu formen. Diese finale DocBook-XML-Quelle kann in HTML, epub, reinen Text, PostScript und PDF umgewandelt werden. (Einige Formate könnten von der Verteilung ausgeschlossen sein.)