Shell Commands

&

More

Uwe Schimanski

Seab@er Software

Goch, den 3. Februar 2020

Copyright © 2018 Uwe Schimanski

Published by Publisher

seabaer-ag.de

Licensed under the Creative Commons Attribution-NonCommercial 3.0 Unported License (the "License"). You may not use this file except in compliance with the License. You may obtain a copy of the License at http://creativecommons.org/licenses/by-nc/3.0. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "as is" basis, without warranties or conditions of any kind, either express or implied. See the License for the specific language governing permissions and limitations under the License.

First printing, May 2018

Inhaltsverzeichnis

Vorwort **xiii**

1 Konfiguration **1**
1.1 Bash . . . 1
1.2 Shell Gedächnis . . . 2
1.3 Befehle mehrzeilig . . . 2
1.4 Tastenzuordnung . . . 3
1.5 Tastaturfunktionen . . . 3
1.6 Prompt . . . 4
1.6.1 Bash . . . 4
1.6.2 Zsh . . . 7
1.7 VI Editor . . . 8
1.7.1 Konfiguration . . . 8
1.7.2 Starten . . . 9
1.7.3 Befehle . . . 9
1.8 Sonderzeichen . . . 13

2 Befehle **15**
2.1 Ordner / Verzeichnisse . . . 15
2.1.1 Anlegen . . . 15
2.1.2 Löschen . . . 15
2.2 Berechtigungen . . . 16
2.2.1 Chmod, Chown und Chgrp . . . 16
2.2.2 Chattr . . . 16
2.2.3 ACL . . . 17
2.2.4 Sticky-Bit . . . 17
2.3 LSOF - Geöffnete Dateien anzeigen . . . 18
2.3.1 Optionen . . . 18
2.3.2 Was hat ein Programm offen . . . 18
2.3.3 Portzugriff . . . 19
2.3.4 Verzeichniszugriff . . . 19
2.3.5 Benutzerzugriff . . . 19
2.3.6 Prozesszugriff . . . 20
2.4 Dateien . . . 20
2.4.1 Suchen . . . 20
2.4.2 Löschen . . . 21
2.4.3 Vergleichen . . . 21
2.4.4 Datei patchen . . . 21
2.4.5 Nur neuere Dateien kopieren . . . 21
2.4.6 Anzahl Dateien . . . 21
2.4.7 Dateien anlegen / Datum aktualisieren . . . 22
2.4.8 Dateien nach Größe / Zeit anzeigen . . . 22
2.4.9 Status Informationen . . . 22

2.4.10 Unveränderliche Dateien . . . 23
2.4.11 Dateien konvertieren . . . 24
2.4.12 Dateianfang anzeigen . . . 24
2.4.13 Dateiinhalt ausgeben . . . 24
2.5 Programme . . . 25
2.5.1 Im Hintergrund ausführen . . . 25
2.5.2 Ausgabe des Programmpfads . . . 25
2.5.3 Programme in einer Sandbox ausführen . . . 25
2.6 Mails . . . 26
2.6.1 Konfiguration . . . 26
2.6.2 Verschicken . . . 26
2.7 Grep . . . 27
2.7.1 In Farbe . . . 27
2.7.2 Klein- / Großschreibung . . . 27
2.7.3 Exakte Suche . . . 27
2.7.4 Mehrere Wörter Suchen . . . 27
2.7.5 Zeilen Nummer . . . 28
2.7.6 Anzahl Übereinstimmungen . . . 28
2.7.7 Alles ausgeben . . . 28
2.7.8 Ausgabe Datei . . . 28
2.8 Links . . . 28
2.8.1 Anlegen . . . 28
2.8.2 Anzeigen . . . 29
2.8.3 Löschen . . . 29
2.9 Größe und Position der Console . . . 29
2.10 Suchen & Ersetzten . . . 30
2.10.1 SED . . . 30
2.10.2 TR . . . 34
2.11 Spaltenweise Operationen . . . 35
2.11.1 AWK . . . 35
2.11.2 CUT . . . 35
2.12 Klammerexpandierung . . . 36
2.13 Ausgabe sortieren . . . 37
2.14 Rechnen . . . 37
2.15 Umgebunsvariablen anzeigen . . . 37
2.16 Prüfsummen . . . 37
2.17 Programmschleife . . . 37
2.18 Merge PDF Dateien . . . 38
2.19 Verzeichnisstruktur mit Tree . . . 38
2.20 Screenshot erstellen . . . 38
2.21 Convert Character Set . . . 38
2.22 Curl . . . 39

3 Dateisystem **41**
3.1 Logical Volume (LVM) Informationen . . . 41
3.2 Festplattenplatz . . . 41
3.3 Dateisystem konvertieren . . . 42
3.3.1 EXT2 nach EXT3 . . . 42
3.4 Festplatten überwachen (SMART) . . . 42
3.5 Wechseldatenträger . . . 43
3.5.1 schließen / auswerfen . . . 43
3.6 USB Geräte . . . 43
3.7 Formatieren . . . 44
3.8 Festplatten / Partitionen anzeigen . . . 44

3.9 Boot Partition kopieren und aktivieren . . . 45
3.10 Iso Image mounten . . . 46
3.11 Filesystem anzeigen . . . 47
3.12 Hdparm . . . 47
3.12.1 Festplatte Speed Test . . . 47
3.12.2 Power Management . . . 47
3.13 Loopback-Device . . . 47
3.14 Festplatten Statistic . . . 48
3.15 Filesystem Read-only Modus . . . 48
3.16 Partitions Informationen . . . 48
3.16.1 Tune2fs . . . 48
3.17 Swap Size . . . 49
3.17.1 Anzeigen . . . 49
3.17.2 Hinzufügen . . . 49
3.18 Software RAID . . . 50
3.18.1 Allgemein . . . 50
3.18.2 Raid Erstellen . . . 51
3.18.3 Raid anzeigen . . . 52
3.18.4 Raid löschen . . . 52
3.18.5 Platte entfernen . . . 53
3.18.6 Dateisysten . . . 53
3.18.7 Raid erweitern . . . 53
3.18.8 Array wiederherstellen . . . 54
3.18.9 Konfiguration sichern . . . 54
3.18.10 Tuning Resync / Rebuild . . . 54
3.19 SSH Mount . . . 55
3.20 Badblocks . . . 55
3.21 Festplatte sicher löschen . . . 55
3.22 BTRFS . . . 56
3.22.1 Speicherplatz Belegung . . . 56
3.22.2 Snapshot anzeigen . . . 56

4 Paketmanager 59
4.1 RPM Paket Manager . . . 59
4.1.1 Paket Installieren . . . 59
4.1.2 Paket update . . . 59
4.1.3 Failed to cache . . . 60
4.1.4 Paket löschen . . . 60
4.1.5 Paket Abfrage . . . 60
4.1.6 Abhängigkeiten Abfrage . . . 60
4.1.7 Install Datum . . . 61
4.2 Zypper (SuSE) . . . 62
4.2.1 Paket Installieren . . . 62
4.2.2 Paket deinstallieren . . . 62
4.2.3 Paket suchen . . . 62
4.2.4 Paket update . . . 63
4.2.5 Repositories verwalten . . . 63
4.2.6 Repository Prioritäten setzten . . . 64
4.2.7 System Upgrade . . . 65
4.2.8 Patche prüfen . . . 66
4.2.9 Benötigte Patche auflisten . . . 66
4.2.10 Patche installieren . . . 67
4.2.11 Alle Patche auflisten . . . 67
4.2.12 Patch Informationen abfragen . . . 67

4.2.13 Empfohlene Pakete installieren . . . 68
4.2.14 Abhängigkeiten prüfen . . . 68
4.2.15 Spzialfälle . . . 68
4.2.16 Proxy Einstellung . . . 69
4.2.17 Alte Kernel löschen . . . 69
4.2.18 Kernel installieren . . . 69
4.2.19 Patch descline (sperren) . . . 70
4.3 Yum (Red Hat) . . . 71
4.3.1 Proxy Einstellung . . . 71
4.3.2 Software installieren . . . 71
4.3.3 Software löschen . . . 71
4.3.4 Software aktualisieren . . . 71
4.3.5 Paket suchen . . . 72
4.3.6 Vorhandene Pakete . . . 72
4.3.7 Installierete Pakete . . . 72
4.3.8 Paket Informationen . . . 73
4.3.9 Repository . . . 73
4.3.10 Repository hinzufügen . . . 74
4.3.11 Update mit Cron . . . 74
4.4 Apt & Apt-get (Debian) . . . 75
4.4.1 Paket installieren . . . 75
4.4.2 Paket suchen . . . 75
4.4.3 Paket löschen . . . 75
4.4.4 Paket Informationen . . . 76
4.5 Pacman (Arch Linux) . . . 77
4.5.1 Konfiguration . . . 77
4.5.2 Repositories . . . 77
4.5.3 Spiegel-Server . . . 77
4.5.4 Update . . . 78
4.5.5 Paket decline . . . 78
4.5.6 Paket Neuinstallieren . . . 78
4.5.7 Befehle . . . 78

5 Netzwerk **81**
5.1 Netzwerkzugriffe erlauben / verbieten . . . 81
5.2 IP-Adresse Static . . . 81
5.2.1 Debian . . . 81
5.3 Datenpakete verfolgen . . . 81
5.4 Wake on Lan (WoL) . . . 82
5.5 Netzwerk mit Ip . . . 82
5.5.1 Syntax . . . 82
5.5.2 IP-Adresse anzeigen . . . 82
5.5.3 Interfaces anzeigen . . . 83
5.5.4 IP-Adresse hinzufügen . . . 83
5.6 ZMD Meldung (SLES) . . . 84
5.7 Netzlaufwerke . . . 84
5.8 Bonding . . . 84
5.8.1 SuSE . . . 84
5.8.2 Red Hat . . . 86
5.9 DNS . . . 86
5.9.1 DNS Server . . . 86
5.9.2 Namensauflösung . . . 87
5.10 Mount Webdav . . . 88
5.10.1 Arch Linux . . . 88

5.10.2 OpenSuSE . . . 88
5.10.3 Mount aufheben . . . 88
5.11 Mount Cifs . . . 89
5.12 DHCP . . . 89
5.12.1 Server . . . 89
5.12.2 Client . . . 89
5.13 Ping . . . 89
5.14 Private IP-Adressen . . . 89
5.15 USB-Wlan Stick . . . 90
5.15.1 Check USB-Stick . . . 90
5.15.2 Config Interface . . . 90
5.15.3 Config Verbindung . . . 90
5.15.4 Verbindung herstellen . . . 91
5.15.5 Verfügbare Netze anzeigen . . . 91
5.15.6 Anzeige Network Informationen . . . 91
5.15.7 Weitere Monitoring Tools . . . 91
5.16 Anzeige WAN-IP . . . 91
5.17 WLAN beim Start verbinden . . . 92
5.18 Netzwerk Check . . . 93
5.18.1 Monitoring . . . 93
5.18.2 Performance . . . 93
5.19 Netzwerkzugriffe erlauben / verbieten . . . 93

6 Archive 95
6.1 Zip & Unzip . . . 95
6.1.1 Archiv erstellen . . . 95
6.1.2 Remove Files . . . 95
6.1.3 Update Files . . . 95
6.1.4 Exclude Files . . . 96
6.1.5 Extract Files . . . 96
6.2 Gzip & Gunzip . . . 96
6.3 Bzip2 & Bunzip2 . . . 97
6.4 Tar . . . 97
6.4.1 Inkrementelles Backup - Dateien suchen / Auspacken . . . 98

7 Rsync & Rsnapshot 99
7.1 Rsync - Verzeichnisse abgleichen . . . 99
7.1.1 Parameter . . . 99
7.1.2 Verzeichnis Synchronisieren . . . 99
7.1.3 Rsync in Scripten . . . 100
7.1.4 Rsync-Daemon . . . 100
7.2 Rsnapshot . . . 101
7.2.1 Konfiguration . . . 101
7.2.2 Sicherungsintervall . . . 101
7.2.3 Sicherungsverzeichnisse . . . 102
7.2.4 Include / Exclude . . . 102
7.2.5 Unterschiede zwischen Snapshots . . . 102

8 System 103
8.1 Prozesse . . . 103
8.1.1 Prozesse auflisten mit ps . . . 103
8.1.2 Prozesse ausflisten mit pstree . . . 103
8.1.3 Prozesse ausflisten mit pgrep . . . 104
8.1.4 Prozesse abkoppeln . . . 104

8.1.5 Prozess löschen . . . 104
8.1.6 Prozess Limits . . . 105
8.2 Module . . . 105
8.2.1 Geladene Module anzeigen . . . 105
8.2.2 Verfügbare Module anzeigen . . . 105
8.2.3 Modul Informationen . . . 105
8.2.4 Module laden / entladen . . . 105
8.2.5 Module automatisch laden . . . 106
8.3 Boot Splash ändern . . . 106
8.3.1 OpenSuSE . . . 106
8.4 Zeitserver . . . 106
8.4.1 NTP . . . 106
8.5 Cronjobs . . . 108
8.5.1 Allgemein . . . 108
8.5.2 Systemweite Jobs . . . 108
8.5.3 Benutzer Jobs . . . 109
8.5.4 Berechtigungen . . . 109
8.6 Init-Skripte verwalten . . . 110
8.6.1 Verwalten mit chkonfig . . . 110
8.6.2 Verwalten mit insserv . . . 111
8.7 Runlevel . . . 112
8.7.1 Init System . . . 112
8.7.2 SystemD . . . 112
8.8 Konsolen Login nicht möglich . . . 112
8.9 Syslog . . . 113
8.10 Sudo (Runas) . . . 114
8.10.1 Syntax . . . 114
8.10.2 Alias . . . 114
8.10.3 Einstellungen . . . 114
8.10.4 Beispiele . . . 115
8.11 Dienste anzeigen . . . 116
8.12 CPU Info . . . 116
8.13 Server Domain . . . 117
8.13.1 Software . . . 117
8.13.2 Konfiguration . . . 117
8.13.3 Dienste starten . . . 120
8.13.4 Domain aufnehmen . . . 120
8.13.5 Testen . . . 120
8.14 Firewall . . . 120
8.14.1 Ausschalten / Deaktivieren . . . 120
8.15 Ports . . . 121
8.15.1 Anzeigen . . . 121
8.15.2 Freigeben . . . 121
8.16 Autostart . . . 121
8.16.1 Gnome . . . 121
8.16.2 KDE . . . 122
8.17 Memory . . . 122
8.18 HugePages . . . 122
8.19 Shutdown / Reboot . . . 123
8.19.1 Shutdown . . . 123
8.19.2 Last Reboot . . . 124
8.20 Anmelden disablen . . . 124
8.21 Hinweistext (MoD) . . . 124

8.22 Systeminformationen . . . 127
8.23 SSL Zertifikate . . . 128
8.23.1 Erstellen . . . 128
8.24 NFS . . . 128
8.24.1 Freigaben anzeigen . . . 128
8.24.2 Fstab . . . 128
8.25 Fstab . . . 129
8.25.1 Beispiel . . . 129
8.25.2 Device-Spec . . . 129
8.25.3 Mount Point . . . 130
8.25.4 FS-Type . . . 130
8.25.5 Options . . . 130
8.25.6 Dump . . . 131
8.25.7 Pass . . . 131
8.25.8 Rebuild Fstab . . . 131
8.26 Fonts . . . 132
8.26.1 Verzeichnisse . . . 132
8.27 Hostname . . . 132
8.27.1 Setzen . . . 132
8.27.2 Abfragen . . . 132
8.28 DD . . . 132
8.28.1 Syntax . . . 133
8.28.2 Optionen . . . 133
8.28.3 Beispiele . . . 133
8.29 Battery . . . 134
8.29.1 Status ausgeben . . . 134
8.30 Hardware Info . . . 134
8.30.1 Hwinfo . . . 134
8.30.2 Inxi . . . 134
8.31 Spracheinstellung . . . 136
8.32 Benachrichtigungen . . . 137
8.33 Kernel . . . 137
8.33.1 Arch Linux / Manjaro . . . 137
8.34 Chroot . . . 138
8.35 Samba . . . 138
8.35.1 Disable SMBv1 . . . 138
8.36 Display Server . . . 139
8.37 Lock Screen . . . 139
8.38 Wallpapers . . . 139
8.39 Mailversand mit Nullmailer . . . 139
8.39.1 Installation . . . 139
8.39.2 Konfiguration . . . 140
8.39.3 Send Mail . . . 140
8.40 Python . . . 140
8.40.1 _atomic_removal is not defined . . . 140
8.41 2 Factor authentication . . . 141
8.41.1 Install . . . 141
8.41.2 Konfiguration . . . 141
8.41.3 Generator . . . 141
8.42 XFCE . . . 142
8.42.1 Whisker Menu . . . 142

9 SystemD **143**
9.1 Startzeit ausgeben . . . 143
9.2 Service / Dienste . . . 143
9.2.1 Auflisten . . . 143
9.2.2 Dienste starten . . . 144
9.2.3 Status . . . 144
9.3 Network . . . 145
9.3.1 Allgemein . . . 145
9.3.2 Konfiguration . . . 145
9.4 Timer - Cron Job . . . 148
9.4.1 Allgemein . . . 148
9.4.2 Service Units . . . 148
9.4.3 Timer . . . 149
9.4.4 Aktivieren . . . 151
9.4.5 Timer Informationen . . . 151
9.4.6 User Timer . . . 151
9.5 Timesyncd - Systemzeit . . . 152
9.5.1 Konfiguration . . . 152
9.5.2 Aktivieren . . . 152
9.5.3 Anzeigen . . . 152
9.6 Journalctl . . . 154
9.6.1 Konfiguration . . . 154
9.6.2 Journal Dateien verkleinern . . . 154
9.6.3 Syslog . . . 155
9.6.4 Status und Überprüfung . . . 155
9.6.5 Anzeigen Journal / Filtern . . . 155
9.6.6 Kernel Meldungen . . . 157
9.6.7 Fortlaufende Ausgabe . . . 158
9.6.8 Ausgabe in Datei . . . 158

10 Remote Verbindung **159**
10.1 VNC Server . . . 159
10.1.1 Verbindung für alle . . . 159
10.1.2 Verbindung für einen . . . 159
10.2 SSH . . . 160
10.2.1 SSH Konfiguration . . . 160
10.2.2 Root Login disable . . . 161
10.2.3 User Verbindungen . . . 161
10.2.4 SSH Banner . . . 161
10.2.5 SSH Dienst . . . 162
10.2.6 X11 Forwarding . . . 162
10.2.7 SSH Key erstellen . . . 162
10.2.8 SSH Key senden . . . 162
10.2.9 SSH Key entfernen . . . 162
10.2.10 Alias . . . 163
10.2.11 TCP-Stealth . . . 163
10.2.12 Match . . . 163
10.2.13 Chroot für Sftp . . . 164
10.2.14 Login Mail . . . 165
10.2.15 Error . . . 165
10.2.16 Beispiele . . . 165
10.3 Displaymanager . . . 166
10.3.1 Konfiguration . . . 166
10.3.2 Dienst starten . . . 166

10.3.3 Root Login GDM . . . 166
10.3.4 Anmelde Bildschirm . . . 166
10.4 SCP . . . 167
10.5 Remote Desktop . . . 168
10.5.1 RDP - Arch Linux . . . 168
10.5.2 RDP - Debian . . . 168

11 Benutzer / Gruppen 169
11.1 Benutzer . . . 169
11.1.1 Anlegen . . . 169
11.1.2 Ändern . . . 169
11.1.3 Löschen . . . 170
11.1.4 Account sperren . . . 170
11.1.5 Anzeigen . . . 170
11.1.6 Kennwort ändern . . . 170
11.1.7 Benutzer Info's . . . 171
11.1.8 Login Fehlversuche . . . 171
11.2 Gruppen . . . 171
11.2.1 Anlegen . . . 171
11.2.2 Ändern . . . 171
11.2.3 Löschen . . . 172
11.2.4 Anzeigen . . . 172
11.3 Logon Zeit begrenzen . . . 172
11.3.1 Voraussetzung . . . 172
11.3.2 Konfiguration . . . 172
11.3.3 Aktivierung . . . 173

12 Drucken 175
12.1 CUPS (Common Unix Print System) . . . 175
12.1.1 Konfiguration . . . 175
12.1.2 Verwaltung - Browser . . . 175
12.1.3 Dienst starten . . . 175
12.1.4 Admin . . . 176
12.1.5 Druckdaten entfernen . . . 176
12.1.6 Servernamen Drucker . . . 176

13 Programme 177
13.1 Kalender in der Shell . . . 177
13.2 Screenfetch . . . 178
13.3 Neofetch . . . 178
13.4 Merge PDF Dateien . . . 178
13.5 Firefox . . . 179
13.5.1 Speicherbedarf . . . 179
13.6 Conky - Desktop aufmotzen . . . 180
13.6.1 Allgemein . . . 180
13.6.2 Conky ausführen . . . 180
13.6.3 Konfiguration . . . 180
13.6.4 Definitionen . . . 180
13.6.5 RSS Feed einbinden . . . 181
13.6.6 Scripte ausführen . . . 182
13.7 Youtube-dl - Video / Musik . . . 183
13.7.1 Konfiguration . . . 183
13.7.2 Video Formate . . . 183
13.7.3 Audio extract vom Video . . . 184

14 Virtualisierung **185**
14.1 KVM (Kernel-based Virtual Machine) . . . 185
14.1.1 Voraussetzungen . . . 185
14.1.2 Installation KVM . . . 185
14.1.3 User . . . 186
14.1.4 Datastore . . . 186
14.1.5 Network . . . 187
14.1.6 Maschine Location . . . 188
14.1.7 Create Disk . . . 188
14.1.8 Info Disk . . . 188
14.1.9 Create Maschine . . . 189
14.1.10 Konfiguration . . . 189
14.1.11 Maschinen anzeigen . . . 189
14.1.12 VM Console . . . 190
14.1.13 Shutdown, Reboot und Start . . . 190
14.1.14 Convert Disk . . . 190
14.1.15 Resize Disk . . . 191
14.1.16 Video RAM . . . 191
14.1.17 Huge Pages . . . 191
14.2 Troubleshooting (KVM) . . . 193
14.2.1 Failed to start network default . . . 193
14.3 Virtual Box . . . 194
14.3.1 Konfiguration . . . 194
14.3.2 Verwaltung Console . . . 194
14.3.3 Gasterweiterung . . . 194

15 Backup **195**
15.1 Borg . . . 195
15.1.1 Installation . . . 195
15.1.2 Repository erstellen . . . 196
15.1.3 Backup erstellen . . . 196
15.1.4 Backup Scripten . . . 196
15.1.5 Key Ex- und Import . . . 197
15.1.6 Backup delete . . . 197
15.1.7 Info Repo / Archiv . . . 197
15.1.8 Mount / Umount . . . 198
15.1.9 Change Passwort . . . 199
15.1.10 Platzhalter . . . 199
15.1.11 Restore . . . 199
15.1.12 Backup überprüfen . . . 200
15.1.13 Backup Rename . . . 200
15.1.14 Borg unlock . . . 200
15.2 Restic . . . 201
15.2.1 Allgemein . . . 201
15.2.2 Installation . . . 201
15.2.3 Repository erstellen . . . 201
15.2.4 Backup . . . 201
15.2.5 Backup Vorhalten . . . 202
15.2.6 List Snapshots . . . 203
15.2.7 List Files . . . 203
15.2.8 Backup Statistik . . . 203
15.2.9 Backup Check . . . 204
15.2.10 Repository Unlock . . . 204

16 Verweise / Links **205**

Vorwort

Ich beschäftige mich schon seit dem Jahr 1993 mit Unix. Unsere CAD Software lief damals auf Unix Maschinen und 1998 habe ich dann Linux auf meinen Rechnern installiert. Damals habe ich SuSE installiert und bin bis heute dieser Distribution treu geblieben.

Die Shell unter Linux ist ein mächtiges Werkzeug. Um mal schnell nach dem passenden Befehl nach zu schlagen, wurde diese Dokumentation geschrieben. Zu den Befehlen gibt es meistens mehrere Parameter, die in dieser Dokumentation nicht alle beschrieben werden. Füer mehr Informationen kann man hierzu die Man Page aufrufen. Alle Befehle und auch die Scripte wurden in der Bash Shell ausprobiert und können auch in einer anderen Shell verwendet werden.

Als OS Systeme wurden von mir OpenSuSE und Manjaro verwendet.

Bei Fragen und Anregungen bin ich unter der folgenden Mail Adresse zu erreichen:
uws@seabaer-ag.de

Kapitel 1

Konfiguration

1.1 Bash

Im persönlichen Home Verzeichnis liegt die versteckte Datei `.bashrc`. In dieser Datei kann man die Bash konfigurieren. In dieser Datei können Aliase, Variablen und Funktionen definiert werden. Die neue Konfiguration kann dann mit `source ./.bashrc` geladen werden.

```
# define alias
alias dir='ls --color=auto --format=vertical
alias ls='ls -lahF --color=tty'
alias df='df -hT'
alias du='du -sch'
# define variables
strBackupPach=/backup/tuxserver
strBackupLog=backup.log
export strBackupPath strBackupLog
export GREP_OPTIONS='--color=auto'
export GREP_COLOR='1;32' # Ansi color light green
# Functions
back() {
 cd $strBackupPath
 vi $strBackupLog
}
# Berechtigungen setzen fuer neuangelegte Objekte\\
umask 022
```

Listing 1.1: Beispiel .bashrc

In der nachfolgenden Tabelle werden die Konfigurationsdateien aufgelistet, die bei einem Login/Aufruf einer Shell in der Reihenfolge verarbeitet werden.

Konfigurations Datei	*Login Shell*	*Interaktive Shell*	*Beschreibung*
/etc/profile	X	-	Systemweit, wird bei einem Update überschrieben.
/etc/profile.local	X	-	Systemweit, bleibt bei einem Update erhalten.
/etc/bash.bashrc	X	-	Systemweit, wird bei einem Update überschrieben.
~/.bashrc	X	X	Benutzerkonfiguration.
~/.alias	X	X	Benutzerkonfiguration. In dieser Datei können die Alias verwaltet werden.
~/.bash_profile	X	X	Benutzerkonfiguration Wird eine der Dateien
~/.bash_login	X	X	gefunden, so wird diese verarbeitet, auch wenn
~/.profile	X	X	die anderen Dateien vorhanden sind.

Tabelle 1.1: Liste der Dateien

Bei einem Logout wird die Datei `~/.bash_logout` verarbeitet.

1.2 Shell Gedächnis

Die bereits eingegebenen Befehle werden in der `/.bash_history` gespeichert. Standardmässig werden die letzten 500 Befehle eingetragen. Mit `uws@tux>echo $HISTSIZE` kann man sich den Wert anzeigen lassen. Möchte man mehr Befehle speichern, so kann das mit `uws@tux>export HISTSIZE=700` gemacht werden.

Die Suchfunktion schaltet man mit `[strg+R]` oder `[strg+s]` ein. Der Prompt verändert sich nun und den zu suchenden Befehl kann eingegebn werden. Die Bash vervollständigt die Eingabe Automatisch. Um weiter zu suchen, einfach die Tastenkombination noch einmal machen. Der gefundene Befehl wird dann mit einem `Enter` abgeschickt oder mit dem drücken der `Esc-Taste` kann man den Befehl noch editieren.

Mit `uws@tux>fc -l -10` werden die 10 letzten Befehle angezeigt.

1.3 Befehle mehrzeilig

Wird eine Anweisung in der Shell zu lang, so kann man sie mit \ am Ende der Zeile und nach einem Return in der zweiten Zeile weiterschreiben. Als Prompt Text wird die Variable `PS2` ausgewertet.

```
uws@tux>PS2="Weiter geht's>"

uws@tux>grep oracle /etc/passwd \
Weiter geht's>echo found
```

Listing 1.2: Beispiel

1.4 Tastenzuordnung

Eine Tastenzuordnung kann man auf zwei Arten realisieren. Entweder man definiert die Tastenzuordnung in der `.bashrc` oder in der `.inputrc`.

```
# F1: zeigt /var/log/messages
bind '"\eOP":"tail -f /var/log/messages\n"'
# F2: wechselt nach /etc
bind '"\eOQ":"cd /etc\n"'
# F3: wiederholt das letzte Wort der aktuellen Zeile
bind '"\eOR":"!#$\e^"'
```

Listing 1.3: Beispiel .bashrc

```
# F1: zeigt /var/log/messages
"\eOP":"tail -f /var/log/messages\n"
# F2: wechselt nach /etc
"\eOQ":"cd /etc\n"
# F3: wiederholt das letzte Wort der aktuellen Zeile
"\eOR":"!#$\e^"
```

Listing 1.4: Beispiel .inputrc

1.5 Tastaturfunktionen

Taste	*Erklärung*
[Strg] + [A]	springt am Anfang der Eingabezeile
[Strg] + [E]	springt am Ende der Eingabezeile
[Alt] + [B]	springt ein Wort nach links in der Eingabezeile
[Esc] + [B]	geht ein Wort zurück
[Alt] + [F]	springt ein Wort nach rechts in der Eingabezeile
[Esc] + [F]	geht ein Wort vor
[Strg] + [K]	löscht von der Cursor-Position aus bis zum Ende der Zeile
[Strg] + [U]	löscht von der Cursor-Poistion aus bis zum Anfang der Zeile
[Strg] + [W]	löscht ein Wort nach links
[Strg] + [T]	vertauscht die beiden Zeichen vor und hinter dem Cursor
[Alt] + [T]	vertauscht die beiden vorangehenden Wörter in der Eingabezeile
[Esc] + [T]	vertauscht die beiden vorangehenden Wörter
[Strg] + [L]	räumt das Terminal Fenster auf
[Tab]	Vervollständigen von Befehlen / Dateinamen oder Verzeichnisse

Tabelle 1.2: Auswahl Tastaturfunktionen

1.6 Prompt

1.6.1 Bash

Den Prompt kann mit der Variable PS1 eingestellt werden.

```
uws@tux>export PS1="[\t] \u@\h:\w>"
[10:26:22] uws@tux:~>
```

Listing 1.5: Setzen PS1

In der nachfolgenden Tabelle ist ein Auszug der Escape Sequencen.

Escape-Sequence	*Beschreibung*
\a	Der Ascii Klingel character (07)
\d	Das Datum in "Wochentag Monat Datumß.B. Mon Mai 22
\e	Der Ascii Escape character (033)
\h	Der hostname bis zu ersten.
\H	Der hostname
\j	The number of jobs currently managed by the shell
\l	The basename of the shell's terminal device name
\n	Newline
\r	Carriage return
\s	Der Name der Shell, der Basisname von $0
\t	Die aktuelle Zeit im 24 Stunden Format HH:MM:SS
\T	Die aktuelle Zeit im 12 Stunden Format HH:MM:SS
\@	Die aktuelle Zeit im 12 Stunden Format am/pm
\u	Den Usernamen des angemeldeten User
\v	Die Version der Bash
\V	Das Release der Bash, Version + Patchlevel
\w	Das aktuelle Arbeitsverzeichnis
\W	Der Basisname des aktuellen Arbeitsverzeichnisses
\!	The history number of this command
\#	The commans number of this command
\$	Ist die UID gleich 0, dann #, sonst $
\nnn	The character corresponding to the octal number nnn
\\	Ein Backslash
\[	Begin a sequence of non-printing characters, which could be used to embed a terminal control sequence into the prompt
\]	End a sequence of non-printing characters

Tabelle 1.3: Auswahl Escape Sequencen

Dem Prompt kann man auch farbig darstellen. Um z.B. dem User root eine andere Farbe zuzuweisen. Die Angabe der Farbe wird durch `\[` und `\]` begrenzt. Es lassen sich der Hintergrund und der Text in Farbe darstellen.

Sequenz	***Farbe***
\[\033[XXm\]	Keine Hintergrundfarbe
\[\033[40;XXm\]	Schwarz
\[\033[41;XXm\]	Rot
\[\033[42;XXm\]	Grün
\[\033[43;XXm\]	Hellbraun
\[\033[44;XXm\]	Blau
\[\033[45;XXm\]	Lila
\[\033[46;XXm\]	Türkis
\[\033[47;XXm\]	Hellgrau

Tabelle 1.4: Hintergrund Farben

Sequenz	***Farbe***
\[\033[0;30m\]	Schwarz
\[\033[1;30m\]	Dunkelgrau
\[\033[0;31m\]	Rot
\[\033[1;31m\]	Hellrot
\[\033[0;32m\]	Grün
\[\033[1;32m\]	Hellgrün
\[\033[0;33m\]	Braun
\[\033[1;33m\]	Gelb
\[\033[0;34m\]	Blau
\[\033[1;34m\]	Hellblau
\[\033[0;35m\]	Lila
\[\033[1;35m\]	Helles Lila
\[\033[0;36m\]	Dunkles türkis
\[\033[1;36m\]	Türkis
\[\033[0;37m\]	Hellgrau
\[\033[1;37m\]	Weiß
\[\033[0m\]	Auf default Farbe setzten

Tabelle 1.5: Text Farben

Die Uhrzeit kann in dem nachfolgenden Beispiel in der rechten oberen Ecke platziert werden. Die Variable `COLUMNS` enthält die Anzahl der Spalten im Terminal.

```
uws@tux>export PS1="\033[s\033[1;\$((COLUMNS-4))f\$(date +%H:%M)\033[u"
```

Listing 1.6: Uhrzeit rechts oben

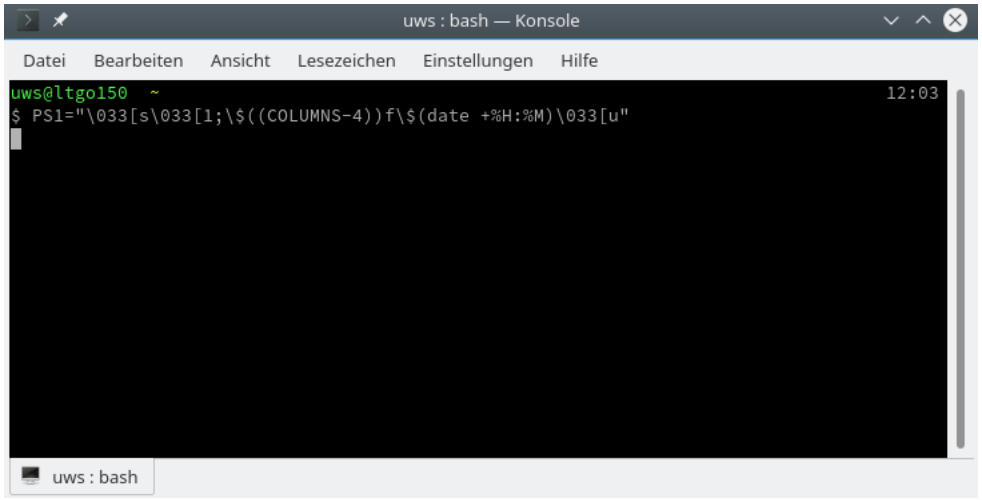

Abbildung 1.1: Prompt mit Zeitangabe

Im nachfolgenden Beispiel wird der Prompt mit der Farbe `grüen` dargestellt. Am Anfang leitet das `\e` die Farbe ein und am Ende der Zeile wird durch ein `\e[m` die Farbe wieder ausgeschaltet.

```
uws@tux>export PS1="\e[1;32m \u@\w:~> \e[m"
```

Listing 1.7: Prompt in Grün

1.6.2 Zsh

Die Farbwerte für die Zsh Shell sind die gleichen wie die der Bash. Sie können in der vorherigen Tabelle nachgeschaut werden.

```
uws@tux>export PS1=$'\e[1;32m%n@%/:=> \e[0m'
```

Listing 1.8: Beispiel Zsh Shell

Abbildung 1.2: Zsh Prompt

In der nachfolgenden Tabelle ist ein Auszug der vorhandenen Escape Sequencen. Die Farbe des Prompts nach

Escape-Sequenz	*Beschreibung*
%/	Arbeitsverzeichnis (pwd)
%	Arbeitsverzeichnis, aber das Home Verzeichnis wird mit angezeigt.
%M	Der ganze Hostname
%n	Der Hostname bis zum ersten Punkt
%@	Uhrzeit im 12 Stunden Format
%T	Uhrzeit im 24 Stunden Format
%*	Uhrzeit im 24 Stunden Format incl. der Sekunden
%n	Username
%N	Name eines Scripts, Source File oder Shell
%w	Datum im day-dd Format
%W	Datum immm/dd/yy Format
%D	Datum im yy-mm-dd Format

Tabelle 1.6: Liste Escape Sequencen

der Shell Background Farbe einstellen. In dem nachfolgenden Beispiel wird abgefragt, ob die Hintergrundfarbe Weiß ist.

```
if [ "${COLORFGBG}" = "0;15" ]; then
 PS1=$'\e[0;30m%n@%m %~\e[0m\n$'
else
 PS1=$'\e[1;32m%n@%m %~\e[1;33m %~\e[0m\n$'
fi
```

Listing 1.9: Beispiel Backgrond Color

1.7 VI Editor

1.7.1 Konfiguration

Die Einstellungen für den VI Editor können global in der Datei /etc/vimrc oder für jeden Benutzer in der Datei /.vimrc vorgenommen werden.

```
" Kommentarzeile
" Syntax Color einschalten
syntax on
set nowrap

" Color config, possible colors
" black, blue, cyan, gra, grey, green, magenta, red, white, yellow, brown
" DarkBlue, DarkGreen, DarkCyan, DarkRed, DarkMagenta, DarkYellow
" LightGray, LightGrey, DarkGray, DarkGrey, LighBlue, LightBlue, LightCyan
" LightRed, LightMagenta, LightYellow
" highlight groups= comment, constant, normal, nontext, special, cursor
hi normal ctermfg=white ctermbg=black guifg=white guibg=black
hi nontext ctermfg=blue ctermbg=black guifg=blue  guibg=black
hi comment ctermfg=green ctermbg=white

" Backup
set backup
set backupdir=~/tmp

" swap file
set dir=~/tmp

" show line numbers
set number

" show editing mode (insert/replace) on the last line
set showmode

" highlight matching brackets
set showmatch

" make searches case insensitive
set ignorecase

" Tab size (Anzeige und Plazierung)
set tabstop=3

" show cursor line and column in the status line
set ruler

" color scheme, files under /usr/share/vim/vim72/colors
color desert
```

Listing 1.10: Beispiel vimrc

1.7.2 Starten

Den Editor `vi` kann man mit verschiedenen Optionen öffnen.

Kommando	*Beschreibung*
vi	Ohne Datei starten
vi <datei>	Starten mit der angegebenen Datei
vi +<datei>	Es wird in einer Kopie gearbeitet und erst bei dem beenden des Editors wird die Datei geschrieben
vi -r<datei>	Die Bearbeitung wird nach einem Absturz an der gleichen Stelle fortgesetzt
vi -R<datei>	Die Datei wird schreibgeschützt geöffnet

Tabelle 1.7: Starten vi

1.7.3 Befehle

Folgende Befehle gibt es im VI-Editor. Dieses ist aber nur ein Ausschnitt der Befehle.

Speichern und Beenden

Kommando	*Beschreibung*
:q	Editor beenden
:q!	Editor beenden ohne zu speichern
:w	Datei speichern
:w!	Speichern erzwingen
:wq oder ZZ	Speichern und beenden
<strg> + z	Shell starten, zurück mit `fg`

Tabelle 1.8: Speichern und Beenden

Kopieren

Kommando	*Beschreibung*
yy	Kopiert die aktuelle Zeile in den Puffer
ny	Kopiert n+1 Zeilen in den Puffer
yw	Kopiert ein Wort in den Buffer

Tabelle 1.9: Kopieren

Sonstiges

Kommando	*Beschreibung*
u	Undo
U	Alle Änderungen der aktuellen Zeile zurücknehmen

Tabelle 1.10: Sonstiges

Navigieren

Kommando	*Beschreibung*
G	An das Ende springen
1G	In die erste Zeile Springen
0	An dem Anfang der Zeile springen
^	An dem Anfang der Zeile springen
$	An das Ende der Zeile springen
b	An den Anfang des letzten Wortes springen
w	An den Anfang des nächsten Wortes springen
e	An das Ende des letzten Wortes springen
h	Ein Zeichen nach links
j	Eine Zeile nach unten
k	Eine Zeile nach oben
l	Ein Zeichen nach rechts
nG	n-te Zeile der Datei
H	Die erste Zeile des Bildschirms
+	Die nächste Zeile des Textanfangs
-	Die vorgerige Zeile des Textanfangs
(	Satzanfang
)	Satzende
{	Absatzanfang
}	Absatzende
[	Am Anfang der Datei springen
]	Am Ende der Datei springen
<strg> + u	Nach oben blättern
<strg> + d	Nach unten blättern
L	In der letzten Zeile der Bildschirmausgabe springen

Tabelle 1.11: Navigieren

Löschen

Kommando	*Beschreibung*
x	Zeichen löschen
r	Zeichen ersetzen
s	Zeichen ersetzen und in den Bearbeitungsmodus wechseln
dd	Aktuelle Zeilen ausschneiden (Zwischenablage)
[n]dd	Die nächsten [n] Zeilen ausschneiden
C	Zeile löschen
V	Zeile markieren
v	Zeichen markieren
d	Markierten Text löschen
cw	Aktuelles Wort ersetzten
dw	Aktuelles Wort löschen, incl. angehängtem Leerzeichen
de	Aktuelles Wort löschen, über Zeile hinaus
dl	Zeichen unter Cursor-Position löschen
dj	Zwei Zeilen löschen, Position ist die 1. Zeile
dk	Zwei Zeilen löschen, Position ist die 2. Zeile
:5,10d	Zeilen 5 bis 10 löschen
:.,$d	Alle Zeilen löschen
nx	N-Zeichen ab Cursorposition
dL	Bis zum unteren Bildschirmrand löschen
d)	Bis zum Absatzende löschen
D	Bis zum Zeilenende löschen

Tabelle 1.12: Löschen

Einfügen

Kommando	*Beschreibung*
p	Zwischenablage nach aktueller Zeile einfügen
P	Zwischenablage vor aktueller Zeile einfügen
R	Text überschreiben
J	Aktuelle Zeile an das Ende der vorherigen Zeile anhängen
o	Eine Zeile nach der aktuellen Zeile einfügen und in den Bearbeitungsmodus wechseln
O	Eine Zeile vor der aktuellen Zeile einfügen und in den Bearbeitungsmodus wechseln
a	Ein Leerzeichen einfügen und in den Bearbeitungsmodus wechseln
A	Zum Ende der Zeile springen und in den Bearbeitungsmodus wechseln
i	In den Bearbeitungsmodus wechseln
I	Am Anfang in den Bearbeitungsmodus wechseln
s<text>	Ein Zeichen wird durch Text ersetzt
S<text>	Eine ganze Zeile wird durch Text ersetzt
ns<text>	N-Zeichen werden durch Text ersetzt
cw<text>	Ein Wort durch Text ersetzt

Tabelle 1.13: Einfügen

Suchen und Ersetzen

Kommando	*Beschreibung*
/Tux	Suchen nach dem Wort Tux
/	Suche wiederholen
n	Suche wiederholen
:s/nach/durch	In aktueller Zeile suchen und ersetzen
:1,7s/nach/durch	In Zeile 1 bis 7 suchen und ersetzen
:%/nach/durch	In allen Zeilen suchen und ersetzen
?<nach>	Die Suche erfolgt rückwärts
?	Wiederholung rückwärts
:s/nach/durch/g	Nur in der aktuellen Zeile wird gesucht und ersetzt

Tabelle 1.14: Suchen und Ersetzen

1.8 Sonderzeichen

Sonderzeichen	*Beschreibung*
;	Trennt Kommandos
:	Dummy-Kommando, führt nichts aus
.	Ohne eigene Subshell das Kommando ausführen. z.B. . datei
#	Kommentar
&	Das Programm wird im Hintergrund ausgeführt
&&	Bedingte Kommandoausführung. z.B. cmd1 && cmd2, dann wird cmd2 nur ausgeführt, wenn cmd1 erfolgreich war
|	Verbindet Kommandos miteinander
||	Bedingte Kommandoausführung, wie &&, nur das cmd2 dann ausgeführt wird, wenn cmd1 einen Fehler auswirft
*	Jokerzeichen, alle
?	Jokerzeichen, genau ein Zeichen
[xyz]	Jokerzeichen, eines aus xyz
[^xyz]	Jokerzeichen, alle ausser xyz
[ausdruck]	Andere Schreibweise für Test
~	Homeverzeichnis
>	Umleiten in einer Datei, Datei wird neu geschrieben
>>	Umleiten in einer Datei, wird an die Datei angehängt
>&	Umleiten der Standardausgabe
<	Einlesen einer Datei
<< ende	Einlesen einer Datei bis ende
(...)	Innerhalb der Klammer werden die Kommandos in einer Shell ausgeführt
{...}	Kommandos gruppieren
{,,}	Zeichenketten zusammenführen
$	Inhalt von Variablen
$* oder $@	Liste der übergebenen Parameter an die Shell
$#	Parameteranzahl
$0	Name des Shell-Programms
$?	Rückgabewert des letzten Kommandos
$!	PID des letzten Hintergrundprozesses
$$	PID der aktuellen Shell
$1 ... $9	Parameter 1 bis 9
$(...)	Kommandosubstitution
${...}	Funktionen zur manipulation von zeichenketten
$[...]	Berechnungen
"..."	Sonderzeichen behalten ihre Wirkung
'...'	Sonderzeichen verlieren ihre Wirkung
Â´...Â´	Kommandosubstitution
\zeichen	Demaskieren, hebt die Wirkung des Sonderzeichens aus

Tabelle 1.15: Sonderzeichen

Kapitel 2

Befehle

2.1 Ordner / Verzeichnisse

2.1.1 Anlegen

Ordner / Verzeichnisse kann man mit dem Befehl `mkdir <name>` anlegen.
Mit dem nachfolgenden Befehl werden alle drei Verzeichnisse gleichzeitig angelegt. Ohne den Parameter `-p` kann man nur die Verzeichnisse anlegen, wenn das vorherige Verzeichnis existiert. In dem Beispiel unten müssten also die Verzeichnisse `daten` und `privat` vorhanden sein.

```
uws@tux>mkdir -p daten/privat/auto
```

Listing 2.1: Create Verzeichnisse

Sollen die Verzeichnisse nebeneinander erstellt werden, so setzt man die Verzeichnisnamen in geschweiften Klammern.

```
uws@tux>mkdir -v {daten,privat,auto}
mkdir: created directory 'daten'
mkdir: created directory 'privat'
mkdir: created directory 'auto'

uws@tux>ls
daten
privat
auto
```

Listing 2.2: Create Verzeichnisse einzeln

2.1.2 Löschen

Möchte man Verzeichnisse mit einem Inhalt löschen, so wird der Parameter `-r` mit angegeben. Der Parameter löscht die Verzeichnisse und entfernt die darin enthaltenen Dateien.

```
uws@tux>rm -rv daten privat auto/
removed directory 'daten'
removed directory 'privat'
removed directory 'auto'
```

Listing 2.3: Verzeichnis löschen

2.2 Berechtigungen

2.2.1 Chmod, Chown und Chgrp

Die Berechtigungen für ein Verzeichnis / Dateien können mit dem Befehl `chmod` gesetzt werden. Der Befehl `chown` setzt den Besitzer des Verzeichnisses und kann auch die Gruppe setzen. Für das setzen der Gruppe gibt es auch noch den Befehl `chgrp`. Mit der Option `-R` werden alle Verzeichnisse rekursiv geändert.

```
uws@tux>chmod -R 775 /u01/app

uws@tux>chown -R uws:users /u01/app

uws@tox>chgrp -R users /u01/app
```

Listing 2.4: Beispiel Berechtigung

7=Lesen, Schreiben und Ausführen
6=Lesen und Schreiben
5=Lesen und Ausführen
4=Nur Lesen
3=Schreiben und Ausführen
2=Nur Schreiben
1=Nur Ausführen
0=Keine Berechtigung

Eine andere Möglichkeit mit `chmod` die Rechte zu vergeben, ist die Angabe der Benutzergruppe. Mit einem Minus Zeichnen werden der Gruppe die Rechte wieder entzogen.

```
uws@tux>chmod g=rw MeineDoku.txt

uws@tux>chmod g-w MeineDoku.txt

uws@tux>chmod g+x MeineDoku.txt
```

Listing 2.5: Beispiel Benutzergruppen

u=Eigentümer(User)
g=Gruppe(Group)
o=Andere(Other)
a=Alle(all)

2.2.2 Chattr

Mit dem Bash Befehl `chattr` (Change Attribute) können Attribute für Dateien oder Verzeichnisse gesetzt werden. Man kann zum Beispiel einzelne Dateien für das Löschen sperren. Mit + wird das Attribute gesetzt und mit - wird das Attribute gelöscht.

```
root@tux># immutable File / Path, can't delete or more
root@tux>chattr +i <file\_name>

root@tux># append mode, File can add Data, not modified
root@tux>chattr -a <file\_name>

root@tux>lsattr
```

Listing 2.6: Beispiel Berechtigung

2.2.3 ACL

Die Zugriffsrechte lassen sich auch über `Access Control Lists (ACL)` setzen. Um ACL nutzen zu können, muss das Dateisystem mit den Optionen `user_xattr` und `acl` eingehängt sein. Mit dem Befehl `setfacl` kann man Benutzer und auch Gruppen dem jeweiligen Objekt zuordnen. Um sich die Berechtigungen anzeigen zu lassen, wird der Befehl `getfacl` verwendet. Hat man ACL Berechtigungen gesetzt, so wird bei einem `ls -l` ein '+' Zeichen hinter den Berechtigungen angezeigt. Mit Dolphin und auch mit dem Konqueror lassen sich die ACL Berechtigungen bequem mittels der GUI setzen.

```
uws@tux>setfacl -m u:oracle:rw MeineDoku.txt

uws@tux>setfacl -m g:privat:rw bild1.jpg

uws@tux>getfacl bild1.jpg
```

Listing 2.7: Beispiel ACL

Man kann ein Verzeichnis mit `chmod 700` auf restriktiven Zugriff setzen um anschließend einem Benutzer oder auch nur einer Gruppe den Zugriff auf diesem Verzeichnis zu erlauben.

2.2.4 Sticky-Bit

Hat man einer Datei das Schreibrecht anderen Usern erteilt, so können diese User diese Datei auch löschen, da das Schreibrecht auch das löschen beinhaltet. Setzt man auf das Verzeichnis das `Sticky-Bit`, so darf nur der Eigentümer die Datei löschen oder umbenennen. Die anderen User dürfen nur noch in diese Datei schreiben. Das wegnehmen des `Sticky-Bits` geschieht anstelle des Plus Zeichens mit einem Minus Zeichen.

```
uws@tux>chmod +t /transfer/data
uws@tux>ls -l /transfer

drwxr-xr-t 2 uws users 296 data

uws@tux>chmod +t MeineDoku.txt

uws@tux>chmod -t /transfer/data
```

Listing 2.8: Beispiel ACL

2.3 LSOF - Geöffnete Dateien anzeigen

2.3.1 Optionen

Einen Überblick über die wichtigsten Optionen sind in der nachfolgenden Tabelle aufgelistet (Aus LinuxUser 04.2016).

Option	*Beschreibung*
-a	Logisches Und
-b	Verhindert, dass Lsof blockierende Funktionen verwendet
-c <Zeichen>	Suchkriterium
+c <Anzahl>	Wie viele Zeichen sollen berücksichtigt werden
+d <Verzeichnis>	Alles wird ausgegeben, was in dem angegebenen Verzeichnis ist
-d <Muster>	Alles ausschließen, was dem Muster entspricht.
+D <Verzeichnis>	Wie +d, nur werden auch Unterverzeichnisse berücksichtigt.
+/-f	Definiert, wie Lsof Pfade interpretieren soll
-i4/-i6	IPv4 oder IPv6-Verbindungen berücksichtigen
-p <PID>	Angabe der PID
-t	Nur PID ausgeben
-u <User>	Nur die Dateien ausgeben, die dem angegebenen User gehören
-U	Unix Domain Sockets verwenden
-T <Key>	TCP/IP-Informationen gemäß Key ausgeben
-s	Dateigröße anzeigen.
-S <Sekunden>	Timeout füer Kernel-Funktionen
+/- <Sekunden>	Aktiviert deb Wiederholungs.
-V	Markiert angeforderte, aber nicht gefundene Befehle, Dateien u.s.w

Tabelle 2.1: Optionen lsof

2.3.2 Was hat ein Programm offen

```
uws@tux>lsof -c <programm_name>

uws@tux>lsof -c bash

COMMAND PID  USER FD  TYPE DEVICE SIZE/OFF NODE    NAME
bash    2823 uws  cwd DIR  8,36       4096 1071099 /home/uws
bash    2823 uws  rtd DIR  0,32        246     256 /
bash    2823 uws  txt REG  0,32     656584     275 /bin/bash
.
.
```

Listing 2.9: Programm Info

In der nachfolgenden Tabelle werden die File-Deskriptoren (FD) mit ihren Beschreibungen aufgelistet.

Name	*Beschreibung*
cwt	Arbeitsverzeichnis (Current Working Directory)
txt	Textdatei
rtd	Wurzelverzeichnis (Root Directory)
mem	Bibliotheken (Memory-mapped File)
mmap	Gerätezugriff (Memory-mapped Device)
ltx	Shared Librarys
err	Zugriffsfehler
pd	Elternverzeichnis (Parent Directory)

Tabelle 2.2: File-Deskriptoren

Für die Type gibt es folgende Namen.

Name	*Beschreibung*
REG	Reguläre locale Datei
DIR	Verzeichnis
PIPE	Pipe
IPv4, IPv6	IP-Verbindung (Socket)
DEL	Gelöschte Datei
BLK	Block Device
CHR	Character Device

Tabelle 2.3: Type

2.3.3 Portzugriff

```
uws@tux>lsof -i :<port_no>

uws@tux>lsof -i :80
```

Listing 2.10: Beispiel Portzugriff

2.3.4 Verzeichniszugriff

```
uws@tux>lsof -D <Verzeichnis>

uws@tux>lsof -D /daten/allgemein
```

Listing 2.11: Beispiel Verzeichniszugriff

2.3.5 Benutzerzugriff

Wird vor dem Usernamen ein ^ vorangestellt, so werden alle offenen Dateien ausgelistet, außer von dem Benutzer mit dem ^

```
uws@tux>lsof -u <username>

uws@tux>lsof -u uws

uws@tux>lsof -u ^root
```

Listing 2.12: Beispiel Benutzerzugriff

2.3.6 Prozesszugriff

```
uws@tux>lsof -p <pid>

uws@tux>lsof -p 2326

uws@tux>lsof -d txt
```

Listing 2.13: Beispiel Prozesszugriff

2.4 Dateien

2.4.1 Suchen

Nach einer Datei in dem System suchen, kann man mit dem Befehl `find` in der Shell machen.
Findet alle Dateien, die größer sind als 10MB. Mit $ wird das Ergebnis von `find` als Parameter dem Befehl `ls` übergeben. Gefundene Dateien können mit einer Aktion weiterverarbeitet werden. Hierzu gibt es die Option `-exec`. Der Programmaufruf muss mit `{}` beendet werden und wird mit einem Semikolon abschlossen. In der Bash muss das Semikolon mit \ maskiert werden. In dem dritten Beispiel werden alle Dateien aufgelistet, die älter als 10 Tage sind.

```
uws@tux>ls -lah $(find / -type f -size +10M)

uws@tux>find /usr -size +10M -exec ls -lah {} \;

uws@tux>find /usr -type f \( -name '*.*' \) -ctime +10 -exec ls {} \;
```

Listing 2.14: Beispiele find

Hier wird nach Dateien gesucht, die neuer sind als die angegebene Referenz Datei.

```
uws@tux>touch -date='10:00' /tmp/ref

uws@tux>find -newer /tmp/ref
```

Listing 2.15: Beispiel newer

Die gefundenen Dateien in einem Tar-Archiv packen.

```
uws@tux>find -iname '*.jpg' -exec tar -rf bilder.tar {} \;
```

Listing 2.16: Beispiel für tar

Anzahl der Dateien in einem Verzeichnis ausgeben.

```
uws@tux>ls | wc -l

uws@tux> find /home/uws/Documents -type f | wc -l
```

Listing 2.17: Beispiel Anzahl Dateien

Nach leeren Dateien suchen und nach einer Bestätigung werden sie gelöscht.

```
uws@tux>find / -empty -ok rm "{}" \;
```

Listing 2.18: Beispiel für löschen

Nach Dateien oder Verzeichnissen suchen, die nicht dem Suchkriterium entsprechen. Die Option `-regex` berücksichtigt den kompletten Pfad, wobei `-name` nur den Namen berücksichtigt ohne den Pfad.

```
uws@tux>find / \! -regex ".*spool.*" -empty -ok rm -rf "{}" \;
```

Listing 2.19: Beispiel für löschen

Dateien finden, die innerhalb der letzten 120 Minuten verändert wurden. Ohne das - vor der Angabe der Zeit (120), wird alles gefunden, was vor 2 Stunden geändert wurde. Mit einem + werden Dateien gefunden, die vor der angegebenen Zeit verändert worden sind.

```
uws@tux>find /etc /usr -mmin -120

uws@tux>find ~ -mmin +120 mmin -1440
```

Listing 2.20: Beispiel für löschen

2.4.2 Löschen

Dateien können mit dem Befehl `rm` gelöscht werden. Wir die Option `-v` mit angegeben, so wird die Ausgabe detailierter. In dem nachfolgenden Beispiel werden alle Dateien gelöscht, die älter als 10 Tage sind.

```
uws@tux>find /daten -type f \( -name '*.jpg' \) -ctime +10 -exec rm {} \;
```

Listing 2.21: Beispiel löschen

2.4.3 Vergleichen

Einen Unterschied von zwei oder drei Dateien kann man mit dem Befehl `diff`[3] anzeigen lassen.

```
uws@tux>diff <datei1> <datei2>

uws@tux>diff3 <datei1> <datei2> <datei3>
```

Listing 2.22: Beispiel vergleichen

2.4.4 Datei patchen

Sollen zwei Dateien auf dem gleichen Stand gebracht werden, so erstellt man als erstes eine Differenz Datei und die so erstellte Datei wird dann in die erste Datei mit `patch` eingefügt.

```
uws@tux>diff datei1 datei2 > d1_d2.diff

uws@tux>patch datei1 < d1_d2.diff
patching file datei1
```

Listing 2.23: Beispiel Datei patch

2.4.5 Nur neuere Dateien kopieren

Mit dem Befehl `cp` und der Option `-u` kann man Dateien kopieren, die ein neueres Datum haben als die auf dem Zielverzeichnis.

```
uws@tux>cp -dpRux /home /mnt/media/home
```

Listing 2.24: Beispiel neuere Dateien kopieren

2.4.6 Anzahl Dateien

Möchte man die Anzahl der Dateien in einem Verzeichnis ermitteln, so kann man das mit der Kombination von `ls` und `wc` erledigen.

```
uws@tux>ls | wc -l
```

Listing 2.25: Beispiel Anzahl Dateien

2.4.7 Dateien anlegen / Datum aktualisieren

Eine leere Datei wird mit dem Befehl `touch` angelegt. Dieser Befehl wertet die `umask` aus und legt die neue Datei mit den Berechtigungen an. Ebenso kann man mit `touch` einer vorhandenen Datei das aktuelle Datum mit Uhrzeit verpassen.

```
uws@tux>touch empty.txt

uws@tux>touch *.txt
```

Listing 2.26: Beispiel touch

2.4.8 Dateien nach Größe / Zeit anzeigen

Mit der Option `-S` werden die Dateien nach Größe absteigend angezeigt. Mit der zusätzlichen Option `-r` wird die sortierreihenfolge umgedreht.In dem ersten Beispiel werden die 15 Größten Dateien angezeigt und in dem zweiten Beispiel wird nach der Datum / Zeit sortiert.

```
uws@tux>ls -lahS | head -n 15

uws@tux>ls -lahtr
```

Listing 2.27: Beispiel Größe / Zeit

2.4.9 Status Informationen

Mit dem Befehl `stat` kann man sich die Status Informationen einer Datei / Filesystem anzeigen lassen.

```
uws@tux>stat myfile.txt
  File: 'myfile.txt'
  Size: 337 Blocks: 8 IO Block: 4096 regular file
Device: fc03h/64515d Inode: 1048633 Links: 1
Access: (0644/-rw-r-r--) UID: ( 502/ uws) GID: ( 502/ uws)
Access: 2016-04-18 09:08:18.605079663 +0200
Modify: 2014-01-22 11:48:29.896951359 +0200
Change: 2014-01-22 11:48:29.909951675 +0200
```

Listing 2.28: Beispiel stat File

Möchte man sich Informationen über ein Device anzeigen lassen, so gibt man die Option `-f` mit an. Wird die Option nicht mit angegeben, so werden die Informationen als normales File ausgegeben.

```
uws@tux>stat -f /dev/sda1
  File: "/dev/sda1"
    ID: 0    Namelen: 255    Type: tmpfs
Block Size: 4096    Fundamental block size: 4096
Blocks: Total: 4641806    Free: 4641760    Available: 4641760
Inodes: Total: 4641806    Free: 4641090
```

Listing 2.29: Beispiel stat Device

Informationen über die Group ID eines Owner sich anzeigen lassen, geschieht mit der Option `--format=%g`. In der nachfolgenden Liste gibt es eine Übersicht der Format Ausgaben.

```
uws@tux>stat --format=%x myfile.txt
2014-01-22 11:48:29.896951359 +0200
```

Listing 2.30: Beispiel stat format

Anstelle von `--format` kann auch ein `-c` genommen werden.

```
uws@tux>stat -f -c%n /dev/sda1
/dev/sda1
```

Format	*Beschreibung*
%a	Zugriffsrechte in Octal
%A	Zufriffsrechte in Human form
%b	Anzahl der belegten Blöcke
%B	Größe in Bytes
%C	SELinux security
%d	Device Nummer in Dezimal
%D	Device Nummer in Hex
%f	Raw Mode in Hex
%F	File Type
%g	Group ID vom Owner
%G	Group Name des Owner
%h	Nummer der hard links
%i	Inode Nummer
%n	Datei Name
%N	Quoted Datei Name
%o	I/O Block Größe
%s	Total size, in Bytes
%t	Major device type in Hex
%T	Minor device type in Hex
%u	User ID des Owners
%U	User Name des Owners
%x	Datum / Zeit des letzten Zugriffs
%X	Datum / Zeit des letzten Zugriffs in Sekunden
%y	Datum / Zeit der letzten Modifikation
%Y	Datum / Zeit der letzten Modification in Sekunden
%z	Datum / Zeit des letzten Änderung

Tabelle 2.4: Formate

```
uws@tux>stat -f -c%c /dev/sda1
4644806
```

Listing 2.31: Beispiel stat format

2.4.10 Unveränderliche Dateien

Für die Dateisysteme `EXT2` und `EXT3` gibt es ein zusätzliches Attribute, die Dateien unveränderlich machen, solange das Attribut gesetzt ist. Mit dem Befehl `lsattr` kann man sich das Attribute anzeigen lassen und der Befehl `chattr` mit dem Attribut `-i` setzt das Attribut.

```
uws@tux>lsattr *.txt

uws@tux>chattr -i *.txt
```

Listing 2.32: Beispiel Unveränderliche Datei

2.4.11 Dateien konvertieren

Um eine Windows-Datei in das Linux / Unix Format zu konvertieren und auch in die andere Richtung, gibt es die Programme `fromdos`, `todos`, `dos2unix` und `unix2dos`. Gegebenenfalls müssen die Programme nachinstalliert werden.

```
uws@tux>fromdos <datei>

uws@tux>unix2dos <datei>
```

Listing 2.33: Beispiel Datei konvertieren

2.4.12 Dateianfang anzeigen

Anzeige der ersten <n> Zeilen einer Datei / Bildschirm Ausgabe.

```
uws@tux>head -10 /var/log/messages
```

Listing 2.34: Beispiel Datei konvertieren

2.4.13 Dateiinhalt ausgeben

Mit dem Befehl `cat` wird der Inhalt einer Datei von oben nach unten ausgegeben. Möchte man die Ausgabe von unten nach oben haben, so gibt es hierfür den Befehl `tac`.

```
uws@tux>tac /var/log/messages | less
```

Listing 2.35: Beispiel Dateiinhalt ausgeben

2.5 Programme

2.5.1 Im Hintergrund ausführen

Um ein Programm im Hintergrund laufen zu lassen, so wird nach dem Befehl das & Zeichen angehängt.

```
uws@tux>tail -f /var/log/messages &
```

Listing 2.36: Beispiele Hintergrund

Hat man mehrere Programme im Hintergrund gestartet, so kann man sie sich mit dem Befehl `jobs` anzeigen lassen.

```
uws@tux>jobs
```

Listing 2.37: Beispiel Jobs

Einen Job wieder in den Vordergrund zu holen, so wird der Befehl `fg` abgesetzt. Sind mehrere Jobs gestartet worden, so wird an dem Befehl `fg` noch ein `%<job_nr>` angehängt, um einen speziellen Job in den Vordergrund zu holen.

```
uws@tux>fg

uws@tux>fg %2
```

Listing 2.38: Beispiel Vordergrund

Soll das Programm wieder in den Hintergrund laufen, so muss das Programm gestoppt werden, um ihn dann in den Hintergrund zu bringen.

```
uws@tux>ctrl+Z

uws@tux>bg
```

Listing 2.39: Beispiel Stoppen

2.5.2 Ausgabe des Programmpfads

Möchte man die Pfadangabe eines Programms sich anzeigen lassen, so gibt es zwei Möglichkiten. In der ersten wird die `${PATH}` Variable ausgewertet.

```
uws@tux>which <ProgrammName>

uws@tux>whereis <ProgrammName>
```

Listing 2.40: Beispiel Programmpfad

2.5.3 Programme in einer Sandbox ausführen

Möchte man Programme oder auch Dienste in einem geschützten Bereich ausführen, kann man sie in einem `chroot-Käfig` starten. In diesem Käfig können Angreifer keinen Schaden anrichten.

```
uws@tux>chroot /nimm/das/als/root ftp -o option1
```

Listing 2.41: Beispiel Sandbox

2.6 Mails

2.6.1 Konfiguration

Lokale Mails können ohne Konfiguration verschickt werden. Sollen die Mail aber an einem Mail Server übergeben werden, so ist dieses zu konfigurieren. Folgende Parameter müssen in der Datei `/etc/postfix/main.cf` verändert werden.

```
myhostname = tux.seabaer-ag.de
mydomain = seabaer-ag.de
myorigin = $mydomian
relayhost = [172.30.250.66]

root@tux>systemctl restart postfix
```

Listing 2.42: main.cf

Soll der Absender umbenannt werden, so sind folgende Einstellungen vorzunehmen.

```
root@tux>tail -n 4 /etc/postfix/main.cf
#
# translating sender e.g. "root@tux" to "uws@seabaer-ag.de"
#
Sender_canonical_maps = /etc/postfix/sender_canonical

root@tux>cat /etc/postfix/sender_canonical
#
# translating user root to uws@seabaer-ag.de
#
root uws@seabaer-ag.de

root@tux>postmap /etc/postfix/sender_canonical

root@tux>systemctl restart postfix
```

Listing 2.43: Absender

2.6.2 Verschicken

Mails kann man mit dem Befehl `mail` verschicken.

```
uws@tux>echo "Eine Mail fuer dich" | mail -s "Du hast Post" uws@localhost

uws@tux>cat mail.txt | mail -s "Mehrzeileige Mail" uws@localhost
```

Listing 2.44: Send Mail

Mit -a kann man Attachements verschicken. Wenn mehrere Attachements verschickt werden sollen, kann man die Option -a mehrmals angeben. Verschiedene Empfänger werden mit einem Komma getrennt angegeben.

```
uws@tux>mail -s "Anhang" -a Image1.jpg -a bhb.pdf uws@seabaer-ag.de,
   jan@yahoo.gr
```

Listing 2.45: Send Attachmend

2.7 Grep

2.7.1 In Farbe

Soll die Ausgabe von grep in Farbe passieren, so gibt man die Option `--color=auto` mit an. Dauerhaft kann man diese Option in der `.bashrc` einschalten.

```
uws@tux>grep --color=auto

uws@tux>cat .bashrc
export GREP_OPTIONS='--color=auto'
export GREP_COLOR='1;32' # hellgruen, ANSI Farbcode
```

Listing 2.46: Grep Color

2.7.2 Klein- / Großschreibung

Möchte man alle Wörter finden, egal, ob Klein- oder Großschreibung, so wird hierzu der Parameter `-i` angegeben.

```
uws@tux>grep -i "rman-08" /home/uws/log/rman_level0.log
RMAN-0815 ****************************
rman-0816  Errors found
```

Listing 2.47: Upper und Lower character

2.7.3 Exakte Suche

Soll nur nach dem genauen Wort gesucht werden, so gibt man die Option `-w` an.

```
uws@tux>grep -w "rman-08" /home/uws/log/rman_level0.log
rman-0816  Errors found
```

Listing 2.48: Exakte Suche

2.7.4 Mehrere Wörter Suchen

Nach mehreren Wörtern kann man mit `egrp` suchen. Die Angabe der Option `-w` ist optional.

```
uws@tux>egrep -w "RMAN-08|rman-08" /home/uws/log/rman_level0.log
RMAN-0815 ****************************
rman-0816  Errors found
```

Listing 2.49: Mehrere Wörter Suchen

Mit grep kann man die Optionen `-E` oder `-e` nehmen. `Egrep` ist ein grep mit der Option `-E`. Gibt man ein `.*` in dem Suchstring ein, so ist es wie ein `und`. Ein `oder` wird mit einem `Pipe` gemacht. Nachfolgend sind einige Beispiele aufgelistet.

```
uws@tux>grep -i "Manager\|Sales" employee.txt
100 Thomas Manager Sales     $5,000
400 Willi  Manager Marketing $2,500

uws@tux>grep -i -E "Manager|Sales" employee.txt

uws@tux>egrep -i "manager|sales" employee.txt

uws@tux>grep -e "Manager" -e "Sales" employee.txt

uws@tux>grep -E "Manager.*Sales" employee.txt
100 Thomas Manager Sales     $5,000
```

Listing 2.50: Mehrere Wörter Suchen

2.7.5 Zeilen Nummer

Mit der Option -n wird auch die Zeilen Nummer mit ausgegeben.

```
uws@tux>grep -n "RMAN-08" /home/uws/log/rman_level0.log
2:RMAN-0815 ********************************
```

Listing 2.51: Zeilen Nummern

2.7.6 Anzahl Übereinstimmungen

Möchte man nur die Anzahl der gefundenen Übereinstimmungen angezeigt bekommen, so nimmt man hierfÃ¼r die Option -c.

```
uws@tux>grep -c "RMAN-08" /home/uws/log/rman_level0.log
1
```

Listing 2.52: Übereinstimmungen

2.7.7 Alles ausgeben

Mit -v wird alles ausgegeben, außer dem Suchstring.

```
uws@tux>grep -v "RMAN-08" /home/uws/log/rman_level0.log
Recovery Manager: Release 11.1.0.6.0 - Production on Mon Apr 29 19:00:00
   2013
Copyright Â© 1982, 2007 Oracle. All rights reserved
```

Listing 2.53: Alles Suchen

2.7.8 Ausgabe Datei

In welchen Dateien sich der Suchstring befindet, wird mit der Option -l ermittelt.

```
uws@tux>grep -l "RMAN-08" /home/uws/log/*.log
/home/uws/log/rman_level0.log
/home/uws/log/rman_level1.log
```

Listing 2.54: Ausgabe Datei

2.8 Links

2.8.1 Anlegen

Links (Verweise) können mit dem Befehl `ln` angelegt werden. Ohne Optionen werden `Hardlinks` angelegt und mit der Option `-s Softlinks`.

```
uws@tux>ln <quelle> <ziel>

uws@tux>ln -s <quelle> <ziel>
```

Listing 2.55: Erstellen Links

Ein `Softlink (Symbolischer Link)` zeigt direkt auf die Datei / Verzeichnis und ein `Hardlink auf` einen Inode. Wenn die Datei umbenannt oder verschoben wird, so bekommt der `Softlink` davon nicht mit, der Hardlink schon. Softlinks können Partitions- und Dateisystem Ã¼bergreifend erstellt werden, Hardlinks nicht. Hardlinks auf Verzeichnisse sollten vermieden werden, sind aber grundsätzlich möglich durch den User root.

2.8.2 Anzeigen

```
uws@tux>ls -l
MeineDatei.pdf > /home/uws/Daten/Bash.pdf

uws@tux>ls -l | grep .-\>
```

Listing 2.56: Links anzeigen

2.8.3 Löschen

```
uws@tux>rm <Datei>
```

Listing 2.57: Links löschen

2.9 Größe und Position der Console

Möchte man die Größe und Position der Console in Erfahrung bringen, so gibt es hierfür den Befehl `xwininfo`, den man in der Console absetzt. In der Ausgabe `geometry` steht dann die Größe und die Position des Fensters. Die Position des Fensters wird von der linken oberen Ecke aus gesehen. Diese Werte können dann z.B. für die Konfiguration eines Gnome Konsolen Fenster gebraucht werden. Hierzu ist dann in der Konfiguration des Fenster unter Befehl der Wert `-geometry 100x150+10+10` einzugeben.

```
uws@tux>xwininfo
.
.
- geometry 100x150+10+10
```

Listing 2.58: Größe und Position

2.10 Suchen & Ersetzten

2.10.1 SED

In einer Datei kann man mit dem Befehl `sed` nach Texten suchen und ersetzten. In dem nachfolgenden Beispiel wird er String `domain` durch `domäne` ersetzt und durch die Umleitung in eine neue Datei geschrieben.

```
uws@tux>sed -e s/domain/dom"ane/g <datei> > <datei.neu>
```

Listing 2.59: Beispiel sed

Syntax Beispiele

```
uws@tux>sed Kommando Textdatei
uws@tux>sed Kommando < Textdatei
uws@tux>Programm | sed Kommando | ...
uws@tux>sed -f Skript ...
uws@tux>sed -e Kommando1 -e Kommando2 -e Kommando3 ...
uws@tux>sed Kommando > Zieldatei
uws@tux>sed Kommando > Zieldatei 2>&1
```

Listing 2.60: Beispiele Syntax

Sed-Optionen

Option	*Beschreibung*
-e	Angabe auszuführender Befehle
-u	Auf Datenpufferung verzichten
-s	Dateien separat behandeln
-r	Erweiterte reguläre Ausdrücke verwenden
-i [Endung]	Sicherungsdatei anlegen
-f <script>	Script-Datei ausführen
-n	Unterdrücken der nicht betroffenen Textbereiche
-v	Versinsabfrage

Tabelle 2.5: SED-Optionen

Möchte man mehrere Kommandos in einem Rutsch ausführen, so kann man die Kommandos in einer Datei schreiben und diese Datei wird dann mit `-f` übergeben. In dieser Datei steht dann in jeder Zeile ein Sed-Befehl.

```
uws@tux>cat myFile
s/Hans//
s/ha/Ha/g
```

Listing 2.61: Beispiel Script-Datei

In der nachfolgenden Tabelle wird eine Auswahl der Editierkommandos aufgelistet.

Kommando	*Beschreibung*
i	Anfügen von Zeilen oberhalb der angegebenen Stelle
a	Anfügen von Zeilen unterhalb der angegebenen Stelle
p	Ausgabe der angegebenen Zeilen
l[länge]	Ausgabe der angegebenen Zeilen optional mit Längenbegrenzung
y	Austausch von Zeichen
q	Sed beenden
c	Ersetzen von Text in der angegebenen Zeile
d	Löschen der angegebenen Zeilen
s	Suchen und ersetzen

Tabelle 2.6: Editierkommandos

Für die Option s gibt es noch folgenden Optionen.

Option	*Beschreibung*
g	Befehl auf alle gefundenen Textstellen in der Zeile anwenden
p	Ergebnis der Aktion ausgeben
w[Datei]	Ergebnis in eine Datei schreiben

Tabelle 2.7: Optionen für s

Nachfolgend die Optionen der Editierkommandos.

Option	*Beschreibung*
=	Ausgabe der Zeilennummer
g	Betrifft alle vorkommen
p	Gibt bei dem Editierkommando s die geänderte Zeile aus
w	Schreiben der bearbeiteten Zeilen in die Datei

Tabelle 2.8: Optionen Editierkommandos

Desweiteren gibt es noch Sonderzeichen für SED.

Zeichen	*Beschreibung*
(	Öffnet eine Anweisung
)	Schließt eine Anweisung
{	Öffnet optionale Anweisung
}	Schließt optionale Anweisung
[	Öffnet Klassenbeschreibung von Zeichen
]	Schließt Klassenbeschreibung von Zeichen
"	Maskiert eine Anweisung und löst Shell-Variablen auf
’	Maskiert eine Anweisung und löst Shell-Variablen nicht auf
‘	Schließt Anweisungsblock ein
.	Ein beliebiges Zeichen außer Zeilenvorschub
,	trennt Parameter, etwa Zeilenangaben
Leerzeichen	Setzt Markierung (t- und b-Befehl)
$	Dokumentenende, letzte Zeile oder Zeilenende
&	Platzhalter für das Suchmuster, das in der Ersetzen-Anweisung mit ausgegeben wird
|	Oder (Abtrennen von regulären Ausdrücken)
/	Trennzeichen in Editierkommandos
^	Anfang der Zeile, aber: `[^Begriff]` = Negierung
\	Demaskieren
!	Nach Zeilenzahl: nicht diese Zeile ausgeben
	Nie oder beliebig oft
+	Muster mindestens einmal vorhanden
=	Ausgabe der Zeilennummern
\n	Zeilenvorschub
\t	Tabulator

Tabelle 2.9: Sonderzeichen

Nun werden in der nächsten Tabelle die Suchmuster und Adressangaben aufgelistet.

Muster	*Beschreibung*
(ohne)	Alle Zeilen
25	Zeile 25
25!	Nicht Zeile 25
10,20	Zeilen 10 bis 20
$	Letzte Zeile
'/Muster/!'	Nicht Suchmuster
Ẑeichen	Zeichen am Anfang
/Zeichenkette/	Zeichenkette
[Zeichnemenge]	Zeichenmenge
[:alpha:]	Beliebige Buchstaben
[:lower:]	Kleinbuchstaben
[:upper:]	Großbuchatben
[:alnum:]	Alphanumerische Zeichen
[:digit:]	Zahlen
[:xdigit:]	Hexadezimalzahlen
[:blank:]	Tabulatoren und Leerzeichen
[:space:]	Leerzeichen
[:cntrl:]	Steuerzeichen
[:print:]	Druckbare Zeichen (ohne Steuerzeichen)
[:graph:]	Sichtbare Zeichen (ohne Leerzeichen)
[:punct:]	Satzzeichen

Tabelle 2.10: Suchmuster / Adressangaben

Nun noch weitere Beispiele.

```
cat textdatei | sed -n '/Hans/p'     # Sucht nach Hans
sed -n '/[Hh]an/p' textdatei         # Alle Namen, die han oder Han
   beinhalten
sed -n '3,5!'p textdatei             # Alle Zeilen, ausser 3 und 5
sed -n '/Hans/!'p textdatei          # Alle Zeilen, ausser die mit Hans
sed -n '/[H|G]/'p textdatei          # Zeilen, die H oder G enthalten
sed -n '/[H]\|[G]/!'p textdatei      # Zeilen, die H oder G nicht enthalten
cat textdatei | sed -n '3p'          # Ausgabe der Zeile 3
cat textdatei | sed -n '$p'          # Ausgabe der letzten Zeile
sed -n '/[H]./,/[J]./!'p textdatei   # Zeilen nicht ausgeben, die H oder J
   enthalten.\\
cat textdatei | sed -n '/[:alnum:]/'p # Alle Zeilen, die alphanumerische
   Zeichen enthalten\\
cat textdatei | sed -n 's/j/J/p'     # Suchmuster nur beim ersten Auftreten
   ersetzten\\
cat textdatei | sed -n 's/j/J/gp'    # Suchmuster bei jedem Auftreten
   ersetzten\\
sed -n 's/Hans//gp' textdatei        # L"oschen des Wortes Hans\\
cat textdatei | sed -n '4s/Hans/Jens/gp'  # Ersetzen von Hans durch Jens in
   der Zeile 4\\
sed -n '/ans/s09/089/gp' textdatei   # Ersetzen von 09 durch 098 bei allen
   Zeilen mit dem Suchmuster ans\\
sed -n '/ans/!s0/089/gp' textdatei   # Ersetzen von 09 durch 098 bei allen
   Zeilen die nicht ans enthalten\\
cat textdatei | sed -n s'/[0-9]\/]//'gp # L"oschen aller Zahlen mitsamt Schr
   "agstrichen\\
```

Listing 2.62: Weitere Beispiele

2.10.2 TR

Es gibt auch noch den Befehl `tr`, mit dem man Strings ersetzten kann. Achtung, der zu ersetzende String muss gleich lang sein, sonst werden die fehlenden Stellen aufgefüllt.
Konvertieren Klein- auf Großschreibung.

```
uws@tux>tr a-z A-Z < datei.old > datei.neu

uws@tux>tr [:lower:] [:upper:] < datei.old > datei.new
```

Listing 2.63: Konvertierung

Ersetzen von Strings.

```
tr '{}' '()' < datei.old > datei.neu
```

Listing 2.64: Ersetzen Strings

Konvertieren Leerzeichen zu Tab. Für jedes Leerzeichen wird ein Tab gesetzt. Sind zwei oder mehr Leerzeichen vorhanden, so gibt es auch die gleiche Anzahl an Tabs.

```
uws@tux>echo "Zeile mit Leerzeichen" | tr [:space:] '\t'
Zeile     mit     Leerzeichen
```

Listing 2.65: Tabs setzen

Soll nur für ein Tab gesetzt werden, auch wenn mehrere Leerzeichen vorkommen, so gibt man die Option `-s` mit an.

```
uws@tux>echo "Zeile mit Leerzeichen" | tr -s [:space:] '\t'
Zeile   mit   Leerzeichen
```

Listing 2.66: Tabs setzen

Setzen von nur einem Leerzeichen.

```
uws@tux>echo "Zeile    mit    viele    Leerzeichen" | tr -s [:space:] ' '
Zeile mit viele Leerzeichen
```

Listing 2.67: Leerzeichen setzen

Löschen von Strings.

```
uws@tux>echo "Die User ID ist 784592" | tr -d 'D'
ie User I ist 784592

uws@tux>echo "Die User ID ist 784592" | tr -d [:digit:]
Die User ID ist

uws@tux>echo "Die User ID ist 784592" | tr -cd [:digit:]
784592
```

Listing 2.68: Strings löschen

Löschen aller non-printable Zeichen.

```
uws@tux>tr -cd [:print:] < file.txt > file_new.txt
```

Listing 2.69: Löschen non-printable

2.11 Spaltenweise Operationen

2.11.1 AWK

Das Program `awk` arbeitet Spaltenweise mit organisierten Informationen.

```
uws@tux>df | awk '/dev/ {summe +=$4} END {print summe " Kb"}'
```

Listing 2.70: Beispiel awk

2.11.2 CUT

Mit dem Kommandozeilen Befehl `cut` können Spalten ausgelesen werden. In dem ersten Beispiel wird aus einer Datei das zweite Zeichen ausgelesen. Einen Bereich wird mit `-c2-4` ausgelesen.

```
uws@tux>cut -c2 uws.txt
w
l
e
```

Listing 2.71: Beispiel cut

Ab einer Position bis zum Ende der Zeile wird folgendermaßen ausgelesen.

```
uws@tux>cut -c2- uws.txt
we is not working
lisabeth is at home
ernd has a car
```

Listing 2.72: Beispiel cut

Vom Anfang einer Zeile bis zur einer Bestimmten Position wird mit z.B. `-8` angegeben. Wird nur das Minus Zeichen angegeben, so wird die ganze Zeile ausgegeben.

```
uws@tux>cut -c-6 uws.txt
Uwe is
Elisab
Bernd
```

Listing 2.73: Beispiel cut

Die Option `-b` ist wie die Angabe von `-c`.

```
uws@tux>uname -a | cut -b 1-11,56-
```

Listing 2.74: Beispiel -b

Verschiedene Felder werden mit der Option `-f` ausgelesen und mit `-d` kann man das Trennzeichen setzten. Einen Bereich kann mit `-f 1-3,6,7` angegeben werden.

```
uws@tux>uname -a | cut -d ' ' -f 2,3
tux 3.1.10-1.13-desktop

@uws@tux>grep -i "nfs" /etc/fstab | cut -d ' ' -f2
/data/mail
/data/pictures
/data/video
```

Listing 2.75: Beispiel Felder

-d = Trennzeichen
-f = Spalte auslesen

Alle Felder ausgeben, außer ein bestimmtes Feld.

```
uws@tux>grep "/bin/zsh" /etc/passwd | cut -d ':' --complement -s -f7
uws:x:1000:1000:uws:/home/uws
```

Listing 2.76: Beispiel Feld

Für die Ausgabe wird ein neues Trennzeichen definiert. Ein Zeilenvorschub wird mit einem `$'\n'` gemacht.

```
uws@tux>grep "/bin/zsh" /etc/passwd | cut -d ':' -s -f1,6,7 --
   outputdelimiter='#'
uws#/home/uws#/bin/zsh
```

Listing 2.77: Beispiel Feldtrenner

2.12 Klammerexpandierung

Man kann nur Zahlen und Buchstaben expandieren.

```
uws@tux>echo {A..Z}
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

uws@tux>echo {1..12}
1 2 3 4 5 6 7 8 9 10 11 12

uws@tux>echo {1..12..2}
1 3 5 7 9 11

uws@tux>echo {a,b,c}{d,e,f}
ad ae af bd be bf cd ce cf

uws@tux>echo {a{1..3},b,c}{d,e,f}
a1d a1e a1f a2d a2e a2f a3d a3e a3f bd be bf cd ce cf

uws@tux>ls -l {,/local}/usr{,/share} # entspricht
ls -l /local/usr
ls -l /local/usr/share
ls -l /usr
ls -l /usr/share

uws@tux>ls -l a?.sh
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 a1.sh
uws@tux>ls -l [ab]*
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 a1.sh
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 b1.sh

uws@tux>ls -l [a-c]*
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 a1.sh
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 b1.sh
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 c1.sh

uws@tux>ls -l [^ab]*
-rw-rw-r-- 1 uws users 0 Sep 9 12:43 c1.sh

uws@tux>echo a?.sh
a1.sh

uws@tux>echo *
a1.sh b1.sh c1.sh
```

Listing 2.78: Beispiele Klammerexpandierung

2.13 Ausgabe sortieren

Die Ausgabe wird sortiert.

```
uws@tux>ls -l ~/bin| sort +4
```

Listing 2.79: Beispiel sortieren

2.14 Rechnen

Mit dem Aufruf `expre` kann in der Shell auch gerechnet werden.

```
uws@tux>expr 5 \* 600
3000
```

Listing 2.80: Beispiel Rechnen

2.15 Umgebunsvariablen anzeigen

Die Umgebungsvariablen können mit dem Aufruf `printenv` angezeigt werden.

```
uws@tux>printenv
```

Listing 2.81: Beispiel Umgebungsvariablen

2.16 Prüfsummen

Eine Prüfsumme für eine Datei kann man mit den Befehlen `md5sum`, `sha1sum`, `sha256sum` oder `sha512sum` erstellt werden.

```
uws@tux>md5sum spiele.iso
cfda02fa72abd850335a9066656322e5 *spiele.iso
```

Listing 2.82: Beispiel md5sum

Den Inhalt einer Variablen kann man folgendermaßen die Prüfsumme ermitteln.

```
uws@tux>echo "$uws" | sha1sum

uws@tux>uws1='echo "$uws" | sh1sum'

uws@tux>echo ${uws1%-} # Ausgabe Pruefsumme ohne abschliessenden '-'
```

Listing 2.83: Beispiel Variable

2.17 Programmschleife

Möchte man ein Programm in einer Schleife ausführen, so kann man das mit dem Befehl `watch` machen. Dieses Programm führt den übergebenen Befehl jede Sekunde aus und gibt ihn auf der Konsole aus. Mit `-n` kann die Intervall Zeit mitgegeben werden.

```
uws@tux>watch -n 2 cat /proc/meminfo
Every 2,0s: cat /proc/meminfo Wed Oct 29 14:01:22 2014

MemTotal:    4055224 kB
MemFree:     2581004 kB
.
.
```

Listing 2.84: Beispiel watch

Gibt man noch die Option `-d=cumulative` mit an, so werden alle Änderungen an der Ausgabe grau hinterlegt. Ohne die Option `cumulative` werden nur die Änderungen grau markiert, sie sich nach dem letzten Aufruf geändert haben.
Soll der auszuführende Befehl mit einem `Pipe (|)` Zeichen weiterverarbeitet werden, so muss die Befehlzeile in '' stehen.

```
uws@tux>watch -d 'ls -l | grep uws'
```

Listing 2.85: Beispiel watch mit Pipe

2.18 Merge PDF Dateien

Mehrere PDF-Dateien zu einer PDF-Datei zusammen fügen, kann mit dem Befehl `pdfunite` gemacht werden.

```
uws@tux>pdfunite in1.pdf in2.pdf in3.pdf out.pdf
```

Listing 2.86: Beispiel Merge PDF

2.19 Verzeichnisstruktur mit Tree

Mit dem Befehl `tree` kann man sich Verzeichnisstrukturen anzeigen lassen. Dieses funktioniert auch mit Tar-Archiven.

```
uws@tux>tree my_tar
Daten
 |
 |- Datei1.pdf
 |_ Datei2.jpg
```

Listing 2.87: Beispiel Tree

2.20 Screenshot erstellen

Einen Screenshot kann man mit dem Programm `import` machen. In dem nachfolgendem Beispiel wird ein Screenshot von dem ganzen Bildschirm angefertigt. Vorher wird 10 Sekunden gewartet.

```
uws@tux>sleep 10;import -window root screen.png
```

Listing 2.88: Beispiel Screenshot

2.21 Convert Character Set

Dateien in ein anderes Character Set zu konvertieren, kann man mit dem Programm `iconv` machen. Mit `iconv -l` werden die verfügbaren Character Sets aufgelistet.

```
uws@tux>iconf -f <from\_char> -t <to\_char>//TRANSLIT <input> -o <output>

uws@tux>iconv -f US-ASCII -t UTF-8//TRANSLIT Index.tex -o Index.new.tex
```

Listing 2.89: Beispiel Convert

2.22 Curl

Mit dem Programm `curl` kann man Dateien downloaden oder auch Infomationen über eine Web Seite erhalten.

```
uws@tux>curl -I -L https://www.apache.org
HTTP/1.1 200 OK
Date: Wed, 25 Jul 2018 13:45:23 GMT
Server: Apache/2.4.18 (Ubuntu)
Last-Modified: Wed, 25 Jul 2018 13:10:17 GMT
ETag: "efc3-571d2999a81b8"
Accept-Ranges: bytes
Content-Length: 61379
Vary: Accept-Encoding
Cache-Control: max-age=3600
Expires: Wed, 25 Jul 2018 14:45:23 GMT
Content-Type: text/html

uws@tux>curl -# http://...  # Progress bar

uws@tux>curl -o <OutputFileName> http://...

uws@tux>curl -v ...         # Detailierte Ausgabe

uws@tux>curl -V ...         # Ausgabe der Version
```

Listing 2.90: Beispiel Curl

Ebenso kann man mit `curl` auch Wetter Informationen sich in der Shell anzeigen lassen.

```
uws@tux>curl wttr.in        # Standort wird ermittelt

uws@tux>curl wttr.in/Goch   # Angabe des Ortes

uws@tux>curl wttr.in/Moon   # Mondphase

uws@tux>curl -s ...         # Silent Mode
```

Listing 2.91: Wetter

Eine Hilfe über Befehle kann man sich folgendermaßen anzeigen lassen.

```
uws@tux>curl cheat.sh/ip
```

Listing 2.92: Hilfe Commands

Kapitel 3

Dateisystem

3.1 Logical Volume (LVM) Informationen

Anzeige der Physical Volumes Informationen.

```
uws@tux>pvscan

uws@tux>pvdisplay /dev/sdb
```

Listing 3.1: Beispiel Physical Volume

Anzeige der Volume Group Informationen.

```
uws@tux>vgscan

uws@tux>vgdisplay /dev/volg1
```

Listing 3.2: Beispiel Volume Group

Anzeige der Logical Volume Information.

```
uws@tux>lvscan

uws@tux>lvdisplay /dev/volg1/system
```

Listing 3.3: Beispiel Logical Volume

Mehr zu diesem Thema gibt es in der separaten Dokumentation `Logical Volume Manager`.

3.2 Festplattenplatz

Die Belegung des Festplattenplatzes kann man mit dem Befehl `df` abfragen. Die Option `-hT` gibt die Größe in humanable an und der Typ wird ausgegeben. Mit dem Befehl `du` wird die Größe des Verzeichnisses angezeigt.

```
uws@tux>df -hT
Dateisystem      Typ    Groesse Benutzt Verf. Verw\%  Eingehaengt auf
/dev/mapper/VG1 ext4   150GB      63GB  84GB  43\%   /
/dev/sda1       ext4    99M       14M   85M  15\%   /boot

uws@tux>du -sch
2,8M  .
2,8M  insgesamt

uws@tux>du -sch * --exclude='u0*' --exclude='home'
```

Listing 3.4: Beispiel Festplattenplatz

3.3 Dateisystem konvertieren

3.3.1 EXT2 nach EXT3

Um ein vorhandenes EXT2 Dateisystem nach EXT3 zu konvertieren, so führt man den Befehl tune2fs mit der Option -j aus. Nach einem Neustart steht das neue Dateisystem zu Verfügung.

```
uws@tux>tune2fs -j /dev/sda2
```

Listing 3.5: Beispiel EXT2 nach EXT3

Ein Vorhandenes EXT3 Dateisystem läst sich auch nach EXT2 konvertieren. Auch hier steht das neue Dateisystem nach einem Neustart zu Verfügung.

```
root@tux>tune2fs -f -O ^has_journal /dev/sda2
```

Listing 3.6: Beispiel EXT3 nach EXT2

3.4 Festplatten überwachen (SMART)

Moderne Festplatten sind mit der SMART Technologie ausgestattet. Die Informationen kan man unter Linux mit dem Programm smartctl abrufen. Für die Überwachung der SMART-Festplatten ist der smartctld Daemon zuständig.

```
root@tux>smartctl -i /dev/sda
smartctl 6.6 2017-11-05 r4594 [x86\_64-linux-4.12.14-lp150.12.4-default] (
   SUSE RPM)
Copyright (C) 2002-17, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF INFORMATION SECTION ===
Model Family:     Samsung based SSDs
Device Model:     Samsung SSD 750 EVO 250GB
Serial Number:    S33SNWAH581202D
LU WWN Device Id: 5 002538 d701c5ba6
Firmware Version: MAT01B6Q
User Capacity:    250.059.350.016 bytes [250 GB]
Sector Size:      512 bytes logical/physical
Rotation Rate:    Solid State Device
Form Factor:      2.5 inches
Device is:        In smartctl database [for details use: -P show]
ATA Version is:   ACS-2, ATA8-ACS T13/1699-D revision 4c
SATA Version is:  SATA 3.1, 6.0 Gb/s (current: 6.0 Gb/s)
Local Time is:    Wed Jul 25 17:49:47 2018 CEST
SMART support is: Available - device has SMART capability.
SMART support is: Enabled

root@tux>smartctl -ia /dev/sda # Ausgabe aller Informationen

root@tux>smartctl -H /dev/sda
smartctl 6.6 2017-11-05 r4594 [x86\_64-linux-4.12.14-lp150.12.4-default] (
   SUSE RPM)
Copyright (C) 2002-17, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF READ SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED
```

Listing 3.7: Beispiel smartctl

Die Anzahl der Parkvorgänge der Köpfe kann man auch mit smartctl abfragen. Die Festplatten sind zwischen 300.000 bis 600.000 Parkvorgänge ausgelegt. Diesen Wert findet man unter der Tabelle Vendor Specific SMART Attributes with Thresholds in der Spalte 193 und hat die Bezeichnung Load_Cycle_Count. Der Wert steht dann in der Spalte RAW_Value. Handelt es sich um eine ATA-Festplatte, so fragt man die Informationen mit der Option -d ata ab.

```
root@tux>smartctl -a /dev/sda

root@tux>smartctl -d ata -a /dev/sda
```

Listing 3.8: Beispiel Parkvorgänge

3.5 Wechseldatenträger

3.5.1 schließen / auswerfen

Wechseldatenträger (DVD, USB) können auch von der Kommandozeile ausgeworfen oder auch geschlossen werden.
Die Syntax für den Befehl `eject` lautet: `eject <option> <gerätedatei/mountpoint>`. Wird nur der Befehl `eject` ausgeführt, so wird das Standardgerät ausgeworfen.

```
uws@tux>eject -v /media/disk

uws@tux>eject -v /dev/sr0
```

Listing 3.9: Beispiel eject

Option	*Beschreibung*
-t	schließen
-v	wortreich
-d	Anzeige Standardgerät

Tabelle 3.1: Optionen für eject

3.6 USB Geräte

Hat man ein USB Gerät (Stick, Festplatte) an den Rechner angeschlossen, so kann man sich mit dem Befehl `lsusb` sich die USB-Ports anzeigen lassen. Ist ein Gerät angeschlossen, so wird er hier aufgelistet, ansonsten gibt es Informationen über die USB Anschlüsse.

```
uws@tux>lsusb
Bus 002 Device 004: ID 03f0:231d Hewlett-Packard Broadcom 2070 Bluetooth
   Combo
Bus 002 Device 003: ID 1d57:fa60 Xenta
Bus 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Listing 3.10: Beispiel lsusb

3.7 Formatieren

Eine Partition kann man mit einem Dateisystem formatieren. Unter Linux gibt es hierzu eine vielzahl an Datei Systemen. Eine Partition wird dann mit `mkfs` formatiert. Vorher ermittelt man das richtige Device mit `fdisk -l`.

```
uws@tux>sudo fdisk -l

Festplatte /dev/sdd: 3,7 GiB, 3933208576 Bytes, 7682048 Sektoren
Disk model: STORAGE DEVICE
Einheiten: Sektoren von 1 * 512 = 512 Bytes
Sektorgroesse (logisch/physikalisch): 512 Bytes / 512 Bytes
E/A-Groesse (minimal/optimal): 512 Bytes / 512 Bytes
Festplattenbezeichnungstyp: dos
Festplattenbezeichner: 0x00000000

Geraet     Boot Anfang    Ende Sektoren Groesse Kn Typ
/dev/sdd1        14476 7698571  7684096  3,7G    b W95 FAT32

uws@tux>mkfs.ext3 /dev/sdd1
```

Listing 3.11: Formatieren

Hier ist eine Auswahl an Dateisystemen, die man mit `mkfs` formatieren kann.

- brtfs
- exfat
- ext2/3/4
- f2fs
- fat
- jfs
- msdos
- ntfs
- reiserfs
- vfat
- xfs

3.8 Festplatten / Partitionen anzeigen

Informationen über Festplatten / Partitionen kann man sich mit dem Befehl `fdisk -l` sich anzeigen lassen. Hiermit bekommt man raus, an welchem Device ein USB Gerät hängt und angesprochen werden kann.

```
root@tux>fdisk -l
Festplatte /dev/sda: 232,9 GiB, 250059350016 Bytes, 488397168 Sektoren
Einheiten: Sektoren von 1 * 512 = 512 Bytes
Sektorgroesse (logisch/physikalisch): 512 Bytes / 512 Bytes
E/A-Groesse (minimal/optimal): 512 Bytes / 512 Bytes
Festplattenbezeichnungstyp: dos
Festplattenbezeichner: 0x853b391a

Gerae      Boot    Anfang      Ende  Sektoren Groesse Kn Typ
/dev/sda1  *         2048    206847    204800    100M  7 HPFS/NTFS/exFAT
/dev/sda2          206848 249648116 249441269    119G  7 HPFS/NTFS/exFAT
```

```
/dev/sda3        249649152 251418623   1769472    864M 27 Verst. NTFS WinRE
/dev/sda4        251418624 488396799 236978176    113G  f W95 Erw. (LBA)
/dev/sda5        251420672 259835903   8415232      4G 82 Linux Swap /
   Solaris
/dev/sda6        259837952 444389375 184551424     88G 83 Linux
/dev/sda7        444391424 488396799  44005376     21G 83 Linux


Festplatte /dev/mmcblk0: 3,8 GiB, 4041211904 Bytes, 7892992 Sektoren
Einheiten: Sektoren von 1 * 512 = 512 Bytes
Sektorgroesse (logisch/physikalisch): 512 Bytes / 512 Bytes
E/A-Groesse (minimal/optimal): 512 Bytes / 512 Bytes
Festplattenbezeichnungstyp: dos
Festplattenbezeichner: 0x00000000

Geraet          Boot Anfang    Ende Sektoren Groesse Kn Typ
/dev/mmcblk0p1         8192 7892735  7884544    3,8G  b W95 FAT32
```

Listing 3.12: Beispiel Festplatten Informationen

3.9 Boot Partition kopieren und aktivieren

Soll die Boot Partition auf einer neuen Festplatte kopiert und aktiviert werden, so sind folgenden Schritte durchzuführen. Als erstes wird der Inhalt der ersten Partition zu den neuen Partition kopiert. In dem unteren Beispiel wird angenommen, das die neue Partition unter /dev/sdb befindet.

```
root@tux>dd id=/dev/sda1 of=/dev/sdba [noerror]
```

Listing 3.13: Beispiel Kopieren Partition

In der Datei /etc/fstab wird die neue Partition hinzugefügt und mit einem Kommentarzeichen am Anfang versehen.

```
root@tux>grep -i "boot" /etc/fstab
/dev/disk/by-id/scsi-SATA_VBOX_HARDDISK_VB9606f499-part1 /boot  reiserfs
   acl,user_xattr 1 2
#/dev/disk/by-id/scsi-SATA_VBOX_HARDDISK_VB1d7291b2-part1 /boot reiserfs
   acl,user_xattr 1 2
```

Listing 3.14: Beispiel fstab

Nun kann die alte Partition abgehangen werden. Danach in der Datei `/etc/fstab` das Kommentarzeichen entfernen und es an der alten Partition hinzufügen.

```
root@tux>umount /boot

root@tux>grep -i "boot" /etc/fstab
#/dev/disk/by-id/scsi-SATA_VBOX_HARDDISK_VB9606f499-part1 /boot  reiserfs
   acl,user_xattr 1 2
/dev/disk/by-id/scsi-SATA_VBOX_HARDDISK_VB1d7291b2-part1 /boot reiserfs  acl
   ,user_xattr 1 2
```

Listing 3.15: Beispiel Umount /boot

Nun kann die neue Partition angehangen werden.

```
root@tux>mount /boot
```

Listing 3.16: Beispiel Mount /boot

Als letztes muss für die neue Platte die Option `bootable` aktiviert werden. Für die alte Platte muss diese Option natürlich deaktiviert werden.

```
root@tux>cfdisk /dev/sdb

root@tux>cfdisk /dev/sda
```

Listing 3.17: Beispiel Bootable

3.10 Iso Image mounten

Unter Linux ist es ganz einfach, ein Iso Image in das System einzubinden.

```
uws@tux>mount -o loop <path_to_iso_image> <mount_path>
```

Listing 3.18: Beispiel mount Iso Image

Wenn das Image File ein Abbild einer Festplatte ist und dort mehrere Partitionen enthalten sind, so muss man als erstes den Start Sektor der Partition ermitteln. Anschließend kann man das Image mit einem `offset` mounten.

```
uws@tux>sudo fdisk -l /image/sdb.img
Festplatte sdb.img: 465,78 GiB, 500107862016 Bytes, 976773168 Sektoren
Einheiten: Sektoren von 1 * 512 = 512 Bytes
Sektorgroesse (logisch/physikalisch): 512 Bytes / 512 Bytes
E/A-Groesse (minimal/optimal): 512 Bytes / 512 Bytes
Festplattenbezeichnungstyp: dos
Festplattenbezeichner: 0x2da52da4

Geraet        Boot    Anfang      Ende  Sektoren Groesse Kn Typ
ltgoxxx.img1           2048  24578047  24576000 11,7G   27 Verst. NTFS WinRE
ltgoxxx.img2 *     24578048  24782847    204800  100M    7 HPFS/NTFS/exFAT
ltgoxxx.img3       24782848 507141264 482358417  230G    7 HPFS/NTFS/exFAT
ltgoxxx.img4      507142144 976773119 469630976  224G    f W95 Erw. (LBA)
ltgoxxx.img5      507144192 511365119   4220928    2G   82 Linux Swap /
   Solaris
ltgoxxx.img6      511367168 846915583 335548416  160G   83 Linux
ltgoxxx.img7      846917632 976773119 129855488 61,9G   83 Linux

uws@tux>mount -t auto -o loop,offsett=$((24782848*512)) /image/sdb.img /mnt/
   sdb4
```

Listing 3.19: Beispiel mount Iso Image Partition

3.11 Filesystem anzeigen

Möchte man sich das verwendete Filesystem einer Partition anzeigen lassen, so kann man das mit dem nachfolgenden Befehl machen. In dem zweiten Beispiel wird das Filesystem einer LVM-Partition ausgegeben.

```
uws@tux>file -s /dev/sdc1
/dev/sdc1: Linux rev 1.0 ext3 filesystem data, UUID=67...

uws@tux>file -s /dev/dm-1
/dev/dm-1: ReiserFS V3.6
```

Listing 3.20: Beispiel Filesystem Info

3.12 Hdparm

Mit `hdparm` kann man sich Informationen der Festplatten sich anzeigen lassen.

3.12.1 Festplatte Speed Test

```
uws@tux>hdparm -t /dev/sda
/dev/sda:
 Timing buffered disk reads: 1542 MB in  3.00 seconds = 513.64 MB/sec
```

Listing 3.21: Beispiel Speed Test

3.12.2 Power Management

Die Abfrage des Power Management der Festplatte kann man auch mit `hdparm` sich anzeigen lassen. Gibt es als Rückgabewert 128 oder 254, so werden die Köpfe oft in die Parkposition gefahren. Mit dem Wert 255 schaltet man dieses Verhalten ab. Gilt nicht für SSD-Platten.

```
uws@tux>hdparm -I /dev/sdb | grep -i "Advanced power management"

root@tux>hdparm -B 255 /dev/sdb
```

Listing 3.22: Beispiel Power Management

Damit dieses auch nach einem Neustart des Rechners gemacht wird, so wird der Befehl in der Datei `/etc/rc.local` eingetragen.

3.13 Loopback-Device

Unter Linux kann man ein Loopback-Device (Container) erstellen, das dann in das System gemounted werden kann. Hierzu wird eine Container Datei erstellt und dann mit dem Loopback-Device verbunden. Die Größenangabe der Datei erfolgt in `kb`. Gibt es im Verzeichnis `/dev` keine `loop*` Einträge, so wird `losetup -f` einmal ausgeführt.

```
root@tux> mkdir /container

root@tux>dd if=/dev/zero of=/container/lda bs=1024 count=102400
102400+0 Datensaetze ein
102400+0 Datensaetze aus
104857600 Bytes (105 MB) kopiert, 0,643267 s, 163 MB/s

root@tux>losetup -f # create /dev/loop*, when noct exist

root@tux>losetup /dev/loop1 /container/lda
```

Listing 3.23: Create Container

Nun kann das Loopback-Device formatiert werden.

```
root@tux>mkfs.ext4 /dev/loop1

root@tux>mkdir -p /media/container

root@tux>mount /dev/loop1 /media/container
```

Listing 3.24: Formatieren

Das Löschen des Loopback-Devices wird mit der Option `-d` vorgenommen.

```
root@tux>umount /media/container

root@tux>losetup -d /dev/loop1
```

Listing 3.25: Löschen Loopback-Device

Damit das Loopback-Device bei einem Neustart des Systems automatisch gemounted wird, so ist in der Datei `/etc/fstab` folgendes hinzugefügt.

```
root@tux>grep -i "loop1" /etc/fstab
/dev/loop1  /media/container  ext4  defaults,loop 0 0
```

Listing 3.26: /etc/fstab

Verbundene Loopback-Devices kann man sich mit der Option `-l` anzeigen lassen.

```
root@tux>losetup -l
NAME        SIZELIMIT OFFSET  AUTOCLEAR RO  BACK-FILE
/dev/loop1          0      0          0  0  /container/lda
```

Listing 3.27: Show Loopback-Device

3.14 Festplatten Statistic

Mit dem Befehl `iostat` kann man sich den Durchsatz der Festplatten anzeigen lassen. Mit der Option `-N` wird der Durchsatz von LVM-Devices ausgegeben. Mit `-m` werden die Werte in MB und mit einer `Tahl` wird die Aktualisierung pro Sekunde angegeben. Gibt man die Option `-human` an, so werden die Werte in k, mb ausgegeben.

```
uws@tux>iostat -m 2
```

Listing 3.28: Statistic

3.15 Filesystem Read-only Modus

Befindet sich das Filesystem im Read-only Modus, so kann man es mit einem `remount` wieder in den normalen Modus versetzten.

```
root@tux>mount -o remount /
```

Listing 3.29: Read-only Modus

3.16 Partitions Informationen

3.16.1 Tune2fs

Informationen über EXT-Partitionen kann man sich mit `tune2fs` sich anzeigen lassen. `Tune2fs` kann nur für die Dateisysteme `ext2,3,4` benutzt werden.

```
root@tux>tune2fs -l /dev/sda1
tune2fs 1.44.3 (10-July-2018)
Filesystem volume name:   <none>
Last mounted on:          /
Filesystem UUID:          24e303ee-9611-4479-81be-c418134dd3cf
Filesystem magic number:  0xEF53
Filesystem revision #:    1 (dynamic)
.
.
.
```

Listing 3.30: Beispiel tune2fs

Für Loopback Devices wird `/dev/loop0` und für ein Software Raid ein `/dev/md0` eingesetzt.

3.17 Swap Size

3.17.1 Anzeigen

Die Größe der Swap Partition kann man sich mit dem folgenden Befehl sich anzeigen lassen.

```
uws@tux>cat /proc/swaps
Filename    Type  Size  Used  Priority
/dev/sda5                             partition 4207612 0 -1

uws@tux>swapon -s
Filename    Type  Size  Used  Priority
/dev/sda5                             partition 4207612 0 -1

uws@tux>free -m [g]
      total   used  free  shared  buff/cache   available
Mem:   3847   2355   257     159        1234        1599
Swap:  4108      0  4108

uws@tux>[h]top
```

Listing 3.31: Show Swap

3.17.2 Hinzufügen

Eine neue Swap Partition kann man folgendermaßen einbinden.

```
root@tux>swapon /dev/sda2
```

Listing 3.32: Swap hinzufügen

3.18 Software RAID

3.18.1 Allgemein

Unter Linux kann man ein Software Raid erstellen. Hierzu gibt es das Programm `mdadm`. Für das Raid kann man Festplatten, Partitionen oder auch Loopback Devices nehmen. Die Konfigurationsdatei mdadm.conf befindet sich im Verzeichnis /etc/mdadm.
Mit `mdadm` kann man folgende Raids erstellen.

Raid Level	*Beschreibung*
Linear	Mehrere Partitionen aneinander hängen
Multipath	Ein Mapping einer Datei auf zwei verschiedenen Pfade auf der leichen Partition
Fauly	Emuliert ein fehlerhaftes RAID für Testfälle
0	Stripping, mehrere Platten werden zu einer großen zusammen gefügt
1	Spiegelung einer Platte
4	Wie Raid 0, aber mit einer zusätzlichen Platte für Paritätsbits
5	Mindestens 3 Platten werden zusammen gefügt und die Paritätsbits werden auf alle Platten verteilt
6	Wie Raid 5, nur das zwei unabhängige Paritätsbits geschrieben werden
10	Kombination von Raid 0 und Raid 1

Tabelle 3.2: Raid Levels

Eine Auswahl der Optionen wird in der nachfolgenden Tabelle aufgelistet. Eine vollständige Liste incl. der Beschreibung kann man in der Manpage nachlesen.

Option	*Beschreibung*
–add	Fügt dem Raid eine Platte / Partition hinzu
–backup-file	Erstellt ein Backup File für eine Raid Erweiterung
–detail	Array Details ausgeben
–fail	Status des Raids ändern
–force	Die Ausführung erzwingen
–help	Anzeige des Hilfe Textes
–level=	Raid Type
–query	Überprüfen, ob das Device ein MD-Device oder zu einem Raid gehört
–raid-device	Setzen der Anzahl der Platten in dem Raid
–remove	Löschen einer Platte / Partition aus dem Raid
–stop	Anhalten / Beenden des Raids
–spare-disk	Angabe der Spare Platte
–test	Die angegebenen Optionen testen
–uuid=	Die UUID des Raids
–verbose	Erweiterte Informationen ausgeben, kann 2 mal gesetzt werden
–zero-superblock	Löschen des Raid Superblocks

Tabelle 3.3: Optionen

3.18.2 Raid Erstellen

Damit wir das Erstellen von einem Raid System für dieses Beispiel machen können, erstellen wir erstmal 4 Container mit einer Größe von 500MB und erstellen dann die Loopback Devices.

```
uws@tux>for a in 1 2 3 4; do dd if=/dev/zero of=/container/ld$a bs=1024
   count=512000;done
512000+0 records in
512000+0 records out
524288000 bytes (524 MB) copied, 1.39163 s, 377 MB/s
512000+0 records in
512000+0 records out
524288000 bytes (524 MB) copied, 2.11694 s, 248 MB/s
512000+0 records in
512000+0 records out
524288000 bytes (524 MB) copied, 2.10103 s, 250 MB/s
512000+0 records in
512000+0 records out
524288000 bytes (524 MB) copied, 2.34669 s, 223 MB/s

uws@tux>losetup -f # Wenn /dev/loop* nicht vorhanden
root@tux>for a in 1 2 3 4; do losetup /dev/loop\$a /container/ld$a; done
```

Listing 3.33: Create Loopback Device

Nun können wir Raid Systeme erstellen. Als erstes erstellen wir ein `Raid 0` System.

```
root@tux>mdadm --create /dev/md/md0 --level=0 --raid-devices=2 /dev/loop1 /
   dev/loop2
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md/md0 started.
```

Listing 3.34: Create Raid 0

Ein Raid System mit einer fehlende Platte, die später hinzugefügt werden kann, wird folgendermaßen erstellt.

```
root@tux>mdadm --create /dev/md/md0 --level=1 --raid-devices=2 /dev/loop1
   missing
mdadm: Note: this array has metadata at the start and
    may not be suitable as a boot device. If you plan to
    store '/boot' on this device please ensure that
    your boot-loader understands ms/v1.x metadata, or use
    --metadada=0.90
Continue creating arry? y
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md/md0 started.

root@tux># Fehlende Platte hinzufuegen
root@tux>mdadm --manage /dev/md/md0 --add /dev/loop2
mdadm: added /dev/loop2
```

Listing 3.35: Create Raid 1

Ein `Raid 5` System mit einer Spare Platte erstellt man folgendermaßen.

```
root@tux>mdadm --create /dev/md/md1 --level=5 --raid-devices=3 /dev/loop1 /
   dev/loop2 /dev/loop3 --spare-devices=1 /dev/loop4
```

Listing 3.36: Create Raid 5

Einem vorhandenen Raid System kann man nachträglich auch eine `Spare` Platte hinzufügen. Durch das Hinzufügen einer weiteren Platte / Device, wird diese automatisch als `Spare` eingebunden.

```
root@tux>mdadm --manage /dev/md/md0 --add /dev/loop3
```

Listing 3.37: Add Spare

3.18.3 Raid anzeigen

Ein erstelltes Raid System kann man mit `mdadm --detail` sich anzeigen lassen.

```
root@tux>mdadm --detail /dev/md/md0
        Version: 1.2
  Creation Time: Tue Oct 28 14:12:36 2014
     Raid Level: raid1
     Array Size: 511680 (499.77 MiB 523,96 MB)
  Used Dev Size: 511680 (499.77 MiB 523,96 MB)
   Raid Devices: 2
  Total Devices: 3
    Persistence: Superblock is persistent

    Update Time: Tue Oct 28 14:12:36 2014
          State: clean
 Active Devices: 2
Working Devices: 3
 Failed Devices: 0
  Spare Devices: 1

           Name: tux:md0  (local to host tux)
           UUID: 9dc2cd57:64bf443e:86eea668:95ceb8ad)
         Events: 17

  Number   Major Minor RaidDevice  State
     0       7     1       0       active sync  /dev/loop1
     1       7     2       1       active sync  /dev/loop2
     2       7     3       -       spare  /dev/loop1
```

Listing 3.38: Show Raid Info

Folgende Raid Zustände gibt es. Mit `examine und scan` zeigt den erkannten Array an.

Zustand	*Beschreibung*
Clean	Normalzustand, es liegen keine Fehler vor.
Degrated	Ein Ausfall liegt vor, eine oder mehrere Platten sind defekt.
Resync	Die Sicherungsinformationen werden geprüft und ggf. neu erstellt.
Rebuild	Die Daten werden wieder hergetellt.

Tabelle 3.4: Raid Zustände

```
root@tux>mdadm --examine --scan
ARRAY /dev/md/Datastore  metadata=1.2 UUID=8c08f95f:a4a0da14:d1386b30:
   eeaa6de4 name=fsgo220:Datastore
```

Listing 3.39: Raid ermiiteln

3.18.4 Raid löschen

Bevor ein Raid gelöscht werden kann, muss das Raid angehalten werden. Danach muss von jeder Platte / Device der `superblock` enfernt werden, die die Platte / Device als Raid-Device festlegt. Das Entfernen des Superblocks muss für jede Platte / Device gemacht werden, die dem Raid zugeordnet war.

```
root@tux>mdadm stop /dev/md/md0
mdadm: stopped /dev/md/md0

root@tux>mdadm --zero-superblock /dev/loop1
root@tux>mdadm --zero-superblock /dev/loop2
root@tux>mdadm --remove /dev/md/md0
```

Listing 3.40: Raid löschen

3.18.5 Platte entfernen

Ist eine Platte in dem Raid Verbund defekt, so muss sie mit `--remove` aus dem Raid Verbund entfernt werden.

```
root@tux>mdadm --mange /dev/md/md0 --remove /dev/loop3
mdadm: hot removed /dev/loop3 from /dev/md/md0
```

Listing 3.41: Platte entfernen

Nach dem Entfernen der defekten Platte und dem hinzufügen einer neuen, so wird das Raid System wieder hergestellt. Den Fortschritt des Rebuilds kann man sich anzeigen lassen.

```
root@tux>mdadm --mange /dev/md/md0 --add /dev/loop4
mdadm: added /dev/loop4

root@tux>watch -n 1 cat /proc/mdstat
```

Listing 3.42: Platte hinzufügen

3.18.6 Dateisysten

Das Raid System kann man einfach formatieren.

```
root@tux>mkfs -t ext4 /dev/md/md0 # oder mkfs.ext4

root@tux>mount /dev/md/mdo /media/data
```

Listing 3.43: Formatieren ext4

Wurde ein Raid 0,5,6 oder 10 erstellt, so sollte das Dateisystem für dieses Raid angepasst werden. Als erstes müssen die `Chunk Size` des Raid Systems ermittelt werden.

```
uws@tux>mdadm -D /dev/md/md0 | grep -i "Chunk Size"

  Chunk Size: 512K
```

Listing 3.44: Formatieren Raid 5

Nun können wir die Parameter per Hand ermitteln.

`block-size`	Die Größe der Dateisystemblöcke in Bytes, heute sind es 4096 Byte (4 Kib)
`stride-size`	Die Chunk Size wird durch die `block-size` geteilt (512Kib/4Kib = 128)
`stride-width`	Hier wird die `stride-size` mal der effektiven Partition genommen. Bei einem Raid5 System mit 4 Platten (128*3=384)

```
root@tux>mkfs.ext4 -b 4096 -E stride=128,stripe-width=384 /dev/md/md0
```

Listing 3.45: Formatieren Raid 5

3.18.7 Raid erweitern

In dem ersten Beispiel wird einem Raid eine Platte hinzugefügt. Danach können wir es mit `grow` erweitern. Vorher wurde ein Raid 5 System mit 3 Devices erstellt.

```
root@tux>mdadm --manage /dev/md/md0 --add /dev/loop4
mdadm: added /dev/loop4

root@tux>mdadm --grow --raid-devices=4 /dev/md/md0 --backup-file=/tmp/md0.
    bck
mdadm: Need to backup 3072k of critical section..
```

Listing 3.46: Raid erweitern

Sollte das System bei der Erweiterung abstürzen, so kann es mit dem Backup File fortgesetzt werden.

```
root@tux>mdadm --manage /dev/md/md0 --continue --backup-file=/tmp/md0.bck
```

Listing 3.47: Raid Absturz

Zum Abschluss muss noch das Dateisystem erweitert werden.

```
root@tux>umount /data # oder /dev/md/md0

root@tux>fsck.ext4 -f /dev/md/md0

root@tux>resize2fs /dev/md/md0

root@tux>mount /data
```

Listing 3.48: Dateisystem erweitern

3.18.8 Array wiederherstellen

Ist das Raid nicht mehr funktionstüchtig, weil zwei Platten aus einem Raid 5 System ausgestiegen sind, verwendet man die ersten Platten, um das Raid zu aktivieren. Danch kann man die ausgestiegenen Platten hinzufügen. Dieser Vorgehen wird auch für ein Raid gebraucht, wenn es aus Loop Devices besteht.

```
root@tux>mdadm --stop /dev/md/md0

root@tux>mdadm --assemble /dev/md/md0 /dev/loop1 /dev/loop2 --force

root@tux>mdadm --re-add /dev/md/md0 /dev/loop3 /dev/loop4
```

Listing 3.49: Raid wiederherstellen

3.18.9 Konfiguration sichern

Die Konfiguration des Raid Systems kann man auf zwei Arten sichern.

```
root@tux>mdadm --detail --scan > /etc/mdadm.conf

root@tux>mdadm --detail --brief /dev/md0 > /etc/mdadm.conf
```

Listing 3.50: Konfiguration sichern

Eine Vollständige Sicherung wird ohne die Parameter `scan` / `brief` durchgeführt.

```
root@tux>mdadm --detail /dev/md0 >> /etc/mdadm_full.conf
```

Listing 3.51: Vollständige Sicherung

3.18.10 Tuning Resync / Rebuild

War eine Festplatte defekt und wurde eine neue Platte in das Raid System eingebunden, so findet dann ein `Resync` / `Rebuild` statt. Dieses kann man beschleunigen, in dem man in der Datei `speed_limit_min` einen neuen Wert setzt. Der Standard Wert ist 1000. In der Datei `speed_limit_max` steht die maximale Geschwindigkeit drin. Meistens ist sie auf 200MiB/s gesetzt. Nach einem Nneustart des Systems werden die Standardwerte wieder hergestellt.

```
root@tux>echo 200000 > /proc/sys/dev/raid/speed_limit_min
```

Listing 3.52: Tunig Rebuild

Um den Fortschritt des Rebuilds nach dem Einbau einer neuen Festplatte sich anzeigen zu lassen, wird die Datei `/proc/mdstat` ausgelesen.

```
root@tux>cat /proc/mdstat | grep recovery
[=>.........] recovery =  9.5% (187339328/1951945600) finish=957.6min speed
   =30710K/sec
```

Listing 3.53: Status Rebuild

3.19 SSH Mount

Mit dem Programm `sshfs` kann man ein Verzeichnis von einem anderen Computer mounten. Standardmäßig wird der Mountpoint mit den Credentials des Users `root` gemounted. Mit `-o gid=` und `uid=` kann man einen anderen User angeben.

```
uws@tux>sshfs <user>@<server>:[pfad] <mount_point>

uws@tux>sshfs -o gid=1000,uid=1000 <user>@<server>:[pfad] <mount_point>
```

Listing 3.54: Mount mit ssh

Mit `fusermount -u` wird der mount Point wieder entfernt.

```
uws@tux>fusermount -u <mount_point>
```

Listing 3.55: Umount

SSH-Parameter können mit `SSHOPT` übergeben werden.

```
uws@tux>sshfs -o SSHOPT=
```

Listing 3.56: SSH-Parameter

3.20 Badblocks

Überprüfen der Festplatte / Partition nach Badblocks.

```
uws@tux>sudo badblocks -v /dev/sda1
```

Listing 3.57: Check Badblocks

Überprüfen mit der Angabe der Block Size, default Wert ist 1024.

```
uws@tux>sudo badblocks -v -b 2048 /dev/sda1
```

Listing 3.58: Block Size

Die Ausgabe in einer Datei speichern.

```
uws@tux>sudo badblocks -v -o badblocks.log /dev/sda1
```

Listing 3.59: Speichern in Datei

3.21 Festplatte sicher löschen

Soll eine Festplatte sicher gelöscht werden, so kann man sie mit Nullen überschreiben. Die Angabe des Parameters `conv` ist optional. Mit diesem Parameter wird angegeben, das mit dem überschreiben weiter gemacht werden soll, wenn Festplatten Fehler vorliegen.

```
uws@tux>dd if=/dev/zero of=/dev/sdf bs=1MB conv=noerror
```

Listing 3.60: Mit Nullen überschreiben

Soll die Festplatte mit Zufallszahlen überschrieben werden, so wird `urandom` anstelle von `zero` genommen. Dieses überschreiben kann bei einer 3TB Platte schon eine Woche dauern, da die Geschwindigkeit nur 10-20Mbytes/s ist.

```
uws@tux>dd if=/dev/urandom of=/dev/sdf bs=1MB conv=noerror
```

Listing 3.61: Zufallszahlen

Schneller geht das überschreiben mit `openssl`. Dieses dauert bei einer 3TB Platte ca. 5 Stunden. `Urandom` erzeugt ein zufälliges Passwort und `openssl` erzeugt den Zufallsstrom durch AES-Verschlüsselung. (Aus der C'T 2/2016 S.161)

```
uws@tux>openssl enc -qes-256-ctr -pass pass:"$(dd if=/dev/urandom bs=128
   count=1 2>/dev/null|base64)" -nosalt </dev/zero >/dev/sdf
```

Listing 3.62: Openssl

Eine Festplatte kann mit dem Befehl `shred` mehrfach überschrieben werden. Die Anzahl des überschreibens wird mit `iterations` angegeben. Der Parameter `-v` steht für `verbose`.

```
uws@tux>shred --iterations=7 -v /dev/sdf
```

Listing 3.63: Shred

3.22 BTRFS

3.22.1 Speicherplatz Belegung

Das Kommando `df` gibt für das Dateisystem `BTRFS` nicht die richtige Menge des belegten Platzes aus. Damit man sich den belegten Platz anzeigen lassen kann, muss das Kommando innerhalb von `btrfs` angegeben werden.

```
uws@tux>df -hT | grep -i "btrfs"
/dev/sda6      btrfs      89G     38G   50G   44% /
/dev/sda6      btrfs      89G     38G   50G   44% /boot/grub2/x86_64-efi
/dev/sda6      btrfs      89G     38G   50G   44% /.snapshots
/dev/sda6      btrfs      89G     38G   50G   44% /boot/grub2/i386-pc
/dev/sda6      btrfs      89G     38G   50G   44% /var/spool
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/libvirt/images
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/mailman
/dev/sda6      btrfs      89G     38G   50G   44% /srv
/dev/sda6      btrfs      89G     38G   50G   44% /opt
/dev/sda6      btrfs      89G     38G   50G   44% /var/opt
/dev/sda6      btrfs      89G     38G   50G   44% /var/log
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/mariadb
/dev/sda6      btrfs      89G     38G   50G   44% /var/crash
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/machines
/dev/sda6      btrfs      89G     38G   50G   44% /usr/local
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/mysql
/dev/sda6      btrfs      89G     38G   50G   44% /tmp
/dev/sda6      btrfs      89G     38G   50G   44% /var/tmp
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/pgsql
/dev/sda6      btrfs      89G     38G   50G   44% /var/lib/named

uws@tux>btrfs filesystem df /
Data, single: total=43.00GiB, used=35.73GiB
System, single: total=32.00MiB, used=16.00KiB
Metadata, single: total=3.00GiB, used=1.69GiB
GlobalReserve, single: total=112.25MiB, used=0.00B
```

Listing 3.64: Speicherplatz Belegung

3.22.2 Snapshot anzeigen

Unter `OpenSuSE` und auch in anderen Distributionen gibt es für die Snapshots das Tool `snapper`. Mit `snapper` lassen sich Dateien auf früheren Snapshots wieder herstellen. Außerdem lassen sich System-Rollbacks durchführen, Systemänderungen rückgängig machen und Snapshots manuell erstellen / Verwalten. Eine GUI für Snapper gibt es unter http://snapper.io.

Die Konfigurations-Datei `root` befindet sich im Verzeichnis `/etc/snapper/configs`. Die Konfiguration kann man sich folgendermaßen anzeigen lassen.

```
rootqtux>snapper list-configs
Config | Subvolume
-------+----------
root   | /
```

Listing 3.65: Show Config

Auflisten der angefertigten Snapshots.

```
root@tux>snapper list # oder auch snapper ls
```

Listing 3.66: Show Snapshots

Kapitel 4

Paketmanager

4.1 RPM Paket Manager

Der Paket Manager `RPM` wurde ursprünglich von Red Hat entwickelt und unter der DNU General Public License (GPL) gestellt. Heute verwenden diesen Paket Manager außer Red Hat noch OpenSuSE, Fedora und CentOS. Die erstellten Pakete haben die Endung RPM.
Der Paket Manager kopiert alle im Paket enthaltenen Dateien an die festgelegte Stelle, führt alle nozwendigen Scripte aus und aktualisiert dann die RPM Datenbank. Die RPM Datenbank handelt es sich um eine handvoll Dateien auf BDB-Basis. Die Datenbank befindet sich im Verzeichnis `/var/lib/rpm`.
Gibt es bei der Installation des Paketes Abhängigkeiten mit anderen Programmen oder Bibliotheken, so werden diese nicht automatisch aufgelöst. Das muss der Anwender dann selber erledigen.

4.1.1 Paket Installieren

Ein Software Paket aus dem Internet, von einer DVD oder von der lokalen Festplatte, wird mit dem Befehl `rpm -i` unstalliert.

```
uws@tux>sudo rpm -i <paketname>
```

Listing 4.1: Paket installieren

Möchte man bei der Installation des Paketes mehr Informationen erhalten, so gibt man die Option `-v` und `-h` an. Mit diesen beiden Optionen werden der Status und der Fortschritt angezeigt.

```
uws@tux>rpm -ivh "http://<domainname>/<paketname>
```

Listing 4.2: Fortschritt anzeigen

Soll ein Paket installiert werden, ohne vorher die Abhängigkeiten aufzulösen, so gibt man die Option `--nodeps` oder `--force` an. Die dann installierte Software wird höchstwarscheinlich nicht funktionieren.

```
uws@tux>rpm -i --nodeps <paketname>
```

Listing 4.3: Nodeps

4.1.2 Paket update

Ist ein Paket installiert und man möchte eine neuere Version installieren, so wird hier die Option `-U` benutzt.

```
uws@tux>rpm -U <paketname>
```

Listing 4.4: Paket update

4.1.3 Failed to cache

Bei einer Installation / Update kommt die Meldung: `Failed to cache rpm database`, so muss man die RPM-DB neu erstellen.

```
uws@tux>sudo rpm --rebuild
```

Listing 4.5: Create RPM-DB

4.1.4 Paket löschen

Um ein installiertes Paket loszuwerden, gibt man die Option `-e` an.

```
uws@tux>rpm -e <paketname>
```

Listing 4.6: Paket deinstallieren

Wird versucht, ein Paket zu deinstallieren, welche andere Programme benötigen, so scheitert dieser Versuch. Mit der Option `--whatrequires` kann man sich anzeigen lassen, welche Pakete die zu deinstallierete Software benötigen. Diese Pakete gilt es dann vorher zu löschen. Mit der Option `--nodeps` oder `--force` werden die Abhängigkeiten ignoriert und das Paket wird deinstalliert.

```
uws@tux>rpm -e --whatrequires <paketname>
```

Listing 4.7: Paket löschen

4.1.5 Paket Abfrage

Die RPM Datenbank wird mit der Option `-q` abgefragt.

```
uws@tux>rpm -q <paketname>

uws@tux>rpm -qpl <paketname>

uws@tux>rpm -q --scripts <paketname>

uws@tux>rpm -qa | grep binutil
```

Listing 4.8: DB Abfragen

Schalter	*Beschreibung*
-a	Alle Pakete auflisten.
-l	Auflisten aller Dateien im Paket
-c	Nur die Konfigurationsdateien anzeigen
-d	Anzeigen der enthaltenen Dokumentation
-i	Anzeige der Meta-Informationen
-p	Abfrage eines nicht installierten Paketes
--scripts	Anzeige der Scriptinhalte
--changelog	Anzeige einer Pakethistorie
--whatrequires	Anzeige, welche Abhängigkeiten es gibt

Tabelle 4.1: Abfrage Schalter

4.1.6 Abhängigkeiten Abfrage

Welche Abhängigkeiten es für ein Paket gibt, kann man mit der folgenden Abfrage sich anzeigen lassen.

```
uws@tux>rpm -q --whatrequires <paketname>
```

Listing 4.9: Abfrage Abhänigkeiten

4.1.7 Install Datum

Das Installationsdatum für ein Paket kann man sich mit der Option `--last` anzeigen lassen.

```
uws@tux>rpm -qa --last | grep firefox

firefox-17.0.6-1.0.1.el6_4.x86_64 Do 16 Mai 2013 15:27:29 CEST
```

Listing 4.10: Install Datum

4.2 Zypper (SuSE)

Mit dem Programm `Zypper` kann man in der Kommandozeile Software installieren, aktualisieren oder auch deinstallieren. Eine Übersicht über die vorhandenen Optionen gibt es mit dem Befehl `zypper --help` oder auch in der `Manpage`. In der Konfigurationsdatei `/etc/zypper/zypp.conf` befinden sich viele hilfreiche Kommentare zu dem Programm.

4.2.1 Paket Installieren

Software wird mit der Option `install` installiert. Ein bestehendes Paket wird mit der Option `-f` erneut installiert.

```
uws@tux>sudo zypper install <paketname>

uws@tux>sudo zypper install -r "openSuSE-10.3-DVD" <paketname>

uws@tux>sudo zypper -f <paketname>
```

Listing 4.11: Software installieren

Ganze Programm Gruppen (Zypper Pattern) kann man auch in einem Rutsch installieren.

```
uws@tux>sudo zypper install -t pattern devel_basis
```

Listing 4.12: Zypper Pattern

Sollen nur die Quellcodes eines Programm Paketes installiert werden, so gibt man die Option `srcpackage` an.

```
uws@tux>sudo zypper install -i srcpackage gimp
```

Listing 4.13: Quellcode installieren

4.2.2 Paket deinstallieren

Ein Programm wird man mit der Option `remove` wieder los.

```
uws@tux>sudo zypper remove <paketname>
```

Listing 4.14: Software deinstallieren

4.2.3 Paket suchen

Die Suche mach einem Paket kann mit der Option `search` gemacht werden.

```
uws@tux>sudo zypper search <paketname>
```

Listing 4.15: Paket suchen

Man kann nicht nur nach einzelnen Paketen suchen, sondern auch nach ganzen Programm Gruppen. Diese Gruppen werden im Zypper Sprachgebrauch `Pattern` genannt. Bei dem nachfolgenden Beispiel werden alle Pakete gesucht, die der Gruppe game angehören.

```
uws@tux>sudo zypper search -t pattern game
```

Listing 4.16: Paket Gruppe

Um sich vorliegende Patche anzeigen zu lassen, so gibt man die Option `patch` an.

```
uws@tux>sudo zypper search -t patch
```

Listing 4.17: Paket Patche

Installierte Pakete kann man sich mit der Option `-i` anzeigen lassen.

```
uws@tux>sudo zypper search -i | less
```

Listing 4.18: Installierte Pakete

4.2.4 Paket update

Mit dem Befehl `zypper lu` kann man überprüfen, ob für die installierten Pakete updates vorliegen. Bei diesem Befehl werden nicht nur die Pakete angezeigt, sondern auch die dazugehörenden Paketquellen. Mit `zypper update` kann man dann die Pakete einspielen.
Sollen alle Pakete auf einmal eingespielt werden, so gibt man `zypper dup` ein. Möchte man nur die Pakete aktualisieren, die vom SuSE Team als kritisch eingestuft wurde, so gibt man `zypper up` ein.

```
uws@tux>sudo zypper lu <paketname>

uws@tux>sudo zypper update <paketname>

uws@tux>sudo zypper dup

uws@tux>sudo zypper up
```

Listing 4.19: Paket update

Mit `zypper update -d` werden die Pakete heruntergeladen, aber nicht installiert. Für die Installation ruft man dann `zypper update` ohne die Option `-d` auf.

```
uws@tux>sudo zypper update -d

uws@tux>sudo zypper update
```

Listing 4.20: Paket download

4.2.5 Repositories verwalten

Unter dem Verzeichnis `/var/cache/zypp/raw` befinden sich jeweils in einem eigenen Verzeichnis pro Repository die Meta-Daten mit den Datei Informationen. Diese Meta-Daten lassen sich mit dem Befehl `zypper refresh` auf den neusten Stand bringen. Außerdem werden im Verzeichnis `/var/cache/zypp/solve` die Informationen abgelegt, welche Abhängigkeiten ein Paket verursacht und wie diese am besten / schnellsten aufzulösen sind. Im Verzeichnis `repodata` liegen die Meta-Informationen, hierbei handelt es sich um Gzip-Dateien, in denen XML-Dateien sich befinden.

```
uws@tux>sudo zypper refresh
```

Listing 4.21: Repository refresh

Die Nummer eines Repositories wird mit `zypper -lr` abgefragt. Mit der Option `-u` werden zusäzlich die Repository-URI angezeigt. Die Option `-p` zeigt die Proirity für die Repositories an. Alle Eigenschaften der Repositories können mit der Option `-a` angezeigt werden.

```
uws@tux>sudo zypper lr
# | Alias              | Name                    |Enabled|Refresh
--+------------------+-----------------------+-------+--------
1 | Update SuSE 11.4 | Updates openSuSE 11.4 | YES   | YES
2 | openSuSE-11.4    | openSuSE-11.4         | YES   | NO
3 | repo-non-oss     | openSuSE-11-4-non-oss | YES   | YES
```

Listing 4.22: List Repositories

Einen `Export` der Eigenschaften der Repositories kann mit der Option `--export` durchgeführt werden. In dem nachfolgenden Beispiel wird der Export in die Datei backup.repo geschrieben.

```
uws@tux>sudo zypper lr --export /daten/backup/backup.repo
Repositories have been successfully exported to /daten/backup/backup.repo
```

Listing 4.23: Export Repositories

Der `Import` wird mit `ar` und der Angabe der Datei gemacht.

```
uws@tux>sudo zypper ar /daten/backup/backup.repo
```

Listing 4.24: Import Repositories

Einen automatischen Check eines Repositories schaltet man mit `zypper mr -R` aus. Mit der Option `mr -d` schaltet ein Repositorie aus.

```
uws@tux>sudo zypper mr -R <repo_nr>

uws@tux>sudo mr -d <repo_name>

uws@tux>sudo mr -d <repo_nr>
```

Listing 4.25: Repository check

Soll verhindert werden, dass jede Installation eines Pakets durchgeführt wird, so richtet man eine Paktesperre ein. Mit der Option `ll`, kann man sich die Paktesperren anzeigen lassen. Zum löschen einer Paketsperre wird die Option `cl` benutzt.

```
uws@tux>sudo zypper al gimp

uws@tux>sudo zypper ll

uws@tux>sudo zypper cl gimp
```

Listing 4.26: Paketsperre

Nach einer Installation löscht der Paketmanager alle Dateien. Möchte man diese aber behalten, so wird dieses mit der Option `mr --kt` eingeschaltet.

```
uws@tux>sudo zypper mr --kt
```

Listing 4.27: Pakete behalten

Einen vergebenen Alias Namen kann man mit `renamerepo` ändern.

```
uws@tux>sudo zypper renamerepo repo-source repo-source-new
```

Listing 4.28: Rename Repository

Ein Repository wird mit `addrepo` hinzugefügt. Nach der Eingabe der URL wird noch ein Alias Namen vergeben.

```
uws@tux>sudo zypper addrepo --name "VLC" http://download.vlc.org VideoLan
Adding repository 'VLC' [done]
Repository 'VLC' successfully added
Enabled: YES
Autorefresh: No
URI: http://download.vlc.org
```

Listing 4.29: Added Repository

Ein nicht mehr benötigtes Repository kann man mit `removerepo` oder auch `rr` löschen.

```
uws@tux>sudo zypper rr VLC
Removing repository 'VLC' [done]
Repository 'VLC' has been removed
```

Listing 4.30: Repository löschen

In der nachfolgenden Tabelle sind alle wichtigen Befehle des Repo-Managers aufgelistet.

4.2.6 Repository Prioritäten setzten

Über die Priorität wird festgelegt, welche Installationsquelle der Paketmanager nutzen soll. Die Standardeinstellung für die Repositories ist 99. Je höher der Wert, desto geringer ist die Priorität. Bieten zwei Repositories dasselbe Paket an, so wird das Paket von der höheren Priorität installiert. Soll zum Beispiel auf das DVD Repository nur dann zugegriffen werden, wenn keine Internetverbindung besteht, so setzt man die Priorität des Repositories auf 100. Aber Achtung, einen kleineren Wert als 99 sollte nicht vergeben werden, da das die Stabilität des Systems beeinträchtigen kann.
In dem nachfolgenden Beispiel wird für alle lokalen Repositories die Priorität auf 100 gesetzt.

Befehl	Kurzschreibweise	Beschreibung
zypper repos	zypper lr	Listet die eingerichteten Paketquellen auf
zypper addrepo <url>	zypper ar <url>	Fügt ein Repository unter dem gewählten Namen hinzu
zypper removerepo <name>	zypper rr <name>	Entfernt das Repository
zypper renamerepo <nr> <name>	zypper nr <nr> <name>	Vergibt einem Repository einen neuen Namen
zypper modifyrepo --enable <name>	zypper mr -e <name>	Aktiviert ein Repository
zypper modifyrepo --disable <name>	zypper mr -d <name>	Schaltet ein Repository aus
zypper modifyrepo --refresh <name>	zypper mr -r <name>	Schaltet die automatische Aktualisierung ein
zypper modifyrepo --no-refresh <name>	zypper mr -R <name>	Schaltet die automatische Aktualisierung aus
zypper modifyrepo --priority <wert> <name>	zypper mr -r <wert> <name>	Setzt die Priorität des Repos auf den angegebenen Wert (kleiner ist höher)

Tabelle 4.2: Befehle Repo-Manager

```
uws@tux>sudo zypper mr -p 100 -t -l
```

Listing 4.31: Setzten Priorotät

4.2.7 System Upgrade

Möchte man ein System Upgrade durchführen, also z.B. von OpenSuSE 12.3 nach 13.1, so kann man das mit `zypper` ganz einfach erledigen. Nachfolgend sind die einzelnen Schritte aufgeführt, um ein System Upgrade durchzuführen.

Achtung: Es ist dafür zu sorgen, das ausreichend Platz für das Upgrade auf dem System zu Berfügung steht. Das Verzeichnis `/boot` sollte mindestens `300MB` groß sein. Auch das / Verzeichnis sollte mindestens über `6GB` freien Platz verfügen. Das gleiche gilt auch für `/var`, wenn es sich in einer separaten Partition befindet.

Als erstes wird überprüft, ob das Repo-Update aktiv ist. Man kann auch den Befehl `zypper lr -uP` nehmen, dann wird zusätzlich noch die Prio ausgegeben.

```
uws@tux>sudo zypper repos -ur
#|Alias       |Name        |Enabled|Refresh|URI
-+-----------+-----------+-------+-------+-----------------------------------

1|repo-update|repo-update|yes     |yes     |http://download.opensuse.org/up..
```

Listing 4.32: Repo check

Steht in der Spalte `Enabled` bei Repo-Update kein `Yes`, so setzt man in der Konsole folgenden Befehl ab.

```
uws@tux>sudo zypper modifyrepo --enable repo-update
```

Listing 4.33: Repo Enabled

Wird kein Eintrag für das Repo-Update angezeigt, so wird mit `zypper addrepo` das fehlende Repository hinzugefügt.

```
uws@tux>sudo zypper addrepo --check --name 'openSUSE-12.3-Update' http://
   download.opensuse.org/update/12.3 repo-update
```

Listing 4.34: Add Repo

Nun können wir das Alte System auf dem letzten Patchlevel heben.

```
uws@tux>sudo zypper refresh

uws@tux>sudo zypper update
end{lstlisting}
Nach erfolgtem Update werden die alten Repositories abgeschaltet.
\listBash
\begin{lstlisting}[captionpos=b, caption=Disable Old Repo, label=lst:bash]
uws@tux>sudo zypper modifyrepo --all --disable
```

Listing 4.35: Letzter Patchlevel

Jetzt können die neuen Repositories hinzugefügt werden.

```
uws@tux>sudo zypper addrepo --name "openSUSE-13.1 OSS" http://download.
   opensuse.org/distribution/13.1/repo/oss repo-13.1-oss

uws@tux>sudo zypper addrepo --name "openSUSE-13.1 Non-OSS" http://download.
   opensuse.org/distribution/13.1/repo/non-oss repo-13.1-non-oss

uws@tux>sudo zypper addrepo --name "openSUSE-13.1 Updates OSS" http://
   download.opensuse.org/update/13.1 repo-13.1-update-oss

uws@tux>sudo zypper addrepo --name "openSUSE-13.1 Updates Non-OSS" http://
   download.opensuse.org/update/13.1-non-oss repo-13.1-update-non-oss
```

Listing 4.36: Add new Repo

Nach diesen ganzen vorarbeiten, können wir nun das Upgrade durchführen.

```
uws@tux>sudo zypper clean -a

uws@tux>sudo zypper ref

uws@tux>sudo zypper dup
```

Listing 4.37: Upgrade System

4.2.8 Patche prüfen

Mit dem Kommando `pchk` (patch-check) wird abgefragt, ob für das System neue Patche zu Verfügung stehen.

```
uws@tux>sudo zypper pchk
Loading repository data ...
Reading installed packages ...
5 patches needed (2 security patches)
```

Listing 4.38: List Patche

4.2.9 Benötigte Patche auflisten

Die benötigten Patche können mit der Option `list-patches` oder `lp` angezeigt werden.

```
uws@tux>sudo zypper lp
Loading repository data ...
Reading installed packages ...

Repository            | Name                   | Version | Category   | Status
----------------------+------------------------+---------+------------+-------
Update openSUSE 11.4 | apache2-mod_apparmor | 4985    |recommended| needed
Update openSUSE 11.4 | kdebase4-openSUSE    | 5043    |recommended| needed
Update openSUSE 11.4 | libxcrypt            | 5049    |security   | needed
```

```
Update openSUSE 11.4 | man-pages            | 5032    |security    | needed
Update openSUSE 11.4 | valgrind             | 5053    |recommended| needed
```

Listing 4.39: Benötigte Patche

Mit der Option `lu` werden alle verfügbaren Paketaktualisierungen angezeigt.

```
uws@tux>sudo zypper lu
Loading repository data ...
Reading installed packages ...
s | Repository          | Name         |Current Vers |Avaible Vers | Arch
--+---------------------+--------------+-------------+-------------+------
V | Update openSUSE 11.4|apparmor-docs |2.5.1.r455-52|2.5.1.r445-63|noarch
```

Listing 4.40: Verfügbare Patche

4.2.10 Patche installieren

Die Patche können mit dem Befehl `zypper patch` installiert werden.

```
uws@tux>sudo zypper patch
Loading repository data ...
Reading installed packages ...
Resolving package dependencies ...

The following NEW patches are going to be installed:
 apache2-mod_apparmor kdebase4-openSUSE libxcrypt man-pages valgrind

The following packages going to be upgraded:
 apparmor-dosc apparmor-parser apparmor-profiles apparmor-utils

18 packages to upgrade
Overall download size: 9.3 MiB. After the operation, 8.6 Mib will be freed.
Continue? [y/n/?] (y):
```

Listing 4.41: Patche installieren

4.2.11 Alle Patche auflisten

Alle Patche werden mit der Option `patches` angezeigt.

```
uws@tux>sudo zypper patches
Loading repository data ...
Reading installed packages ...
Catalog              | Name           |Version | Category     | Status
--------------------+---------------+--------+-------------+----------
Update openSUSE 11.4| Mesa           | 4546   | recommended | installed
Update openSUSE 11.4| ModemManager   | 4453   | recommended | installed
Update openSUSE 11.4| MozillaFirefox| 4195   | recommended | installed
Update openSUSE 11.4| MozillaFirefox| 4457   | security    | installed
```

Listing 4.42: Alle Patche auflisten

4.2.12 Patch Informationen abfragen

Informationen über die einzelnen Patche kann man mit `info -t patch` abfragen.

```
uws@tux>sudo zypper info -t patch MozillaFirefox
Loading repository data ...
Reading installed packages ...

Informationen zu MozillaFirefox
```

```
Name: MozillaFirefox
Version: 5020
Arch: noarch
Vendor: maint-coord@suse.de
Status: installed
Category: security
Created On: Fri Aug 19 19:47:02 2011
Reboot Required: No
Package Manager Restart Required: No
Interactive: No
Summary: MozillaFirefox: Update to Firefox 6
Description:
MozillaFirefox was updated to version 6
```

Listing 4.43: Patch Informationen

4.2.13 Empfohlene Pakete installieren

Für bereits installierte Pakete findet Zypper mit der Option `install-new-recommends (inr)` neu hinzugefügte empfohlene Pakete.

```
uws@tux>sudo zypper inr
Installierte Pakete lesen...

Die folgenden NEUEN Pakete werden installiert.
```

Listing 4.44: Empfohlene Pakete

4.2.14 Abhängigkeiten prüfen

Mit `zypper verify (ve)` kann man Paketabhängigkeiten prüfen und diese dann installieren. Startet ein installiertes Programm nicht mehr und es gibt eine Fehlermeldung, das etwas fehlt, so kann dieses geprüft werden.

```
uws@tux>sudo zypper -e nodeps mozilla-xulrunner190
uws@tux>firefox
Could not find compatible GRE between version 1.9.0 and 1.9.0

uws@tux>sudo zypper ve
Installierte Pakete lesen...
Einige der Anhaegigkeiten der installierten Pakete sind beschaedigt. Um
   diese Abhaengigkeitenzu reparieren ...
```

Listing 4.45: Abhängigkeiten prüfen

4.2.15 Spzialfälle

Sollen nur Pakete installiert werden, die kein Neustart, Bestätigungsmeldung anzeigen oder eine Lizenz angenommen werden soll, so gibt man die Option `--skip-interaktiv` an. In der legalen Grauzone bewegt sich die Option `-l`. Diese Option weist den Paketmanager an, alle Lizenzklauseln zu akzeptieren.

```
uws@tux>sudo zypper update --skip-interactive
```

Listing 4.46: Update

Möchte man nur den Source Code eines Pakets herunterladen, so gibt es hierzu die Option `source-install (si)`. Nach dem Herunterladen kann man das Paket von Hand übersetzen.

```
uws@tux>sudo zypper si -d inkscape
```

Listing 4.47: Download Source Code

Soll ein Paket mit einer bestimmten Versionsnummer deinstalliert werden, so geschieht das mit dem nachfolgenden Befehl. Anstelle von dem = kann man auch < setzen, dann werden alle Versionen deinstalliert, die kleiner sind als die angegebene Nummer. Genauso funktioniert auch das einspielen einer bestimmten Version eines Paketes.

```
uws@tux>sudo zypper remove "amarok=2.2"

uws@tux>sudo zypper install "amarok=2.2.0-2"
```

Listing 4.48: Paket deinstalliern

Um zu erfahren, wer das Paket gebaut hat, ob es installiert ist oder aus welcher Bibliothek es stammt, so wird hierbei die Option `info` erwendet.

```
uws@tux>sudo zypper info libxinel
```

Listing 4.49: Paket Info

4.2.16 Proxy Einstellung

Gibt es einen Proxy Server im Netzwerk, so muss dieser in Yast (Netzwerkdienste - Proxy) eingetragen werden, incl. der Portnummer wie z.B. `http://192.168.1.18:80`.
Ist der Proxy nicht eingetragen, können keine Updates heruntergeladen werden und es kommt die Fehlermeldung `couldn't connect to host`.

4.2.17 Alte Kernel löschen

Als erstes lassen wir uns anzeigen, welche Kernel Versionen installiert sind. Der nachfolgende Befehl listet alle Kernel Versionen sortiert nach Datum auf.

```
uws@tux>sudo rpm -pa -last kernel-\*
kernel-sysms-3.11.10-11.1.x86_64          Wed May 21 13:55:49 2014
kernel-default-devel-3.11.10-11.1-x86_64 Wed May 21 13:55:47 2014
kernel-desktop-devel-3.11.10-11.1-x86_64 Wed May 21 13:55:46 2014
kernel-source-3.11.10-11.1.noarch         Wed May 21 13:55:42 2014
kernel-devel-3.11.10-11.1.noarch          Wed May 21 13:55:42 2014
kernel-desktop-3.11.10-11.1.x86_64        Wed May 21 13:55:42 2014
kernel-sysms-3.11.10-11.1.x86_64          Tue Feb 11 07:29:26 2014
kernel-default-devel-3.11.10-7.1.x86_64   Tue Feb 11 07:29:24 2014
kernel-desktop-devel-3.11.10-7.1.x86_64   Tue Feb 11 07:29:23 2014
kernel-source-3.11.10-7.1-noarch          Tue Feb 11 07:29:16 2014
kernel-devel-3.11.10-7.1-noarch           Tue Feb 11 07:29:11 2014
kernel-desktop-3.11.10-7.1.x86_64         Tue Feb 11 07:28:48 2014
```

Listing 4.50: Kernel auflisten

Nun können wir die alten Kernel Versionen löschen.

```
uws@tux>sudo zypper rm kernel-desktop-devel-3.11.10-7.1 \
kernel-devel-3.11.10-7.1 \
kernel-desktop-3.11.10-7.1
```

Listing 4.51: Alte Kernel löschen

4.2.18 Kernel installieren

Mit eeinem zypper Command, lassen sich alle verfügbaren Kernel Versionen anzeigen.

```
uws@tux>sudo zypper se -s 'kernel*'
S | Name            | Type        | Version          | Arch   | Repository
--+----------------+------------+-----------------+--------+--------------
v | kernel-default | package     | 2.6.32.10-0.4.1 | x86_64 | Alternativ...
i | kernel-default | package     | 2.6.32.10-0.4.1 | x86_64 | (System Pack...
```

```
  | kernel-default | srcpackage | 2.6.32.10-0.4.1 | x86_64 | Alternativ...
```

Listing 4.52: List Kernel Versionen

Installieren eines Kernels.

```
uws@tux>sudo zypper in kernel-default-2.6.12.10-0.4.1
```

Listing 4.53: Kernel installieren

Damit man auf dem System verschiedene Kernel installieren und betreiben kann, muss die Option `multiversion` in der `/etc/zypp/zypp.conf` gesetzt sein. Standardmäßig ist diese Option gesetzt.

```
uws@tux>grep -i "multiversion" /etc/zypp/zypp.conf
multiversion = provides:multiversion(kernel)
```

Listing 4.54: Option multiversion

4.2.19 Patch descline (sperren)

Läst sich ein Patch nicht installieren, so kann man ihn sperren, so das er nicht mehr installiert wird. Eine eingerichtete Sperre kann man mit `zypper locks` sich anzeigen lassen.

```
uws@tux>zypper locks
# | Name               | Type  | Repository
--+-------------------+-------+-----------
1 | OpenSUSE-2016-923 | patch | (any)
```

Listing 4.55: Show Locks

Einen Patch sperren kann man mit `zypper addlock` vornehmen. Bei der Option `-t` wird der Typ angegeben. Folgende Typen gibt es: `package, patch, pattern, product`.

```
uws@tux>sudo zypper addlock -t patch OpenSUSE-2016-923
```

Listing 4.56: Patch sperren

Eine Sperre kann mit `zypper rl` wieder aufgehoben werden.

```
uws@tux>sudo zypper rl OpenSUSE-2016-923
```

Listing 4.57: Patch Sperre aufheben

4.3 Yum (Red Hat)

Um Software unter Red Hat zu installieren, gibt es das Programm `yum` in der Shell.

4.3.1 Proxy Einstellung

Damit man Software installieren kann, sollte man in der `.bashrc` für den User `root` folgende Variablen hinzufügen.

```
root@tux>grep http .bashrc
export http_proxy=http://10.18.14.81:80/
export ftp_proxy=http://10.18.14.81:80/
export HTTP_PROXY=http://10.18.14.81:80/
export FTP_PROXY=http://10.18.14.81:80/
```

Listing 4.58: Proxy Variablen

4.3.2 Software installieren

Mit `install` oder auch `groupinstall` werden Pakete installiert. Mit der Option `-y` wird das Paket ohne Rückfrage installiert.

```
root@tux>yum groupinstall kde-desktop gnome-dektop

root@tux>yum -y install samba.x86_64
```

Listing 4.59: Software installieren

4.3.3 Software löschen

Mit `remove` kann man installierte Pakete löschen und mit `groupremove` ganze Paketgruppen.

```
root@tux>yum remove firefox.x86_64

root@tux>yum groupremove gnome-desktop
```

Listing 4.60: Software deinstallieren

4.3.4 Software aktualisieren

Ein Paket wird mit `update` aktualisiert. Ohne die Angabe des Software Pakets, werden alle Applikationen aktualisiert. Gibt man noch die Option `-y` mit an, so werden die Updates ohne nachfragen installiert.

```
root@tux>yum update firefox.x86_64

root@tux>yum -y update
```

Listing 4.61: Software aktualisieren

Vor einem Update kann man sich die Pakete anzeigen lassen, die aktualisert werden.

```
root@tux>yum list updates
Loaded plugins: downloadonly, refresh-packagekit, security
Updated Packages
coreutils.x86_64 8.4-19.0.1.el6_4.2 ol6_latest
coreutils-libs.x86_64 8.4-19.0.1.el6_4.2 ol6_latest
gdb.x86_64 7.2-60.el6_4.1 ol6_latest
glx-utils.x86_64 9.0-0.8.el6_4.3 ol6_latest
.
```

Listing 4.62: Updates auflisten

Ganze Paketgruppen lassen sich mit `groupupdate` aktualisieren.

```
root@tux>yum groupupdate 'Graphical Internet'
Dependencies Resolved
Upgrade 5 Package(s)
Is this ok [Y/N]: y
Running Transaction
Updating : evolution-data-server-3.0.2-1.el6.x86_64     1/10
Updating : evolution-3.0.2-3.el6.x86_64                 2/10
Updating : evolution-NetworkManager-3.0.2-3.el6.x86_64 3/10
Updating : evolution-help-3.0.2-3.el6.x86_64            4/10
Updating : empathy-3.0.2-3.el6.x86_64                   5/10
Cleanup  : evolution-NetworkManager-3.0.2-3.el6.x86_64 6/10
Cleanup  : evolution-help-3.0.2-3.el6.x86_64            7/10
Cleanup  : evolution-3.0.2-3.el6.x86_64                 8/10
Cleanup  : empathy-3.0.2-3.el6.x86_64                   9/10
Cleanup  : evolution-data-server-3.0.2-1.el6.x86_64    10/10
```

Listing 4.63: Paketgruppen aktualisieren

4.3.5 Paket suchen

Möchte man den Paketnamen eines Programms wissen, so wird hierzu der Befehl `yum search` benutzt.

```
root@tux>yum search samba
Loaded plugins: downloadonly, refresh-packagekit, security
========== N/S Matched: samba ==========
samba-client.x86_64: Samba client programs
samba-common.i686: Files used by both Samba servers and clients
samba-common.x86_64: Files used by both Samba servers and clients
samba-doc.x86_64: Documentation fort he Samba suite
```

Listing 4.64: Paket suchen

4.3.6 Vorhandene Pakete

Alle vorhandenen Pakete in der Yum Database werden mit `list` angezeigt. Als erstes werden die installierten Pakete angezeigt und danach die Available Packages.

```
root@tux>yum list | less
Loaded plugins: downloadonly, refresh-packagekit, security
```

Listing 4.65: Vorhandene Pakete anzeigen

4.3.7 Installierete Pakete

Installierte Pakete können mit `list installed` angezeigt werden.

```
root@tux>yum list installed | less
Loaded plugins: downloadonly, refresh-packagekit, security
Installed Packages
ConsoleKit.x86_64       0.4.1-3.el6 @anaconda-OracleLinuxServer
ConsoleKit-libs.x86_64 0.4.1-3.ol6 @anaconda-OracleLinuxServer
.
.
```

Listing 4.66: Installierte Pakete

4.3.8 Paket Informationen

Informationen über Pakete können mit `yum info` angezeigt werden.

```
root@tux>yum info firefox
Loaded plugins: downloadonly, refresh-packagekit, security
Installed Packages
Name        : firefox
Arch        : x86_64
Version     : 17.0.6
Release     : 1.0.1.el6_4
Size        : 29M
Repo        : installed
From repo   : ol6_latest
Summary     : Mozilla Firefox Web Browser
URL         : http://www.mozilla.org/projects/firefox/
License     : MPLv1.1 or GPLv2+ or LGPLv2+
Description : Mozilla Firefox is an open-source web browser, designed for
            : standards compliance, performance and portability.
```

Listing 4.67: Paket Informationen

4.3.9 Repository

Mit `yum repolist` werden nur die Repositories ausgegeben, die für das System aktiviert sind.

```
root@tux>yum repolist
Loaded plugins: downloadonly, refresh-packagekit, security
repo_id        repo_name                                  status
ol6_UEK_latest Latest Unbreakable Enterprise Kernel ..      176
ol6_latest     OracleLinux 6Server latest (x86_64)       21.522
```

Listing 4.68: Aktive Repositories

Alle Repositories werden mit `yum repolist all` angezeigt.

```
root@tux>yum repolist all
Loaded plugins: downloadonly, refresh-packagekit, security
repo_id         repo_name                                  status
ol6_UEK_base    Unbreakable Enterprise Kernel for Oracle.. disabled
ol6_UEK_latest  Latest Unbreakable Enterprise Kernel ..    enabled:176
OL6_ga_base     Oracle Linux 6Server GA installation       disabled
ol6_latest      OracleLinux 6Server latest (x86_64)        enabled:21.522
.
.
```

Listing 4.69: Alle Repositories

Im Verzeichnis `/etc/yum.repos.d` gibt es die Datei `public-yum-ol6.repo`. In dieser Datei befinden sich alle Repositories und hier kann man sie auch aktivieren. Ein Update der Datei kann man folgendermaßen machen.

```
root@tux>cd /etc/yum.repos.d

root@tux>mv public-yum-ol6.repo public-yum-ol6.repo.disabled

root@tux>wget http://public-yum.oracle.com/public-yum-ol6.repo
```

Listing 4.70: Update Repositories

4.3.10 Repository hinzufügen

Um ein neues Repository hinzufügen zu können, muss vorher die Release Informationen heruntergeladen und installiert werden.

```
root@tux>wget http://download.fedoraproject.org/pub/epel/6/x86_64/
   epel_release-6-8.noarch.rpm

root@tux>rpm -ivh epel_release-6-8.noarch.rpm

root@tux>yum repolist
```

Listing 4.71: Added Repositories

Erfolgt nach `yum repolist` die Fehlermeldung: `Error: Cannot retrieve metalink for repository: epel. Please verify its path and try again`, so ist die Variable `https_proxy` zu setzten.

```
root@tux>export https_proxy=https://10.18.41.81:80/
```

Listing 4.72: Proxy setzten

4.3.11 Update mit Cron

Sollen die Updates vollautomatisch heruntergeladen und installiert werden, so kann man das am einfachsten mit dem Programm `yum-cron` erledigen. Dieses Paket muss vorher installiert werden.

```
root@tux>yum install yum-cron
```

Listing 4.73: Installieren yum-cron

Nach erfolgreicher Installation befindet sich die Konfigurationsdatei unter `/etc/sysconfig` mit dem Namen `yum-cron`. Nach erfolgter Konfiguration muss der Daemon für `yum-cron` eingerichtet werden.

```
root@tux>service yum-cron start

root@tux>chkonfig yum-cron on
```

Listing 4.74: Enable yum-cron

Nun befindet sic him Verzeichnis `/etc/cron.daily` die Datei `0yum.cron`.

4.4 Apt & Apt-get (Debian)

Seit Debian 8 ist der Nachfolger von `apt-get` das Programm `apt`. `Apt` steht für `Advanced Packing Tool` und ist das Frontend für `Dpkg`, den Debian Package Manager. Auch mit `dpkg` kann man Software installieren, die zwar als Deb-Paket vorliegen, sich aber in keinem Repository befinden.
In der nachfolgenden Tabelle sind die Unterschiede zwischen Apt-get und Apt aufgelistet.

Funktion	***Apt-get***	***Apt***
Paket installieren	apt-get install `Paket`	apt install `Paket`
Paket entfernen	apt-get remove `Paket`	apt remove `Paket`
Paket incl Konfiguration entfernen	apt-ge purge `Paket`	apt purge `Paket`
Quelle aktualisieren	apt-get update	apt update
Pakete aktualisieren (ohne Entfernen oder Neuinstallation)	apt-get upgrade	apt upgrade
Pakete aktualisieren (mit Entfernen und Neuinstallation)	apt-get dist-upgrade	apt full-upgrade
Nicht mehr benötigte Abhängigkeiten entfernen	apt-get autoremove	apt autoremove
Paket suchen	apt-get search `Paket`	apt search `Paket`
Paketinformationen anzeigen	apt-cache show `Paket`	apt show `Paket`
Aktive Paketquellen anzeigen	apt-cache policy	apt policy
Vorhandene und installierte Paketversion anzeigen	apt-cache policy `Paket`	apt policy `Paket`
Paketquellen bearbeiten	-	apt edit-cources
Pakete nach Kriterien auflisten	dpgk --get-selections	apt list
Paketstatus setzen / ändern	echo Paket hold | dpkg --set-selections	apt-mark `Paket`

Tabelle 4.3: Unterschiede Apt & Apt-get

4.4.1 Paket installieren

```
uws@tux>sudo apt install inxi
Reading package lists ... Done
Building dependency tree
Reading state information ... Done
.
.
```

Listing 4.75: Paket installieren

4.4.2 Paket suchen

```
uws@tux>sudo apt search inxi

uws@tux>sudo apt content inxi
```

Listing 4.76: Paket suchen

4.4.3 Paket löschen

```
uws@tux>sudo apt remove <paket-name>

uws@tux>sudo apt purge <paket-name>
```

Listing 4.77: Paket löschen

4.4.4 Paket Informationen

```
uws@tux>sudo apt show inxi
Package: inxi
State: installed
Automatically installed: no
```

Listing 4.78: Paket Informationen

4.5 Pacman (Arch Linux)

Unter Arch Linux wird Software mit Kommandozeilen Werkzeug `pacman` eingespielt. Sollen Pakete aus dem AUR-Repository installiert werden, so nimmt man hierzu den Befehl `yaourt`.

4.5.1 Konfiguration

Für die Konfiguration von `Pacman` gibt es in dem Verzeichnis `/etc` die Datei `pacman.conf`. Eine Farbige Ausgabe der Meldungen erhält man, wenn die Option `color` eingeschaltet wurde.

```
uws@tux>cat /etc/pacman.conf
#RootDir     = /
#DBPath      = /var/lib/pacman/
#CacheDir    = /var/cache/pacman/pkg/
#LogFile     = /var/log/pacman.log
#GPGDir      = /etc/pacman.d/gnupg/
HoldPkg     = pacman glibc
#XferCommand = /usr/bin/curl -C - -f %u > %o
#XferCommand = /usr/bin/wget -passive-ftp -c -o %o %u
#CleanMethod = KeepInstalled
#UseDelta    = 0.7
Architecture = auto
```

Listing 4.79: Auszug pacman.conf

Die Download-Geschwindigkeit lässt sich mit der Option `--limit-rate` festlegen. Diese Option wird dem Parameter `XferCommand` hinzugefügt.
`XferCommand = /usr/bin/wger --passive-ftp --limit-rate=40k -c -o %o %u`

4.5.2 Repositories

Es gibt für Offizielle Arch Linux Pakete sechs verschiedene Repositories, wo die Pakete zugeordnet wurden. Weitere Repositories kann man sich in der Datei `/etc/pacman.conf` definieren, indem man dort einen

Repo Name	*Beschreibung*
Core	Enthält Grundlegende Programme
Extra	Enthält zusätzliche Programme, wie z.B. KDE oder Gnome
Testing	Enthält neuere Versionen, die aber noch nicht getestet worden sind
Community	Enthält Programme, die ursprünglich nur als PKGBUILD im AUR zu Verfügung standen und nach einer Testphase übernommen wurden
Multilib	Enthält 32 Bit Programme für 64 Bit Systeme
Multilib-testing	Wie Multilib, jedoch sind die Programme noch nicht ausreichend getestet worden

Tabelle 4.4: Repository Namen

Repo-Namen und Server spezifiziert. Eine Liste von `inoffizieller` Repositories findet man im englischen Wiki http://wiki.archlinux.org/index.php/Unofficial_user_repositories.

4.5.3 Spiegel-Server

Es gibt verschiedene Spiegel-Server von dem Hauptserver ftp.archlinux.org. Die Liste der Spiegel-Server befindet sich unter `/etc/pacman.d` und hat den Namen `mirrorlist`. Unter https://www.archlinux.de/?page=MirrorStatus kann man sich die Geschwindigkeit der Spiegelserver anzeigen lassen. Eine aktuelle Liste der Spiegel-Server kann man sich unter https://www.archlinux.org/mirrorlist erstellen lassen. Die Antwortzeiten der Spiegel-Server kann man mit `rankmirrors` oder auch mit `pacman-mirrors -g` testen.

```
uws@tux>sudo pacman-mirrors -c Germany  # update Germany Mirror Server
```

Listing 4.80: Spiegelserver

Die `mirrorlist` wird vom System immer wieder mal aktualisiert. Man kann sich auch eine eigene Mirroliste erstellen und diese in der `pacman.conf` in den entsprechenden Repositories Abschnitten einfügen.

```
[core]
Include = /etc/pacman.d/mirrorlist
Include = /etc/pacman.d/myMirrorlist
```

Listing 4.81: Eigene Mirrorliste

4.5.4 Update

Mit dem Befehl `pacman -Syu` kann man eine komplette System Aktualisierung vornehmen. Möchte man aber nur eine Aktualisierungen für die installierten Pakete vornehmen, so kann das mit `checkupdates` geschehen.

```
uws@tux>sudo checkupdates
```

Listing 4.82: Checkupdates

4.5.5 Paket decline

Möchte man ein Paket von der Aktualisierung ausschließen, so kann man das mit `--ignore` machen. Eine ganze Paketgruppe wird mit `--ignoregroup` ausgeschlossen. Soll ein Paket dauerhaft von einer Aktualisierung ausgeschlossen werden, so gibt man in der `/etc/pacman.conf` unter `IgnorePkg` die Pakete an, die nicht aktualisiert werden sollen.

```
uws@tux>sudo pacman -Syu --ignore=<PaketName>   # einzelne Pakete
   ausschliessen

uws@tux>sudo pacman -Syu ..ignoregroup=<PaketName>  # ganze Paket Gruppe
   ausschliessen

uws@tux>grep -i "IgnorePkg" /etc/pacman.conf
IgnorePkg=<PaketName>
```

Listing 4.83: Ignore Paket

4.5.6 Paket Neuinstallieren

Möchte man ein Programm neu installieren, aber es sind noch fragmente auf der Festplatte und die Installation schlägt fehl, so kann man die Installation mit `overwrite` erzwingen.

```
uws@tux>sudo pacman -S <PaketName> --overwrite
```

Listing 4.84: Programm Neuinstallation

4.5.7 Befehle

In der nachfolgenden Tabelle sind einige der wichtigsten Befehle aufgelistet. Die Angabe der Paketnamen muss immer `klein` geschrieben werden.

Befehl	*Beschreibung*
pacman -Syu	Eine komplette Systemaktualisierung wird durchgeführt
pacman -S <paket1> <paket2>	Ein oder auch mehrere Pakete installieren
pacman -Sy	Lokale DB aktualisieren
pacman -Su	Alle installieren Pakete aktualisieren
pacman -Syy	Loakle DB neu aufbauen
pacman -Syuu	Alle installierten Pakete downgraden (z.B. von Testing nach Core/Extra)
pacman -Ss <paket>	Nach Paketen suchen. Es wird nach Name oder Beschreibung gesucht
pacman -Sg	Suche nach Paketgruppen
pacman -Sg <paketgruppe>	Inhalt einer Paketgruppe anzeigen
pacman -Q	Alle installierten Paket incl. der Versionsnummer anzeigen
pacman -Qs <paket>	Nach installierten Paketen suchen
pacman -R	Paket deinstallieren
pacman -Rd <paket>	Pakete deinstallieren, ohne auf Abhängigkeiten zu achten
pacman -Rdd <paket>	Pakete deinstallieren und alle Abhängigkeitsprüfungen abschalten
pacman -Rs <paket>	Paket mit allen Abhängigkeiten deinstallieren
pacman -Rss <paket>	Paket mit allen Abhängigkeiten und deren Abhängigkeiten deinstallieren
pacman -D <paket>	Status eines installierten Programms ändern ohne eine Neuinstallation
pacman -Qi <paket>	Info über ein installiertes Paket anzeigen
pacman -Si <paket>	Info über ein zu installierendes Paket anzeigen
pacman -Sw <paket>	Paket herunterladen und noch nicht installieren
pacman -S testing/<paket>	Ein Paket aus einem bestimmten Repositorie installieren
pacman -U <paket>	Ein lokales Paket installieren
pacman -Runs <paket>	Ein Paket mit allen Anbhängigkeiten deinstallieren
pacman -Qdt	Verwaiste Pakete anzeigen, die nicht mehr von anderen Paketen benötigt werden
pacman -Qet	Pakete anzeigen, die ausdrücklich installiert wurden, aber nicht von anderen benötigt werden
pacman -Scc	Leert den lokalen Speicher /var/cache/pacman/pkg
pacman -Sc	Löscht nicht mehr benötigte oder alte Pakete aus /var/cache/pacman/pkg und aus /var/lib/pacman
pacman -Qi <paket>	Alle installierten Dateien des Paketes anzeigen
pacman -Qm	Pakete anzeigen, die sich in keinem aktivierten Repository sich befinden
pacman -Qu	Anzeige, für welche von den Installierten Paketen Updates gibt
pacman -Qk	Überprüft alle Pakete auf fehlende Dateien
pacman -Fy	Die lokale Datenbank wird aktualisiert
pacman -Fs <paket>	Das Paket suchen, das die Datei enthält
pacman -Fsx <regex>	Suche nach regulären Ausdrücken, sonst wie -Fs
pacman -Fi <paket>	Alle Dateien des Paketes anzeigen
pacman-optimize	Die Pacman DB optimieren

Tabelle 4.5: Pacman Befehle

Kapitel 5

Netzwerk

5.1 Netzwerkzugriffe erlauben / verbieten

Um den Zugriff auf verschiedene Dienste zu erlauben oder zu verbieten, gibt es die Dateien `/etc/hosts.allow` und `/etc/host.deny`. Soll zum Beispiel SSH von allen Systemen erlaubt sein, HTTP nur von einem IP-Bereich und Telnet generell verboten, so sind in den Dateien folgendes einzutragen.

```
uws@tux>cat /etc/hosts.allow
sshd: ALL
http: LOCAL, 192.168.0.0/255.255.255.0

uws@tux>cat /etc/hosts.deny
telnet: ALL
```

Listing 5.1: Dienste erlauben / verbieten

5.2 IP-Adresse Static

5.2.1 Debian

In der Datei `/etc/network/interfaces` folgende Zeilen hinzufügen.

```
iface eth0 inet static
  address 10.81.12.150
  netmask 255.255.255.0
  gateway 10.81.12.1
  dns-search localnet
  dns-nameserver 10.81.12.200
```

Listing 5.2: Interfaces

Die Zeile `iface eth0 inet dhcp` aus der Datei auskommentieren. In diesem Beispiel hat das Interface den Namen `eth0`. Er kann bei ihnen auch anders lauten.

5.3 Datenpakete verfolgen

Mit dem Programm `traceroute` kann man sich die IP-Router Datenpakete zum ziel ermitteln.

```
uws@tux>sudo traceroute seabaer-ag.de
raceroute to seabaer-ag.de (81.169.145.156), 30 hops max, 60 byte packets
 1  fritz.box (192.168.10.105)  8.999 ms  9.491 ms  9.895 ms
 2  * * *
 3  crmfra11-et-27-58.qsc.de (87.234.11.41)  32.285 ms  34.017 ms  34.442 ms
 4  atuin.rzone.de (80.81.192.110)  34.473 ms  38.441 ms  38.932 ms
 5  vl481.fiddlersriddle.as6724.net (81.169.144.121)  51.265 ms  52.706 ms
    52.737 ms
```

```
 6  w9c.rzone.de (81.169.145.156)  45.131 ms  36.733 ms  41.337 ms
```

Listing 5.3: Traceroute

5.4 Wake on Lan (WoL)

Einen ausgeschalteten Computer kann man mittels eines `Magic Pakets` anschalten. Hierzu gibt es den Befehl `wol`.

```
uws@tux>sudo wol -h <broadcast> <macadresse>

uws@tux>sudo wol 172.30.100.255 00:00:00:00:00:00
```

Listing 5.4: WoL

5.5 Netzwerk mit Ip

Der Nachfolger der `net-tools (ifconfig, Route, Arp)` ist das Programm `ip` aus dem Paket `iproute2`. Die `net-tools` werden nicht mehr weiterentwickelt. Das Programm `ip` gibt es schon seit 1999 und bietet auch den Support für IPv6.

5.5.1 Syntax

Die Syntax des Commands `ip` lautet:
`ip [options] object command`
In der Tabelle sind die Objekte aufgeführt, die man verwenden kann.

Objekt	Beschreibung
link oder l	Netzwerk Device
address oder a / addr	Netzwerk Adresse Device
addrlabel oder addrl	Label Konfiguration
neighbour oder n / neigh	ARP oder NDISC cache Eintrag
route oder r	Routing Tabelle
rule oder ru	Regeln
maddress oder m / maddr	Multicast
mroute oder mr	Multicast routing cache
tunnel oder t	Tunnel über IP
xfrm oder x	Framework für IPsec Protokol

Tabelle 5.1: Liste Dialog

Eine Hilfe über die Objekte kann man auch mit `ip object help` sich anzeigen lassen.

```
uws@tux>ip a help
```

Listing 5.5: Hilfe

5.5.2 IP-Adresse anzeigen

Die IP-Adresse kann man sich mit dem Befehl `ifconfig` anzeigen lassen. Diesen Befehl kann man nur als `root` ausführen. Als Nachfolger gibt es den Befehl `ip`, den auch der normale User ausführen kann.

```
uws@tux># Lange form: ip address
uws@tux>ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
   default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

```
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group
    default qlen 1000
.
.

uws@tux># IPv6 = -6 and show only device wlan0
uws@tux>ip -4 a wlan0
```

Listing 5.6: IP-Adresse anzeigen

5.5.3 Interfaces anzeigen

Um sich anzeigen zu lassen, welche Interfaces zu Verfügung stehen, wird der Befehl `ip link show` benutzt.

```
uws@tux>ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
   DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode
   DORMANT group default qlen 1000
    link/ether 8c:70:5a:1b:4b:b0 brd ff:ff:ff:ff:ff:ff
3: eth0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state
    DOWN mode DEFAULT group default qlen 1000
    link/ether e4:11:5b:54:c7:0c brd ff:ff:ff:ff:ff:ff
```

Listing 5.7: Interfaces anzeigen

5.5.4 IP-Adresse hinzufügen

Eine IP-Adresse oder auch eine zweite IP-Adresse kann man einem Interface folgendermßen hinzufügen.

```
uws@tux>sudo ip addr add 10.81.12.200/24 dev eth0
```

Listing 5.8: IP-Adresse hinzufügen

5.6 ZMD Meldung (SLES)

Wenn in der /var/log/messages die Meldung zmd: NetworkManagerModule (Warn): Failed to connect to NetworkManager auftaucht, so besagt diese Meldung, das die Netzwerkschnittstellen nicht mit dem NetworkManager konfiguriert sind. Damit diese Meldung nicht mehr in der /var/log/messages auftaucht, is die Datei /usr/lib/zmd/modules/Novel-Zenworks.Zmd.NetworkManager.dll umzubenennen. Danach ist der zmd Daemon mit /etc/init.d/novel-zmd restart neu zu starten.

```
uws@tux>cd /usr/lib/zmd/modules

uws@tux>sudo mv Novell.Zenworks.Zmd.NetworkManager.dll Novel.Zenworks.Zmd.
   NetworkManaget.dll.ori

uws@tux>sudo /etc/init.d/novell-zmd restart
```

Listing 5.9: ZMD

5.7 Netzlaufwerke

Netzlaufwerke können unter KDE / Gnome mit smb://<user>@<server>:<freigabe> angezeigt werden. CIFS Freigaben sollten immer mit der Option sync gemounted werden, da es sonst zu Datenverlusten kommen könnte, wenn die Netzverbindung zusammen bricht.

5.8 Bonding

Sind in dem Rechner zwei oder mehrere Netzwerkkarten eingebaut, so kann man diese zu einer Virtuellen Netzwerkkarte zusammenfügen (Bonding). Unter SuSE befinden sich die Dateien im Verzeichnis /etc/sysconfig/network und unter Red Hat /etc/sysconfig/networkscripts. Nach der Konfiguration des Bondings muss das Netzwerk einmal neu gestartet werden.

5.8.1 SuSE

Unter SuSE wird in der Datei ifcfg-<id_network> (MAC Adresse oder auch ethx) die Parameter auf BOOTPROTO='none' und STARTMODE='off' abgeändert.

```
uws@tux>egrep -w "BOOTPROTO|STARTMODE" ifcfg-eth0
BOOTPROTO='none'
STARTMODE='off'
```

Listing 5.10: Parameter setzten

Nun kann die Datei ifcfg-bond0 mit folgendem Inhalt erstellt werden.

```
uws@tux>cat ifcfg-bond0
BOOTPROTO='static'
BROADCAST='192.168.1.255'
IPADDR='192.168.1.10'
NETMASK='255.255.255.0'
NETWORK='192.168.1.1' # Gateway
REMOTE_IPADDR=
STARTMODE='onboot'
BONDING_MASTER='yes'
BONDING_MODULE_OPTS='mode=active-backup miimon=100'
BONDING_SLAVE0='eth0'
BONDING_SLAVE1='bus-pci-0000:05:03.1' # PCI Kennung
BONDING_SLAVE2='...'
```

Listing 5.11: ifcfg-bond0

Folgende Bonding Module Opts gibt es:

Mode	*Beschreibung*
ballance-rr oder 0	Round-robin Regel: Die Pakete werden sequentiell vom ersten verfügbaren Slave bis zum letzten übertragen. Dieser Modus bietet Lastverteilung und Fehlertoleranz
active-backup oder 1	Nur ein Slave im Bond ist aktiv. Fällt ein Salve aus, wird der andere Salve aktiviert
balance-xor oder 2	Übertragene Pakete basierend auf dem Ergebnis einer XOR Operation der MAC Adresse des Senders und Empfängers
broadcast oder 3	Über alle Slaves werden die Pakete übertragen. Dieser Modus bietet Fehlertoleranz
802.3ad oder 4	Das Bonding wird nach dem IEEE Standard 802.3ad - Link Aggregation Control Protocol erstellt. Der Switch muss dieses unterstützen und auch konfiguriert werden
balance-tlb oder 5	Adaptive transmit Load balancing -der Datenverkehr wird durch einen Salve erledigt. Falls der Empfang fehl schlägt, Ã¼bernimmt ein anderer Slave
balance-alb oder 6	Adaptive Load balancing - Beinhaltet balance-tlb und zusätzlich Lastverteilung für IPV4-Datenverkehr

Tabelle 5.2: Bonding Mode

5.8.2 Red Hat

```
uws@tux>cat ifcfg-eth0
DEVICE=eth0
MASTER=bond0
SLAVE=yes
ONBOOT=yes
BOOTPROTO=none
TYPE=Ethernet
NM_CONTROLLED=no
USERCTL=no
HWADDR=00:1c:c4:E0:75:66 # MAC Addresse
NAME="System eth0"
```

Listing 5.12: ifcfg-eth0

```
uws@tux>cat ifcfg-bond0
DEVICE=bond0
BONDING_OPTS="mode=1 updelay=0 miimon=100 downdelay=0"
TYPE=Bond
BONDING_MASTER=yes
BOOTPROTO=none
IPADDR=192.168.1.10
NETMASK=255.255.255.0
BROADCAST=192.168.1.255
PREFIX=24
GATEWAY=192.168.1.1
DEFROUTE=yes
DNS1=192.168.1.110 # optional
DNS2=192.168.1.111 # optional
NM_CONTROLLED=no
USERCTL=no
IPV4_FAILURE_FATAL=yes
IPV6INIT=no
NAME="Bond connection 1"
UUID=c819e8a8-4852-4266-9q0b-f06060e8aef7 # optional
ONBOOT=yes
```

Listing 5.13: ifcfg-bond0

```
uws@tux>cat /etc/modprobe.d/bonding.conf
alias bond0 bonding
# wenn nicht in ifcfg_bond0 gesetzt worden ist,
# hier die options eintragen.
# Gilt nicht fuer Red Hat Systeme
options bond0 mode=0 miimon=100 use_carrier=1

uws@tux>modprobe bonding

uws@tux>service network restart

uws@tux>cat /proc/net/bonding/bond0
```

Listing 5.14: bonding.conf

Weitere Informationen über das Bonding gibt es unter `/usr/share/doc/kernel_docx.x.x/Documentation/networking/bonding.txt`.

5.9 DNS

5.9.1 DNS Server

In der Datei `/etc/resolf.conf` werden die DNS Server eingtragen.

```
uws@tux>cat /etc/resolf.conf
nameserver 192.168.1.110
nameserver 8.8.8.8
```

Listing 5.15: resolf.conf

Hier eine kleine Liste von freien Nameservern.

Server	*Beschreibung*
144.76.34.109	opennameserver.org (Germany)
144.76.133.38	opennameserver.org (Germany)
46.101.154.113	Free Public DNS
78.47.206.86	powerdns.com
141.16.33.174	ZyWall DNS
37.235.1.174	FreeDNS
37.235.1.177	FreeDNS
64.6.64.6	Verisign
64.6.65.6	Verisign

Tabelle 5.3: Nameserver

5.9.2 Namensauflösung

Informationen über einen Rechner kann man mit `dig` abfragen. Dieser Rechner / Server kann auch im Internet stehen.

```
uws@tux>dig <domain_server> <server_name> [any, A, MX, SIG]

uws@tux>dig www.seabaer-ag.de
; <<>> DiG 9.9.4-rpz2.13269.14-P2 <<>> www.seabaer-ag.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode; QUERY, status: NOERROR, id: 14122
;; flags; qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY; 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4000
;; QUESTION SECTION:
;www.seabaer-ag.de IN A

;;ANSWER SECTION:
www.seabaer-ag.de. 7200 IN CNAME seabaer-ag.de.
seabaer-ag.de      7200 IN A     81.169.145.156

;; Query time: 30 msec
;; SERVER: 192.168.1.110#53(192.168.1.110)
;; WHEN: Tue Jun 23 10:33:21 CEST 2014
;; MSG SIZE rcvd: 76
```

Listing 5.16: Namen auflösen

Mit dem Befehl `nslookup` kann man auch Informationen über einen Rechner erhalten, aber das Programm wird nicht mehr weiterentwickelt und wurde durch `dig` abgelöst.

```
uws@tux>nslookup tux.seabaer-ag.de
Server:  192.168.1.110
Address: 192.168.1.110#53

Name: tux.seabaer-ag.de
Address: 192.168.1.35
```

Listing 5.17: nslookup

Ebenso kann man mit dem Befehl `host` Informationen erhalten.

```
uws@tux>host www.seabaer-ag.de
www.seabaer-ag.de is an alias for seabaer-ag.de.
seabaer-ag.de has address 81.169.145.156
seabaer-ag.de has IPv6 address 2a01:238:20a:202:1090::144
seabaer-ag.de mail is handled by 5 smtp.rzone.de.
```

Listing 5.18: host

5.10 Mount Webdav

5.10.1 Arch Linux

Für das Mounten eines Webdav Shares muss als erstes das Paket `davfs2` installiert werden.

```
uws@tux>sudo pacman -Sy davfs2
```

Listing 5.19: Install davfs2

Im Home Verzeichnis des Benutzers wird das Verzeichnis `.davfs2` angelegt und in diesem die Datei `secrets`.

```
uws@tux>mkdir .davfs2
uws@tux>cd .davfs2
uws@tux>cat secrets
https://webdav.mc.gmx.net "UserName" "Passwort"
```

Listing 5.20: Datei secrets

Nun kann in der Datei `/etc/fstab` das zu verbindende Share eingetragen werden.

```
uws@tux>cat /etc/fstab | grep -i "webdav"
https://webdav.mc.gmx.net /cloud/gmx davfs users,noauto,uid=uws,file_mode
   =600,dir_mode=700 0 1
```

Listing 5.21: Datei fstab

Zum Abschluß den User noch in der Gruppe `network` aufnehmen, damit er das Recht hat den Share einzubinden.

```
uws@tux>sudo usermod -aG network uws

uws@tux>mount /cloud/gmx
```

Listing 5.22: Guppe und Mount

5.10.2 OpenSuSE

Unter OpenSuSE wird das paket `wdfs` installiert.

```
uws@tux>sudo zypper install wdfs
```

Listing 5.23: Install wdfs

Nun kann das Webdav Share verbunden werden.

```
uws@tux>wdfs https://webdav.mc.gmx.net /cloud/gmx -o username=<WebdavUser> -
   o password=<Passwort>
```

Listing 5.24: Mount Share

5.10.3 Mount aufheben

Soll die Verbindung zu dem Webdav Share wieder beendet werden, so wird das mit `fusermount` gemacht.

```
uws@tux>fusermount -u /cloud/gmx
```

Listing 5.25: fusermount

5.11 Mount Cifs

Ab der OpenSuSE Version 13.1 wurde der `Standard-Authentifizierungsalgorithmus` auf `ntlmssp` geändert. Gibt es bei einem Mount mit einem Cifs-Share die Fehlermeldung: ttfamily mount erro(95): Operation not supportet, so trägt man in der `/etc/fstab` den Parameter `sec=ntlmv2` ein. Sollte es auch mit diesem Parameter nicht funktionieren, so trägt man `sec=ntlm` ein.

5.12 DHCP

5.12.1 Server

Die Konfiguration von DHCP wird in dem Verzeichnis `/var/lib/dhcp` abgelegt. Welche IPAdressen schon vergeben worden sind, kann man sich mit dem nachfolgenden Beispiel sich anzeigen lassen.

```
uws@tux>sudo egrep "lease|hostname|hardware|\}" /var/lib/dhcp/dhcp.leases
```

Listing 5.26: Anzeige IPAdressen

5.12.2 Client

Wie auch bei dem DHCP Server, liegen die Informationen über DHCP in dem Verzeichnis `/var/lib/dhcp`. Informationen über das Netz kann man sich folgendermaßen anzeigen lassen.

```
uws@tux>sudo arp -a
rbgo110.seabaernet (192.168.30.10) at 34:31:c4:66:e0:84 [ether] on br0
fsgo002.seabaernet (192.168.30.20) at 00:08:9b:c2:ef:8d [ether] on br0

uws@tux>cat /proc/net/arp
IP address HW type Flags HW address Mask Device
192.168.30.10 0x1 0x2 34:31:c4:66:e0:84 * br0
192.168.30.20 0x1 0x2 00:08:9b:c2:ef:8d * br0

uws@tux>ip neighbor show
192.168.30.10 dev br0 lladdr 34:31:c4:66:e0:84 REACHABLE
192.168.30.20 dev br0 lladdr 00:08:9b:c2:ef:8d STALE
```

Listing 5.27: Info Netzwerk

5.13 Ping

Ausgeben von erreichbaren Geräten.

```
root@tux>for i in $(seq 10 30); do ping -c 1 "192.168.30.$i" | grep "bytes
    from"; done
64 bytes from 192.168.30.10: icmp_seq=1 ttl=64 time=39.8 ms
64 bytes from 192.168.30.20: icmp_seq=1 ttl=255 time=1.50 ms
```

Listing 5.28: Ping

5.14 Private IP-Adressen

Für den privaten Bereich sind Adressbereiche aus dem öffentlichen Adressraum ausgespart. Diese Adressbereiche werden nicht in das Internet durchgereicht.
Folgende Adressbereicht (IPv4) gibt es:

Adresse	*Beschreibung*
10.0.0.0 bis 10.255.255.255	Class A Netz
172.16.0.0 bis 172.31.255.255	Class B Netz
192.168.0.0 bis 192.168.255.255	Class C Netz

Tabelle 5.4: Private Adressen

5.15 USB-Wlan Stick

Möchte man auf einem Raspberry PI einen USB-Wlan Stick in Betrieb nehmen, so sollte der Stick vom Betriebssystem erkannt und die Treiber geladen worden sein. Danach kann man das Interface und die Credentials der Wlan Verbindung konfigurieren. Getestet unter Raspian.

5.15.1 Check USB-Stick

Als erstes schauen wir nach, ob der USB-WLAN Stick erkannt wurde.

```
root@tux>lsusb

root@tux>dmesg | grep -i -E "sb|usb 1|usbcore|wlan"

root@tux>journalctl -b -p err
```

Listing 5.29: Check USB-Stick

5.15.2 Config Interface

Nun Konfigurieren wir das Interface für das Wlan.

```
uws@tux>sudo cat /etc/network/interfaces

auto wlan0
allow-hotplug wlan0
iface wlan0 inet manual
 wpa-ap-scan 1
 wpa-scan-ssid 1
 wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

Listing 5.30: Config Interface

5.15.3 Config Verbindung

In der Config-Datei /etc/wpa_supplicant/wpa_supplicant.conf steht das Passwort im Klartext drin. Man kann es auch verschlüsselt ablegen. Um den Schlüssel zu erstellen, gibt es das Programm wpa_passphrase.

```
uws@tux>sudo wpa_passphrase "<SSID>" "<passwd>" >> /etc/wpa_supplicant/
   wpa_supplicant.conf
```

Listing 5.31: Passwort verschlüsseln

```
uws@tux>sudo cat /etc/wpa_supplicant/wpa_supplicant.conf
country=DE
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
 SSID="Seabaer.net"
 key_mgmi=WPA-PSK
 proto=WPA2
 pairwise=CCMP
```

```
 group=CCMP
 psk="<passwort>"
}
```

Listing 5.32: wpa_supplicant.conf

5.15.4 Verbindung herstellen

Nun kann die Verbindung von Hand aufgebaut werden.

```
uws@tux>wpa_supplicant -iwlan0 -Dnl80211 -c/etc/wpa_supplicant/
   wpa_supplicant.conf

root@tux># other connection
root@tux>ifup wlan0
```

Listing 5.33: Verbindung herstellen

5.15.5 Verfügbare Netze anzeigen

Die zu Verfügung stehenden Netze kann man sich mit `iwlist` anzeigen lassen.

```
uws@tux>iwlist wlan0 scan | grep ESSID
```

Listing 5.34: Verfügbare Netze

5.15.6 Anzeige Network Informationen

```
uws@tux>sudo iwconfig wlan0

uws@tux>sudo nmcli connection show
uws@tux>sudo nmcli connection show "Seabaer.net"
uws@tux>sudo nmcli dev show wlan0
uws@tux>sudo nmcli def wifi
uws@tux>sudo nmcli -f GENERAL,WIFI-PROPERTIES dev show wlan0

uws@tux>sudo watch -n 1 cat /proc/net/wireless
```

Listing 5.35: Netzwerk Info

5.15.7 Weitere Monitoring Tools

```
uws@tux>sudo apt install wavemon # Debian

uws@tux>sudo zypper install wavemon # SuSE

uws@tux>sudo yum install wavemon # RedHat

uws@tux>sudo pacman -S wavemon # Arch Linux

uws@tux># install nmcli for debian
uws@tux>sudo apt install network-manager-gnome

uws@tux>sudo watch -n 1 cat /proc/net/wireless
```

Listing 5.36: Weitere Tools

5.16 Anzeige WAN-IP

Die zugewiesene WAN-IP des Internet Providers kann man sich folgendermaßen anzeigen lassen.

```
uws@tux>curl ipinfo.io/ip
```

Listing 5.37: Anzeige WAN-IP

5.17 WLAN beim Start verbinden

Soll eine WLAN Verbindung nach dem Starten des Systems aufgebaut werden, so kann das mit dem Network Manager gemacht werden.Die Konfigurationsdateien befinden sich in dem Verzeichnis `/etc/NetworkManager`. Unter `system-connection` sind die Einstellungen fÃ¼r die Netzwerkverbindungen. Um eine WLAN-Verbindung bei dem Booten zu aktivieren, muss im `Network Manager` für die entsprechenden Verbindungen die Einstellungen aufgerufen werden. Unter dem Reiter `Allgemeine Einstellungen` muss `Alle Benutzer dürfen sich mit diesem Netzwerk verbinden` aktiviert sein. In dem Reiter `WI-FI-Sicherheit` muss das Kennwort mit der Option `Passwort speichern und alle Benutzer zur Verfügung stellen (unverschlüßselt)` konfiguriert sein. Das Passwort wird dann in den entsprechenden Dateien unter `/etc/NetworkManager/system-connection` eingetragen. Nur der User `root` hat Zugriff auf diese Dateien.

5.18 Netzwerk Check

5.18.1 Monitoring

Für das Anzeigen der Netzwerk Auslastung gibt es das Programm `iptraf`, das man nachträglich installieren muss.

5.18.2 Performance

Mit dem Programm `iperf` kann man die Netzwerk Performance messen. Auf einem Rechner wird das Programm als Server gestartet und von dem anderen Client aus wird dann die Performance gemessen. Ebenso gibt es das Programm `netperf` für die Performance Messung. Beide Programme muss man gegebenen falls nach installieren.
Als erstes starten wir den Server.

```
uws@tux>iperf -s
------------------------------------------------------------
Server listing on TCP port 5001
TCP window size: 85.3 kByte (default)
------------------------------------------------------------
```

Listing 5.38: Server start

Von den anderen Client starten wir die Performance Messung.

```
uws@tux1>iperf -c tux
------------------------------------------------------------
Client connection to tux, TCP port 5001
TCP window size: 16.0 kByte (default)
------------------------------------------------------------
```

Listing 5.39: Client start

5.19 Netzwerkzugriffe erlauben / verbieten

Um den Zugriff auf verschiedene Dienste zu erlauben oder zu verbieten, gibt es die Dateien `/etc/hosts.allow` und `/etc/host.deny`. Soll zum Beispiel SSH von allen Systemen erlaubt sein, HTTP nur von einem IP-Bereich und Telnet generell verboten, so sind in den Dateien folgendes einzutragen.

```
uws@tux>cat /etc/hosts.allow
sshd: ALL
http: LOCAL, 192.168.0.0/255.255.255.0

uws@tux>cat /etc/hosts.deny
telnet: ALL
```

Listing 5.40: Dienste erlauben / verbieten

Kapitel 6

Archive

6.1 Zip & Unzip

Mit dem Programm `zip` kann man Dateien in einem Archive zusammen packen. Das Programm vereint die Funktionen der Programme `gzip` und `tar` miteinander, ist aber kompatibel zu den anderen Packern (Winzip, Winrar,...).
Die Syntax des Programms ist: `zip [option] zipfile dateien/verzeichnis`

6.1.1 Archiv erstellen

Mit der Option `-v` werden bei dem packen Informationen ausgegeben.

```
uws@tux>zip -v MyArchiv.zip *.txt
 adding: banner.txt   (in=730) (out=186) (deflated 75%)
 adding: login.txt    (in=159) (out=90) (deflated 43%)
 .
 .
total bytes=30575, compressed=10693 -> 65% savings
```

Listing 6.1: Create Archiv

Wird die Option `-m` mit angegeben, so werden nach den erstellen des Archives die Original Dateien gelöscht. Ist das Verzeichnis anschließend leer, so wird das Verzeichnis auch gelöscht.
Möchte man ein ganzes Verzeichnis und deren Unterverzeichnisse in einem Archiv packen, so gibt man die Option `-r` mit an.

```
uws@tux>zip -r -v MyArchiv.zip /daten
```

Listing 6.2: Create Archiv rekursiv

6.1.2 Remove Files

Dateien aus einem Zip-Archiv entfernen, wird mit der Option `-d` gemacht.

```
uws@tux>zip -d MyArchiv.zip banner.txt
```

Listing 6.3: Delete File

6.1.3 Update Files

Um Dateien in einem Zip-Archiv zu aktualisieren, kann mit der Option `-u` gemacht werden.

```
uws@tux>zip -u MyArchiv.zip config.txt
```

Listing 6.4: Update Files

6.1.4 Exclude Files

Um Dateien auszuschließen, gibt man die Option -x an.

```
uws@tux>zip -x MyArchiv.zip link.txt
```

Listing 6.5: Exclude Files

6.1.5 Extract Files

```
uws@tux>unzip MyArchiv.zip
```

Listing 6.6: Extract alle Files

6.2 Gzip & Gunzip

Das Programm gzip verkleinert eine Datei und hängt an diese Datei die Endung .gz an. Das Programm kann keine Archive erstellen wie zip. Möchte man ein Archiv erstellen, so muss das mit dem Aufruf tar geschehen. Mit der Option -v werden bei dem packen Informationen ausgegeben.

```
uws@tux>lst
MeineDoku.txt
uws@tux>gzip MeineDoku.txt
uws@tux>ls
MeineDoku.txt.gz
```

Listing 6.7: Datei packen

Möchte man eine andere Dateiendung haben, so wird das mit der Option -s (Suffix) gemacht.

```
uws@tux>gzip -s .gzip MeineDoku.txt
uws@tux>ls
MeineDoku.txt.gzip
```

Listing 6.8: Andere Dateiendung

Der Kompressionsfaktor wird mit den Werten -1 bis -9 eingestellt, wobei -1 schneller komprimiert und -9 langsamer, aber dafür mit besserer Kompressionsrate. Als Standardfaktor ist -6 eingestellt.

```
uws@tux>gzip -3 .gzip MeineDoku.txt
uws@tux>export GZIP="-3"
```

Listing 6.9: Kompressionsfaktor

Komprimierte Textdateien braucht man nicht erst auspacken, um sie zu lesen. Hierfür gibt es zless und zcat.

```
uws@tux>zless MeineDoku.txt.gz
#
# Hier geht es los
#
```

Listing 6.10: zless

Um Komprimierte Dateien zu entpacken, kann man gzip -d angeben oder die Dateien werden mit gunzip entpackt. Ist die Zieldatei schon vorhanden, so fragt gzip nach, ob die vorhandene Datei ersetzt werden soll. Mit der Option -f (force) wir die vorhandene Zieldatei ohne Nachfrage überschrieben.

```
uws@tux>gunzip MeineDoku.txt.gz

uws@tux>gzip -d MeineDoku.txt.gz
```

Listing 6.11: Gunzip

6.3 Bzip2 & Bunzip2

Das Programm `Bzip2` komprimiert die Dateien besser als `Gzip`. Wie für Gzip gibt es die gleichen Optionen für Bzip2. Bzip2 kann zusätzlich das Original behalten und legt eine neue komprimierte Datei an. Dieses Verhalten erreicht man mit der Option `-k`. Der Kompressionsfaktor wird mit `export BZIP2="-3"` eingestellt.

```
uws@tux>bzip2 MeineDoku.txt
uws@tux>ls
MeineDoku.txt.bz2

uws@tux>bzip2 -k MeineDoku.txt
uws@tux>ls
MeineDoku.txt
MeineDoku.txt.bz2
```

Listing 6.12: Datei packen

Um Dateien zu entpacken, gibt es das Programm `bunzip2`. Anders als `gunzip`, fragt bunzip2 nicht nach, wenn die Zieldatei vorhanden ist, sondern bricht die Aktion ab. Mit dem Schalter `-f` läst sich dieses Verhalten abschalten.

```
uws@tux>bunzip MeineDoku.txt.bz2
```

Listing 6.13: Datei Entpacken

6.4 Tar

Mit dem Befehl `tar (tape archiver)` werden Dateien in einem Container zusammengefasst. Diesen Container kann man anschließend packen oder dieses auch gleich mit `tar` erledigen.

```
uws@tux># Ein ganzen Verzeichnis packen
uws@tux>tar -cvf container.tar Documents/

uws@tux> Packen und komprimieren
uws@tux>tar -cvzf container.tar.gz Documents/

uws@tux># Container entpacken
uws@tux>tar -xvzf container.tar.gz
```

Listing 6.14: Tar

Parameter	*Beschreibung*
c	Erstellen (create)
v	Im Hintergrund arbeiten
f	Dateiname
r	Datei zu einem Archiv hinzufügen
z	Archiv wird mit Gip gepackt / entpackt
j	Archiv wird mit Bzip2 gepackt / entpackt
x	Entpackt das Archiv
t	Inhalt des Archives anzeigen
J	Archiv wird mit XY gepackt / entpackt

Tabelle 6.1: Tar Parameter (Auswahl)

Soll mit `tar` ein Backup des Systems erfolgen, kann es nützlich sein, Verzeichnisse von dem Backup auszunehmen. Hierfür gibt es die Option `--exclude`. Möchte man `tar` mitteilen, das die Sicherung per `ssh` auf einem anderen Rechner abgelegt werden soll, so gibt es hierfür die Option `--rsh-command`.
In dem nachfolgenden Beispiel wird das Backup auf dem Rechner San im Verzeichnis /backup abgelegt. Die Datei hat dann den Namen FullBackup_2010_07_30.tar. Ab dem Wurzelverzeichnis werden alle Dateien, außer die Dateien im Verzeichnis /proc in dem Archiv abgelegt.

```
uws@tux>tar -cvf user@san:/backup/FullBackup_$(daten '+%Y_%m_%d').tar --rsh-
    command=/usr/bin/ssh --exclude=/proc /

uws@tux>tar -tjf /daten/backup/FullBackup.tar.bz2 | less
```

Listing 6.15: Beispiel

6.4.1 Inkrementelles Backup - Dateien suchen / Auspacken

In dem vorherigen Beispiel werden die Dateien inkrementel gesichert. Mit dem nachfolgenden Befehlen kann man nach Dateien suchen und auch Dateien wieder zurückspielen.

```
uws@tux>backuppath=/home/uws/backup
uws@tux>for archiv in ${backuppath}/backup-*.tgz; do tar -tzf $archiv -C /
>| gep bild145.jpg
>done

uws@tux>for archiv in ${backuppath}/backup-*.tgz; do
>tar -xzf $archiv -C /daten/bilder/bild145.jpg
>done
```

Listing 6.16: Dateien suchen / Auspacken

Kapitel 7

Rsync & Rsnapshot

7.1 Rsync - Verzeichnisse abgleichen

7.1.1 Parameter

Hier eine Auswahl an Parameter für `rsync`.

Parameter	*Beschreibung*
a	Archiv Modus, Abkürzung für -rlptgoD
v	Ausführliche Ausführung
n	Testlauf
b	Löscht die Dateien auf dem Ziel nicht, sondern hängt den Suffix ~ an die Dateien an.
z	Dateien auf dem Ziel werden komprimiert
u	Überschreibt keine Datei auf dem Ziel, wenn der Zeitstempel neuer ist als auf der Quelle. Hierbei müssen beide die gleiche Systemzeit haben
--delete	Löscht Dateien auf dem Ziel, wenn sie nicht mehr auf der Quelle vorhanden sind."
--suffix	Definiert einen neuen Suffix
--stats	Noch mehr Informationen bei der Ausführung
--partial	Rsync speichert die übertragenen Teilstücke als Hardlink ab und Speichert nach erfolgreichem Übertragen die Datei
--progress	Ausgabe des Fortschritt Status
--backup-dir	Sichert die zu löschenden Dateien auf dem Ziel in dem angegebenen Verzeichnis
--bwlimit	Angabe der Bandbreite in Kilobit/Sekunde

Tabelle 7.1: Rsync Parameter (Auswahl)

7.1.2 Verzeichnis Synchronisieren

Um ein Quell Verzeichnis mit einem Zielverzeichnis abzugleichen, gibt es den Befehl `rsync`.In dem nachfolgenden Beispiel werden die Dateien mit der Option -a rekursiv kopiert und mit der Option -z komprimiert. Die Bandbreite wird auf 30 Kilobit/Sekunde beschränkt. Wird bei der Angabe des Source Verzeichnisses kein Slash angehangen, so wird in dem Zielverzeichnis das Source Verzeichnis angelegt. In dem unten aufgeführten Beispiel wird in dem Verzeichnis /home/uws das Quell Verzeichnis uws angelegt.

```
uws@tux>rsync --stats -avz --bwlimit=30 /home/uws backup:/home/uws
```

Listing 7.1: Bandbreite

Auf dem Ziel werden die Dateien gelöscht, wenn sie nicht mehr auf der Quelle vorhanden sind.

```
uws@tux>rsync -av --delete /home/uws uws@baer.local.net:/daten/backup
```

Listing 7.2: Dateien löschen

Auf dem Ziel werden die Daten nicht gelöscht, sondern mit einem Suffix versehen.

```
uws@tux>rsync -avb --suffix=.bak --delete /home/uws uws@local.net:/daten/
    backup
```

Listing 7.3: Suffix

Die zu löschenden Dateien werden auf dem Ziel in einem angegebenen Verzeichnis gesichert. Diese Angabe ist entweder relativ zum Home-Verzeichnis (--backup-dir=old/) schreibt nach (~/old) oder eine absolute Pfadangabe.

```
uws@tux>rsync -av --delete --backup-dir=/daten/backup/old /home/uws uws@baer
    .local.net:/daten/backup
```

Listing 7.4: Backup Directory

Verzeichnisse und Dateien, die nicht mit synchronisiert werden sollen, werden mit `--exclude` angegeben.

```
uws@tux>rsync -av --exclude ".ssh/" --exclude ".*" /home/uws uws@baer.local.
    net:/daten/backup
```

Listing 7.5: Exclude

7.1.3 Rsync in Scripten

Möchte man `rsync` in einem Script einsetzen und ein Vollautomatisches Backup erstellen, so braucht man hierzu ein `SSH-Schlüsselpaar`, da man sonst das Kennwort eingeben muss.
Das SSH-Schlüsselpaar wird mit dem Befehl `ssh-keygen -t dsa` und der Angabe einer leeren Passphrase erstellt. Den öffentlichen Schlüssel (`/.ssh/id_dsa.pub`) muss auf dem entfernten Rechner in die Datei `~/.ssh/authorized-keys` eingetragen werden. Hierzu gibt es den Befehl `ssh-copy-id -i <remote.host> 32 <user>@<remote.host>`, der dieses erledigt. In der Datei `authorized-keys` können sich mehrere Schlüssel befinden.
Da wir einen Key ohne Passphrase erstellt haben, gibt es ein Sicherheitsrisiko. Um dieses Risiko zu minimieren, kann man für bestimmte Keys nur einzelne Kommandos erlauben. Hierbei wird in der Datei `/.ssh/autohrized_keys` am Anfang des Keys (`ss-dss ..`) der Befehl mit den dazugehörenden Optionen in der Form `command="rsync Optionen"` eingetragen. Das alles wird in einer Zeile vorgenommen.

```
uws@tux>ssh-keygen -t dsa}
Generating public/private dsa key pair.
Enter the File in which to save the key (~/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ~/.ssh/id_dsa.
Your public key has been saved in ~/.ssh/id_dsa.pub.
The key fingerprint is:
aa:cc:dd:ee:gg:kk:pp:zz:ok:12:r5:f5:g7:3d:t9:g5 uws@tux
uws@tux>ssh-copy-id -i ~/.ssh/id-dsa.pub baer 32 uws@baer
29
Password:
Now try logging into the machine, with "ssh baer", and check in:
     .ssh/authorized_keys
to make sure we haven't added extra keys that you weren't expecting.
```

Listing 7.6: SSH Key erstellen

7.1.4 Rsync-Daemon

Rsync kann auch als Daemon `rsyncd` laufen. Der Standard Port für rsync ist 873, der in der Datei `/etc/services` steht. Der `rsync Daemon` wird auf modernen Systemen mit `xinetd` gestartet. Im Verzeichnis `/etc/xinetd` befindet sich die Datei `rsync`. In dieser Datei kann man die Anzahl der Verbindungen pro IP-Adresse einschränken. Hierfür ist die Option `per_source` zuständig.
Für die Konfiguration des rsync Daemons ist die Datei `/etc/rsync.conf` zuständig. In dieser Datei sind alle Freigaben aufgelistet.

```
uws@tux>cat /etc/rsync.conf
#Global OPTIONS

motd file = /etc/motd.rsync
uid = nobody
gid = nogroup
max connections = 50
socket options = SO_KEEPALIVE
timeout = 1200
log file = /var/log/rsync.log
transfer logging = true
hosts deny = *.local.org 192.168.10.13 210.135.22.66/24

Module Options
[ftp]
  comment = public archive
  path = /daten/ftp
  use chroot = yes
  max connections = 10
[privat_zone]
  comment = Meine Daten
  path = /home/uws
  secrets file = /etc/rsyncd.uws.secrets
  auth users = uws
  uid = uws
```

Listing 7.7: rsync.conf

Parameter	*Beschreibung*
motd file	Begrüßungstext des Servers
uid	User ID-Nr. nach dem Verbindungsaufbau einen Kindprocess starten
gid	Group ID-Nr. nach dem Verbindungsaufbau einen Kindprocess starten
max connections	Anzahl der Clients, die eine Verbindung aufbauen dürfen
host deny	IP-Adressen oder ganze Bereiche den Verbindungsaufbau zu verbieten
secrets file	Passwort Datei, wo der Name und das Passwort drin stehen z.B. uws:<passwd>

Tabelle 7.2: Parameter rsync.conf

7.2 Rsnapshot

Mit dem Tool `Rsnapshot` werden die Daten in verschiedenen Versionen abgespeichert. Standardmäßig geschieht das im Verzeichnis `/.snapshots`. Für die Konfiguration wird die Datei `/etc/rsnapshot.conf` benutzt. In der `crontab` werden nun zwei Einträge angelegt, einen Aufruf rsnapshot mit der Option `hourly`, die andere mit der Option `daily`. So entstehen im Snapshotverzeichnis verschiedene Unterverzeichnisse `daily.0, daily.1, ...` und `hourly.0, hourly.1, ....`

7.2.1 Konfiguration

Das Snapshotverzeichnis wird in der Datei `/etc/rsnapshot.conf` festgelegt. Bei der Angabe des Verzeichnisses ist darauf zu achten, das als Abschluß ein Slash (/) gemacht wird. In dieser Datei dürfen keine Leerzeichen gemacht werden, nur Tabs.

7.2.2 Sicherungsintervall

Bei der Angabe des Snapshot Intervalls handelt es sich um die Anzahl der Snapshots, die erstellt werden. Bei dem unten ausgelisteten Beispiel werden 6 Stündliche Snapshots erstellt (hourly.1 bis hourly.6), 7 Tägliche Snapshots (daily.1 bis daily.7).

```
Intervall   hourly    6
Intervall   daily     7
Intervall   weekly    4
Intervall   monthly   6
```

Listing 7.8: Snapshot Intervall

7.2.3 Sicherungsverzeichnisse

Die Angabe nach dem Quellverzeichnisses wird für das Snapshotverzeichnis benutzt. In dem unten aufgeührten Beispiel wird das localhost bei einem stündlichen Snapshot an /.snapshot/hourly.1 angehangen.

```
backup      /home/      localhost/
backup      /etc/       localhost/
backup      /usr/local  localhost/
```

Listing 7.9: Sicherungsverzeichnis

7.2.4 Include / Exclude

Möchte man Verzeichnisse aus dem Snapshot ausklammern, so wird eine Datei erstellt, in dem alle Verzeichnisse aufgelistet sind, die nicht gesichert werden sollen. Diese Datei wird dann bei `exclude_file` angegeben.

```
uws@tux>grep exclude_file /etc/rsnapshot.conf
# The include_file and exclude_file parameters. If enabled, simply get
#exclude_file /path/to/exclude/file_name

uws@tux>cat exclude
# exclude path
/home/uws/Downloads
/home/uws/tmp
```

Listing 7.10: Exclude

7.2.5 Unterschiede zwischen Snapshots

Mit der Option `diff` kann man sich die Unterschieden zwischen zwei Snapshots sich anzeigen lassen.

```
uws@tux>rsnapshot diff daily.0 daily.1
```

Listing 7.11: Diff

Kapitel 8

System

8.1 Prozesse

8.1.1 Prozesse auflisten mit ps

Prozesse werden mit dem Befehl `ps` aufgelistet. Ruft man den Befehl mit dem Parameter `x` auf, so werden nur die eigenen Prozesse ausgelistet.

```
uws@tux>ps x
  PID TTY    STAT  TIME  COMMAND
 6466 ?      S     0:00  sshd: uws@pts/0
 6467 pts/0  Ss    0:00  -bash
11882 pts/0  R+    0:00  ps x
```

Listing 8.1: Eigene Prozesse

In der ersten Tabelle sind einige Parameter aufgelistet und in der zweiten Tabelle der Status der Prozesse.

Parameter	*Beschreibung*
a	Auflistung der Prozesse aller Benutzer
l	Lange Ausgabe mit zusätzlichen Informationen
w	Verlängert die Spalte Command für die Ausgabe, darf auch mehrfach verwendet werden
x	Eigene Prozesse ausgeben
u	Anzeige des Eigentümers, Rechenzeit und Speicheranteil des Prozesses

Tabelle 8.1: PS Parameter (Auswahl)

STAT	*Beschreibung*
S	Sleeping
R	Running
D	Dead
Z	Zombie

Tabelle 8.2: STAT (Auswahl)

8.1.2 Prozesse ausflisten mit pstree

Mit `pstree` werden die Prozesse als Baumstruktur angezeigt. Bei dieser Anzeige kann man den Zusammenhang zwischen `Eltern` und `Kind` Prozess sehen.

```
uws@tux>pstree -A uws
sshd---bash---pstree
```

Listing 8.2: pstree

Parameter	*Beschreibung*
a	Anzeige der Parameter, mit der die Prozesse laufen
h	Highlight für den eigenen Prozess
H	Mit der Angabe der Prozessnummer wird auch dieser im Highlight dargestellt
p	Ausgabe der Prozessnummer (PID)
u	Ausgabe des Benutzers
A	Die Ausgabe wird mit Ascii formatiert

Tabelle 8.3: Pstree Parameter (Auswahl)

8.1.3 Prozesse ausflisten mit pgrep

Der Befehl `pgrep` vereint die Befehle `ps` und `grep`. Mit dem Parameter `-lf` wird der vollständige Befehlsauruf inklusive aller Argumente angezeigt.

```
uws@tux>pgrep -lf ssh
6419 sshd: uws [priv]
6466 sshd: uws@pts/0
```

Listing 8.3: pgrep

8.1.4 Prozesse abkoppeln

Wird ein Programm oder ein Befehl in einer `Bash` gestartet und schließt man dann soft die `Bash`, so wird der Prozess gelöscht, da er an dem Mutterprozess (`Bash`) hing. Möchte man das verhindern, so wird das Programm / Befehl mit `nohup` aufgerufen. Der nun so erzeugte Prozess läuft nun ganz eigenständig und wird auch nicht beendet, wenn die Bash geschlossen wird.

```
uws@tux>nohub find . -name "*.mp3" > /tmp/MusikListe.txt
```

Listing 8.4: Nohub start

8.1.5 Prozess löschen

Muss ein Prozess beendet werden, so verwendet man den Befehl `kill`. Eine Auflistung aller Anweisungen erhält man mit dem Parameter `-l`. Mit Befehl kann man auch mehrere Prozesse löschen. Mehrere Prozesse kann man aber auch einfacher löschen mit dem Befehl `killall`. Anstelle der Prozessnummer erwartet der Befehl den Namen des Prozesses. Mit dem Parameter `-i` wird der Interaktive Modus eingeschaltet und es erfolgt eine Nachfrage, ob man den Prozess auch wirklich löschen möchte.

```
uws@tux>kill -9 6466

uws@tux>killall -i ssh
```

Listing 8.5: Kill Prozess

Anweisung	*Beschreibung*
SIGHUB	Nach dem beenden des Prozesses wird er direkt neu gestartet
SIGTERM	Dem zulöschenden Prozess wird die Möglichkeit gegeben, hinter sich aufzuräumen
SIGKILL	Hier wird der Prozess gelöscht, ohne Rücksicht auf Verluste
SIGSTOP	Der Prozess wird angehalten, bis man ihn mit `SIGCONT` wieder startet

Tabelle 8.4: Anweisung

Eine zweite Möglichkeit einen Prozess zu löschen gibt es mit dem Befehl `pkill`. Mit dem Parameter `-u` kann man gezielt Prozesse anderer Benutzer

```
uws@tux>pkill -9 -u uws
```

Listing 8.6: pkill

8.1.6 Prozess Limits

Die maximale Anzahl der Prozesse kann man sich mit `ulimit` anzeigen lassen.

```
uws@tux>ulimit -u
31482
```

Listing 8.7: Max. Prozesse

8.2 Module

8.2.1 Geladene Module anzeigen

Mit dem Befehl `lsmod` kann man sich die geladenen Module anzeigen lassen.

```
uws@tux>lsmod | grep dm
dm_mod  92428 0
```

Listing 8.8: geladene Module anzeigen

8.2.2 Verfügbare Module anzeigen

Mit dem Befehl `modprobe -l` werden alle verfügbaren Module angezeigt. Die Module befinden sich im Verzeichnis `/lib/modules/<kernelversion>/kernel`.

```
uws@tux>modprobe -l | less
```

Listing 8.9: Verfügbare Module anzeigen

8.2.3 Modul Informationen

Informationen über ein geladenes Modul erhält man mit dem Befehl `modinfo`.

```
uws@tux>modinfo <ModulName>
```

Listing 8.10: Modul Info

8.2.4 Module laden / entladen

Module können mit dem Befehl `modprobe` zur Laufzeit geladen oder entladen werden. Der Befehl löst alle Abhängigkeiten auf und lädt auch die Module, die das zu ladende Modul braucht. Mit der Option `-v` wird eine ausführliche Ausgabe der Aktion angezeigt. Um ein Modul zu entladen, gibt man die Option `-r` an.

```
uws@tux># laden
uws@tux>modprobe -v <ModulName>

uws@tux># entladen
uws@tux>modprobe -r <ModulName>
```

Listing 8.11: Modul laden / entladen

8.2.5 Module automatisch laden

Um Module bei einem Systemstart automatisch zu laden, trägt man den Befehl in die Datei

`/etc/init.d/boot.local` ein.

```
uws@tux>cat /etc/iit.d/boot.local
#! /bin/sh
#
# Copyright (C) 2002 SuSE Linux AG Nuernberg, Germany. All rights reserved.
#
.
.
#
modprobe dm-mirror
```

Listing 8.12: Modul automatisch laden / entladen

8.3 Boot Splash ändern

8.3.1 OpenSuSE

Den Boot Splash kann man folgendermaßen ändern.

- Download des neuen Boot Themes
- Entpacken der Datei
- Die ausgepackten Dateien, incl. der Verzeichnisse, werden in das Verzeichnis `/etc/bootsplash/themes` kopiert
- In der Datei `/etc/sysconfig/bootsplash` das neue Theme der Variable `THEME=` eintragen
- In der Console als `root mkinitrd` ausführen
- Reboot

8.4 Zeitserver

8.4.1 NTP

Mit dem Dienst `NTP` (Network Time Protocol) ist es möglich, einen Zeitserver auf zusetzten. Ist auf dem Rechner der Dienst `ntp` konfiguriert, so kann man mit den nachfolgenden Befehlen Informationen über die Zeitsynchronisation abfragen.

```
uws@tux>ntpq -p

uws@tux>watch ntpq -np
```

Listing 8.13: Zeitsynchronisation abfragen

Der Dienst `ntp` kann manuell gestartet werden, um eine Synchronisation mit den Zeitservern vorzunehmen.

```
uws@tux>/etc/init.d/ntpd [start] [restart] [force-reloaded] [status] # for
   init startup

uws@tux>sudo systemctl restart ntpd # for systemD

uws@tux>service ntpd force-reloaded # for oracle linux
```

Listing 8.14: Zeitsynchronisation starten

Die Systemzeit sollte bei jedem Starten und Herunterfahren des Rechners in die Realtime Clock des Rechners geschrieben werden. Falls das nicht geschieht, so kann man dieses mit dem Befehl `hwclock` vornehmen.

```
uws@tux>hwclock --show

uws@tux>hwclock --systohc

uws@tux>hwclock --help
```

Listing 8.15: Realtime Clock

In der Datei `/etc/ntp.conf` werden die abzufragenden Zeitserver eingetragen. Sind mehrere Zeitserver konfiguriert, so kann man mit `prefer` einen bevorzugten Zeitserver definieren.

```
------schnipp------
server <zeitserver1> prefer}
server <zeitzerver2>
server <zeitzerver3>
------schnapp------
```

Listing 8.16: Zeitserver

Der Client kann sich die Uhrzeit auch manuell von dem Zeitserver holen. Hierzu wird in der Console der Befehl `ntpdate` genommen.

```
uws@tux>sudo ntpdate <zeitserver>
```

Listing 8.17: Zeitserver kontaktieren

Folgende Zeitserver gibt es im Internet und können verwendet werden. Es ist nur eine Auswahl an Zeitservern.

Name	***URL***
TU-Berlin	ntps1-0.cs.tu-berlin.de
	ntps1-1.cs.tu-berlin.de
Uni Erlangen	ntp0.fau.de
	ntp1.fau.de
	ntp2.fau.de
	ntp3.fau.de
Evetel	ntp0.evetel.de
	ntp1.evetel.de
Freenet	ntp0.freenet.de
	ntp1.freenet.de
T-Online	ntp1.t-online.de
Web.de	ntp.web.de
Allgemein	0.de.pool.ntp.org
	1.de.pool.ntp.org
	2.de.pool.ntp.org
	3.de.pool.ntp.org

Tabelle 8.5: Zeitserver (Auswahl)

8.5 Cronjobs

8.5.1 Allgemein

Für bestimmte Zeitpunkte kennt das Programm `Cron` auch vordefinierte Labels. In der nachfolgenden Tabelle sind die Labels aufgelistet. Sie können anstatt der Zeitangabe angegeben werden. Neben der einfachen

Label	*Bedeutung*	*Entspricht*
@yearly, @annualy	Jährliche Ausführung	0 0 1 1 *
@monthly	Monatliche Ausführung	0 0 1 * *
@weekly	Wöchentliche Ausführung	0 0 * * 0
@daily, @midnight	Tägliche Ausführung	0 0 * * *
@hourly	Stündliche Ausführung	0 * * * *
@reboot	Nach dem Systemstart	-

Tabelle 8.6: Cron Label

Zeitangabe versteht `Cron` auch Bereichsangaben und auch Teiler. Mehrere Angaben werden durch ein Komma voneinander getrennt eingegeben. Ebenso kann man anstelle der Ziffernangabe für die Monate und Tage auch die englischen Abkürzungen verwenden.

```
0,15,30,45 * * * * w>/var/log/users
*/4 * * * * w>/var/log/users
```

Listing 8.18: Beispiel Cronjob

8.5.2 Systemweite Jobs

Die Systemweiten Cron-Jobs werden in der Datei `/etc/crontab` eingetragen. Der Aufbau der Datei ist der gleiche wie bei dem Benutzer Job. Der einzige Unterschied besteht darin, dass vor dem Befehl der Benutzername steht, der den Befehl ausführen soll.

```
uws@tux>cat /etc/crontab
SHELL=/bin/bash
HOME=/
PATH=/usr/bin:/usr/sbib:/sbin:/bin
MAILTO=root
-*/15 * * * * root /usr/logrotate
```

Listing 8.19: Beispiel System Job

Die drei Variablen am Anfang der Datei sind für folgendes definiert.

Variable	*Beschreibung*
SHELL	Angabe der zu verwendenden Shell
HOME	Das Home Verzeichnis
PATH	Pfade, in den nach Programmen gesucht wird
MAILTO	Empfänger der Cron-Job Meldungen

Tabelle 8.7: Variablen

Möchte man selber Systemweite Jobs erstellen, so sollte dieses nicht in der /etc/crontab geschehen, da bei einem Update des Programms Cron die Datei überschrieben wird. Hierfür gibt es das Verzeichnis /etc/cron.d. In diesem Verzeichnis sollten die selbsterstellten Job-Tabellen abgelegt werden.
Die meisten Distributionen legen immer wieder kehrende Jobs in den Verzeichnissen /etc/cron.hourly, /etc/cron.daily oder /etc/cron.monthly ab.

8.5.3 Benutzer Jobs

Benutzer Cronjobs werden mit dem Befehl crontab -e erstellt. Die Syntax in der Tabelle ist Minute, Stunde, Tag, Monat, Wochentag. Der Sonntag hat die 0 oder die 7, Montag die 1 u.s.w.

```
uws@tux>crontab -e
# Minute Stunde Tag(Monat) Monat  Tag(Woche)  Befehl
# (0-59) (0-23) (1-31)     (1-12) (0-7; 1=Mo)
   5      *      *          *      *          rm /var/log/uws.log
   */5    *      *          *      *          rm /var/log/uws.log
   2/5    *      *          *      *          rm /var/log/uws.log
   59     23     *          *      0          cp /var/log/messages /var/log/
      messages.back
   0      0      *          *      *          cp /var/log/messages /var/log/
      messages.back
   20,30  1      *          *      1-5        cp /var/log/messages /var/log/
      messages.back
```

Listing 8.20: Beispiel Benutzer Job

Der erste Befehl (beginnend mit 5...) wird 5 Minuten nach jeder vollen Stunde, der zweite alle 5 Minuten (die Schrittweite wird durch */Schrittweite angegeben), der dritte Job läuft ab der 2 Minute los und dann alle 5 Minuten. Der vierte Job startet einmal pro Woche sonntags um 23:59 Uhr, der fünfte täglich um 00:00 Uhr und der sechste montags bis freitags jeweils um 01:20 und 01:30 ausgeführt.
Die Con-Jobs werden im Verzeichnis /var/spool/cron/tabs mit dem jeweiligen Benutzernamen als Datei abgespeichert.

8.5.4 Berechtigungen

In der Datei /etc/cron.allow stehen die User drin, die Cron Jobs erstellen können. Benutzer, die keine Cron Jobs erstellen sollen, stehen in der Datei /etc/cron.deny drin.

```
uws@tux>cat /etc/cron.deny

guest
```

Listing 8.21: Berechtigung

8.6 Init-Skripte verwalten

Die Startskripte befinden sich im Verzeichnis `/etc/init.d`. Jeder Dienst, der in einem bestimmten Runlevel Starten oder Stoppen soll, bekommt einen Link in den Verzeichnissen `rc0.d` bis `rc6.d`. Die Startlinks bekommen als ersten Buchstaben ein `S` und die Stoplinks ein `K`. Als zweites kommt dann eine Zahl, an welcher Stelle / Reihenfolge der Dienst gestartet oder gestoppt werden soll.
Der Nachfolger für das Verwalten von Init-Skripten ist `SystemD`.

8.6.1 Verwalten mit chkonfig

Im Kopfbereich der Startskripte sind zwei Zeilen mit Kommentarzeichen enthalten, die für die Verwaltung mit `chkconfig` zuständig sind.

```
# chkconfig: 2345 55 25
# description: Oracle startup and shutdown
```

Listing 8.22: Beispiel Start Script

In der ersten Zeile steht nach dem Doppelpunkt, in welchem Runlevel der Dienst starten soll. In dem obigen Beispiel in den Runlevels 2, 3, 4 und 5. Danach wird die Startposition festgelegt und zum Schluss kommt die Stoppposition. In der Zweiten Zeile steht eine Beschreibung des Dienstes. Die Beschreibung kann auch mehrzeilig ausgeführt werden, sofern jede Zeile mit einem Backslash endet, bis auf die letzte Zeile.
Ruft man den Befehl `chkconfig` ohne Parameter auf, so werden alle Dienste angezeigt, in welchen Runlevels sie laufen.

```
uws@tux>chkconfig

uws@tux>chkconfig --ist <dienstname>
```

Listing 8.23: List Dienste

Um einen Dienst zu installieren, so wird der Parameter `-add` angegeben.

```
uws@tux>chkconfig --add <dienstname>
```

Listing 8.24: Dienst hinzufügen

Mit der Option `-del` kann man den Dienst löschen. Es werden nur die Links in der Verzeichnissen gelöscht, der Dienst wird weder gestoppt noch gestartet.

```
uws@tux>chkconfig --del <dienstname>
```

Listing 8.25: Dienst löschen

FÃ¼r das ändern eines Dienstes braucht man nicht das Init-Script zu ändern, sondern dieses geschieht mit der Option `-level`.

```
uws@tux>chkconfig --level 3 <dienstname> off [on]
```

Listing 8.26: Dienst ändern

In dem Beispiel oben wird der Startlink aus dem Runlevel 3 entfernt und ein Stopplink in dem Runlevel erstellt. Sollen die Änderungen in mehrere Runlevels durchgeführt werden, so wird gibt man die Levels ohne Leerzeichen an (23). Den alten Stand kann man mit der Option `reset` wiederherstellen, sofern man nicht die Init-Scripte angepasst hat.

```
uws@tux>chkconfig <dienstname> reset
```

Listing 8.27: Dienst wiederherstellen

8.6.2 Verwalten mit insserv

OpenSuse und auch die Suse Enterprise Produkte benutzen für die Verwaltung der Init-Scripte den `insserv` Befehl. `Insserv` kalkuliert bei jedem Lauf selbstständig, an welcher Position das Script gestartet oder gestoppt werden soll. Hierbei werden auch Abhängigkeiten der Dienste berücksichtigt. In dem Init-Script steht die Konfiguration für den `insserv` zwischen den Zeilen `### BEGIN INIT INFO` und `### END INIT INFO`.

```
### BEGIN INIT INFO
# Provides: sshd
# Required-Start: $network $remote_fs
# Required-Stop: $network $remote_fs
# Default-Start: 3 5
# Default-Stop: 0 1 2 6
# Description: Start the SSH Daemon
### END INIT INFO
```

Listing 8.28: Insserv

In der ersten Zeile steht, welchen Dienst das Script zu Verfügung stellt. In den Zeilen zwei und drei stehen die Abhängigkeiten drin, welche für die Berechnung des Automatischen Starten oder Stoppens zuständig sind. Diese Boot-Facilities stehen in der Datei `/etc/insserv.conf` drin. Gibt man in dem Init-Script die Boot-Facilitie `$all` an, so wird der Dienst erst nach allen anderen Diensten gestartet.
Die letzten drei Zeilen geben an, in welchen Runlevels der Dienst gestartet oder gestoppt werden soll. Sowie eine Beschreibung des Dienstes.
Um einen Dienst mit `insserv` zu erstellen, so ruft man das Programm mit dem Dienstnamen auf. Das Programm erstellt automatisch dir Start- und Stopplinks in den definierten Runlevels.

```
uws@tux>sudo insserv <dienstname>
```

Listing 8.29: Dienst erstellen

Einen Dienst kann man mit der Option `-r` wieder entfernen.

```
uws@tux>sudo insserv -r <dienstname>
```

Listing 8.30: Dienst entfernen

Möchte man einen Dienst auch in einem zusätzlichen Runlevel anlegen, so geschieht das mit der nachfolgenden Befehlszeile.

```
uws@tux>sudo insserv <dienstname>,start=3
```

Listing 8.31: Dienst modifizieren

Die Standardeinstellung für den Dienst kann man mit der Option `-d` wieder herstellen.

```
uws@tux>sudo insserv -d <dienstname>
```

Listing 8.32: Dienst wiederhestellen

Leider kann `insserv` keine Liste aller Dienste ausgeben, hierfür kann man das Programm `chkconfig --list` nehmen.

8.7 Runlevel

8.7.1 Init System

Den aktuellen Runlevel kann man sich mit `runlevel` sich anzeigen lassen.
Folgende Runlevel gibt es:

Runlevel	*Beschreibung*
0	Shutdown
S	Single User
1	Single User ohne Netzwerk
2	Lokaler Mehrbenutzerbetrieb ohne Netzwerk
3	Mehrbenutzerbetrieb incl. Netzwerk
4	Nicht definiert
5	Wie Runlevel 3, aber mit grafischer Oberfläche
6	Reboot

Tabelle 8.8: Runlevel

```
uws@tux>runlevel

N 5
```

Listing 8.33: Aktuellen Runlevel

8.7.2 SystemD

Unter SystemD kann die Verwaltung der Runlevel mit dem Programm systemctl steuern und auch abfragen. Anstelle der Runlevel Nummern kommen nun Targets zum Einsatz. In der nachfolgenden Tabelle ist eine Gegenüberstellung der Runlevels mit der Targets.

Runlevel	*Target*
0	poweroff.target oder Symbolik Link runlevel0.target
1	rescure.target oder runlevel1.target
3	multi-user.target oder runlevel3.target
5	graphical.target oder runlevel5.target
6	reboot.target oder runlevel6.target
emergency	emergency.target

Tabelle 8.9: Runlevel / Target

```
uws@tux># Show active runlevel
uws@tux>sudo systemctl get-default
graphical.target

uws@tux># change default runlevel
uws@tux>sudo systemctl -set-default multi-user.target

uws@tux># change runlevel
uws@tux>sudo systemctl isolate multi-user.target
```

Listing 8.34: Runlevel

8.8 Konsolen Login nicht möglich

Kann man sich nicht mehr an einer Konsole (tty0 - tty7) anmelden und es erscheint in der Konsole die Meldung, das `Module kann nicht geladen werden`, so muss man die `/etc/pam.d/login` überprüfen.

Bei einem 32bit Linux sollte die pam_limits.so nach /lib/security zeigen und bei einem 64bit nach /lib64/security.

8.9 Syslog

Alle Meldungen werden in die message.log geschrieben, sofern es sich nicht um ein SystemD System handelt. Standardmäßig wird jede Stunde in die Datei eine Meldung geschrieben. Soll diese Zeit verändert werden, so wird dieses in die Datei syslog-ng.conf.in gemacht. Den Wert von stats(3600) ist auf die gewünschte Update Intervall zu setzten. Die Angabe erfolgt in Sekunden.

```
#
# global options
#
options { long_hostname(off); sysnc(0); perm(0640); stats(1200); };

uws@tux>SuSEconfig -module syslog-ng
Starting SuSEconfig, the SuSE Configuration Tool ...
Running Module syslog-ng only
Reading /etc/sysconfig and updating the system ...
Executing /sbin/conf.d/SuSEconfig.syslog-ng ...
Checking //etc/syslog-ng/syslog-ng.conf.SuSEconfig file: ok
Installing new //etc/syslog-ng/syslog-ng.conf
Finished

uws@tux>/etc/init.d/syslog restart
Shutting down syslog services     done
Starting syslog services          done
```

Listing 8.35: Syslog

Auf RedHat Systemen wird in der Datei /etc/rsyslog.conf das $ModLoad immark aktiviert und der Parameter $MarkMessagePeriod 1200 eingefügt. Die Angabe der Zeit erfolgt in Sekunden.

8.10 Sudo (Runas)

Mit dem Befehl `sudo` kann man Programme unter einem anderen Benutzer, z.B. als root, ausführen. Für die Konfiguration ist die Datei `/etc/sudoers` zuständig, die man mit visudo editieren kann. Eigene Konfigurationsdateien legt man am besten im Verzeichnis `/etc/sudoers.d` ab, da sonst bei einem Update die `/etc/sudoers` Datei überschrieben wird. Es werden nur Dateien eingelesen, die keine Extension haben. Mit `visudo -f <Dateiname>` können die eigenen Dateien editiert werden. Bei dem beenden von `visudo` wird ein Syntax check durchgeführt. Die Rechte für die Dateien sollten auf `0440` gesetzt werden.

8.10.1 Syntax

Die Syntax kann man sich mit dem folgenden Satz beschrieben:

- Wer darf
- Von was
- Als was
- Mit welchen Einschränkungen /Eigenschaften
- was

```
User Hostname = (ZielUser) Einstellungen: Befehl, Befehl
```

Listing 8.36: Syntax sudo

Anstelle eines Users kann man auch eine Gruppe angeben, die mit einem vorangestellten Prozentzeichen definiert werden.

8.10.2 Alias

Man kann Aliase definieren, in dem mehrere Befehle zusammengesetzt werden. Die erlaubten Aliasnamen enthalten nur Großbuchstaben, Ziffern oder den Underscore. Die Aliasnamen beginnen immer mit einem Großbuchstaben.
Die Syntax hierzu lautet:

```
Aliastype Aliasname = Befehl, Befehl
```

Listing 8.37: Syntax Alias

Folgende Alias Typen gibt es:

- User_Alias Benutzerliste
- Runas_Alias Zielbenutzerliste
- Host_Alias Hostnameliste
- Cmnd_Aliad Befehlsliste
- was

```
Cmnd_Alias CMD_SHUTDOWN = /sbin/shutdown, !/sbin/init *, /sbin/init 0
```

Listing 8.38: Beispiel Alias

In dem Beispiel kann ein Shutdown durchgeführt werden, aber kein /sbin/init Befehl, bis auf /sbin/init 0. Die Negation wird mit einem vorangestellten Ausrufezeichen eingeleitet

8.10.3 Einstellungen

In der folgenden Übersicht gibt es die Ausführungsbeschränkungen.

Einschalten	*Ausschalten*	*Beschreibung*
PASSWD	NOPASSWD	Abfrage des Passwortes
SETENV	NOSETENV	Weitergabe der Umgebungsvariablen
EXEC	NOEXEC	Shellescapes verhindern

Tabelle 8.10: Ausführungsbeschränkungen

8.10.4 Beispiele

Alle User dürfen einen Befehl ohne Passwort Abfrage ausführen, außer der User harry.

```
ALL, !harry ALL=(root) NOPASSWD:/bin/info.sh
```

Listing 8.39: Beispiel harry

Alle User der Gruppe `admin` dürfen alle Befehle ohne Passwort Abfrage ausführen, nur nicht Befehle für das Netzwerk und System Update.

```
%admin ALL=(root) NOPASSWD:ALL,!CMD_NETWORK,!CMD_UPDATE
```

Listing 8.40: Beispiel admin

Alias CMD_UPDATE definieren.

```
Cmnd_Alias CMD_UPDATE = /usr/bin/zypper, /usr/bin/zypper pchk, !/usr/bin/
   zypper patch
```

Listing 8.41: Beispiel Alias

8.11 Dienste anzeigen

Welche Dienste automatisch gestartet werden, kann man mit dem Befehl `chkconfig` überprüfen.

```
uws@tux>chkconfig -t | grep on
acpid on
alsasound on
apache2 on
.
.
```

Listing 8.42: Show Services

Laufende Dienste können mit dem Befehl `systemctl` oder `service` überprüft werden.

```
uws@tux>systemctl -a | grep running
proc-sys-fs-binfmt_misc.automount load active running Arbitary Executable .
acpid.service                     load active running ACPI Event Daemon
apache2.service                   load active running apache
.
.
uws@tux>service -status-all | grep running
Checking for service acpid         running
Checking for httpd2:               running
.
.
```

Listing 8.43: Show Services

Den Status eines Dienstes kann man sich mit `systemctl status` sich anzeigen lassen.

```
root@tux>systemctl status cron.service
cron.service - Command Scheduler
     Loaded: loaded (/usr/lib/systemd/system/cron.service; enabled)
     Active: active (running) since Mon, 2013-07-15 12:30:00 CEST; 18h ago
   Main PID: 1290 (cron)
     CGroup: name=systemd:/system/cron.service
```

Listing 8.44: Service Status

8.12 CPU Info

Informationen über die CPU stehen in `/proc/cpuinfo`.

```
uws@tux>cat /proc/cpuinfo
processor         : 0
vendor_id         : GenuineIntel
cpu family        : 6
model             : 26
model name        : IntelÂ® XeonÂ® CPU X5550 @ 2.67GHz
.
.
```

Listing 8.45: CPU Info

Auch mit dem Befehl `lscpu` kann man sich Informationen über die CPU's anzeigen lassen.

```
uws@tux>lscpu
Architecture:   i686
CPU op-mode(s): 32bit, 64bit
Byte Order:     Little Endian
CPU(s):         2
.
.
```

Listing 8.46: CPU Info

8.13 Server Domain

8.13.1 Software

Damit man einen Server/Computer in einer Domain aufnehmen kann, muss folgende Software installiert sein.

- Redhat-config-samba # Nur bei Red Hat Systemen
- Samba-common
- Samba-lient
- Samba
- Pam_krb5
- Krb-workstation
- Krb-libs
- krbafs

8.13.2 Konfiguration

Die Dateien `krb5.conf`, `smb.conf`, `nsswitch.conf`, `ntp.conf` und `hosts` müssen nun konfiguriert werden.

```
root@tux>cat /etc/krb5.conf
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log

[libdefaults]
default_realm = GO.SEABAER-AG.DE
kdc_timesync = 1
ccache_type = 4
dns_lookup_realm = false
dns_lookup_kdc = false
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true

[realms]
GO.SEABAER-AG.DE = {
kdc = gode001.go.seabaer-ag.de
kdc = gode002.go.seabaer-ag.de
admin_server = gode001.go.seabaer-ag.de
default_domain = go.seabaer-ag.de
}

[domain_realm]
.go.seabaer-ag.de = GO.SEABAER-AG.DE
# .example.com = EXAMPLE.COM
# example.com = EXAMPLE.COM
```

Listing 8.47: krb5.conf

```
root@tux>cat /etc/samba/smb.conf
[global]
netbios name = tux
workgroup = GO
realm = GO.SEABAER-AG.DE
password server = gode001.go.seabaer-ag.de, gode002.go.seabaer-ag.de,
#wins server =
passdb backend = tdbsam
security = ADS
encrypt passwords = true
log level = 0
idmap backend = idmap_rid:OPENITSOLUTIONS=10000-100000000
idmap uid = 10000-100000000
idmap gid = 10000-100000000
allow trusted domains = no
template shell = /bin/nologin
client use spnego = yes
client ntlmv2 auth = yes
winbind use default domain = yes
winbind enum users = yes
winbind enum groups = yes
winbind nested groups = yes
restrict anonymous = 2
domain master = no
local master = no
preferred master = no
os level = 0
# log level = 10

template homedir = /home/%U
template shell = /bin/nologin

log file = /var/log/samba/%m.log
max log size = 5000

printcap name = /etc/printcap
load printers = no

[homes]
comment = Home Directories
browseable = no
writeable = yes

[printers]
comment = All Local Printers
path = /var/spool/samba
browseable = no
guest ok = no
writeable = no
printable = yes
```

Listing 8.48: smb.conf

```
root@tux>cat /etc/nsswitch.conf
#
# /etc/nsswitch.conf
#
# An example Name Service Switch config file. This file should be
# sorted with the most-used services at the beginning.
#
# The entry '[NOTFOUND=return]' means that the search for an
# entry should stop if the search in the previous entry turned
# up nothing. Note that if the search failed due to some other reason
# (like no NIS server responding) then the search continues with the
# next entry.
#
# Valid entries include:
#
# nisplus   Use NIS+ (NIS version 3)
# nis       Use NIS (NIS version 2), also called YP
# dns       Use DNS (Domain Name Service)
# files     Use the local files
# db        Use the local database (.db) files
# compat    Use NIS on compat mode
# hesiod    Use Hesiod for user lookups
# [NOTFOUND=return] Stop searching if not found so far
#
# To use db, put the "db" in front of "files" for entries you want to be
# looked up first in the databases
#
# Example:
#passwd:  db files nisplus nis
#shadow:  db files nisplus nis
#group:   db files nisplus nis

passwd:   compat winbind
shadow:   compat
group:    compat winbind

#hosts:   db files nisplus nis dns
hosts:    files dns

# Example - obey only what nisplus tells us...
#services:  nisplus [NOTFOUND=return] files
#networks:  nisplus [NOTFOUND=return] files
#protocols: nisplus [NOTFOUND=return] files
#rpc:       nisplus [NOTFOUND=return] files
#ethers:    nisplus [NOTFOUND=return] files
#netmasks:  nisplus [NOTFOUND=return] files

bootparams: nisplus [NOTFOUND=return] files

ethers:     files
netmasks:   files
networks:   files
protocols:  db files
rpc:        db files
services:   db files

netgroup:   nisplus
publickey:  nisplus
automount:  files nisplus
```

```
aliases:     files nisplus
```

Listing 8.49: nsswitch.conf

```
uws@tux>grep -i "go.seabaer-net.de" /etc/ntp.conf
server gode001.go.seabaer-ag.de
server gode002.go.seabaer-ag.de
```

Listing 8.50: ntp.conf

In der /etc/hosts Datei wird der Servr Name eingetragen.

```
uws@tux>grep -i "go.seabaer-net.de" /etc/hosts
192.158.10.200  fsgo200.go.seabaer-net.de fsgo200
```

Listing 8.51: hosts

8.13.3 Dienste starten

Nun müssen die Dienste smb und winbind gestartet werden.

```
uws@tux>sudo service smb start
uws@tux>sudo chkconfig smb on

uws@tux>sudo service winbind start
uws@tux>sudo chkconfig winbind on
```

Listing 8.52: Dienste starten

8.13.4 Domain aufnehmen

Nach erfolgreicher Konfiguration kann der Server in die Domain aufgenommen werden.

```
uws@tux>sudo net ads join -U <user_with_domain_rights>
```

Listing 8.53: Domain aufnehmen

8.13.5 Testen

Nun können Benutzer,Gruppen und Informationen der Domain abgefragt werden.

```
uws@tux>sudo kinit uws@go.seabaer-ag.de # Domain eventuell Gross schreiben

uws@tux>sudo klist -e

uws@tux>sudo wbinfo -u # Domain User List
uws@tux>sudo wbinfo -g # Domain Gruppen
uws@tux>sudo wbinfo -t
```

Listing 8.54: Tests

8.14 Firewall

8.14.1 Ausschalten / Deaktivieren

Die lokale Firewall kann mit chkconfig oder service ausgeschaltet werden.

```
uws@tux>sudo service ipchains off
uws@tux>sudo service iptables off

uws@tux>sudo chkconfig ipchains off
uws@tux>sudo chkconfig iptables off
```

Listing 8.55: Ausschalten

Unter Oracle Linux (Red Hat) den Wert `SELINUX` auf `disabled` stellen, der sich in der Datei `/etc/selinux/conf` befindet. Danach den Rechner neu starten oder `/usr/sbin/setenforce 0` eingeben. Den Status kann man mit `/usr/sbin/getenforce` abfragen. Mit `setenforce` kann man die Firewall im laufenden Betrieb ausschalten.

8.15 Ports

8.15.1 Anzeigen

Einen belegten Port kann man mit `netstat` sich anzeigen lassen.

```
uws@tux>sudo netstat -anp | grep tcp
tcp   0   0.0.0.0.0:80  0.0.0.0:*   Listen  1658/nginx
```

Listing 8.56: belegten Port anzeigen

Offene Ports kann man sich mit `nmap` anzeigen lassen. Ohne die Angabe der Ports, die `nmap` prüfen soll, werden die Ports 1 bis 1024 gescannt, sowie die Ports, die in der Datei `/usr/share/nmap/nmap-service` stehen.

```
uws@tux>sudo nmap -v localhost
uws@tux>sudo nmap fritz.box
```

Listing 8.57: offene Ports anzeigen

Parameter	*Beschreibung*
-T	Geschwindigkeits des Scans, 0 für niedrig und 5 für sehr hoch
-traceroute	Gibt aus, über welche Ports und Protokolle das Ziel zu erreichen ist
-p	Angabe der Ports. Default ist 1-1024 z.b: -p -900; -p 22,80
-sV	Versionsscan, es wird versucht, die Applikation heruaszufinden, die an dem Port lauscht
-sU	UDP Scan

Tabelle 8.11: Parameter nmap

8.15.2 Freigeben

Den belegten Port kann man mit `fuser` wieder freigeben.

```
uws@tux>sudo fuser -k -n tcp 80
```

Listing 8.58: belegten Port freigeben

8.16 Autostart

8.16.1 Gnome

Pfad	*Beschreibung*
/etc/xdg/autostart	Systemweit
/usr/share/gnome/autostart	Systemweit
~/.config/autostart	User

Tabelle 8.12: Gnome Pfade

8.16.2 KDE

Pfad	*Beschreibung*
/etc/xdg/autostart	Systemweit
/usr/share/gnome/autostart	Systemweit
~/.kde4/Autostart	User (KDE4)
~/.kde4/shutdown	User (KDE4)
~/.config/autostart	User (KDE5)
~/.config/autostart-scriptscriptfont	User (KDE5)
~/.config/plasma-workspace/env	User (KDE5)
~/.config/plasma-workspace/shutdown	User (KDE5)

Tabelle 8.13: KDE Pfade

8.17 Memory

Die aktuelle Speicherbelegung kann man sich mit dem folgenden Statement abfragen.

```
uws@tux>egrep 'Mem|Cache|Swap' /prov/meminfo
MemTotal:   66007312 kB
MemFree:    15150272 kB
Cached:     62775044 kB
SwapCached:      200 kB
SwapTotal: 52428792 kB
SwapFree:  52301204 kB
```

Listing 8.59: Memory Info

Auch mit dem Befehl `free`, kann man sich den Speicherverbrauch anzeigen lassen.

```
uws@tux>free -m -t
```

Listing 8.60: Memory Info mit free

Eine schöne graphische Anzeige gibt es mit dem Programm `htop`. Dieses Programm kann aus den Repositories nachinstalliert werden, da es meistens nicht mit installiert wird.

8.18 HugePages

Die HugePages Einstellungen kann man mit folgendermaßen abfragen.

```
uws@tux>egrep 'Huge' /proc/meminfo
HugePagesTotal:  0
HugePages_Free:  0
HugePages_Rsvd:  0
HugePagesize:  2048 kB
```

Listing 8.61: HugePages anzeigen

Die Größe der HugePage kann man mit dem folgenden Script ermitteln.

```
uws@tux>cat HugePages.sh
#!/bin/bash
#
# Setting check for HugePages
# The Script is from:
# docs.oracle.com/cd/E37670_01/E37355/html/ol_config_hugepages.html
#
KERN=`uname -r | awk -F. '{printf("%d.%d\n",$1,$2);}'`
```

```
#
# Find HugePage size
#
HPG_SZ=`grep Hugepagesize /proc/meminfo | awk {'print $2'}`
NUM_PG=1
for SEG_BYTES in `ipcs -m | awk {'print $5'} | grep "[0-9][0-9]*"`
do
  MIN_PG=`echo "$SEG_BYTES/($HPG_SZ*1024)" | bc -q`
  if [ $MIN_PG -gt 0 ]; then
    NUM_PG=`echo "$NUM_PG+$MIN_PG+1" | bc -q`
  fi
done
#
# Finish and output the results
#
case $KERN in
  '2.4') HUGETLB_POOL=`echo "$NUM_PG*$HPG_SZ/1024" | bc -q`;
         printf "\nRecommend setting: vm.hugetlb_pool = $HUGETLB_POOL\n";;
  '2.6')  printf "\nRecommend setting: vm.nr_hugepages = $NUM_PG\n";;
  *)      printf "\nUnrecognized kernel version $KERN. Exiting.\n";;
esac

uws@tux>./HugePages.sh

Recommended setting: vm.nr_hugepages = 3848
```

Listing 8.62: HugePage Größe

Einschalten der HugePages werden dann mit:

```
uws@tux>echo 3848 >/proc/sys/vm/nr_hugepages
```

Listing 8.63: HugePage aktivieren

Damit dieses auch den Reboot überlebt, so trägt man in der `/etc/sysctl.conf` folgende Zeile ein.

```
uws@tux>grep huge /etc/sysctl.conf

vm.nr_hugepages = 3848
```

Listing 8.64: HugePage reboot

Zum Abschluss der Konfiguration muss noch der Wert in der Datei `/etc/security/limits.conf` gesetzt werden. Hierzu wird der Wert der `HugePages` mal den Wert der `HugePageSize` genommen. In diesem Beispiel also 3848*2048=7880704.

```
uws@tux>egrep memlock /etc/security/limits.conf
oracle soft memlock 7880704
oracle hard memlock 7880704
```

Listing 8.65: HugePage limits.conf

8.19 Shutdown / Reboot

Einen `Shutdown` des Systems kann man mit einem init 0 erledigen und einen `Reboot` des Systems kann man mit einem init 6 einleiten. Nachfolgend möchte ich weiter auf die Befehle `shutdown` und `reboot` eingehen.

8.19.1 Shutdown

Soll das System nach einer Wartezeit herunterfahren, so gibt man dem Befehl die Zeit in Sekunden an.

```
uws@tux>sudo shutdown 10
```

Listing 8.66: Shutdown Wartezeit

Alternativ kann man auch die Uhrzeit angeben, wann das System heruntergefahren werden soll.

```
uws@tux>sudo shutdown -r 14:30
```

Listing 8.67: Shutdown Uhrzeit

Folgende Optionen kann man mit angeben.

Option	*Beschreibung*
-H	Das System anhalten
-P	Das System ausschalten
-r	Das System neu starten
-c	Abbruch des Neustarts

Tabelle 8.14: Option Shutdown

8.19.2 Last Reboot

Möchten man wissen, wann das System das letzte Mal neu gestartet worden ist, so kann man das mit den folgenden Befehlen sich anzeigen lassen.

```
uws@tux>uptime
07:22 1:35 an, 4 Benutzer, Durchschnittslast: 0,02, 0,14, 0,12

uws@tux>last reboot | head -n 1
reboot system boot 3.11.10-21-deskt Mon Oct 6 05:45 - 07:22 (1:37)

uws@tux>who -b
      Systemstart 2014-10-06 05:45
```

Listing 8.68: Letztes Reboot

8.20 Anmelden disablen

Möchte man für Wartungsarbeiten an dem System das Anmelden verbieten, so kann man im Verzeichnis `/etc` die Datei `nologin` erstellen. Danach können sich die User nicht mehr an dem System anmelden, außer `root` natürlich. Gibt es im Verzeichnis `/etc` auch die Datei `nologin.txt`, so wird dieser Inhalt dem User angezeigt, wenn er sich anmelden möchte.

8.21 Hinweistext (MoD)

In der Datei `/etc/motd` (message of the day) kann man einen Text platzieren, der bei der Anmeldung an dem System angezeigt wird.

```
uws@tux>cat /etc/motd
Have a lot of fun ...
```

Listing 8.69: Hinweistext

Einen Text vor der Anmeldung kann man in den Dateien `/etc/issue` (Konsole) und `/etc/issue.net` (Netzlogin) platzieren.

```
uws@tux>cat /etc/issue
Welcome to openSUSE 13.1 "Bottle" - Kernel \r (\l).

uws@tux>cat /etc/issue.net
Welcome to openSUSE 13.1 "Bottle" - Kernel %r (%t).
```

Listing 8.70: Hinweistext vor Login

Folgende Escape Codes gibt es:

Escape	Beschreibung
\b	Ausgabe der Baudrate
\d	Ausgabe des aktuellen Datums
\s	Ausgabe des System Namens, Name des OS System
\l	Ausgabe des aktuelle tty Name
\m	Ausgabe des Architecture Identifier
\n	Ausgabe des Hostnamens
\o	Ausgabe der Domain
\r	Ausgabe der Release Nummer des Systems
\t	Ausgabe der aktuellen Zeit
\u	Ausgabe der Anzahl der angemeldeten User
\U	Ausgabe des Textes "1 User" oder "Users"
\v	Ausgabe der Version des OS Systems, Build Date und mehr

Tabelle 8.15: Escape Codes

Man kann auch vor jedem SSH-Login ein Script ausführen, das Systeminformationen auf dem Schirm ausgibt. Hierzu habe ich ein Script auf Github , `https://github.com/lfelipe1501/Arch-MOTD/blob/master/update_motd.sh` gefunden.

```
#!/bin/env bash
#Script to update motd with relevant information.

#Define output file
#motd="/etc/motd"
motd="/home/$USER/bin/motd"

# Collect information
HOSTNAME=`uname -n`
KERNEL=`uname -r`
CPU=`awk -F '[ :][ :]+' '/^model name/ { print $2; exit; }' /proc/cpuinfo`
ARCH=`uname -m`
PACMAN=`checkupdates | wc -l`
DETECTDISK=`mount -v | fgrep 'on / ' | sed -n 's_^\(/dev/[^ ]*\) .*$_\1_p'`
DISC=`df -h | grep $DETECTDISK | awk '{print $5 }'`
MEMORY1=`free -t -m | grep "Mem" | awk '{print $6" MB";}'`
MEMORY2=`free -t -m | grep "Mem" | awk '{print $2" MB";}'`
MEMPERCENT=`free | awk '/Mem/{printf("%.2f% (Used) "), $3/$2*100}'`

#Time of day
HOUR=$(date +"%H")
if [ $HOUR -lt 12  -a $HOUR -ge 0 ]
then    TIME="Morning"
elif [ $HOUR -lt 17 -a $HOUR -ge 12 ]
then    TIME="Afternoon"
else    TIME="Evening"
fi

#System uptime
uptime=`cat /proc/uptime | cut -f1 -d.`
upDays=$((uptime/60/60/24))
upHours=$((uptime/60/60%24))
upMins=$((uptime/60%60))
upSecs=$((uptime%60))
```

```
#System load
LOAD1=`cat /proc/loadavg | awk {'print $1'}`
LOAD5=`cat /proc/loadavg | awk {'print $2'}`
LOAD15=`cat /proc/loadavg | awk {'print $3'}`

#Color variables
#W="\033[00;37m"
W="\033[0m"
B="\033[01;36m"
R="\033[01;31m"
G="\033[01;32m"
N="\033[0m"

#Clear screen before motd
cat /dev/null > $motd

echo -e "
$G .--.      .--.
$G |   \    /   |                                    $B _ _
$G |  . \_/ .  |  __ _ _ __  _  __ _ _ __  ____  $B| (_)_ __  _   ___  __
$G |  |\    /|  | / _' | '_ \| |/ _' | '__|/    \ $B| | | '_ \| | | \ \/ /
$G |  | \_/ |  || (_| | | | | | | (_| | |   | () | $B| | | | | | |_| |>  <
$G |  |      |  | \__,_|_| |_| |\__,_|_|   \____/ $B|_|_|_| |_|\__._/_/\_\\
$G '--'      '--'           |___|

                        Based on
" > $motd

echo -e "
        $B. $W
       $B/#\ $W                        _      $B _ _
      $B/###\ $W       __ _ _ __ ___| |__   $B| (_)_ __  _   ___  __
     $B/#####\ $W    / _' | '__/ __| '_ \ $B| | | '_ \| | | \ \/ /
    $B/##.-.##\ $W  | (_| | | | (__| | | |$B| | | | | | |_| |>  <
   $B/##(   )##\ $W  \__,_|_|  \___|_| |_|$B|_|_|_| |_|\__._/_/\_\\
  $B/#.--   --.#\ $W
 $B/'            '\
" >> $motd
echo -e "$G-----------------------------------------------------------------"
   >> $motd
echo -e "$W   Good $TIME$A You're Logged Into $B$A$HOSTNAME$W! "      >>
   $motd
echo -e "$G-----------------------------------------------------------------"
   >> $motd
echo -e "$B    KERNEL $G:$W $KERNEL $ARCH                                  "
   >> $motd
echo -e "$B       CPU $G:$W $CPU                                           "
   >> $motd
echo -e "$B    MEMORY $G:$W $MEMORY1 / $MEMORY2 - $MEMPERCENT              "
   >> $motd
echo -e "$B  USE DISK $G:$W $DISC (Used)                                   "
   >> $motd
echo -e "$G-----------------------------------------------------------------"
   >> $motd
echo -e "$B  LOAD AVG $G:$W $LOAD1, $LOAD5, $LOAD15             " >> $motd
echo -e "$B    UPTIME $G:$W $upDays days $upHours hours $upMins minutes
   $upSecs seconds " >> $motd
#echo -e "$B PROCESSES $G:$W You are running $PSU of $PSA processes        "
    >> $motd
echo -e "$B    PACMAN $G:$W $PACMAN packages can be updated                "
   >> $motd
```

```
echo -e "$B     USERS $G:$W `users | wc -w` users logged in            " >>
    $motd
echo -e "$G-----------------------------------------------------------------"
   >> $motd
echo -e "  $W Please do not share your login and use$G sudo$W for$R root$W
   access " >> $motd
echo -e "$G-----------------------------------------------------------------"
   >> $motd
echo -e "$N" >> $motd
```

Listing 8.71: MoD Beispiel

Um nun das Script bei jedem Login auszuführen, sind nachfolgend die verschiedenen Dateien zu ändern. Dieses ist für Arch Linux und andere Distributionen, die auf Arch Linux basieren.

```
uws@tux># Remove Line: session optional pam_motd.so motd=/etc/motd
uws@tux># from file /etc/pam.d/system-login

uws@tux># Add two lines into file /etc/pam.d/sshd
uws@tux>grep -i motd /etc/pam.d/sshd

session  optional  pam_exec.so  stdout /usr/bin/UpdateMotd.sh
session  optional  pam_motd.so  motd=/etc/motd
```

Listing 8.72: MoD Beispiel

8.22 Systeminformationen

Mit dem Befehl `uname` kann man sich die Systeminformationen, wie zum Beispiel die Kernel Version, anzeigen lassen. Ein Aufruf von `uname` ohne Parameter, entspricht ein `uname -s`.
In der nachfolgenden Tabelle stehen die Parameter und welche Informationen sie ausgeben.

Parameter	*Beschreibung*
-a/--all	Ausgabe aller Informationen
-i/--hardware-platform	Ausgabe der Prozessorfamilie
-m/--machine	Ausgabe der Hardwareplattform
-n/--nodename	Ausgabe des Hostnamens
-o/--operating-system	Ausgabe des Betriebssystems
-p/--processor	Ausgabe des Prozessornamens
-r/--kernel-release	Ausgabe der vollständigen Kernel-Versionsnummer
-s/--kernel-name	Ausgabe des Kernel Namens
-v/--kernel-version	Ausgabe Erstellungsdatum des Kernels

Tabelle 8.16: Parameter

```
uws@tux>uname -s
Linux

uws@tux>uname -r
3.11.10-21-desktop
```

Listing 8.73: Hinweistext

8.23 SSL Zertifikate

8.23.1 Erstellen

SSL Zertifikate kann man mit `openssl` erstellen. In den nachfolgenden Beispiel wird ein Zertifikate für HTPPS Verbindungen für das Programm Shellinabox erstellt.

```
root@tux>cd /etc/shellinabox

root@tux>openssl genrsa -des3 -out server.key 1024

root@tux>openssl req -new -key server.key -out server.csr

root@tux>cp server.key server.key.org

root@tux>openssl rsa -in server.key.org -out server.key

root@tux>openssl x509 -req -days 365 -in server.cst -signkey server.key -out
    server.crt

root@tux>cat server.crt server.key > certificate.pem
```

Listing 8.74: Zertifikat erstellen

8.24 NFS

8.24.1 Freigaben anzeigen

Freigegebene NFS Shares kann man sich mit `showmount -e` anzeigen lassen.

```
root@tux>showmount -e tux01
Export list for tux01
/daten/backup tux
/daten/sw     (everyone)
```

Listing 8.75: Freigaben anzeigen

8.24.2 Fstab

Sollen NFS Shares bei dem Start des Rechners verbunden werden, so werden die Shares in der `/etc/fstab` eingetragen.

```
root@tux>grep nfs /etc/fstab
<server>:/mail /mail nfs rsize=8192,wsize=8192,timeo=14,intr 0 0
```

Listing 8.76: Fstab

Option	*Beschreibung*
rsize	Max. bytes read
wsize	Max. bytes write
timeo	Zeit NFS Client wartet auf Anfrage
intr	Erlauben, NFS Anfragen zu unterbrechen, wenn der Server ausfällt

Tabelle 8.17: Optionen

8.25 Fstab

8.25.1 Beispiel

```
uws@tux>cat /etc/fstab
#device-spec  mount_point fs-type options                          dump
   pass
/dev/sda1     /           ext4    default                            1  1
LABEL=home    /home       ext4    default                            0  0
fsgo100:/data /data       nfs     rsize=8192,wsize=8192,timeo=14,intr 0  0
.
.
```

Listing 8.77: Beispiel /etc/fstab)

8.25.2 Device-Spec

Die Geräte (Festplatten, DVD) können entweder als Gerätedatei, Label, UUID, NFS oder CIFS angesprochen werden.
Labels kann man für die verschiedenen Dateisysteme setzen. Für `ext2/3/4` können die Programme `e2label` oder `tune2fs` genommen werden. Für `Reiserfs` das Programm `reiserfstune`, `xfs` das Programm `xfs_admin` und für `btrfs` das Programm `btrfs`.

```
uws@tux>sudo e2label /dev/sda2 mydata
uws@tux>sudo tune2fs -L mydata /dev/sda2

uws@tux>sudo reiserfstune -l mydata /dev/sda2

uws@tux>sudo xfs_admin -L "mydata" /dev/sda2

uws@tux>sudo btrfs filesystem label /dev/sda2 mydata
```

Listing 8.78: Labels setzen)

Die UUID kann man sich mit `blkid` anzeigen lassen.

```
uws@tux>sudo blkid
/dev/sda1: UUID:"89..." LABEL="root" TYPE="ext4"
/dev/sda2: UUID:"cd..." LABEL="mydata" TYPE="ext4"
/dev/sda3: TYPE="swap"  UUID="c1..."
```

Listing 8.79: Anzeige UUID)

Mit dem Parameter `-t` lassen sich die Einträge Filtern. An dem Parameter hängt man noch den Wert `TYPE`, `LABEL`, `UUID` oder `SEC_TYPE` an.

```
uws@tux>sudo blkid -t TYPE=ext4
/dev/sda1: UUID:"89..." LABEL="root" TYPE="ext4"
/dev/sda2: UUID:"cd..." LABEL="mydata" TYPE="ext4"
```

Listing 8.80: Anzeige Filtern)

Filter auf die Gerätedatei.

```
uws@tux>sudo blkid /dev/sda2
/dev/sda2: UUID:"cd..." LABEL="mydata" TYPE="ext4"
```

Listing 8.81: Filter Gerät)

Es gibt noch den Parameter `-o`. Hierbei wird als Wert `List`, `Value`, `device`, `udev` oder `full` mit angegeben.

```
uws@tux>sudo blkid -o list
```

Listing 8.82: List)

8.25.3 Mount Point

Man kann überall im System ein Device einbinden. Möchte man temporär ein Device einbinden, so kann das direkt unter `/mnt` gemacht werden. USB Geräte werden automatisch unter `/var/media` eingebunden.

8.25.4 FS-Type

In der nachfolgende Liste ist eine Auswahl der Dateisysteme angegeben.

Type	*Beschreibung*
auto	Automatisches Erkennen des Dateisystems
ext2/3/4	Ext-Dateisysteme
iso9660	Für CD und DVD
udf	DVD, hat iso9660 abgelöst
jfs	Journaled File System
nfs	NFS Share
ntfs/ntfs-3g	Windows NTFS Filesystem, ntfs-3g mit Schreibzugriff
reiserfs	Reiser FS
swap	Für Swap Partition
ramfs	Ramdisk
tmpfs	Ramdisk, nutzt auch den Swap-Platz
vfat	FAT 16/32
xfs	XFS
brtfs	BTRFS

Tabelle 8.18: Liste FS-Type

8.25.5 Options

Mehrere Optionen werden durch ein Kommata getrennt angegeben und ohne Leerzeichen.

Option	Beschreibung
defaults	Es wird rw, suid, dev, exec, auto, nouser und async gesetzt
dev/nodev	Gerätedateien werden interpretiert
exec/noexec	Ausführen von Dateien erlauben
gid	Setzten der Gruppen-ID, gid=users
uid	Setzen der User-ID, uid=username
suid/nosuid	Das SUID-Bit wird berücksichtigt
auto/noauto	Wird beim booten automatisch eingehängt
ro/rw	Setzen von Read only oder Read Write
sw	Das SWAP-Dateisystem verwenden
sync/async	Alles wird direkt auf dem Datenträger geschrieben. Für USB-Geräte ist sync sinnvoll und async fÃ¼r hohen Datenaufkommen
user/nouser	Einhängen dürfen alle User und Aushängen nur der User, der das Gerät eingehangen hat
users	Jeder darf das Gerät mounten
user,rw,umask=000	Für FAT-Systeme
noatime/nodiratime	Zugriffszeiten nicht in der Inodetabelle speichern. Wird fÃ¼r SSD empfohlen
errors=remount-ro	Bei einem Fehler wird das Gerät im Read only Modus engehangen
noauto,x-systemd.automount	Automatisches Einbinden durch Systemd bei einem ersten Zugriff. Kann verwendet werden bei Systemen, die beim Booten verzögern (NFS, SSHFS, ...)
discard	Wird bei SSD's für Trim verwendet
rsize	Max. bytes read
wsize	Max. bytes write
timeo	Zeit NFS Client wartet auf Anfrage
intr	Erlauben, NFS Anfragen zu unterbrechen, wenn der Server ausfällt

Tabelle 8.19: Optionen

8.25.6 Dump

Soll das Backup Programm `dump` das System sichern, so steht hier ein Wert `ungleich 0`.

8.25.7 Pass

Hier wird definiert, in welcher Reihenfolge die Datenträger überprüft werden sollen. Eine `0` steht für `keine Überprüfung`, eine `1` für als `erstes überprüfen` und eine `2` wird nach `1` überprüft.

8.25.8 Rebuild Fstab

Achtung: Wenn das System noch läuft, keinesfalls ausschalten oder ein Reboot machen. Am besten ist es, wenn man für das Format der fstab Datei eine andere zu Verfügung hat.
Auslesen der `/etc/mtab`.

```
uws@tux>sudo cat /etc/mtab >> /etc/fstab.rebuild
```

Listing 8.83: Mtab

Ist die `/etc/mtab` nicht vorhanden, so kann man die `/proc/mounts` auslesen.

```
uws@tux>sudo cat /proc/mounts >> /etc/fstab.rebuild
```

Listing 8.84: Mounts label

Nicht zu vergessen, die `Swap` Partition.

```
uws@tux>sudo cat /proc/swaps >> /etc/fstab.rebuild
```

Listing 8.85: Swap Partition

Die neu erstellt Datei kann nun editiert werden.

8.26 Fonts

8.26.1 Verzeichnisse

In den Verzeichnissen können Unterverzeichnisse angelegt werden, da alle Verzeichnisse durchsucht werden. In den Dateien `/etc/fonts/fonts.conf` und für `OpenSuSE` auch noch `/etc/fonts/suse-font-dirs.conf` sind die Verzeichnisse aufgelistet, die durchsucht werden sollen.
Folgende Verzeichnisse sind im Suchpfad:
/usr/share/fonts/<dir>
/usr/X11R6/lib/X11/fonts/<dir>
/opt/kde3/share/fonts/<dir>
/usr/local/share/fonts/<dir>
~.fonts/<dir>

8.27 Hostname

8.27.1 Setzen

Der Hostname steht in der Datei `/etc/hostname` oder auch `/etc/HOSTNAME`. Man kann zwar diese Datei editieren, aber einfacher geht das mit dem Befehl `hostname`.

```
uws@tux>sudo hostname tux1
uws@tux>sudo hostname -F /etc/hostname
#hostname
tux1
```

Listing 8.86: Hostname setzen

8.27.2 Abfragen

Ebenso kann man sich mit dem Befehl `hostname` den `Hostnamen, FQDN` und `Domain` anzeigen lassen.

```
uws@tux>hostname
tux

uws@tux>hostname -f
tux.seabaer.de

uws@tux>hostname -d
seabaer.de
```

Listing 8.87: Hostname abfragen

8.28 DD

Mit dd kann man ein Bit-genaues kopieren von Festplatten, Partitionen und Dateien durchführen. Das Kopieren funktioniert auch mit CD/DVD, aber nur mit Daten CD/DVD.
Achtung: dd wird sofort ausfegührt, es erfolgt keine Sicherheitsabfrage.
Hinweis: Bevor man einen Festplatte kopiert, sollte man sie vorher abhängen, damit während des auslesen keine neue Daten geschrieben werden.

8.28.1 Syntax

Die Syntax des Befehls dd lautet:
dd if=<source> of=<destination> <options>
Man benötigt Root-Rechte, wenn man Platten / Partitionen kopieren möchte. Bei CD/DVD braucht man immer Root-Rechte.
if = Input File z.B. /dev/sda
of = Output File z.B. /dev/sdb

8.28.2 Optionen

Die Angabe von Bytes und Blocks muss ganzzahlig sein. Ohne Angabe eines Suffix bei Bytes, wird die Zahl als Byte interpretiert.

Option	*Beschreibung*
status=progress	Zeigt den Fortschritt an
obs=Bytes	Es werden Blöcke in Bytes geschrieben
ibs=Bytes	Es werden Blöcke in Bytes gelesen
bs=Bytes	Es werden Blöcke in Bytes gelesen und geschrieben
count=Blocks	Gibt an, wie viele Blöcke gelesen / geschrieben werden
seek=Blocks	Gibt an, wie viele Blöcke am Anfang des Schreibvorgangs übersprungen werden sollen
skip=Blocks	Gibt an, wie viele Blöcke am Anfang gelesen übersprungen werden sollen

Tabelle 8.20: DD Optionen

Suffixe für Bytes / Blocks Das gleiche Muster gilt auch für TG, T, PB, P, EB, E, ZB, Z, YB,Y

Suffix	*Beschreibung*
KB	1000 Bytes
K	1024 Bytes
MB	1000000 Bytes (1000*1000)
M	1049576 Bytes (1024*1024)
GB	1000000000 Bytes (1000*1000*1000)
G	1073741824 Bytes (1024*1024*1024)

Tabelle 8.21: Suffix

8.28.3 Beispiele

Es werden die ersten zehn 1024 Byte großen Blöcke kopiert.

```
uws@tux>sudo dd if=/dev/sda1 Of=/dev/sdb1 bs=1K count=10
```

Listing 8.88: Beispiel 1

Es werden 2000 Byte große Blöcke kopiert, wobei beim Einlesen die ersten 50 Blöcke (50*2000=100000 Bytes) übersprungen werden, d.h. es wird erst bei Byte 100001 mit dem Lesen begonnen.

```
uws@tux>sudo dd if=/dev/sda1 Of=/dev/sdb1 ibs=2KB obs=2KB skip=50
```

Listing 8.89: Beispiel 2

Ein Backup von einer Festplatte und die erstellte Datei wird gezippt.

```
uws@tux>sudo dd if=/dev/sda bs=32K | gzip -9 > /backup/image.img.gz
```

Listing 8.90: Beispiel 3

Ein Restore von einem Image.

```
uws@tux>sudo cat /backup/image.img.gz |gunzip -d -c | dd of=/dev/sda bs=32K
```

Listing 8.91: Beispiel 4

Backup des Master Boot Records (MBR)

```
uws@tux>sudo dd if=/dev/sda Of=/backup/mbr.img bs=512 count=1
```

Listing 8.92: Beispiel 5

Restore eines MBR.

```
uws@tux>sudo dd if=/backup/mbr.img of=/dev/sda
```

Listing 8.93: Beispiel 6

8.29 Battery

8.29.1 Status ausgeben

Informationen über den Batterie Status kann man mit `upower` oder auch mit `acpi` abfragen.

```
uws@tux>upower -e # --enumerate
/org/freedesktop/UPower/devices/line_power_ac
/org/freedektop/UPower/devices/battery_BAT0

uws@tux>upower -i /org/freedektop/UPower/devices/battery_BAT0 # --show-info

uws@tux>sudo acpi -I -b

uws@tux>sudo ls /sys/class/power_supply/BAT0
```

Listing 8.94: Battery Status

8.30 Hardware Info

8.30.1 Hwinfo

Informationen über die Hardware kann man sich mit dem Befehl `hwinfo` sich anzeigen lassen. In dem nächsten Beispiel wird Informationen über die Wlan Karte abgefragt.

```
uws@tux>hwinfo --wlan --short
network:
   wlan0     Intel Centrino Advanced-N 6205 AGN
```

Listing 8.95: Wlan Karte Info

Hardware Items (Auszug):
All, bios, Bluetooth, bridge, camera, cdrom, chipcard, cpu, dist, dsl, joystick, keyboard, memory, modem, monitor, mouse, network, partition, pci, pcmcia, pppoe, printer, scanner, scsi, sound, usb, usb-ctrl, wlan, xen, zip.

8.30.2 Inxi

Bei Arch Linux, RHEL und Debian gibt es das Programm `inxi` für die Hardware Abfrage. Das Programm `inxi` ist ein Bash Script. Unter OpenSuSE kann man `inxi` nachinstallieren. Die Software gibt es unter: https://software.opensuse.org/download.html?project=utilities&package=inxi Unter OpenSuSE muss außerdem `sensors` und `xdpyinfo` auch installiert werden.

```
uws@tux>inxi -C
CPU: Quad core intel Core i7 860 (-HAT-MCP-) cache: 8192 KB
     Clock speeds: max: 2794 MHz 1: 1463 MHz 2: 1197 MHz ...
```

Listing 8.96: Inxi

Mit der Option `-xxx` wird eine erweiterte Ausgabe angezeigt. Das Wetter kann man mit `-w` sich anzeigen lassen. Nach der Option kann man `Ort,Staat` angeben.

```
uws@tux>inxi -W Berlin,Berlin
Weather: Conditions: 81 F (27 C) - Partly Cloudy Time: June 16, 04:06 AM

uws@tux># Top 10 Ram Verbrauch
uws@tux>inxi -c2 -t m10

uws@tux># Top 10 der Processe
uws@tux>inxi -c2 -t c10

uws@tux># Top 10 Ram und Processe
uws@tux>inxi -c2 -t cm10
```

Listing 8.97: Beispiele

Option	*Beschreibung*
-A	Show Audio
-b	Basic output
-B	Battery Data
-C	CPU output
-d	Optical Drive
-D	Hard Disk
-f	All CPU flags
-F	Full output INXI
-G	Graphic Card
-i	WAN IP
-I	Process, uptime, memory
-m	Memory (RAM)
-M	Machine Data
-n	Advanced Network Card
-N	Network Card
-o	Show unmounted Partition
-p	Full Partition Information
-P	Partition Information
-r	Distro repository data
-R	Show Raid Info
-s	Show Sensors (temperatur)
-S	Systeminformation
-u	Show Partitions UUID
-w	Weather, local
-W	Weather, location
-v	Detail 0-7, 7=> full detail
-c	Color Schema 1-32
-t	?

Tabelle 8.22: Inxi Optionen

8.31 Spracheinstellung

Die eingestellte Sprache kann man in der Variable `LANG` abfragen.

```
uws@tux>echo $LANG
de_de.UTF-8
```

Listing 8.98: Sprache

8.32 Benachrichtigungen

Möchte man eine Benachrichtigung ausgeben, die im Panel angezeigt wird, so kann man das mit `notify-send` erledigen. Ein Zeilenumbruch kann man mit `\n` vornehmen.

```
uws@tux>notify-send "<Ueberschrift" "<Hinweistext>" [-t <dauer>]

uws@tux>notify-send "Hinweis" "'cat /etc/passwd'" -t 5000
```

Listing 8.99: Benachrichtigung

Folgende Optinen gibt es:

Option	*Beschreibung*
-t,--expire-time	Angabe der Zeit in Millisekunden
-u,--urgency=	Dringlichkeitsstufe (low, normal, critical)
-i,--icon=	Angabe eines Icons

Tabelle 8.23: notify-send Optionen

8.33 Kernel

8.33.1 Arch Linux / Manjaro

Um einen neuen Kernel zu installieren, wird in der Shell der Befehl `mhwd-kernel` benutzt. In dem nachfolgenden Beispiel wird ein neuer Kernel installiert, ohne den alten Kernel zu löschen. Im Grub Menu kann man dann zwischen den verschiedenen Kernel auswählen.

```
uws@tux>sudo mhwd-kernel -i linux410 # install kernel version 4.10, for
   version 4.8 ist 48
```

Listing 8.100: Kernel installieren

Um den aktuellen Kernel bei der Installation zu entfernen, gibt man `rmc` (remove current) mit an.

```
uws@tux>sudo mhwd-kernel -i linux410 rmc
```

Listing 8.101: Kernel installieren und alten löschen

Einen installierten Kernel kann man auch nachträglich löschen.

```
uws@tux>sudo mhwd-kernel -r linux48
```

Listing 8.102: Kernel löschen

Eine Liste mit den verfügbaren Kernel kann man sich mit `-l` anzeigen lassen. Bei `-li` wird die Kernel aufgelistet, die auf dem System installiert sind.

```
uws@tux>sudo mhwd-kernel -l
available kernels:
   * linux41
   * linux410
   * linux411
   * linux44
   * linux48
   * linux49

root@tux>mhwd-kernel -li
Currnetly running: 4.4.70-1-MANJARO (linux44)
The following kernels are installed in your system:
   * linux44
```

Listing 8.103: List Kernel

Gibt es Probleme mit der `Initramfs`, so kann man die Dateien mit dem nachfolgenden Befehl neu erstellen lassen. Der nachfolgende Befehl erstellt die Dateien `/boot/initramfs-linux.img` und `/boot/initramfs-linux-fallback.img`.

```
uws@tux>sudo mkinitcpio -p linux44
```

Listing 8.104: Kernel Dateien erstellen

8.34 Chroot

Mit `chroot` ist es möglich, von z.B. einer Linux Live CD auf die `root` Umgebung des installierten Systems zugreifen zu können, um Wartungsarbeiten durchzuführen. Man kann auch z.B. dann Kennwörter ändern.

```
root@tux>mkdir -p /mnt/new/root

root@tux>mount /dev/sdxx /mnt/new/root

root@tux>cd /mnt/new/root
root@tux>mount -t proc proc proc/
root@tux>mount --rbind /sys sys/
root@tux>mount --rbind /dev dev/

root@tux>#optional
root@tux>mount --rbind /run run/

root@tux>chroot /mnt/new/root /usr/bin/bash
root@tux>#kommen folgende error Meldungen:
chroot: cannot run command '/usr/bin/bash': Execformat error
root@tux>#oder
chroot:'/usr/bin/bash': permission denied
root@tux>#dann remount mit exec und dann ein chroot
root@tux>mount -o remount,exec /mnt/new/root

root@tux>chroot /mnt/new/root /usr/bin/bash

root@tux>#optional: laden der bash config und setzen des Promps
root@tux>source /etc/profile
root@tux>source ~./bashrc
root@tux>export PS1="(chroot) $PS1"
```

Listing 8.105: Chroot

Wird in der Chroot-Umgebung ein Internet Zugang benötigt, so ist in der Datei `/etc/resolv.conf` ein DNS Server einzutragen.

```
root@tux>cat /etc/resolv.conf
# Generated by resolvconf
Nameserver 192.168.1.10
```

Listing 8.106: Internetverbindung

8.35 Samba

8.35.1 Disable SMBv1

Aus Sicherheitsgründen sollte die Samba Version 1 abgeschaltet werden. Hierzu wird in der Datei `smb.conf` folgendes eingefügt.

```
uws@tux>grep -i "min protocol" /etc/samba/smb.conf
min protokol = SMB2
```

```
uws@tux># Restart samba services
root@tux>systemctl restart smb
```

Listing 8.107: smb.conf

Unter Windows wird hierzu die Powershell aufgerufen und folgende zwei Befehle abgesetzt.

```
$>Set-SmbServerConfiguration -EnableSMB1Protocol $false -Force
$>Disable-WindowsOptionalFeature -Online -FeatureName smbb1protocol
```

Listing 8.108: Windows

8.36 Display Server

Möchte man wissen, welcher Display Server (Wayland oder X11) verwendet wird, so kann man sich das mit dem folgenden Befehl sich anzeigen lassen.

```
uws@tux>echo $XDG_SESSION_TYPE
X11
```

Listing 8.109: Display Server

8.37 Lock Screen

Wenn die Bildschirmsperre abgestürzt ist, muß man sich an einer Console anmelden und den nachfolgenden Befehl ausführen.

```
uws@tux>loginctl unlock-session 2
```

Listing 8.110: Unlock Screen

8.38 Wallpapers

Die Hintergrundbilder befinden sich in den folgenden Verzeichnissen.
Plasma (KDE):
/usr/share/wallpapers/<name>/contents/images
XFCE:
/usr/share/backgrounds
/usr/share/bitmaps/backgrounds

8.39 Mailversand mit Nullmailer

Mit dem Programm Nullmailer kann man Mail verschicken, aber keine Mails empfangen. Bei der Installation wird gegenbenen falls ein vorhandenes Mailsystem gelöscht.

8.39.1 Installation

Unter Debian und Arch Linux kann man Nullmailer folgendermaßen installieren. Bei der Installation werden einige Parameter abgefragt, die man später auch mit `dpkg-configure nullmailer` auf Debian Systeme machen kann.

```
uws@tux># Debian
uws@tux>sudo apt install nullmailer

uws@tux># Arch Linux
uws@tux>sudo pacman -S nullmailer
```

Listing 8.111: Installation

8.39.2 Konfiguration

Im Verzecihnis /etc/nullmailer gibt es folgende Konfigurations Dateien.

Datei	Beschreibung
defaultdomain	Die Domain wird angehangen, wenn der Hostname keinen Punkt enthält
defaulthost	Hier steht der Hostname drin, wird aber nur ausgelesen, wenn die Datei mailname leer ist
mailname	Hier szeht der Hostname drin und sollte nicht leer sein
adminaddr	Hier stehen die Mail Adressen drin, die an @localhost oder in mailname gerichtet sind
remotes	Hier wird der SMTP Server mit den Parametern beschrieben
allmailfrom	Ab Version > 1.13, überschreibt den vollständigen Absender

Tabelle 8.24: Konfiguration Dateien

Welche Parameter für SMTP gesetzt werden können, kann man sich mit /usr/lib/nullmailer/smtp -help anzeigen lassen. Nachfolgend sind Beispiele der Konfigurations Dateien. Nur die Dateien adminaddr und mailname können mehrzeilig sein.

```
uws@tux>cat /etc/nullmailer/defaultdomain
example.net

uws@tux>cat /etc/nullmailer/defaulthost
tux

uws@tux>cat /etc/nullmailer/mailname
tux

uws@tux>cat /etc/nullmailer/adminaddr
harry.hirsch@wald.de
paulchen.panther@comic.com

uws@tux>cat /etc/nullmailer/remotes
smtp.gmail.com smtp --port=465 --ssl --autj-login --user=nobody --pass=
   password
```

Listing 8.112: Beispiel Konfiguration

8.39.3 Send Mail

In dem nachfolgenden Beispiel wird eine Mail verschickt.

```
uws@tux>cat mail.txt | mail -s "Betreff: Mail 4 u" hans.wurst@web.de
```

Listing 8.113: Send Mail

8.40 Python

8.40.1 _atomic_removal is not defined

Kommt die Fehlermeldung _atomic_removal is not defined, wenn z.B. nach einer Installation die alten Dateien gelöscht werden sollen, so ist in der Datei /usr/lib/python3.5/weakref.py die Zeilen 109 und 117 abzuändern.

```
Line 109: def remove(wr, selfref=ref(self), _atomic_removal=
   _remove_dead_weakref):
Line 117: _atomic_removal(d, wr.key)
```

Listing 8.114: weakref.py

8.41 2 Factor authentication

8.41.1 Install

Für SSH z.B. kann man eine zwei Factor Authentifizierung einrichten. Hierzu wird das Paket `libpam-google-authenticator` installiert.

```
uws@tux>sudo pacmn -S libpam-google-authenticator
```

Listing 8.115: Installation

8.41.2 Konfiguration

In der Datei `/etc/pam.d/sshd` wird in der ersten Zeile folgendes eingetragen.

```
uws@tux>grep -i "google" /etc/pam.d/sshd
auth required pam_google_authenticator.so
```

Listing 8.116: Datei sshd

In der datei `/etc/ssh/sshd_config` wird der Parameter `ChallengeResponseAuthentication` auf yes gesetzt.

```
uws@tux>grep -i "ChallengeResponse" /etc/ssh/sshd_config
ChallengeResponseAuthentication yes
```

Listing 8.117: Datei sshd_config

Nun wird zum Abschluß der Secret Key File erstellt. Hierzu installiert man vorher `grencode`, um ein QR Code zu generieren,

```
uws@tux>sudo pacmn -S greencode

uws@tux>google-authenticator
.... Y
.... Y
.... Y
.... N
.... Y
```

Listing 8.118: Secret Key File

8.41.3 Generator

Auf dem Smartphone kann nun `FreeOtp` installiert werden, um dann den QR Code zu scannen. Es wird automatisch die Daten für die Verbindung erstellt.

8.42 XFCE

8.42.1 Whisker Menu

Installation

Um das Whisker Menu für die Panel Leiste zu aktivieren, muss das Paket `xfce4_whiskermenu_plugin` installiert werden. Vorher muss unter Debian noch ein Repository eingebunden werden.

```
uws@tux># Nur ausfuehren, wenn add-apt-repository nicht installiert ist
uws@tux>sudo apt install --yes software-properties-common

uws@tux># Repository hinzufuegen
uws@tux>sudo add-apt-repository ppa:gottcode/gcppa

uws@tux># Update Repository DB
uws@tux>sudo apt update

uws@tux># Install whisker Menu
uws@tux>sudo apt install --yes xfce4-whiskermenu-plugin
```

Listing 8.119: Install

Dann kann das Whisker Menu dem Panel hinzugefügt werden. Hierzu mit der rechten Maustaste auf das Panel klicken und Leiste | Neue Elemente hinzufügen. Das Whisker Menu auswählen und anschließend an die richtige Position verschieben. Danach kann man das alte Menu aus der Leiste entfernen.

Kapitel 9

SystemD

9.1 Startzeit ausgeben

Die Startzeit des Systems kann man sich mit `systemd-analyse` anzeigen lassen.

```
uws@tux>systemd-analyze
Startup finished in 4.560s(Kernel) + 1.363s(initrd) + 18.960s(userspace) =
   24.883s
```

Listing 9.1: Startzeit Info

Eine ausführlichere Anzeigen bekommt man, wenn man an dem Befehl den Parameter `blame` anhängt.

```
uws@tux>systemd-analyze blame
```

Listing 9.2: Ausführliche Startzeit

Die Ausgabe kann man sich auch als Graphik abspeichern und sich dann anzeigen lassen.

```
uws@tux>systemd-analyze plot > sysboot.svg
```

Listing 9.3: Startzeiten als Graphik

Protokollausgaben des Kernels sich anzeigen lassen. Wenn `Kernel` und `Initramfs` zusammen 15 Sekunden bei dem Start benötigen, ist alles in Ordnung.

```
uws@tux>dmesg --show-delta --color-always | less -R
```

Listing 9.4: Startzeiten

9.2 Service / Dienste

Services / Dienste können mit dem Befehl `systemctl` verwaltet werden.

9.2.1 Auflisten

Wird `systemctl` ohne Parameter aufgerufen, werden alle Services und Dienste aufgelistet. Wird als Parameter `--type` angegeben, so werden nur die Dienste angezeigt, welche als Wert übergeben worden sind.

```
uws@tux>systemctl

uws@tux># mit Filter
uws@tux>systemctl --type=service
```

Listing 9.5: Dienst auflisten

Alle verfügbaren Typen kann man sich mit dem Befehl `systemctl -t help` anzeigen lassen. Folgende Typen gibt es:

- service
- socket
- target
- device
- mount
- automount
- swap
- timer
- path
- slice
- scope

9.2.2 Dienste starten

```
uws@tux>sudo systemctl start <service_name>

uws@tux>sudo systemctl enable <service_name>
```

Listing 9.6: Dienst starten

9.2.3 Status

Den Status eines Service / Dienstes wird folgendermaßen abgefragt.

```
uws@tux>systemctl status network.service
  NetworkManager.service - Network Manager
  Loaded: loaded (/usr/lib/systemd/system/NetworkManager.service; enabled;
     vendor preset: disabled)
 Drop-In: /usr/lib/systemd/system/NetworkManager.service.d
          |-NetworkManager-ovs.conf
  Active: active (running) since Fri 2018-09-14 20:18:26 CEST; 48min ago
    Docs: man:NetworkManager(8)
Main PID: 1402 (NetworkManager)
   Tasks: 4 (limit: 4915)
  CGroup: /system.slice/NetworkManager.service
          |-1402 /usr/sbin/NetworkManager --no-daemon
          |-2123 /sbin/dhclient -d -q -sf /usr/lib/nm-dhcp-helper -pf /var/
             run/dhclient-wlan0.pid -lf /var/lib/NetworkManager/dhclient-86
             d45f4d-e672-4c04-b6e9-a7ae4115053d->
```

Listing 9.7: Dienst Status

9.3 Network

9.3.1 Allgemein

Mit dem SystemD Serverice `systemd-networkd` kann man das Netzwerk starten. Die Konfigurationsdateien befinden sich in dem Verzeichnis `/etc/systemd/network`. Alle hier abgelegten Dateien werden bei dem Start des Rechners ausgelesen.

9.3.2 Konfiguration

In der Konfigurationsdatei gibt es folgende Abschnitte: `Match`, `Link`, `Network`, `Address`, `Route` und `DHCP`. In den nachfolgenden Tabellen werden die Optionen für die einzelnen Abschnitte aufgelistet. In dem Abschnitt `Match` wird festgelegt, auf welchem Device die Konfiguration angewendet werden.
`[Match]`

Parameter	Beschreibung
Name	Name des Netzwerk Adapters. Mehrere Adapter können mit einem Leerzeichen angegeben werden. Ebenso sind Wildcards zulässig z.B. enp*
MACAdress	MAC Adresse des Netzwerk Adapters z.B. MACAdress=01:23:45:67:89:ab 00-11-22-33-44-55 AABB.CCDD.EEFF
Host	Hostname oder Maschinen ID
Virtualization	Überprüft, ob das Netzwerk in einer virtuellen Maschine läuft. z.B. Virtualization=false

Tabelle 9.1: Match

`[Link]`

Parameter	Beschreibung
MACAdress	Nüzlich für MAC-Adress-Spoofing
MTUBytes	Setzt einen MTU Wert, wenn Jumbo frames benutzt werden
Multicast	Erlaubt das Multicast auf dem Device

Tabelle 9.2: Link

`[Network]`

Parameter	Beschreibung
DHCP	DHCP aktivieren. False, true, ipv4 und ipv6 sind mögliche Werte
DHCPServer	Wenn aktiviert, DHCPv4 Server wird gestartet. False oder True
MulticastDNS	Aktiviert multicast DNS. False, true oder resolve sind möglich
DNSSEC	Aktiviert DNSSEC. False, true oder allow-downgrade
DNS	Angabe der DNS Server
Domains	Eine Liste der Domänen, die mithilfe der DNS-Server auf diesem Link aufgelöst werden sollen.
IPForward	Wenn aktiviert, alle eingehenden Pakete werden auf anderen Netzwerk devices weitergeleitet. False, true, ipv4 oder ipv6
IPv6PrivacyExtensions	Konfiguriert die Verwendung von stateless temporären Adressen, die sich im Laufe der Zeit ändern. False, true, prefer-public oder kernel

Tabelle 9.3: Network

`[Address]`

Parameter	Beschreibung
Address	Angabe der IP Adresse des Client. Diese Angabe muss gemacht werden, wenn DHCP ausgeschaltet ist.

Tabelle 9.4: Address

`[Route]`

Parameter	Beschreibung
Gateway	Angabe des Gateways, z.B. des Routers. Diese Angabe muss gemacht werden, wenn DHCP ausgeschaltet ist.
Destination	Obligatorisch das Zielpräfix der Route, möglicherweise gefolgt von einem Schrägstrich und der Präfixlänge

Tabelle 9.5: Route

Hinweis:
Man kann die Address und Gateway Angaben auch in dem Abschnitt `Network` machen. Wird dort nur die Addresse angegeben, so muss in dem Route Abschnitt das Gateway stehen.
`[DHCP]`

Parameter	Beschreibung
UseDNS	Steuert, ob die vom DHCP-Server angekündigten DNS-Server verwendet werden. True oder False
Anonymize	Wenn true, folgen die an den DHCP-Server gesendeten Optionen dem RFC7844 (Anonymitätsprofile für DHCP-Clients), um die Offenlegung identifizierender Informationen zu minimieren
UseDomains	Steuert, ob der vom DHCP-Server empfangene Domänenname als DNS-Suchdomäne verwendet wird. Wenn Route festgelegt ist, wird der vom DHCP-Server empfangene Domänenname nur zum Weiterleiten von DNS-Abfragen, nicht jedoch zum Suchen verwendet. Diese Option kann manchmal die Auflösung lokaler Namen bei Verwendung von systemd-resolved beheben

Tabelle 9.6: DHCP

```
uws@tux>cat /etc/systemd/network/20-static-lan.network
# Network configuration for systemd
# Valid values for each section predefined
# More infos:
# https://wiki.archlinux.org/index.php/Systemd-networkd
#
   ==============================================================================

[Match]
Name=enp4s0
#MACAddress=
#Host=
#Virtualization

[Link]
#MACAddress=
#MTUBytes=
#Multicast

[Network]
DNS=192.168.1.100
#DHCP=
#DHCPServer=
#MulticastDNS=
#DNSSEC=
#DOMAINS=
#IPForward=
#IPv6PrivacyExtensions=

[Address]
Address=192.168.1.152/24

[Route]
Gateway=192.168.1.101
#The destination prefix of the route, possible followed by a slash and the
   prefix length
#If Destination is not present in [Route] section this section is treated as
    a default route.
#Destination=

[DHCP]
#Bevorzugtes device, von klein nach gross
RouteMetric=10
#UseDNS=
#Anonymize=
#UseDomains=
```

Listing 9.8: Beispiel Network static

9.4 Timer - Cron Job

9.4.1 Allgemein

Mit SystemD kann man auch Cron Jobs erstellen, die dann zu einem definierten Zeitpunkt starten. Beispiel Scripte für `Timer` und `Services` kann man in den Verzeichnissen /lib/systemd/system anschauen. Ist das Verzeichnis nicht vorhanden, kann man unter /usr/lib/systemd/system nachschauen. Die selbst geschriebenen Scripte werden unter /etc/systemd/system abgelegt. Diese sind Systemweit. Scripte, die für alle User zu Verfügung stehen sollen, werden unter /etc/systemd/user abgelegt, die der User dann aktivieren kann. Eigene Timer werden unter /.config/systemd/user abgelegt. Damit man eigene Timer aktivieren kann, muss der Process /usr/lib/systemd/systemd --user laufen.
Ein Job wird immer durch zwei Dateien bestimmt. Einer `Service Unit` und einer `Timer` Datei. In der Service Unit wird das zu startende Programm definiert und in der Timer Datei wird die Startzeit konfiguriert.

9.4.2 Service Units

Eine Service Unit Datei erinnert stark an INI-Dateien unter Windows. Eine einfache Unit Datei besteht aus zwei Sektionen. In der ersten Sektion `[Unit]` wird der Dienst mit Description beschrieben. In der Sektion `[Service]` befinden sich die Einstellungen für das auszuführende Programm. Mit der Option `ExecStart` wird das zu startende Programm inklusive des Pfades angegeben. Nachfolgend ein Beispiel einer Unit Datei.

```
uws@tux>cat backup.service
[Unit]
Description=Backup des Home Verzeichnisses
[Service]
ExecStart=/usr/local/bin/backup.sh
```

Listing 9.9: Beispiel Service Unit

In der folgenden Tabelle werden die Optionen für die `Unit` Sektion aufgelistet. Dieses ist nur eine Übersicht und weitere Informationen erhält man mit der `man page (man systemd.unit)`. Die Informationen sind von https://wiki.archlinux.de/title/Systemd/Timers.

Optionen	Beschreibung
Description	Beschreibung der Unit, Pflichtangabe
Documentation	Eine URL kann hier angegeben werden
Requires	Erforderliche Unit
Requisite	Weitere Units können hier angegeben werden
Wants	Benötigte Unit
BindsTo	An der Unit gebunden, fällt sie aus, wir die Timer Unit nicht ausgeführt
PartOf	Beim Ausfall eines Service stoppt die Timer Unit
Conflicts	
Before	Start vor der Unit
After	Start nach der Unit
OnFailure	Ersatz Unit, die bei Ausfall genutzt werden soll
PropagatesReloadTo	Ähnelt OnFailure
ReloadPropagatedFrom	Ähnelt OnFailure
JoinsNamespaceOf	Nur angeben, wenn PrivateNetwork oder PrivateTmp gesetzt sind
RequiresMountFor	Checked, ob Devices auch verbunden (gemounted) sind
OnFailureJobMode	Legt das Verhalten für OnFailure fest. fail, replace (default), replace-irreversibly, isolte, flush, ignore-dependencies, ignore-requirements können angegeben werden
IgnoreOnIsolate	true/false, bei True wird nicht mehr gestoppt bei Isolierung einer anderen Unit
StopWhenUnneeded	true/false, bei True werden nicht mehr benötigte Units deaktiviert
RefuseManualStart	true/false, bei True wird eine User Eingabe verlangt, bevor die Unit aktiviert werden kann

Continued on next page

continued from previous page.

Optionen	Beschreibung
RefuseManualStop	true/false, bei True wird eine User Eingabe verlangt, bevor die Unit deaktiviert werden kann
AllowIsolate	true/false, um unbestimmte Systemzustände zu vermeiden
DefaultDependencies	true/false, bevor der Timer gestartet werden kann, muss das System komplett hochgefahren sein
JobTimeouSec	Stellt bei gestaffelten Jobs ein Time-out zu Verfügung
JobTimeoutAction	Stellt bei gestaffelten Jobs ein Time-out zu Verfügung
JobTimeoutRebootArgument	Stellt bei gestaffelten Jobs ein Time-out zu Verfügung
StartLimitAction	Ein Ersatzservice wird gestartet, wenn StartLimitintervalSec, StartLimitBurst überschritten wird
RebootArgument	Optionales Reboot

Tabelle 9.7: Optionen Unit Sektion

9.4.3 Timer

Um eine Service Unit zu starten, wird ein Timer benötigt. Die Timer Datei ähnelt im Aufbau der Service Unit. Anstelle der Sektion Service gibt es in der Timer Datei den Abschnitt `Timer`. Wenn der Parameter `Unit` leer bleibt, startet SystemD einen Service, der wie der Timer heißt.

```
uws@tux>cat backup.timer
[Unit]
Description=Daily backup
[Timer]
OnCalendar=18:00
OnUnitActiveSec=24h
Persistent=true
Unit=backup.service
[Install]
WantedBy=timers.target
```

Listing 9.10: Beispiel Timer

Optionen	Beschreibung
OnCalendar	Kalendarische Zeitanbage
OnBootSec	Zeitspanne seit dem Booten
OnActiveSec	Zeitspanne seit der Aktivierung der Timer Unit
OnStartupSec	Zeitspanne seit dem Start von systemd
OnUnitActiveSec	Zeitspanne seit der Timer den letzten Job ausgelöst hat
OnUnitinactiveSec	Zeitspanne der inaktivität seit Beendigung des letzten Jobs
AccuracySec	Zeitspanne zum ausführen der Unit, default ist 1 Minute
RandomizedDelaySec	Damit mehrere Timer nicht zu gleicher Zeit ausgeführt werden
Persistent	true, ein versäumter Job wird bei dem nächsten Start nachgeholt
WakeSystem	Das System aus dem suspend Mode aufwecken
RemainAfterElapse	true/false, hält den Timer aktiv

Tabelle 9.8: Optionen Timer Sektion

Die Angabe der Wochentage erfolgt in English und ist optional. Jeder Rubrik kann mit einem Komma (,), zwei Punkten (..) oder mit einem Stern (*) ersetzt werden. Nachfolgend einige Beispiele. Weitere Informationen mit `man systemd.time`.

```
# minutely, hourly, daily, monthly, weekly, yearly, quarterly, semiannually
OnCalendar=weekly
```

```
OnCalendar=Thu,Fri 2019-*-1..4 11:15:00

OnCalendar=*-*-* *:*:00

OnCalendar=*-01,04,07,10-01 00:00:00

OnBootSec=2d 1h 30m
```

Listing 9.11: Beispiele

Folgende Einheiten können benutzt werden. Ohne Angabe einer Einheit wird die Angabe als Sekunden gewertet.

- usec, us
- msec, ms
- seconds, second, sec, s
- minutes, minute, min, m
- hours, hour, hr, h
- days, day, d
- weeks, week, w
- months, month, M
- years, year, y

Weitere Informationen mit `man systemd.unit` und `man systemd.special`.

Optionen	Beschreibung
Alias	Alias Namen der Unit
WantedBy	Pflichtangabe. basic.target, bluetoth.target, getty.target, multi-user.target, printer.target, sockets.target
RequiredBy	entspricht WantedBy
Also	Installiert oder Deinstalliert andere Units bei Aktivierung dieser Unit

Tabelle 9.9: Optionen Install Sektion

Weitere Informationen mit `man systemd.service`.

Optionen	Beschreibung
ExecStart	Pflichtangabe, Angabe eines Befehls oder eines Scripts mit absoluter Pfadangabe
ExecStartPre	Der Befehl / Script wird vor ExecStart ausgeführt
ExecStartPost	Der Befehl / Script wird nach ExecStart ausgeführt
ExecReload	Rekonfiguration des Services
ExecStop	Wird benötigt, wenn der Service zum Beenden spezielle Befehle braucht
PermissionsStartOnly	false/true, bei True kann man ExecStart mit anderen Benutzerrechten ausführen lassen
TimeoutSec	Timeout für den Service
TimeoutStartSec	Timeout für den Service
TimeoutStopSec	Timeout für den Servcice
RuntimeMaxSec	Hier wird die Laufzeit des Service begrenzt, default ist unbegrenzt
WatchdogSec	Überwachung des Services

Continued on next page

continued from previous page.

Optionen	Beschreibung
Restart	on-success, on-failure, on-abnormal, on-watchdog, on-abort, always - Startet den Service neu
RestartSec	Wird für Restart benötigt

Tabelle 9.10: Optionen Service Sektion

9.4.4 Aktivieren

Nachdem die Service Unit und der dazugehörende Timer erstellt worden sind, müssen sie noch aktiviert und gestartet werden.

```
uws@tux>sudo systemctl enable backup.timer

uws@tux>sudo systemctl start backup.timer
```

Listing 9.12: Timer aktivieren

9.4.5 Timer Informationen

Eine Übersicht aller Timer werden mit `systemctl list-timers` angezeigt. Hier wird angezeigt, wann die Timer starten und wann sie das letzte mal gestartet worden sind. Den Output eines Services kann man sich mit `journalctl -u backup` anzeigen lassen.

```
uws@tux>sudo systemctl list-timers
NEXT                          LEFT          LAST
   PASSED    UNIT                         ACTIVATES
Wed 2019-10-02 00:00:00 CEST  5h 16min left Tue 2019-10-01 18:16:52 CEST  26
   min ago logrotate.timer              logrotate>
Wed 2019-10-02 00:00:00 CEST  5h 16min left Tue 2019-10-01 18:16:52 CEST  26
   min ago man-db.timer                 man-db.se>
Wed 2019-10-02 00:00:00 CEST  5h 16min left Tue 2019-10-01 18:16:52 CEST  26
   min ago shadow.timer                 shadow.se>
Wed 2019-10-02 18:31:53 CEST  23h left      Tue 2019-10-01 18:31:53 CEST  11
   min ago systemd-tmpfiles-clean.timer systemd-t>

4 timers listed.
Pass --all to see loaded but inactive timers, too.
```

Listing 9.13: Timer auflisten

9.4.6 User Timer

SystemD Timer funktionieren auch fÃ¼r normale User, falls die Distribution dieses zuläst. Die Service Units und Timer Dateien werden in dem Verzeichnis `~/.config/systemd/user` erstellt. Sollen hingegen alle User die Units und Timer aktivieren können, so müssen die Dateien in dem Verzeichnis `/etc/systed/user` abgelegt werden. Die Timer laufen normalerweise nur, wenn der User eingelogt ist. Diese Verhalten kann man mit `loginctl enable-linger <UserName>` ändern. An den Unit Dateien brauchen keine Änderungen gemacht werden, man muss nur bei dem `systemctl` Kommando den parameter `--user` mit angeben.

```
uws@tux>systemctl --user daemon-reload

uws@tux>systemctl --user enable backup.timer

uws@tux>systemctl --user start backup.timer
```

Listing 9.14: User Timer

9.5 Timesyncd - Systemzeit

Mit `timesyncd` kann man sich die Zeit von einem Zeitserver holen. Der timesyncd Daemon läuft mit minimale Privilegien. Bei jedem Abgleich der Systemzeit wird in dem Verzeichnis `/var/lib/systemd/timesync` die Datei `clock` geschrieben. Dieses kann bei einem Read only Verzeichnis / System zu Problemen führen.
Es gibt auch noch die Möglichkeit die Zeit mit den Programmen `ntp`oder auch `chrony` abzugleichen.

9.5.1 Konfiguration

In der Datei `/etc/systemd/timesyncd.conf` werden die Server für den Abgleich definiert. Ebenso können noch andere Parameter gesetzt werden, die für den Daemon benötigt werden.

```
uws@tux>sudo cat /etc/systemd/timesyncd.conf
#  This file is part of systemd.
#
#  systemd is free software; you can redistribute it and/or modify it
#  under the terms of the GNU Lesser General Public License as published by
#  the Free Software Foundation; either version 2.1 of the License, or
#  (at your option) any later version.
#
# Entries in this file show the compile time defaults.
# You can change settings by editing this file.
# Defaults can be restored by simply deleting this file.
#
# See timesyncd.conf(5) for details.

[Time]
NTP=0.pool.ntp.org 1.pool.ntp.org 2.pool.ntp.org
FallbackNTP=ntp0.fau.de ntp1.fau.de ntp1.freenet.de
#FallbackNTP=0.manjaro.pool.ntp.org 1.manjaro.pool.ntp.org 2.manjaro.pool.
   ntp.org 3.manjaro.pool.ntp.org
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
```

Listing 9.15: tymesyncd.conf

9.5.2 Aktivieren

Mit dem nachfolgenden Befehl wird der `timesyncd` Service aktiviert und gestartet.

```
uws@tux>sudo timedatectl set-ntp true
```

Listing 9.16: Aktivieren

9.5.3 Anzeigen

Um den Service zu überprüfen, gibt man an dem Befehl `timedatectl` den Parameter `status` mit an.

```
uws@tux>timedatectl status
               Local time: Di 2018-08-28 10:31:39 CEST
           Universal time: Di 2018-08-28 08:31:39 UTC
                 RTC time: Di 2018-08-28 08:31:39
                Time zone: Europe/Berlin (CEST, +0200)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

Listing 9.17: Status timesyncd

Eine erweiterte Ausgabe von Informationen erhält man mit `timesync-status`.

```
uws@tux>timedatectl timesync-status
       Server: 213.172.105.106 (0.pool.ntp.org)
Poll interval: 34min 8s (min: 32s; max 34min 8s)
         Leap: normal
      Version: 4
      Stratum: 2
    Reference: D5AC600E
    Precision: 1us (-20)
Root distance: 1.479ms (max: 5s)
       Offset: -999us
        Delay: 24.865ms
       Jitter: 1.416ms
 Packet count: 302
    Frequency: +15,726ppm
```

Listing 9.18: Erweiterte Ausgabe

Konfigurations Einstellungen erhält man mit dem Parameter `show-timesync --all`.

```
uws@tux>timedatectl show-timesync --all
LinkNTPServers=
SystemNTPServers=0.pool.ntp.org 1.pool.ntp.org 2.pool.ntp.org
FallbackNTPServers=ntp0.fau.de ntp1.fau.de ntp1.freenet.de
ServerName=0.pool.ntp.org
ServerAddress=213.172.105.106
RootDistanceMaxUSec=5s
PollIntervalMinUSec=32s
PollIntervalMaxUSec=34min 8s
PollIntervalUSec=34min 8s
NTPMessage={ Leap=0, Version=4, Mode=4, Stratum=2, Precision=-20, RootDelay
   =854us, RootDispersion=1.052ms, Reference=D5AC600E, OriginateTimestamp=
   Tue 2018-08-28 10:18:39 CEST, ReceiveTimestamp=Tue 2018-08-28 10:18:39
   CEST, TransmitTimestamp=Tue 2018-08-28 10:18:39 CEST,
   DestinationTimestamp=Tue 2018-08-28 10:18:39 CEST, Ignored=no PacketCount
   =302, Jitter=1.416ms }
Frequency=1030595
```

Listing 9.19: Konfiguration Info

9.6 Journalctl

Der nachfolger des Init-Systems `SysVinit` ist seit mehreren Jahren `Systemd`. Dabei wurde der Log Service `Rsyslog` durch das `Journal` von Systemd ersetzt. Dadurch gibt es auf den Systemd Maschinen keine `/var/log/messages` oder `/var/log/syslog` mehr. Die neuen Journal Dateien haben nun ein Binär Format. Bei der Anzeige des Journals kommt der Pager `Less` um Einsatz. Der Zugriff mit `journalctl` benötigt keine `Root-Rechte`.
Wird dem User die Gruppe `systemd-journal` zugewiesen, so kann er alle Informationen abrufen, ansonsten nur die Meldungen für den User.

9.6.1 Konfiguration

Mit der Konfigurationsdatei `/etc/systemd/journald.conf` kann man das Journal konfigurieren. Nachfolgend wird ein Teil des `journald.conf` angezeigt.

```
[Journal]
#Storage=auto
#Compress=yes
#Seal=yes
#SplitMode=uid
#SyncIntervalSec=5m
#RateLimitInterval=30s
#RateLimitBurst=1000
#SystemMaxUse=
#SystemKeepFree=
#SystemMaxFileSize=
#SystemMaxFiles=100
```

Listing 9.20: Auszug journald.conf

Variable	*Wert*	*Beschreibung*
Storage	persistent	Die Log-Dateien landen in /var/log/journal
Storage	volatite	Die Log-dateien gibt es nur bei laufender Sitzung,bei einem Reboot / Shutdown werden sie gelöscht
SystemMaxUse	2000M	Maximale Größe der Journal Dateien
SystemKeepFree	1000M	Wieviel Platz mindestens frei bleiben soll

Tabelle 9.11: journald.conf

Die Einheiten sind K,M,G und T.
Weitere Informationen gibt es mit man `journald.conf`.

9.6.2 Journal Dateien verkleinern

Die Journal Dateien können auf einer Größe zu Recht gestutzt werden oder alle Einträge die älter sind als die angegebene Zeit werden gelöscht. Nachfolgend einige Beispiele. Im ersten Beispiel werden die Journal Dateien auf 100MB verkleinert.

```
uws@tux>journalctl --vacuum-size=100M
```

Listing 9.21: Journal verkleinern(100MB)

Mit `--vacuum-time` kann man eine Zeitangabe machen. In dem nächsten Beispiel werden alle Einträge gelöscht, die älter als 1 Monat sind.

```
uws@tux>journalctl --vacuum-time=1month
```

Listing 9.22: Journal shrink Time

9.6.3 Syslog

Soll weiterhin das `Syslog` benutzt werden, so sind die Variablen `ForwardToSysLog=yes` und `MaxLevelSyslog=debug` zu setzten. Zusätzlich muss der `Rsyslog-Daemon` laufen.
Meldungen vom Syslog können mit Omjournal in Journald überführt werden.

9.6.4 Status und Überprüfung

Den Status des Daemons kann man mit `systemctl status` abfragen.

```
uws@tux>sudo systemctl status systemd-journald
 systemd-journald.service - Journal Service
  Loaded: loaded (/usr/lib/systemd/system/systemd-journald.service); static;
  Active: active (running) since Tue 2017-05-09 14:29:01 CEST; 6h ago
    Docs: man:system-journald.service(8)
     Man:journald.conf(5)
Main PID: 423 (system-journal)
  Status: "Processing requests..."
   Tasks: 1 (limit: 512)
.
.
```

Listing 9.23: Status eines Dienstes

Die Größe des Journals kann mit `--disk-usage` abgefragt werden.

```
uws@tux>sudo journalctl --disk-usage
Archived and active journals take up 656.1M on disk.
```

Listing 9.24: Größe Journal

Die Intrigität des Journals wird mit `--verify` durchgeführt.

```
uws@tux>sudo journalctl --verify
PASS: /var/log/a0848146a8854c698d28901e824/system@f5 ... .journal
```

Listing 9.25: Intrigität Journal

Ob die richtige Zeitzone eingestellt ist, kann man mit `timedatectl status` abgefragt werden. Mit `timedatectl set-timezone <zone>` kann man eine neue Zeitzone setzen.

```
uws@tux>sudo timedatectl status
root@tux>timedatectl status
      Local time: Tue 2017-05-09 20:37:07 CEST
  Universal time: Tue 2017-05-09 18:37:07 UTC
        RTC time: Tue 2017-05-09 20:37:07
       Time zone: Europe/Berlin (CEST, +0200)
 Network time on: no
NTP synchrinized: yes
 RTC in local TZ: yes
```

Listing 9.26: Zeitzone

9.6.5 Anzeigen Journal / Filtern

Wird `journalctl` ohne Optionen aufgerufen, so werden alle Einträge zur Laufzeit und dem letzten Reboot angezeigt. Bei jedem Neustart des Rechners wird die Zeile `-- Reboot --` angezeigt. Nur Error Meldungen kann man sich mit `-p err` anzeigen lassen. Es werden alle Meldungen angezeigt, nicht nur die letzten nach einem Reboot.

```
uws@tux>sudo journalctl -b -p err
-- Logs begin at Tue 2017-03-14 10:56:44 CET, end at Tue 2017-05-09 20:37
   ...
May 09 14:28:58 tux kernel: tpm_tis 00:01: A TPM error (7) occurred att ...
```

```
May 09 14:29_03 tux audspd[1040]: No plugins found, exiting
```

Listing 9.27: Boot Errors

Anstelle der Angabe `-p err`, kann man auch `-p 3` angeben.

Nummer	*Beschreibung*
0	emerg
1	alert
2	crit
3	err
4	warning
5	notice
6	info
7	debug

Tabelle 9.12: Error Code

Alle protokollierten Bootvorgänge, die sich im Journal befinden, können mit `--list-boots` angezeigt werden.

```
uws@tux>sudo journalctl --list-boots
-22 b76be92c47054d9ab8a2fld8e14ba35d Tue 2017-03-14 10:56:44 CET-Tue
   2017-...
-21 b508318148d245446ad90843f9400591 Fri 2017-03-17 12:07:21 CET-Fri
   2017-...
```

Listing 9.28: Alle Bootvorgänge

Mit der Option `-b` wird das Journal seit dem letzten Reboot angezeigt. Gibt man noch eine Zahl an, so werden dann diese Meldungen ausgegeben. Gibt man z.B. eine -1 an, so wird das vorletzte Journal angezeigt.

```
uws@tux>sudo journalctl -b -1
```

Listing 9.29: List last Boot

Nach einer Zeit oder auch Zeitraum kann man das Journal auch Filtern. Das Datum / Zeit Format ist grundsätzlich `YYYY-MM-DD HH:MM:SS`.

```
uws@tux>sudo journalctl --since "2017-04-13 11:15:22"

uws@tux>sudo journalctl --since "2017-04-13 11:15:22" --until "2017-04-15
   11:15:22"

uws@tux>sudo journalctl --since yesterday

uws@tux>sudo journalctl --since 10:00 --until "now"
```

Listing 9.30: Meldungen nach Datum

Nach Komponenten kann man das Journal auch filtern. In dem nächsten Beispiel wird nach dem Apache Service gefiltert.

```
uws@tux>sudo journalctl -o short-monotonic -u apache2.service -u php-fpm.
   service --since yesterday

uws@tux>sudo pidof apache2          # search pid, otherwise
uws@tux>sudo ps aux | grep apache2  # search pid
uws@tux>sudo journalctl PID=<pid>

uws@tux>sudo id -u www-data
uws@tux>sudo journalctl _PID=<UID> --since today
```

Listing 9.31: Meldungen nach Service

Alle möglichen Filter kann man sich mit man `system.journal-fileds` sich anzeigen lassen. Meldungen von Anwendungen können auch gefiltert werden, es muss nur der Name des Programms, incl der Pfadangabe angegeben werden.

```
uws@tux>sudo journalctl /usr/bin/vlc
```

Listing 9.32: Meldungen nach Programm

9.6.6 Kernel Meldungen

Kernel Meldungen konnte man früher mit `dmesg` sich anzeigen lassen. Nun wird die Option `-k` angegeben. Die Informationen von früheren Reboots können mit `-b -n` ausgegeben werden. Die Option `-n` bewirkt, das nur die angegebene Anzahl an Zeilen ausgegeben werden. Möchte man alle Meldungen von dem gewählten Reboot sehen, so lässt man die Option weg.

```
uws@tux>sudo journalctl -k

uws@tux>sudo journalctl -k -b -2 -n 10
```

Listing 9.33: Kernel Meldungen

Anzeigen der Kernel Meldungen, die das Initramfs absetzt.

```
uws@tux>journalctl -b -o short-monotonic
```

Listing 9.34: Kernel Meldung von Initramfs

9.6.7 Fortlaufende Ausgabe

Früher wurde eine fortlaufende Ausgabe von `dmesg` mit `tail -f [-n10]` gemacht. Nun genügt die Option `-f`, um das gleiche zu machen. Auch hier kann man mit `-n <zahl>` die anzuzeigende Zeilen angeben.

```
uws@tux>sudo journalctl -f -n 5
-- Logs begin at Tue 2017-03-14 10:56:44 CET. -
May 09 20:45:01 tux systemd[5508]: Stopped target Timers.
May 09 20:45:01 tux systemd[5508]: Received SIGRTMIN+24 from PID 5547 (kill)
    .
May 09 20:45:01 tux systemd[5512]: pam_unix(system-user:session): session...
May 09 20:45:01 tux systemd[1]: Stopped User Manager for UID 0.
May 09 20:45:01 tux systemd[1]: Removed slice User Slice of root.

uws@tux>sudo journalctl -u apache2 -f
```

Listing 9.35: Fortlaufende Anzeige

9.6.8 Ausgabe in Datei

Für die Ausgabe der Meldungen in einer Datei muss der Pager abgestellt werden.

```
uws@tux>sudo journalctl -b -p err --no-pager > MyJournal.txt
```

Listing 9.36: Ausgabe in Datei

Andere Ausgabe Formate können mit der Option `-o` oder auch `--output` angegeben werden. Die Standardausgabe heiät `short`. Mit `-o verbose` wird eine umfangreichere Ausgabe erstellt. Mit `-o cat` wird die Ausgabe gegenüber `short` nochmalig gekürzt. Eine Präzision des Zeitstempels wird mit `-o short-monotonic` gemacht. Eine Ausgabe im `Json` Format kann mit `-o json` oder besser mit `-o json-pretty` gemacht werden. Ein Export im Binär Format wird mit `-o export` durchgeführt.

Kapitel 10

Remote Verbindung

10.1 VNC Server

Wie immer gibt es verschiedene Möglichkeiten, den VNC Server einzurichten. Die Entscheidung, welche Möglichkeit zur Anwendung kommen soll, hängt von den Einsatzbedingungen ab.

10.1.1 Verbindung für alle

OpenSuSE

Es gibt zwei Möglichkeiten, allen Anwendern den Zugriff auf dem Computer zu erlauben. Dazu rufen wir das Programm `YAST` auf und unter Punkt Administration `von einem entfernten Rechner` wird der Punkt `Verwaltung via entfernten Rechner (remote) erlauben` aktiviert. Bevor wir den Befehl `rcxdm restart` ausführen, wird in der Datei `/etc/opt/kde3/share/config/kdm/kdmrc` unter dem Punkt `[Xdmcp]` die Variable `Enable` auf `true` gesetzt. Danach kann der Dienst `rcxdm` neu gestart werden. Der Computer ist nun auf den Ports 5901 (VNC) und 5801 (VNC Ã¼ber Browser) zu erreichen.
In der zweiten Möglichkeit kann man andere Auflösungen für VNC freischalten. Der Port 5x01 liefert die Auflösung 1024x768, 5x02 1280x1024 und 5x03 1600x1200. Hierzu rufen wir wieder das Programm `YAST` auf und unter `Netzwerkdienste - Netzwerkdienste (xinetd)` werden alle VNC Dienste (`vnc1, vnc2, vnc3, vnchttp1, ..`) aktiviert. Auch hier muss sie Datei `kdmrc` editiert und mit `rcxdm` neu gestartet werden.
Der VNC Server muss im Runlevel 5 laufen, da es sonst keinen laufenden X-Server gibt.

10.1.2 Verbindung für einen

Anstatt generell die Freigabe des Computers zu machen, kann auch jeder einzelne Benutzer den seinen Desktop freigeben. Hierbei wird der VNC Server von Hand gestartet. Nach der Eingabe des Befehls `vncserver` wird man nach einem Passwort gefragt. Dieses Passwort wird bei dem Verbindungsaufbau abgefragt. Bei dem erfolgreichen starten des Servers erhält man die Meldung, `'the new 'X' desktop is tux:5'`. Anschließend ist die Datei `/.env/xstartup` zu editieren. In der letzten Zeile steht der anzuzeigende Desktop. Standardmäßig ist der `twm` aktiviert. Möchte man einen anderen Desktop, so kann man für `KDE` den Wert `startkde`, für `Gnome` `gnome`, für `XFCE` dann `startxfce4` und bei `Openbox` `openbox`. Das `&` `Zeichen` am Ende darf nicht gelößcht werden.
Den VNC Server kann man mit dem Befehl `vncserver -kill tux:5` beenden.
Möchte man das starten des VNC Servers automatisieren, so ist der Aufruf in der `.profile` Datei vorzunehmen.

10.2 SSH

Eine Remote Verbindung mittels `SSH` ist einer `Telnet` Verbindung vorzuziehen, da im Gegensatz zu einer `Telnet` Session die Übertragung verschlüsselt durchgeführt wird. In der nachfolgenden Tabelle ist ein Auszug der Optionen für SSH (aus Linux User 11.2017).

Option	Beschreibung
-2	Nur Protokollversion 2
-4	Nur IPv4-Adressen
-6	Nur IPv6-Adressen
-b IP	IP-Adresse verwenden
-C	Kompression verwenden
-F Datei	Angabe alternative Konfigurationsdatei
-i Datei	Angabe der Private-Key Datei
-p Port	Angabe der Portnummer
-v	Ausgabe der Meldungen
-t	Interaktive Menüs
-w Rechner:Port	Weiterleiten der Standardein- und -ausgabe vom lokalen System zum entfernten System
-X	Display ungeschützt umleiten
-Y	Display geschützt umleiten
-N	Für Tunnelzwecke, kein Command

Tabelle 10.1: Optionen SSH

10.2.1 SSH Konfiguration

Die Konfiguration von SSH wird in der Datei `sshd_config` im Verzeichnis `/etc/ssh` vorgenommen. In der nachfolgenden Tabelle sind einige Optionen aufgelistet.

Schlüssel	Wert	Beschreibung
PermitRootLogin	no / yes	Root-Zugang erlauben
AllowGroups	Gruppe	Nur Benutzer in Gruppe dürfen sich anmelden.
AllowUsers	User	Nur Benutzer User darf sich anmelden
PasswordAuthentication	no / yes	Unterbindet das Anmelden mit einem Passwort
ChallengeResponseAuthentication	no / yes	Nur über einen Schlüssel anmelden
PermitEmptyPasswords	no / yes	Keine leeren Kennwörter erlauben
Port	Nummer	Abweichenden Port festlegen
ListenAddress	IP	Netzwerkschnittstelle benutzen, je IP-Adresse eine Zeile
MaxSessions		Auf kritischen Systemen so niedrig wie möglich (1)
MaxStartups		Auf kritischen Systemen so niedrig wie möglich (1)
ServerAliveInterval	Sekunden	Nach welcher Zeit der SSH Server ein Null Paket sendet

Tabelle 10.2: Optionen sshd_config

In der nachfolgenden Tabelle sind die wichtigsten Client Dateien für SSH aufgelistet. Sie befinden sich alle im Verzeichnis `~/.ssh`.

Name	Beschreibung
config	Benutzerdefinerte Einstellungen, überschreibt die Systemeinstellungen
known_hosts	Liste der Systeme, wo schon eine Verbindung aufgebaut wurde und der Schlüssel akzeptiert worden ist.
authorized_keys	Öffentlicher Schlüssel, wenn der Rechner als Server fungiert.
id_rsa	Privater Schlüssel
id_rsa.pub	Öffentlicher Schlüssel

Tabelle 10.3: Client Dateien

10.2.2 Root Login disable

In der Datei `/etc/ssh/sshd_config` ist der Eintrag `PermitRootLogin` auf `no` zu setzten, um das Root Login zu verbieten. Für `PermitRootLogin` kann man folgende Parameter setzen:

Parameter	Beschreibung
yes	Anmelden am System erlaubt
no	Anmelden am System nicht erlaubt
without-password	Das Anmelden ist ohne Passwort möglich
forced-commands-only	Anmelden mit public key möglich, aber nur in Verbindung mit der command Option.

Tabelle 10.4: Parameter PermitRootLogin

10.2.3 User Verbindungen

Der Eintrag `AllowUsers` in der Datei `/etc/ssh/sshd_config` listet alle zugelassenen User auf, die eine SSH Verbindung aufbauen dürfen.
Die Syntax für den Eintrag lautet: `AllowUsers <user1> <user2>`
Ebenso muss die Variable `PasswordAuthentication` auf `yes` gesetzt werden.

10.2.4 SSH Banner

Möchte man bei einer SSH Verbindung ein Banner anzeigen, so muss in der Datei `/etc/ssh/sshd_config` die Zeile `#Banner` auskommentiert und der Pfad zur Banner Datei angegeben werden.

```
uws@tux>grep Banner /etc/ssh/sshd_config
Banner /etc/ssh/sshd-banner

uws@tux>cat /etc/ssh/sshd-banner
**************************************************
WARNING:
========
Only authorized User to used this system.
Are you not authorized User, leave this session.
**************************************************
```

Listing 10.1: Beispiel Banner

10.2.5 SSH Dienst

Bei jeder Änderung an der SSH Konfiguration muss der Dienst neu gestartet werden.

```
root@tux>/etc/init.d/sshd restart

root@tux>service sshd restart

root@tux>#unter SystemD
root@tux>systemctl restart sshd
```

Listing 10.2: Start SSH Service

10.2.6 X11 Forwarding

Möchte man die Graphische Ausgabe auf seinem Rechner haben, so schaltet man auf dem Zielrechner das `X11Forwarding` ein. In der Datei `/etc/ssh/sshd_config` werden die Parameter `X11Forwarding yes`, `X11DisplayOffset 10` und `X11UseLocalhost yes` aktiviert.

```
uws@tux>ssh -X -C uws@tux01

uws@tux01>export DISPLAY='tux:10.0'

uws@tux01>xclock &
```

Listing 10.3: Beispiel X11 Forwarding

10.2.7 SSH Key erstellen

Einen Schlüssel kann man mit `ssh-keygen` erstellen. Es werden zwei Schlüssel erstellt, ein `Privater` und ein `Public` Schlüssel. Den Public Schlüssel kann an andere Remote Systeme gesendet werden. Lässt man bei der Passwort Abfrage das Passwort leer, so kann man sich später an dem Remote System ohne Passwort anmelden. Kann gut in Scripte verwendet werden.

```
uws@tux>ssh-keygen -t rsa -b 4096
```

Listing 10.4: Create SSH Key

-t => Typ des Schlüssels, hier RSA
-b => Länge des Schlüssels, hier 4096 Bits. Je länger, desto schwerer zu entschlüsseln.

10.2.8 SSH Key senden

Den erstellen Public Key Schlüssel kann man mit `ssh-copy-id` zu den Remote Systemen übertragen.

```
uws@tux>ssh-copy-id <user>@<hostname>
```

Listing 10.5: Send SSH Key

10.2.9 SSH Key entfernen

Hat sich der SSH Key eines Remote Zieles verändert, so kann man den SSH Key aus der Datei `known_host` mit dem folgenden Befehlt entfernen.

```
uws@tux>ssh-keygen -R tux10 -f /home/uws/.ssh/known_hosts
```

Listing 10.6: Delete SSH Key

10.2.10 Alias

In der Datei /.ssh/config können Verbindungen definiert werden, die dann mit einem Alias aufgerufen werden. Die config-Datei darf nur für den User bearbeitbar sein. Es muss unbedingt mit chmod 600 auf diese Datei gemacht werden, sonst kommt es zu einer Fehlermedung.

```
uws@tux>cat ~/.ssh/config
Host Remote1
 HostName 10.81.3.16
 User Carl
 CheckHostIP no

uws@tux>ssh Remote1
```

Listing 10.7: Beispiel Alias

10.2.11 TCP-Stealth

Wird in der Datei /etc/ssh/sshd_config die Option TCPStealthSecret auf dem Server / Client aktiviert, so klappt es mit der Verbindung. Ansonsten wird sie abgelehnt. Der eingetragene Wert ist ein Autorisierung-Token. Nur wenn beide gleich sind, wird eine Verbindung ausgebaut. Diesen Modus kann man nur einschalten, wenn auch der Kernel mit TCPStealthSecret kompiliert worden ist.

10.2.12 Match

Mit Match kann man Parameter gezielt setzen. Entweder fÃ¼r User, Gruppen, Computer, Ports, Adressen oder Local Adressen. Die Syntax eines Match Blocks ist:

```
Match <Condition> <Value>
    Parameter <new Value>
    Parameter <new Value>
```

Listing 10.8: Syntax Match

In dem Block kann man folgende Parameter setzen.

AcceptEnd	AllowAgentForwarding	AllowGroups
AllowStreamLocalForwarding	AllowTcpForwarding	AllowUsers
AuthenticationMethods	AuthorizedKeyCommand	AuthorizedKeyCommandUser
AuthorizedKeysFile	AuthorizedPrincipalsCommand	AuthorizedPrincipalsCommandUser
AuthorizedPrincipalsFile	Banner	ChrootDirectory
DenyGroups	DenyUsers	ForceCommand
GatewayPorts	GSSAPIAuthentication	HostbasedAcceptKeyTypes
HostbasedAuthentication	HostbasedUsesNameFromPacketOnly	IPQoS
KbdInteractiveAuthentication	KerberosAuthentication	MaxAuthTries
MaxSessions	PasswordAuthentication	PermitEmptyPasswords
PermitOpen	PermitRootLogin	PermitTTY
PermitTunnel	PermitUserRC	PubkeyAcceptedKeyTypes
PubkeyAuthentication	RekeyLimit	RevokedKeys
RhostsRSAAuthentication	RSAAuthentication	StreanLocalBindMask
StreamLocalBindUnlink	TrustedUserCAKeys	X11DisplayOffsett
X11Forwarding		

Tabelle 10.5: Parameter Match Block

Als Beispiel wird ein Anmelden als Root ohne Passwort erlaubt und ein User muss sich mit Kennwort anmelden.

```
PermitRootLogin no
PasswordAuthentication no
AllowUsers paul root

Match Address 192.168.70.0/24,192.168.8.0/24,127.0.0.1
      PermitRootLogin without-password
      PasswordAuthentication yes
```

Listing 10.9: Beispiel Match

10.2.13 Chroot für Sftp

Mit `ChrootDirectory` kann man wie in Linux eine Chroot Umgebung für Sftp-User einrichten. In dieser Umgebung können die Sftp-User nicht die anderen Verzeichnisse sehen und auch nicht dorthin wechseln.

```
root@tux># create group sftpuser
root@tux>groupadd sftuser

root@tux># create user paul
root@tux>useradd -d /sftp/paul -G sftpuser -s /bin/false paul

root@tux># create directories and permissions
root@tux>mkdir -p /sftp/paul
root@tux>chown -R root:root /sftp
root@tux>chmod -R 775 /sftp
root@tux>mkdir -p /sftp/paul/daten
root@tux>chown paul:sftuser /sftp/paul/daten
root@tux>chmod 775 /sftp/paul/daten

root@tux># edit /etc/ssh/sshd_config
# disable next
#Subsystem sftp /usr/lib/ssh/sftp-server

# new value
Subsystem sftp internal-sftp

# ChrootPermission root:root 775
# Directorys under Chroot, paul:sftpuser 775
Match User paul
      ChrootDirectory /sftp/paul
      X11Forwarding no
      AllowTcpForwarding no
      PermitTTY no
      Banner /etc/ssh/sftp.txt
      ForceCommand internal-sftp

root@tux># restart sshd service
root@tux>systemctl restart sshd
```

Listing 10.10: Beispiel SFTP Chroot

10.2.14 Login Mail

Möchte man eine Mail erhalten, wenn sich ein User mittels `ssh` auf dem Rechner angemeldet hat, so kann man das mit einem Script machen, das sich im Verzeichnis /etc/profile.d befindet. Voraussetzung ist ein Mail System wie der nullmailer. Damit identifiziert werden kann, ob das Login per ssh erfolgt ist, wird geprüft, ob die Variable `SSH_CLIENT` nicht leer ist.

```
root@tux>cat /etc/profile.d/ssh_mail.sh
#!/bin/env bash
#
# Prerequisites: nullmailer is installed and configured
# Copy this file to /etc/profile.d - thats it
#
MailFile=/tmp/ssh_mail.txt
ssh_ip=$(echo $SSH_CLIENT | awk '{print $1}')
ssh_user=$(who -m | awk '{print $1}')
ssh_console=$(who -m | awk '{print $2}')
ssh_login_date=$(who -m | awk '{print $3}')
ssh_login_time=$(who -m | awk '{print $4}')
ssh_port=$(echo $SSH_CLIENT | awk '{print $3}')

printf "SSH Login on $(hostname)\n\n" > $MailFile
printf "   User: $ssh_user\n" >> $MailFile
printf " SSH IP: $ssh_ip\n" >> $MailFile
printf "   Date: $ssh_login_date $ssh_login_time\n" >> $MailFile
printf "Console: $ssh_console\n" >> $MailFile
printf "   Port: $ssh_port\n" >> $MailFile

if [ -n "$SSH_CLIENT" ]
then
  cat $MailFile | mail -s "Login on $(hostname) from $ssh_user" uws@seabaer-
     ag.de
fi
```

Listing 10.11: Beispiel ssh_mail.sh

10.2.15 Error

Taucht in der /var/log/messages folgender Fehler auf: `sshd syslogin_perform_logout:logout() returned an error`, so ist die Datei `utmp` im Verzeichnis /var/run neu anzulegen.

```
root@tux>rm /var/run/utmp

root@tux>touch /var/run/utmp
```

Listing 10.12: Recreate utmp

10.2.16 Beispiele

```
uws@tux># Dateien packen und senden
uws@tux>tar -czvf - <path> | ssh user@host 'cd <path> && tar -xzpvf -'

uws@tux># Dateien holen und entpacken
uws@tux>ssh user@host 'cd <path> && tar -czvf - <path>' | tar -xzfv -
```

Listing 10.13: Beispiele

10.3 Displaymanager

10.3.1 Konfiguration

Die Variable `DISPLAYMANAGER_REMOTE_ACCESS` wird auf `yes` gesetzt, die sich in der Datei `/etc/sysconfig/displayman` befindet. Danach in der Datei `/etc/X11/xdm/Xaccess` das Kommentarzeichen aus der Zeile `* #any host can get a login` entfernen.

10.3.2 Dienst starten

Nach einer Änderung in der Datei `displaymanager` müssen folgende Schritte für OpenSuSE gemacht werden.

```
root@tux>SuSEconfig

root@tux>rcxdm restart
```

Listing 10.14: Reconfig

Bei Änderungen in der Datei `Xaccess` muss der Dienst neu gestartet werden.

```
root@tux>kdm Stopp

root@tux>kdm start

root@tux># SystemD Systeme
root@tux>systemctl restart kdm
```

Listing 10.15: Restart KDM Service

10.3.3 Root Login GDM

Standardmäßig kann man sich nicht mit dem User `root` an einem `Oracle Linux Server` anmelden (GUI). Damit man sich anmelden kann, muss in der Datei `/etc/pam.d/gdm` die folgende Zeile auskommentiert werden:

```
auth required pam_succeed_if.so user != root quit
```

Listing 10.16: Datei gdm

10.3.4 Anmelde Bildschirm

Damit man bei einer Remote Anmeldung (GUI) die Login Oberfläche angezeigt bekommt, fügt man in der Datei `/etc/gdm/custom.conf` im Abschnitt `[xdmcp]` den Eintrag `Enable=true` ein.

10.4 SCP

Um Dateien zwischen von einem Linux Rechner auf einem anderen Rechner zu kopieren, so gibt es hierzu den Befehl `scp`. In den nachfolgenden Beispielen werden zuerst Dateien von einem Rechner zu einem anderen kopiert. Im zweiten Beispiel wird eine Verbindung zu dem Zielrechner mit einem anderen Benutzername hergestellt. Und im letzten Beispiel werden Dateien von einem entfernten Rechner kopiert.

```
uws@tux>scp /home/uws/transfer/*.jpg tux01:/home/uws/transfer

uws@tux>scp /home/uws/transfer*.jpg jan@tux01:/home/jan/transfer

uws@tux>scp jan@tux01:/home/jan/bin/*.sh uws@tux:/home/uws/bin
```

Listing 10.17: Beispiele scp

In der nachfolgenden Tabelle sind einige Optionen Optionen für scp aufgelistet.

Option	Beschreibung
-C	Kompremieren
-F	Datei Alternative Konfigurationsdatei
-i	Datei Privater Schlüssel Datei
-l	Wert Bandbreite, Angabe in kbit/s
-o	Option SSH-Optionen
-P	Port Portangabe, Standard Port 22
-p	Dateiattribute übertragen
-q	Meldungen einschränken
-r	Rekursiv kopieren, Symlinks folgen
-v	Ausführliche Meldungen

Tabelle 10.6: Optionen scp

10.5 Remote Desktop

10.5.1 RDP - Arch Linux

Um eine Remote Desktop Verbindung mittels des Protokolls `rdp` herzustellen, so kann das mit dem Programm `xrdp` gemacht werden.

Installation

Das Paket liegt den meisten Distributionen für die Installation bereit. Für Arch Linux und deren Derivate wird das Paket aus dem AUR installiert.

```
uws@tux>yay -S xrdp

uws@tux>yay -S xorgxrdp-git
```

Listing 10.18: Installation

Nun muss der Dienst noch gestartet und eingerichtet werden.

```
uws@tux>sudo systemctl start xrdp

uws@tux>sudo systemctl start xrdp-sesman

uws@tux>sudo systemctl enable xrdp

uws@tux>sudo systemctl enable xrdp-sesman
```

Listing 10.19: Dienste

Konfiguration

In der Datei $HOME/.xinitrc muss die folgende Zeile abgeändert werden.

```
uws@tux>grep "local dbus" .xinitrc
  #local dbus_args=(--sh-syntax --exit-with-session)
  local dbus_args=(--sh-syntax)
```

Listing 10.20: xinitrc

Und zum Schluss muss die Datei `/etc/X11/Xwrapper.config` angelegt werden. In dieser Datei wird dann folgende Zeile eingetragen.

```
uws@tux>cat /etc/X11/Xwrapper.config
allowed_users=anybody
```

Listing 10.21: Xwrapper.config

10.5.2 RDP - Debian

Auch unter Debian kann man mit `xrdp` eine Remote Desktop Verbindung bereitstellen. Hierbei ist xrdp und tigervncserver zu installieren.

```
uws@tux>sudo apt install --yes tigervncserver xrdp

uws@tux>sudo systemctl enable xrdp
uws@tux>sudo systemctl start xrdp
```

Listing 10.22: Installation

Kapitel 11

Benutzer / Gruppen

11.1 Benutzer

11.1.1 Anlegen

Neue Benutzer werden mit dem Befehl `useradd` angelegt. Eine Auswahl der Optionen folgt in der nachfolgenden Liste. Die `default` Werte stehen in der Datei `useradd`, die sich im Verzeichnis `/etc/defaults` befindet.

Option	*Beschreibung*
-c	Einen Kommentar mit angeben.
-d	Angabe des Home Verzeichnisses.
-e	Datum, wann das Konto abläuft im Format YYYY-MM-DD.
-f	Nach wie vielen Tagen des inaktiven Kontos, nach Ablauf des Kennwortes, das Konto gesperrt werden soll.
-G	Angabe der Gruppen.
-g	Angabe der Hauptgruppe.
-m	Erstellen des Home Verzeichnisses.
-p	Passwort.
-r	Systemaccount.
-s	Angabe der Shell.
-u, --uid	Angabe der UUID, z.B. 500

Tabelle 11.1: Optionen

```
root@tux>useradd <Benutzername>

root@tux>useradd <Benutzername> -u 500
```

Listing 11.1: User Add

11.1.2 Ändern

Einen angelegten Benutzer kann man mit dem Befehl `usermod` bearbeiten. Hierbei gelten die gleichen Optionen wie bei dem Anlegen eines neuen Benutzers.

```
root@tux>usermod -c "Home User" <Benutzername>
```

Listing 11.2: Modify User

Gruppen können einem User mit `-G` zugewiesen werden. In der Kombination mit `-a` werden die angegebenen Gruppen dem User addiert.

```
root@tux>usermod -aG users,kvm,libvirt <Benutzername>
```

Listing 11.3: Add Group

Die UUID eines Users kann mit -u geändert werden.

```
root@tux>usermod -u <uuid> <Benutzername>
```

Listing 11.4: Modify UUID

11.1.3 Löschen

Benutzer werden mit dem Befehl userdel gelöscht. Gibt man die Option -r mit an, so wird das Home Verzeichnis mit gelöscht.

```
root@tux>userdel -r <Benutzername>
```

Listing 11.5: Delete User

11.1.4 Account sperren

Einen User Account kann man sperren, in dem man die Default Shell austauscht. Wird der Eintrag auf /usr/sbin/nologin gesetzt, ist ein Anmelden am dem System nicht mehr möglich.

```
uws@tux>grep pi /etc/passwd
pi:x:1000:1000:Raspbian Account:/home/pi:/usr/sbin/nologin
```

Listing 11.6: Nologin

Einen Account kann man auch mit passwd -l <UserName sperren. Mit passwd -u <UserName> kann man den Account wieder entsperren.

```
uws@tux># Account lock
uws@tux>sudo passwd -l pi

uws@tux># Account unlock
uws@tux>sudo passwd -u pi
```

Listing 11.7: Passwd

11.1.5 Anzeigen

Angelegte Benutzer kann man mit dem nachfolgenden Befehl sich anzeigen lassen. Mit $4 wird die GroupID ausgegeben.

```
uws@tux>awk -F ':' '{print $1}' /etc/passwd

uws@tux>awk -F ':' '{print $1 "-" $4}' /etc/passwd
```

Listing 11.8: List User

Für die UID und Gruppen des angemeldeten Users gibt es auch die Variablen UID und GROUPS.

```
uws@tux>echo $UID

uws@tux>echo $GROUPS
```

Listing 11.9: Show UID / GROUPS

SubsectionAngemeldete User In der folgenden Tabelle stehen die Befehle drin, die angemeldete User anzeigen.

11.1.6 Kennwort ändern

Das Kennwort kann man mit dem Befehl passwd ändern.

```
root@tux>passwd <BenutzerName>
Changing password for <Benutzer>
New Password:
Reenter password:
```

Listing 11.10: Change Passwort

Befehl	Beschreibung
w	Zeigt an, wer ist angemeldet und was sie tun.
who	Zeigt an, wer ist eingelogt.
last	Welche Benutzer waren angemeldet

Tabelle 11.2: Optionen

11.1.7 Benutzer Info's

Informationen über die Benutzer kann man mit dem Programm `finger` sich anzeigen lassen.

```
uws@tux>finger uws

Login: uws            Name: Uwe Schimanski
Directory: /home/uws Shell: /bin/bash
On since Fri Oct 14 12:26 (CEST) on pts/0
No Mail
No Plan
```

Listing 11.11: Info's User

11.1.8 Login Fehlversuche

Mit `faillog -a` werden alle fehlversuche für einen Login pro User angezeigt. Mit `faillog -u` <UserName> werden nur die fehlversuche des angegebenen Users angezeigt.

```
uws@tux>sudo faillog -a
Login        Fehlver. Maximum Letzter                    Auf

root             4        0   01/27/19 18:54:58 +0100   192.168.70.
bin              0        0   01/01/70 01:00:00 +0100
daemon           0        0   01/01/70 01:00:00 +0100
mail             0        0   01/01/70 01:00:00 +0100
ftp              0        0   01/01/70 01:00:00 +0100
http             0        0   01/01/70 01:00:00 +0100
uuidd            0        0   01/01/70 01:00:00 +0100
dbus             0        0   01/01/70 01:00:00 +0100
nobody           0        0   01/01/70 01:00:00 +0100

uws@tux>sudo faillog -u uws
Login        Fehlver. Maximum Letzter                    Auf

uws             22        0   02/20/19 08:39:08 +0100   109.40.64.8
```

Listing 11.12: Fehlversuche

11.2 Gruppen

11.2.1 Anlegen

Neue Gruppen lassen sich mit dem Befehl `groupadd` anlegen.

```
root@tux>groupadd <GruppenName>
```

Listing 11.13: Create Group

11.2.2 Ändern

Eine bestehende Gruppe kann man mit dem Befehl `groupmod` ändern. In der nachfolgenden Tabelle werden ein paar Optionen aufgelistet.

Option	Beschreibung
-g	Ändern der Group ID.
-A	Hinzufügen einen Benutzers für die Gruppe.
-R	Lößchen eines Benutzers aus der Gruppe.

Tabelle 11.3: Optionen

```
root@tux>groupmod -A <BenutzerName> <Gruppen>
```

Listing 11.14: Modify Group

11.2.3 Löschen

Eine Gruppen kann man mit dem Befehl `groupdel` löschen.

```
root@tux>groupdel <GruppenName>
```

Listing 11.15: Delete Group

11.2.4 Anzeigen

```
uws@tux>awk -F ':' '{print $1}' /etc/group
```

Listing 11.16: Delete Group

11.3 Logon Zeit begrenzen

Möchte man die Anmeldezeit für einen User festlegen, so kann man das mittels der `times.conf` im Verzeichnis `/etc/security` bewerkstelligen.

11.3.1 Voraussetzung

Das Paket `libpam-modules` muss installiert sein.

11.3.2 Konfiguration

Die Syntax in der `times.conf` ist:

```
<services>;<ttys>;<user>;<time> # Kommentar
```

Listing 11.17: Syntax times.conf

Beispiel:

```
Login;*;paul;MoFr1200-1700
```

Listing 11.18: Beispiel

Das Format für die Zeitangabe ist HHMM-HHMM.
Format für die Tage: Folgende Operatoren gibt es: Ein weiteres Beispiel:

```
Login;tty*&!pts*;du|ich;!al1000-2400
```

Listing 11.19: Beispiel

Die User `du` und `ich` dürfen sich an allen Tagen in der Zeit zwischen 10:00-24:00 nicht an den virtuellen Consolen anmelden. An Terminals (pty*) gilt die Sperre nicht.

Kürzel	*Tag*
Mo	Montag
Tu	Dienstag
We	Mittwoch
Th	Donnerstag
Fr	Freitag
Sa	Samstag
So	Sonntag
Wk	Wochentags
Wd	Wochenende
Al	Alle Tage
MoWk	Alle Wochentage, außer Montag
AlFr	Alle Tage, außer Freitag

Tabelle 11.4: Liste Tage

Operatoren	*Beschreibung*
!	Logisches nicht
&	Logisches und
|	Logisches oder
	Platzhalter für alles

Tabelle 11.5: Liste Operatoren

11.3.3 Aktivierung

Damit die `times.conf` auch aktiv wird, so muss das Zeitmodul `pam_time.so` in der Datei `common_auth` im Verzeichnis `/etc/pam.d` geladen werden.

```
root@tux>grep pam_time common_auth
account  required  pam_time.so
```

Listing 11.20: Common_auth

Kapitel 12

Drucken

12.1 CUPS (Common Unix Print System)

12.1.1 Konfiguration

Die Konfiguration von CUPS kann mit dem Programm `cupsctl` vorgenommen werden. Die Konfigurations Datei befindet sich im Verzeichnis `/etc/cups` und trägt den Namen`cupsd.conf`. Eine Hilfe bekommt man mit `man cupsd.conf` angezeigt.
Nach Änderung an der Konfiguration muss CUPS neu gestartet werden. In der nachfolgenden Tabelle sind einige der Konfigurartions Schalter aufgelistet.

Schalter	*Beschreibung*
--[no-]debug-logging	Das Logging ein- oder ausschalten.
--[no-]remote-admin	Remote Administration ein- oder ausschalten.
--[no-]remote-any	Internet Zugriff erlauben oder ausschalten.
--[no-]remote-printers	Remote Drucker anzeigen
--[no-]share-printers	Drucker freigeben.
--[no-]user-cancel-any	Benutzer dürfen Jobs löschen.
-E Veschlüsselung	einschalten
-u <username>	Benutzername definieren.
-h <server[:port]>	Server Adresse.

Tabelle 12.1: Schalter

12.1.2 Verwaltung - Browser

Die Verwaltung der Drucker unter CUPS kann man mit dem Browser durchführen. Die Verwaltungsseite wird mit der URL `http://localhost:631` aufgerufen. Diese Seite läst sich natürlich auch von einem anderen Rechner aus aufrufen.

12.1.3 Dienst starten

Der Dienst wird mit `cups restart` neu gestartet. Außer `restart` gibt es auch noch `stop` und `stop`.

```
root@tux>cups <restart> <stop> <start>
```

Listing 12.1: Synatx cups

12.1.4 Admin

Standardmäßig darf nur der erste angelegte Benutzer Cups Administrieren. Möchte man noch andere Benutzer die Administration erlauben, so sind diese Benutzer zu der Gruppe `lpadmin` hinzuzufügen.

```
root@tux>adduser <username> lpadmin
```

Listing 12.2: Add User to group

12.1.5 Druckdaten entfernen

Abgearbeitet Druckdaten kann man mit dem nachfolgenden Befehl entfernen.

```
root@tux>cupsctl AutoPurgeJobs=yes PreserveJobFiles=no PreserveJobHistory=no
```

Listing 12.3: Delete Print Job

12.1.6 Servernamen Drucker

Soll bei jedem Drucker auch der Servernamen mit angezeigt werden, so gibt man folgendes ein.

```
uws@tux>cupsctl BrowseShortNames=no
```

Listing 12.4: Modify Cups

Kapitel 13

Programme

13.1 Kalender in der Shell

In der Bash kann man sich einen Kalender anzeigen lassen mit dem Befehl `cal`. Cal reicht zurück bis zum 1. Januar 0001.

```
uws@tux>cal
June 2012
Su Mo Tu We Th Fr Sa
1 2
3 4 5 6 7 8 9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
```

Listing 13.1: Beispiel cal

Gibt man dem Befehl `cal` noch den Parameter `-y` auf dem Weg, so wird das aktuelle Jahr ausgegeben. Mit `cal 06 2005` wird der Juni 2005 ausgegeben. Die Angabe `cal -3` gibt den aktuellen Monat mit dem vorherigen und nachfolgenden Monat aus. Bei dem Parameter `-j` werden die Tage von 1 bis 365 gezählt und ausgegeben.

13.2 Screenfetch

Mit dem Programm `Screenfetch` kann man sich in der Console eine Übersicht seines Systems anzeigen lassen.

```
root@tux>zypper install screenfetch

root@tux>pacman -Sy screenfetch
```

Listing 13.2: Installation

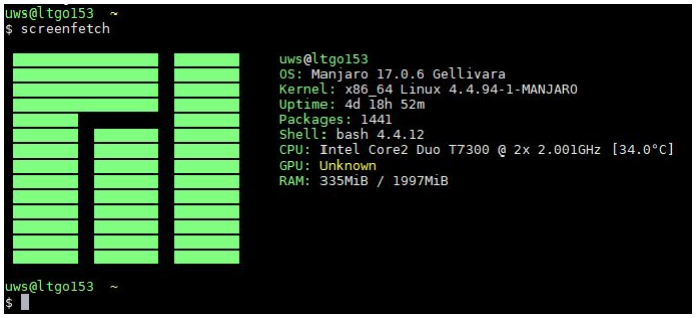

Abbildung 13.1: Screenfetch

13.3 Neofetch

Als alternative zu `Screenfetch` gibt es auch das Programm `neofetch`. Der Unterschied zu Screenfetch ist, das man Neofetch auch konfigurieren kann. Die Konfiguration befindet sich im Verzeichnis /.config/neofetch.

Abbildung 13.2: Neofetch

13.4 Merge PDF Dateien

Sollen mehrere PDF Dateien zu einer PDF Datei zusammen gefügt werden, so kann das mit `pdfunite` gemacht werden.

```
uws@tux>pdfunite in1.pdf in2.pdf in3.pdf out.pdf
```

Listing 13.3: Merge PDF-Dateien

13.5 Firefox

13.5.1 Speicherbedarf

Hat man im Browser mehrere Tabs offen, so steigt der RAM Verbrauch stark an. Auf Systemen, die nicht genug Hauptspeicher haben, wird dann Speicher in den Swap Bereich ausgelagert und das System läuft dann schleppend. Damit der RAM Verbrauch optimiert wird, werden in der Config drei Parameter gesetzt. Der erste Parameter regelt die Anzahl der Webseiten im Cache, die durch den Vor-/Zurückbutton angezeigt werden. Der zweite Parameter regelt den Speicher, den alle offenen Taby belegen dürfen.

```
about:config

browser.sessionhistory.max_entries = 5
browser.sessionhistory.max_total_viewers = 3
```

Listing 13.4: Parameter

Wert	*Beschreibung*
-1	Automatik
0	Es werden keine Seiten im RAM gespeichert
1	32MB
2	128MB
5	256MB
8	1/2/4GB

Tabelle 13.1: Werte Viewers

Bei dem letzen Parameter kann man einstellen, das beim Minimieren der nicht mehr benötigte Speicher sofort frei gegeben werden soll. Ist der Eintrag nicht vorhanden, so wird ein Boolean Eintrag angelegt.

```
about:config

config.trim_on_minimize = true
```

Listing 13.5: Speicher

13.6 Conky - Desktop aufmotzen

13.6.1 Allgemein

Mit dem Programm `conky` kann man sich auf dem Desktop einen Systemmonitor erstellen. Conky ist in fast allen Distributionen enthalten und kann mit dem jeweiligen Paketmanager installiert werden. Auch RSS-Feeds lassen sich mit conky auf dem Desktop darstellen. Es gibt über 250 Objekte und Fähigkeiten, die das Programm mitbringt. Ab der Conky Version 1.10 hat sich die Syntax geändert und auch einige Variablen sind entfallen. Für die Konfigurationsdatei wird nun die LUA-Syntax verwendet.

13.6.2 Conky ausführen

Ruft man `conky` ohne einen Parameter auf, so wird die Konfiguration aus dem Verzeichnis /etc/conky/-conky.conf verwendet. Unter Arch Linux befindet sich die Datei unter /usr/share/doc/conky-1.10.8_pre. Befindet sich im Home Verzeichnis auch eine .conkyrc Datei, so wird diese ausgewertet. Mit dem Parameter `-c` und der Angabe des Pfades und des Datei Namens wird dann diese Datei ausgewertet.

```
uws@tux>conky -c ~/.config/conky/.conkyrc

uws@tux># Kopiert die Standard Konfig in das Verzeichnis
uws@tux>conky -C ~/.config/conky/conky.conf
```

Listing 13.6: Conky ausführen

13.6.3 Konfiguration

In der Konfigurationsdatei gibt es zwei Abschnitte. In dem ersten Abschnitt `conky.config` werden die Grundsätzlichen Einstellungen vorgenommen und in dem Abschnitt `conky.text` wird definiert, was angezeigt werden soll.

```
-- Kommentare werden mit zwei - eingeleitet
conky.config = {
    alignment = 'bottom_right',
    background = false,
    ...
};
conky.text = [[
    ${scroll 16 $nodename - $kernel}
    $hr
    ${color grey}Uptime:$color $uptime
    ...
]]
```

Listing 13.7: Abschnitte Beispiel

Welche variablen es gibt, kann man unter `conky.sourceforge.net/variables.html` sich anzeigen lassen.

13.6.4 Definitionen

Das Aussehen und die Position des Conky Fensters werden in dem ersten Abschnitt definiert. Größenangabe erfolgen in Pixeln Angaben. Hier ein paar Definitionen. Was man dann auf dem Desktop ausgeben möchte, wird dann im Abschnitt `conky.text` festgelegt. Hier wird dann auch die Position der Elemente definiert. Die Angaben stehen dann zwischen den `[[ und ]]`. Nachfolgend eine kurze Liste der Variablen, die als Standard zu Verfügung stehen. Hier ein kleines Beispiel aus der C'T 18.26.

```
congy.text = [[
${scroll 26 $nodename - $sysname $kernel on $machine | }
$hr
${color grey}RAM Usage:$color $mem/$memmax - $memperc% ${membar 4}
${color grey}Swap Usage:$color $swap/$swapmax - $swapperc% ${swapbar 4}
${color grey}CPU Usage:$color $cpu% ${cpubar 4}
```

Definition	*Beschreibung*
alignment = 'top_right'	Angabe der Position des Fensters
gap_y = -10	Verschieben des Conky Fensters nach oben oder unten
gap_x = 5	Verschieben des Conky Fensters nach rechts oder links
minimum_height = 100	Minimum der Höhe
minimum_width = 40	Minimum der Breite
xinerama_head = 1	Auf welchem Monitor soll die Ausgabe erscheinen bei mehr als einen Bildschirm
update_interval = 1.0	Aktualisierungszeit, die Angabe erfolgt in Sekunden. 1.0 ist ein guter Wert
own_window_argb_visual = true	Transparente Conky Anzeige einschalten
own_window_argb_value = 0	Bei 0 komplett durchsichtig und höhere Zahlen durchsichtigkeit abnehmend
double_buffer = true	Bei flackernder Bildschirm Ausgabe einschalten
font = 'DejaVu Sans Mono:size=12'	Schriftart setzen, alle Linux Schriften können verwendet werden
draw_borders = true	Rahmen um das Fenster darstellen
border_width = 3	Stärke des Rahmens
stippled_borders = 1	Gepunktete Umrandung
border_inner_margin = 5	Abstand zum Rand, Angabe in Pixel

Tabelle 13.2: Definitionen

Variable	*Beschreibung*
$nodename	Hostname
$kernel	Kernel
$memperc%	Speicherverbrauch in Prozent
$swap	Swap Größe
$swapmax	Maximale Swap Größe
$swapperc	Aktueller Swap Verbrauch in Prozent
$cpu%	Prozessorlast in Prozent
$fs_used	Belegten Platz der Festplatte
$fs_size	Größe der Festplatte
$hr	Horizontale Linie
scroll	Ausgabe als Laufband

Tabelle 13.3: Variablen

```
$hr
${color grey}File Systems: / ${color${fs_used /\/${fs_size /} ${fs_bar 6 /}
```

Listing 13.8: Beispiel

Ein Bild oder auch ein Logo kann man mit `image` einbinden. Die zwei Zahlen hinter dem Parameter `-p` sind die Werte für den Abstand des Bildes in Pixel x und y.

```
${image .config/conky/MyLogo.jpg -p 170,25}
```

Listing 13.9: Bild einfügen

13.6.5 RSS Feed einbinden

Auch RSS Feeds kann man mit Conky ausgeben. In dem nächsten Beispiel werden die Feeds alle 15 Minuten aktualisiert und es werden 10 Feeds ausgegeben.

```
${rss http://www.heise.de/newsticker/heise.rdf feed_title}
${rss http://www.heise.de/newsticker/heise.rdf item_title 1}
```

Variable	*Beschreibung*
feed_title	Titel des Feeds
item_title 1	Titel des aktuellsten Feed
item_titles 10	Anzeige von 10 Feeds
item_desc 1	Ausgabe der Beschreibung
maximum_width	Maximal Breite der Ausgabe, der Rest wird abgeschnitten

Tabelle 13.4: RSS Feed

```
${rss http://www.heise.de/newsticker/heise.rdf item_desc 1}
${rss http://www.heise.de/newsticker/heise.rdf 15 item_titles 10}
```

Listing 13.10: Beispiel RSS Feed

13.6.6 Scripte ausführen

13.7 Youtube-dl - Video / Musik

Mit dem Programm `youtube-dl` kann man die Videos von Youtube downloaden oder auch aus den Videos die Audio Spur extrahieren.

13.7.1 Konfiguration

Es gibt eine Systemweite Konfigurationsdatei und eine benutzerspezifische. Die Systemweite Datei `youtube-dl.conf` befindet sich im Verzeichnis /etc und die benutzerspezifische unter /.config/youtube-dl/config. Man kann auch verschiedene Konfigurationen erstellen und diese dann mit dem Parameter `--config-location` verwenden. Mit dem Parameter `--ignore-config` werden die Konfigurationsdateien nicht berücksichtigt.

```
uws@tux>cat audio.conf
# Location und entfernen der Youtube-ID aus dem Namen
-o "~/Music/%(title)s.%(ext)s"
# Nur Audio download
-x
# Audioformat
--audio-format mp3
# Audioqualitaet
--audio-quality 320
# Incl. ID3-Tags
--add-metadata
# Ignoriert Fehler in Playlisten
-i
# Vorhandene nicht ueberschreiben
-w
```

Listing 13.11: Beispiel Konfiguration

```
uws@tux>youtube-dl --config-location ~/.config/youtube-dl/config/video.conf
   youtube-url
```

Listing 13.12: Beispiel alternative Konfiguration

Man kann sich auch verschiedene Aliase anlegen mit den Parametern.

```
uws@tux>alias yta="youtube-dl -x --audio-format mp3 --audio-quality 320 --
   add-metadata -o '/daten/music/%(title)s320.%(ext)s'"
```

Listing 13.13: Beispiel Alias

13.7.2 Video Formate

Wird youtube-dl mit dem Parameter `-F` aufgerufen, so werden alle verfügbaren Formate des Videos ausgegeben und mit der Angabe des Format Codes wird dann das Video heruntergeladen.

```
uws@tux>youtube-dl -F  https://www.youtube.com/watch?v=uekZpkYf7-E
[youtube] uekZpkYf7-E: Downloading webpage
[youtube] uekZpkYf7-E: Downloading video info webpage
[info] Available formats for uekZpkYf7-E:
format code  extension  resolution note
249          webm       audio only DASH audio   57k , opus @ 50k, 2.02MiB
250          webm       audio only DASH audio   76k , opus @ 70k, 2.68MiB
140          m4a        audio only DASH audio  128k , m4a_dash container,
   mp4a.40.2@128k, 5.01MiB
171          webm       audio only DASH audio  137k , vorbis@128k, 4.73MiB
251          webm       audio only DASH audio  146k , opus @160k, 5.28MiB
278          webm       256x144    144p   96k , webm container, vp9, 24fps,
   video only, 3.66MiB
```

```
160          mp4        256x144    144p  109k , avc1.4d400c, 24fps, video
   only, 3.48MiB
242          webm       426x240    240p  222k , vp9, 24fps, video only, 7.15
   MiB
133          mp4        426x240    240p  353k , avc1.4d4015, 24fps, video
   only, 8.28MiB
243          webm       640x360    360p  419k , vp9, 24fps, video only,
   14.05MiB
244          webm       854x480    480p  762k , vp9, 24fps, video only,
   25.83MiB
134          mp4        640x360    360p  768k , avc1.4d401e, 24fps, video
   only, 18.12MiB
135          mp4        854x480    480p 1218k , avc1.4d401e, 24fps, video
   only, 30.53MiB
247          webm       1280x720   720p 1558k , vp9, 24fps, video only,
   44.72MiB
136          mp4        1280x720   720p 1860k , avc1.4d401f, 24fps, video
   only, 48.42MiB
248          webm       1920x1080  1080p 2716k , vp9, 24fps, video only,
   68.04MiB
137          mp4        1920x1080  1080p 3188k , avc1.640028, 24fps, video
   only, 69.19MiB
17           3gp        176x144    small , mp4v.20.3, mp4a.40.2@ 24k, 3.08
   MiB
36           3gp        320x180    small , mp4v.20.3, mp4a.40.2, 8.60MiB
18           mp4        640x360    medium , avc1.42001E, mp4a.40.2@ 96k,
   25.05MiB (best)

uws@tux>youtube-dl -f 137 https://www.youtube.com/watch?v=uekZpkYf7-E
[youtube] uekZpkYf7-E: Downloading webpage
[youtube] uekZpkYf7-E: Downloading video info webpage
[download] Destination: Severija - Zu Asche, Zu Staub (Psycho Nikoros) - (
   Official Babylon Berlin O.S.T.)-uekZpkYf7-E.mp4
[download] 100% of 69.19MiB in 00:40
```

Listing 13.14: Anzeige Video Formate

13.7.3 Audio extract vom Video

In dem nächsten Beispiel wird die Audio Spur aus einem Video extrahiert und als MP3 Datei gespeichert.

```
uws@tux>youtube-dl --extract-audio --audio-format mp3 --audio-quality 512k
   https://www.youtube.com/watch?v=uekZpkYf7-E
[youtube] uekZpkYf7-E: Downloading webpage
[youtube] uekZpkYf7-E: Downloading video info webpage
[youtube] uekZpkYf7-E: Downloading js player vfl07ioI6
[download] Destination: Severija - Zu Asche, Zu Staub (Psycho Nikoros) - (
   Official Babylon Berlin O.S.T.)-uekZpkYf7-E.webm
[download] 100% of 5.28MiB in 00:03
[ffmpeg] Destination: Severija - Zu Asche, Zu Staub (Psycho Nikoros) - (
   Official Babylon Berlin O.S.T.)-uekZpkYf7-E.mp3
Deleting original file Severija - Zu Asche, Zu Staub (Psycho Nikoros) - (
   Official Babylon Berlin O.S.T.)-uekZpkYf7-E.webm (pass -k to keep)
```

Listing 13.15: Audio extract

Kapitel 14

Virtualisierung

14.1 KVM (Kernel-based Virtual Machine)

14.1.1 Voraussetzungen

Die CPU muss die Virtualisierung unterstützen. Hierzu kann man die Eigenschaften der CPU abfragen. Intel CPU's müssen die Option `vmx` unterstützen und AMD CPU's `svm`.

```
root@tux>egrep '(vmx|svm)' /proc/cpuinfo
```

Listing 14.1: Abfrage CPU Info

Zusätzlich kann man nachschauen, ob die Kernel Module geladen worden sind.

```
root@tux>lsmod | grep kvm

kvm_intel   241664  0
kvm         704512  1  kvm_intel
irqbypass    16384  1  kvm
```

Listing 14.2: Kernel Module

14.1.2 Installation KVM

Außer dem Paket `KVM` werden noch andere nützliche Programme für das erstellen und verwalten der Virtuellen Maschinen benötigt. Für `RedHat` Systeme lautet der Befehl für die Installation `yum install`. Nach der Installation sollte ein Neustart des Systems erfolgen.
Folgende Pakete sind für KVM nützlich:

- qemu-kvm
- qemu-system-x86 => Für Intel X86 Systeme
- qemu-system => Für andere Architekturen (PowerPC, ARM, Alpha, MIPS, Sparc, OpenRisc)
- virtinst
- libvirt-daemon-system
- libvirt-clients
- libvirt-viewer
- libvirt-manager => GUI für KVM
- bridge-utils
- genisoimage

- libosinfo-bin
- openbsd-netcat => Zugriff per SSH

```
root@tux>zypper install kvm

root@tux>zypper install qemu-kvm python-virtinst libvirt libvirt-python virt
   -mamanger libguestfs-tools
```

Listing 14.3: Installation

Ob die Installation erfolgreich war, kann man mit dem folgenden Befehl überprüfen.

```
uws@tux># User mit session, root mit system
uws@tux>virsh -c qemu:///session
Willkommen bei virsh, dem interaktiven Virtualisierungsterminal.

Tippen Sie:  'help' fuer eine Hilfe zu den Befehlen
       'quit' zum Beenden

virsh #
```

Listing 14.4: Check Install

14.1.3 User

Damit auch ein normaler User Virtuelle Maschinen erstellen und verwalten kann, muss er Mitglied in den Gruppen `libvirt`, `libvirt-qemu`, `qemu`, `kvm` sein. In den Distributionen sind nicht alle Gruppen vorhanden. In Arch Linux sind es zum Beispiel `kvm` und `libvirt`. Unter Debian sind es die Gruppen `kvm`, `libvirt`, `libvirt-qemu`.

14.1.4 Datastore

Einen Datastore kann man mit einer XML-Datei erstellen.

```
uws@tux>cat Datastore.xml
<pool type='dir'>
 <name>VmstoreSSD</name>
 <uuid>1c2b3c2d-0140-1e49-a571-840f4c88210d</uuid>
 <capacity unit='bytes'>0</capacity>
 <allocation unit='bytes'>0</allocation>
 <available unit='bytes'>0</available>
 <source>
 </source>
 <target>
  <path>/vmstore/machines</path>
  <permissions>
   <mode>0755</mode>
   <owner>-1</owner>
   <group>-1</group>
  </permissions>
 </target>
</pool>

uws@tux># Create Datastore
uws@tux>sudo virsh pool-create /vmstore/config/datastore/Datastore.xml
Pool VmstoreSSD aus Datastore.xml erstellt
```

Listing 14.5: Datastore

14.1.5 Network

Nach der Installation wurde standardmäßig eine Netzwerk Konfiguration mit dem Namen `default` erstellt. Dieses kann man in der Verwaltungskonsole sich anzeigen lassen.

```
uws@tux>virsh -c qemu:///system
Willkommen bei virsh, dem interaktiven Virtualisierungsterminal.

Tippen Sie:  'help' fuer eine Hilfe zu den Befehlen
'quit' zum Beenden

virsh #net-list --all
Name       Status Automatischer Start Bleibend
-------------------------------------------
default    Aktiv  yes                  yes
```

Listing 14.6: Show Netzwerk

Sollte die `default` Konfiguration nicht gestartet sein, so kann man sie folgendermaßen einschalten.

```
uws@tux>virsh net-start default
Network default started
```

Listing 14.7: Einschalten Netzwerk

Soll ein Netzwerk automatisch gestartet werden, so macht man das mit `net-autostart <NetworkName>`. Für das Abschalten gibt man noch die Option `--disable` mit an.

```
uws@tux>virsh -c qemu:///system
Willkommen bei virsh, dem interaktiven Virrtualisierungsterminal.

Tippen Sie:  'help' fur eine Hilfe zu den Befehlen
'quit' zum Beenden

virsh #net-autostart default

virsh #net-autostart default --disable
```

Listing 14.8: Autostart Netzwerk

Um eine neue Konfiguration zu erstellen, muss man von einem Template (Vorlage) die neue Konfiguration erstellen. Das bedeutet, das vor dem erstellen in Virsh die Datei vorhanden sein muss. Hierzu können wir das default.xml aus dem Verzeichnis /etc/libvirt/qemu/networks verwenden. Wir kopieren es und anschließend ändern wir es ab. Bei dem erstellen der neuen Netzwerk Konfiguration ist die Angabe von `--file` optional.

```
uws@tux>sudo cp /etc/libvirt/qemu/networks/default.xml /etc/libvirt/qemu/
   networks/myNetwork.xml

uws@tux>sudo vi /etc/libvirt/qemu/networks/myNetwork.xml

uws@tux>virsh -c qemu:///system
Willkommen bei virsh, dem interaktiven Virtualisierungsterminal.

Tippen Sie:  'help' fuer eine Hilfe zu den Befehlen
'quit' zum Beenden

virsh #net create --file /etc/libvirt/qemu/networks/myNetwork.xml
```

Listing 14.9: Create Netzwerk

Die erstellten Virtuellen Maschinen haben Standard mäßig nur Netzzugriff auf die anderen Virtuellen Maschinen und zur Host Maschine. Sollen die Virtuellen Maschinen auch Zugriff auf das VLAN haben, so muss eine Netzwerk Brücke eingerichtet werden.
Hierzu wird in der Konfigurationsdatei, z.B. für `eth0`, die Zeile `BRIDGE=br0` eingefügt und wenn dort eine IP-Adresse eingetragen ist, wird diese auskommentiert.
Nun wird eine Konfigurationsdatei `ifcfg-br0` für die Netzwerk Brücke eingerichtet.

```
root@tux># For RedHat and SuSE. in Debian /etc/network/interfaces
root@tux>cat /etc/sysconfig/network/ifcfg-br0
DEVICE="br0"
BOOTPROTO="static"
IPADDR="xxx.xxx.xxx.xxx"
NETMASK="255.255.25.0"
ONBOOT="yes"
TYPE="Briodge"
NM_CONTROLLED="no"
```

Listing 14.10: Einrichten Netzwerk Brücke

In der Datei `/etc/sysctl.conf` wird das `IP forwarding` aktiviert. Danach muss ein Neustart der Maschine gemacht werden.

```
root@tux>grep "ipv3" systemctl.conf
inet.ipv3.ip_forward=1
```

Listing 14.11: IP forwarding

14.1.6 Maschine Location

Standard mäßig werden die Maschinen im Verzeichnis `/var/lib/libvirt/images` erstellt / abgelegt. Es sollte also ausreichend Platz vorhanden sein.

14.1.7 Create Disk

Eine Disk für die Virtuelle Maschine kann man mit `qemu_img create` erstellen. In der Tabelle sind die Formate aufgeführt, die man erstellen kann.

Formate	*Beschreibung*
raw	Standard Format. Verfügt aber nicht über spezielle Funktionen wie Komprmierung, Snapshot usw. Performance mäßig am schnellsten
qcow2	KVM Format mit den speziellen Funktionen
qed	Ein Qemu Format. Die Leistung ist besser als von Qcow2
qcow	Ist der Vorgänger von qcow2
vmdk	Wird von VmWare verwendet
vdi	Wird von Virtual Box verwendet
vpc	Ist die erste Generation von Virtual PC

Tabelle 14.1: Disk Formate

```
uws@tux>qemu-img create -f qcow2 <Path2Disk>/sda.qcow2 40G
Formatting 'sda.qcow2', fmt=qcow2 size=42949672960 cluster_size=65536
   preallocation=full lazy_refcounts=0ff refcount_bits=16
```

Listing 14.12: Create Disk

14.1.8 Info Disk

Informationen über eine Disk erhält man mit `qemu-img info`.

```
uws@tux>qemu-img info <Path2Disk>/sda.qcow2
mage: sda.qcow2
file format: qcow2
virtual size: 40G (42949672960 bytes)
disk size: 40G
cluster_size: 65536
Format specific information:
```

```
    compat: 1.1
    lazy refcounts: false
    refcount bits: 16
    corrupt: false
```

Listing 14.13: Info Disk

14.1.9 Create Maschine

Für `OpenSuSE` lautet der Befehl für das Erstellen einer Maschine `vm-install` und für `RedHat` Systeme `virt-install`. Nachfolgend ein Beispiel, wie unter RedHat eine Maschine erstellt wird.

```
root@tux>virt-install \
-n myLinux \
--description "My own Linux Machine" \
--os-type=Linux \
--os-variante=debian7 \
--ram=4096 \
--vcpus=2 \
--disk path=/vmstore/machine/myLinux/myLinux.img,bus=virtio,size=20 \
--graphics none \
--cdrom /vmstore/images/debian7_x86_64.iso \
--network bridge:br0
```

Listing 14.14: Erstellen einer Maschine

Option	*Beschreibung*
-n	Name der Maschine
--description	Beschreibung der Maschine
--os-type	Linux, Solaris, Unix oder Windows
--os-variante	Distribution type. Z.B. rhel7, centos6, debian7, suse13, win2k, win2k8, win7, win8
--ram	Memory in MB
--vcpus	Anzahl Virtuelle CPU
--disk	Lage der Virtuellen Maschine und Größe der Festplatte in GB
--graphics	None für die Benutzung einer Text console auf dem VM serial Port, anstelle der Graphischen VNC Windows. Ist der xmanager aktiv, so kann der Parameter ignoriert werden.
--cdrom	Installations Medium
--network	Angabe des Netzwerk Devices

Tabelle 14.2: Beschreibung

14.1.10 Konfiguration

Die Konfiguration der virtuellen Maschinen wird in dem Verzeichnis `/etc/libvirt/qemu` abgelegt. Mit dem Befehl `virsh edit <VmName>` kann die Konfigurationsdatei bearbeitet werden.

```
root@tux>virsh edit myLinux
```

Listing 14.15: Edit Maschine

14.1.11 Maschinen anzeigen

Mit dem Befehl `virsh list -all` kann man sich die Virtuellen Maschinen anzeigen lassen.

```
root@tux>virsh list --all
ID Name State
----------------------------
1 myLinux running
```

Listing 14.16: Auflistung Maschinen

Weitere Informationen über die virtuelle Maschine können mit `virsh dominfo` abgefragt werden.

```
root@tux>virsh dominfo myLinux
Id:             1
Name:           myLinux
UUID:           58083ae7-37dr-46c2-63d9-bd5dc49f642
OS Type:        Linux
State:          running
CPU(s):         2
CPU time:       305.4s
Max memory:     2097152 KiB
Used memory:    1032452 KiB
Persitent:      yes
Autostart:      disable
Manged save:    no
Security model: selinux
Security DOI:   0
Security Label: system_u:system_r:virt_t:s0:c968,c799 (permissive)
```

Listing 14.17: Auflistung Maschinen

Die CPU und Memory Belegung kann man sich mit `virt-top` anzeigen lassen.

14.1.12 VM Console

Anmelden an der virtuellen Maschine kann mit dem Kommando `virsh console` gemacht werden. Die Console kann dann mit `ctrl` + wieder verlassen.

```
root@tux>virsh console myLinux
```

Listing 14.18: Verbinden Maschine

14.1.13 Shutdown, Reboot und Start

```
root@tux>virsh shutdown myLinux

root@tux>virsh reboot myLinux

root@tux>virsh start myLinux
```

Listing 14.19: Verbinden Maschine

14.1.14 Convert Disk

Eine Disk kann man in einem anderen Format konvertieren. Die Unterstützten Formate kann man sich mit dem folgenden Befehl sich anzeigen lassen.

```
root@tux>qemu-img -h | tail -n1
```

Listing 14.20: Auflistung Formate

OVA-Dateien sind nichts anderes als Tar-Archive.

```
root@tux>tar -xvf <ova_file>
```

Listing 14.21: Entpacken OVA-Datei

Mit `qemu-img convert` kann man das Ausgangsformat konvertieren.

```
root@tux>qemu-img convert -o qcowl <vmdk.file> output sdb.qcow2
```

Listing 14.22: Convert VMDK-Datei

14.1.15 Resize Disk

Eine Disk wird mit `qemu-img resize` vergröß ert. Im nachfolgenden Beispiel wird die Disk um 10GB vergröß ert.

```
root@tux>qemu-img resize <disk_image> +10G
```

Listing 14.23: Resize Disk

14.1.16 Video RAM

Möchte man den Speicher für die Video Card ändern, so sucht man in der Konfiguration nach der folgenden Zeile und ändert den Wert bei `vgamem`.

```
root@tux>virsh edit <MachineName>
 <model type="qxl" ram="131072" vram="131072" vgamem=16384" heads="1"
    primary="yes"/>
```

Listing 14.24: Change Video RAM

14.1.17 Huge Pages

Um `Huge Pages` für die Virtuellen Maschinen zu Verfügung zu stellen, sind folgende Schritte zu machen. Als erstes wird in der Datei `/etc/fstab` die nachfolgende Zeile eingefügt.

```
root@tux>grep huge /etc/fstab
hugetlbfs /dev/hugepages hugetlbfs mode=1770,gid=78 0 0
```

Listing 14.25: Edit fstab

Nun mounten wir die `hugepages` und kontrollieren das erfolgreiche verbinden.

```
root@tux>mount /dev/hugepages

root@tux>mount | grep huge
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,mode=1770,gid=78)
```

Listing 14.26: Mount Hugepages

Nun schauen wir nach, wie groß die Hugepages sind.

```
root@tux>grep Hugepagesize /proc/meminfo
Hugepagesize: 2048kB
```

Listing 14.27: Größs e Hugepages

Die Anzahl der Hugepages wird wie folgt berechnet: `RAM der virtuellen Maschine / 2`. Der ermittelte Wert wird dann aufgerundet. Beispiel: `2GB RAM => 2048/2=1024  1050`. Dieser Wert wird nun eingetragen und kontrollieren es. Laufen mehrere Maschine auf dem Host, so zählt man den RAM von allen Maschinen zusammen.

```
root@tux>echo 1050 > /proc/sys/vm/nr\_hugepages

root@tux>grep HugePages\_Total /proc/meminfo
HugePages\_Total: 1050
```

Listing 14.28: Check Hugepages

Nun wird die Maschine mit der Option `mem-path` gestartet.

```
root@tux>qemu-system-x86_64 -enable-kvm -m 2048 -mem-path /dev/hugepages -
hda <disk> ..

root@tux>grep HugePages /proc/meminfo
```

Listing 14.29: Start Maschine

Für das Dauerhafte einrichten der Huge Pages, wird in der Datei `/etc/sysctl.d/40-hugepages.conf` die folgende Zeile eingefügt.
Hinweis: Nicht für Arch Linux Systeme.

```
root@tux>cat /etc/sysctl.d/40-hugepages.conf
vm.nr_hugepages = 1050
```

Listing 14.30: Enable Hugepages

In der `/etc/default/qemu-kvm` wird der Wert von `KVM_HUGEPAGES` von `0` auf `1` gestellt und anschließs end der Libvirt Dienst neu gestartet.
Hinweis: Gilt nur für Debian oder Red Hat Systeme.

```
root@tux>grep KVM\_HUGEPAGES /etc/default/qemu-kvm
KVM\_HUGEPAGES = 1 # 0 = disable, 1 = enable

root@tux>systemctl restart libvirt-bin

root@tux># or
root@tux>service libvirt-bin restart
```

Listing 14.31: Neustart Service

In der Konfiguration der Virtuellen Maschinen wir der nachfolgende Abschnitt eingefügt.

```
root@tux>virsh edit <machine>
<memoryBacking>
  <hugepages/>
</memoryBacking>
```

Listing 14.32: Konfiguration Maschine

14.2 Troubleshooting (KVM)

14.2.1 Failed to start network default

Bei dem starten des Netzwerkes für die VM's kommt es zu einer Fehlermeldung.

```
root@tux>virsh net-start default
error: Failed to start network default
error: internal error: Failed to initialize a valid firewall backend
```

Listing 14.33: Error Meldung

Unter Arch Linux (Manjaro) überprüfen wir die Firewall.

```
root@tux>systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
  loaded: Loaded (/usr/lib/system/firewalld.service;enabled;Vendorpreset:
     disabled)
  Active: active (running) since Thu 2017-03-09 19:15:39 CET;40s ago
    Docs: man:firewalld(1)
Main PID: 590 (firewalld)
   Tasks: 2 (limit 4915)
  CGroup: /system.slice/firewalld.service
          590 /usr/bin/python -ES /usr/bin/firewalld -nofork -nopid

root@tux>pacman -Syu ebtables dnsmasq

root@tux>systemctl restart libvirtd
```

Listing 14.34: Check Firewall

Sollte keine Firewall installiert sein, so installiert man die Firewall nach.

```
root@tux>pacman -Syu firewalld
```

Listing 14.35: Installation Firewall

14.3 Virtual Box

14.3.1 Konfiguration

Mit der Variable VBOX_USER_HOME kann man die Ablage der Konfiguration an einer zentralen Stelle ablegen, wo alle Benutzer Zugriff darauf haben. Normalerweise werden die erstellten Maschinen nur für den jeweiligen Benutzer angezeigt, der die Maschinen erstellt hat, da die Konfiguration standardmäß ig im Home Verzeichnis abgelegt wird.

```
uws@tux>echo $VBOX_USER_HOME
/vm/configuration
```

Listing 14.36: Setzen Variable

14.3.2 Verwaltung Console

Um virtuelle Maschinen auf der Console zu verwalten, gibt es das Programm VBoxManage.

```
uws@tux># Start VM's
uws@tux>VBoxSDL --startvm "Windows 7" # Alternative zu VBoxManage
uws@tux>VBoxManage startvm "Windows 7"

uws@tux># List VM's
uws@tux>vboxmanage list vms

uws@tux># Info ueber VM's
uws@tux>VBoxManage showinfo "Windows 7"

uws@tux># Control
uws@tux>VBoxManage controlvm "Windows 7" pause
uws@tux>VBoxManage controlvm "Windows 7" resume
uws@tux>VBoxManage controlvm "Windows 7" reset
uws@tux>VBoxManage controlvm "Windows 7" poweroff
uws@tux>VBoxManage controlvm "Windows 7" acpipowerbutton

uws@tux># Export / Import
uws@tux>VBoxManage export "Windows 7" -o win7.ovf
uws@tux>VBoxManage import win7.ovf
```

Listing 14.37: Beispiele Verwaltung

Eine Disk kann man folgendermaßen erstellen / konvertieren.

```
uws@tux># Create Disk, Size in MB
uws@tux>VBoxManage createmedium disk --filename disk_sdb.vdi --size 30000

uws@tux># Konvert Disk
uws@tux>dd if=/dev/sdc of=disk.raw
uws@tux>VBoxManage convertfromraw disk.raw disk_sdc.vdi --format VDI
```

Listing 14.38: Create Disk

14.3.3 Gasterweiterung

Wenn die Installation der VirtualBox Gasterweiterung mit einer Fehlermeldung nicht vollständig durchgeführt wird, so müssen die Kernel Source noch installiert werden. Eventuell müssen auch die Module gcc, make, automake und autoconf installiert werden.

```
uws@tux>sudo zypper install kernel-source kernel-symbol

uws@tux>sudo zypper install gcc make automake autoconf
```

Listing 14.39: Kernel Source

Kapitel 15

Backup

15.1 Borg

Das Backup Programm `Borg` ist ein fork von dem Programm `Attic`. Borg und Attic sind in Python geschrieben und Attic wird nicht mehr weiterentwickelt.
Alle Borg Variablen gibt es hier:
https://borgbackup.readthedocs.io/en/stable/usage.html#
Ein Auswahl von Borg Variablen:

Variable	*Beschreibung*
BORG_REPO	Wenn definiert, kann man die Abkürzung ::archive benutzen
BORG_PASSPHRASE	Wenn definiert, erfolgt keine Abfrage des Kennwortes mehr
BORG_DISPLAY_PASSPHRASE	
BORG_KEYS_DIR	Standard ist ~/.config/borg/keys. Ablage der Keys
BORG_SECURITY_DIR	Standard ist ~/.config/borg/security. Hier werden die Security Daten abgelegt
BORG_CACHE_DIR	Standard ist ~/.cache/borg. Zwischenspeicher, benötigt bei großen Repositories viel Platz

Tabelle 15.1: Auswahl Variablen

15.1.1 Installation

Unter Arch Linux kann man das Programm aus dem Repository installieren. Ebenso kann man die Software von dem Hersteller herunterladen und händisch installieren. Hierbei muss man sich sekbst um ein Update des Programms kümmern.

```
root@tux>pacman -Sy borg

uws@tux>wget https://github.com/borgbackup/borg/releases/download/1.1.0b5/
   borg-linux64

root@tux># je nach Distribution auch unter /usr/local/bin
root@tux>cp borg-linux64 /usr/bin/borg
root@tux>chown root:root /usr/bin/borg
root@tux>chmod 755 /usr/bin/borg
```

Listing 15.1: Installation

15.1.2 Repository erstellen

Als erstes muss für ein Backup ein Repository eingerichtet werden. Standardmäßig ist die Verschlüsselung des Backups eingeschaltet. Es wird eine Verschlüsselung mit einem `repokey` genommen. Um den Mode zu ändern, gibt man ein `-e <mode>` bei dem anlegen des Repositories an. Ausser `repokey` gibt es noch `none` und `keyfile`. Bei `repokey` befindet sich der Schlüssel in der Datei `<repo_path/config`. Bei `keyfile` wird die Datei im Home-Verzeichnis unter `~/.config/borg/keys` abgespeichert. Diese Dateien sollten gesichert werden, um sie bei einem Defekt wieder herstellen zu können. Die Datenverschlüsselung wird mit einem 256-bit AES gemacht.

none	keine Verschlüsselung
keyfile	Schlüssel in ~/.config/keys
repokey	Schlüssel im Repository

```
uws@tux>borg init -e keyfile /repo/path
```

Listing 15.2: Repository erstellen

15.1.3 Backup erstellen

```
uws@tux>borg create [-v] [--stats] [--list] [--progress] /repo/path::Sunday
   /daten/Bilder /daten/Music
```

Listing 15.3: Backup erstellen

-v, --info	Ausführliche Ausgabe, log Level INFO
-n, --dry-run	Testlauf
-s, --stats	Statistik am Ende des Backups
--list	Zeigt die Dateien an, die im Backup landen
-p, --progress	Anzeige des Fortschritts
-C, --compression	None, zlib, lz4, lzma. Es kann nach der Angabe der Kompression auch noch der Level angegeben werden z.B. zlib,8. Standard ist der Level 6.
-e, --exclude	Ausschließen von Dateien / Verzeichnissen. Kann mehrfach angegeben werden.

15.1.4 Backup Scripten

Bei jeder An- und Abfrage des Repositories wird nach dem Kennwort gefragt. Wird die Variable `BORG_PASSPHRASE` exportiert, so entfällt die Passwort Abfrage. Dieses kann in Backup Scripten verwendet werden.

```
uws@tux>cat borg_example.shabox
#!/usr/bin/env bash

REPOSITORY=/backup/borg_bck
export BORG_PASSPHRASE='myKennwort'
BACKUPHOST=$(echo $HOSTNAME | cut -d'.' -f1)

borg create -v --stats \
  $REPOSITORY::$BACKUPHOST-$(date +%Y_%m_%d) \
  /daten/Scripts \
  /daten/Programming \
  --exclude /daten/Programming/Test

backup_exit=$?
```

```
# More examples for --exclude
# --exclude '/home/*/.cache'
# --exclude '*.pyc'

# Vorhaltezeit
# Die Angabe des Prefix ist notwendig, da dadurch sichergestellt ist,
# das nur die Backups geloescht werden, die von der Maschine stammen.
# Es werden sonst alle Backups geloescht.

borg prune -v --list $REPOSITORY --prefix $BACKUPHOST \
  --keep-daily=7 --keep-weekly=4 --keep-monthly=6

prune_exit=$?

# Ermitteln des hoechsten Exit Codes
global_exit=$(( backup_exit > prune_exit ? backup_exit : prune_exit ))

if [ ${global_exit} -eq 1];
then
  echo "Backup and/or Prune finished with warning"
fi

if [ ${global_exit} -gt 1];
then
  echo "Backup and/or Prune finished with error"
fi
exit ${global_exit}
```

Listing 15.4: Backup Script

15.1.5 Key Ex- und Import

```
uws@tux>borg key export --paper /repo/path > repo_sec.paper

uws@tux>borg key import --paper /repo/path
```

Listing 15.5: Ex- / Import Key

15.1.6 Backup delete

Mit dem ersten Beispiel wird nur das Backup vom Montag gelöscht und bei dem zweiten Beispiel werden alle Backups ud auch das Repository gelöscht.

```
uws@tux>borg delete /repo/path::Montag

uws@tux>borg delete /repo/path
```

Listing 15.6: Backup delete

15.1.7 Info Repo / Archiv

Mit `borg list /repo/path::<backup_name>` wird der Inhalt des Backups angezeigt. Also alle Dateien, die sich in dem Archiv sich befinden. Ohne Angabe des Backups werden nur die erstellten Backups angezeigt.

```
uws@tux>borg list /repo/path
fsgo002-2018_08_07  Tue, 2018-08-07 20:00:03 [6
   c1c6af082db16a502f1f81adcb07beb1a1cd977fdb17fcb04fd8951baf2c6f9]
fsgo002-2018_08_08  Wed, 2018-08-08 07:57:38 [
   d4353a6c7a515dd127241488a202d1d336db1e5628cadcefce35916d210a03e5]
fsgo002-2018_08_09  Thu, 2018-08-09 20:00:05 [06
   c0586d25ecdc9eedbf88c14d80a11cc6d42453423ca247dbfd2878d2498eed]
```

```
fsgo002-2018_08_10  Fri, 2018-08-10 20:00:05 [8
   c75710dc92de49bfcaed53286fe2d0d281cff89c7ca0f9a0e9e482068822c91]
```

Listing 15.7: List Repo

```
$ borg info /repo/path
Enter passphrase for key /repo/path:
Repository ID: 26
   d1581f08af88066cfdd976649edb70866d3d4304ffa673abcba73057a45538
Location: /repo/path
Encrypted: Yes (repokey)
Cache: /home/uws/.cache/borg/26
   d1581f08af88066cfdd976649edb70866d3d4304ffa673abcba73057a45538
Security dir: /home/uws/.config/borg/security/26
   d1581f08af88066cfdd976649edb70866d3d4304ffa673abcba73057a45538
------------------------------------------------------------------------------

                       Original size      Compressed size    Deduplicated
                          size
All archives:               902.14 GB            848.43 GB            151.90
   GB

                       Unique chunks         Total chunks
Chunk index:                  220795              1195021
```

Listing 15.8: Info Repo

```
uws@tux>borg info /repo/path::Montag
Enter passphrase for key /repo/path:
Name: Montag
Fingerprint: 9a8cc406...
Hostname: tux
Username: uws
Time (start): Mon, 2017-05-22 19:15:03
Time (end):   Mon, 2017-05-22 19:15:33
Command line: /usr/bin/borg create -v --stats /repo/path::Montag ...
Number of files: 81
               Original size    Compressed size    Deduplicated size
This archive:        2.88 MB           2.88 MB              26.62 kB
All archives:       60.58 MB          60.61 MB              56.49 MB

               Unique chunks    Total chunks
Chunk index:             700             823
```

Listing 15.9: Info Archiv

15.1.8 Mount / Umount

Man kann das Repository oder ein Archiv mounten. Hierfür muss das Python Module `llfuse` installiert sein.

```
root@tux>pip install llfuse

uws@tux>borg mount /repo/path /path/to/mount

uws@tux>borg mount /repo/path::Montag /path/to/mount

uws@tux>borg umount /path/to/mount
```

Listing 15.10: Mount / Umount

15.1.9 Change Passwort

```
uws@tux>borg init --encryption=keyfile -v /repo/path # create new repository

uws@tux>borg key change-passphrase -v /repo/path
```

Listing 15.11: Change Passwort

15.1.10 Platzhalter

{hostname}	Name der Maschine, kurz
{fqdn}	Name der Maschine, lang
{now}	Aktuelle locale Datum und Zeit
{utynow}	Aktuelle UTC Datum und Zeit
{user}	User Name
{pid}	Process ID
{borgversion}	Version des Programms z.B. 1.0.8rc1
{borgmajor}	Hauptbersion z.B 1
{borgminor}	Unterversion z.B. 1.0
{borgpatch}	Patch Version z.B. 1.0.8

Beispiele:

```
uws@tux>borg create /repo/path::{hostname}-{user}-{utcnow} ...

uws@tux>borg create /repo/path::{hostname}-{now:%Y_%m_%d_%H:%M:%S} ...

uws@tux>borg prune --prefix '{hostname}-' ...
```

Listing 15.12: Platzhalter

15.1.11 Restore

Es gibt mehrere Wege, Dateien aus dem Backup wieder herzustellen. Für ein Restore ist es am besten, vorher ein neues Verzeichnis anzulegen, worin der Restore abgelegt werden kann. Steht man in dem Produktiv Verzeichnis, werden die Daten bei einem Restore überschrieben.
Für ein Restore kann man auch das Repository / Backup mounten und dann mit dem Explorer die gewünschten Daten wieder herstellen.

```
uws@tux>mkdir -p /restore/data && cd /restore/data
```

Listing 15.13: Create Restore Path

Nun kann man mit `extract` die Daten aus dem Backup wieder herstellen.
In dem ersten Beispiel werden alle Daten aus dem Backup geholt.

```
uws@tux>borg extract -v --list /repo/path/::Montag
```

Listing 15.14: Restore all

Dateien aus einem bestimmten Verzeichnis wieder herstellen.

```
uws@tux>borg extract -v --list /repo/path::Montag /daten/scripts
```

Listing 15.15: Restore Path

Dateien von einem Restore ausschließen.

```
uws@tux>borg extract -v --list /repo/path::Montag /daten/scripts -exclude '
    *.o'
```

Listing 15.16: Restore mit exclude

15.1.12 Backup überprüfen

Mit `borg check` kann man das Backup Repository überprüfen. Die Überprüfung des Repositories kann unter Umständen sehr lange dauern, hängt von der Größe des Repositories ab.
Gibt man `--repository-only` mit an, wird nur das Repository überprüft und nicht auch noch die Backups. Dieses geht erheblich schneller. Wird bei `--archives-only` kein Backup Name mit angegeben, so werden alle Backups überprüft.

```
uws@tux>borg check -v --repository-only /repo/path

uws@tux>borg check -v --archives-only /repo/path::Montag
```

Listing 15.17: Backup überprüfen

15.1.13 Backup Rename

Möchte man ein Backup umbenennen, so kann man das mit `borg rename` machen.

```
uws@tux>borg rename /repo/path::Montag Monday
```

Listing 15.18: Backup umbenennen

15.1.14 Borg unlock

Bei jedem Backup wird bei dem Starten eine Lock Datei in dem Repository angelegt. Bei einem Absturz des Backups kann es vorkommen, das die Lock Datei nicht gelöscht wurde. Mit `borg break-lock` kann man die Sperre aufheben.

```
uws@tux>borg break-lock /repo/path
```

Listing 15.19: Sperre aufheben

15.2 Restic

15.2.1 Allgemein

Mit dem Programm `restic` können in einer Shell ein Backup von Verzeichnissen erstellt werden. Das Programm macht auch Incrementelle Backups und versteht deduplication. Das Backup wird in dem Repository verschlüsselt abgelegt. Restic kann das Backup auch in der Cloud ablegen.

15.2.2 Installation

Das Programm ist in den meisten Repositorys der Distributionen enthalten und können über den Paketmanager installiert werden.

```
uws@tux># Arch Linux
uws@tux>sudo pacman -S restic

uws@tux># Debian
uws@tux>sudo apt install restic

uws@tux># Fedora
uws@tux>sudo dnf install restic

uws@tux># RHEL und CentOS
uws@tux>sudo yum install yum-plugin-copr
uws@tux>sudo yum copr enable copart/restic
uws@tux>sudo yum install restic
```

Listing 15.20: Installation

15.2.3 Repository erstellen

Das Kennwort für das Repository wird auch darin abgelegt. Als erstes wird das Repository initalisiert.

```
uws@tux>restic init --repo /run/media/restic/repo
enter password for new backend:
enter password again:
created restic backend 0815b4711c at /run/media/restic/repo

Please noze that knowledge of your password is required to access the
   repository.
Losing your password means that your data is irrecoverable lost.
```

Listing 15.21: Repository erstellen

15.2.4 Backup

Ein Backup wird mit dem Parameter `backup` gemacht. Mit der Option `--tag` kann man noch ein Tag definieren. Eine Ausführliche Ausgabe des Backups erhält man mit `-v, --verbose`. Mit `--verbose=2` gibt es eine noch detailierte Ausgabe.

```
uws@tuc>restic -r /run/media/restic/repo --tag "First Backup" backup /daten
```

Listing 15.22: Backup Commandozeile

In einem Script kann man benötigte Parameter in Variablen definieren.

Variable	Beschreibung
RESTIC_REPOSITORY	Angabe des Repositories oder Angabe mit -r
RESTIC_PASSWORD_FILE	Passwort Datei
RESTIC_PASSWORD	Das Passwort des Repositories

Continued on next page

continued from previous page.

Variable	Beschreibung
AWS_ACCESS_KEY_ID	Amazon S3 access key ID
AWS_SECRET_ACCESS_KEY	Amazon S3 secret access key
ST_AUTH	Auth URL für Keystone vi authentication
ST_USER	Username für Keystone vi authentication
ST_KEY	Passwort für Keystone vi authentication

Tabelle 15.2: Restic Variablen

Beispiel eines einfachen Scripts.

```
uws@tux>cat BackupWithRestic.sh
#!/bin/env listBash
export RESTIC_REPOSITORY="/run/media/restic/Repo"
export RESTIC_PASSWORD="StrongSecretPassword"

restic -v -v --Tag "Restic Backup" backup /daten >> /daten/var/log/
   BackupRestic.log
# List Snapshots
restic snapshots
```

Listing 15.23: Script Example

15.2.5 Backup Vorhalten

Man kann mit `forget` und weiteren Parametern definieren, wie lange Backups vorgehalten werden sollen. Mit dem Parameter `--dry-run` werden die Backups nicht gelöscht, sondern nur angezeigt, was gelöscht werden soll. Gibt man hinter `forget` die ID des Snapshots mit an, so wird nur der Snapshot gelöscht.

Command	Beschreibung
-l, --keep-last=	Behalte n Snapshots
-H, --keep-hourly=	Behalte die letzten n Stunden Snapshots
-d, --keep-daily=	Behalten die letzten n Tages Snapshots
-w, --keep-weekly=	Behalte die letzten n Wochen Snapshots
-m, --keep-monthly=	Behalte die letzten n Monats Snapshots
-y, --keep-yearly=	Behalte die letzten n Jahres Snapshots
--hostname=	Lösche nur die Snapshots von dem Host
-n, --dry-run	Es erfolgt nur die Anzeige, was gelöscht werden soll

Tabelle 15.3: Forget Commands

```
uws@tux>restic -r /run/media/restic/Repo forget --dry-run --keep-last=2
enter password for repository:
repository 3b92a80b opend successfully, password is correct
Applying Policy: keep the last 2 snapshots snapshots
snapshots for (host [tux], paths [/daten])

keep 2 Snapshots
.
.
```

Listing 15.24: Forget Beispiel

Das `forget` löscht die Daten nicht aus dem Repository, sondern erst der Parameter `prune`.

```
uws@tux>restic -r /run/media/restic/Repo prune
```

Listing 15.25: Prune

15.2.6 List Snapshots

Alle erstellten Snapshots können mit der Option snapshots angezeigt werden.

```
uws@tux>restic -r /run/media/restic/Repo snapshots
enter password for repository:
repository 3b92a80b opened successfully, password is correct
ID        Time                 Host      Tags        Paths
----------------------------------------------------------------------
9b431633  2019-03-24 18:01:19  svgo226   Program     /daten/program/bin
39c1ce35  2019-03-25 08:51:15  svgo226   Daten       /daten/program/bin
56daf1ba  2019-03-25 19:23:58  svgo226   Privat      /fsgo002/daten/Privat
.
.
```

Listing 15.26: List alle Snapshots

Man kann sich die Ausgabe auch Filtern nach Host, Tag oder Path.

```
uws@tux>restic -r /run/media/restic/Repo snapshots --tag Program
enter password for repository:
repository 3b92a80b opened successfully, password is correct
ID        Time                 Host        Tags        Paths
----------------------------------------------------------------------
9b431633  2019-03-24 18:01:19  svgo226     Program     /daten/program/bin
----------------------------------------------------------------------
1 snapshots
```

Listing 15.27: List Snapshots by Filter

Eine Gruppierte Anzeige der Snapshots erhält man mit der Option --group-by.

```
uws@tux>restic -r /run/media/restic/Repo snapshots --group-by tags
enter password for repository:
repository 3b92a80b opened successfully, password is correct
snapshots for (tags [Public]):
ID        Time                 Host        Tags        Paths
----------------------------------------------------------------------
afee555e  2019-04-30 19:07:50  svgo226     Public      /fsgo002/public
66c14051  2019-05-31 19:07:53  svgo226     Public      /fsgo002/public
.
.
```

Listing 15.28: List Snapshots Gruppiert

15.2.7 List Files

Eine Auflistung, welche Dateien sich in dem Backup befinden, bekommt man mit `ls`.

```
uws@tux>restic -r /run/media/restic/Repo ls -l <RepoID>
uws@tux># Anstelle der RepoID kann man auch latest nehmen.
```

Listing 15.29: List Backup Files

15.2.8 Backup Statistik

Informationen über die Backups kann man mit `stats` erhalten.

```
uws@tux>restic -r /run/media/restic/Repo stats
enter password for repository:
repository 3b92a80b opend successfully, password is correct
scanning...
Stats for all snapshots in restore-size mode:
  Total File count:   1722
```

```
        Total Size:   135.936 MiB
```

Listing 15.30: Backup Informationen

15.2.9 Backup Check

Das Repository kann man mit der Option `check` überprüfen.

```
uws@tux>restic -r /run/media/restic/Repo stats
using temporary cache in /tmp/restic-check-cache-259726251
enter password for repository:
repository 3b92a80b opend successfully, password is correct
created new cache in /tmp/restic-check-cache-259726251
create exclisuÂ´ive lock for repository
load indexes
check all packs
check snapshots, trees and blobs
no errors where found
```

Listing 15.31: Backup Check

15.2.10 Repository Unlock

Stüerzt das Programm während eines Backups ab, so ist das Repository gesperrt. Mit der Option `unlock` kann man sie Sperre entfernen.

```
uws@tux>restic -r /run/media/restic/Repo unlock
enter password for repository:
repository 3b92a80b opened successfully, password is correct
successfully removed locks
```

Listing 15.32: Repository Unlock

Kapitel 16

Verweise / Links

Beschreibung	*Verweis / Link*
Bash Prompt How To	http://www.ltdp.org/HOWTO/Bash-Prompt-HOWTO
SSH-Artikel	Sicher gesendet - ssh und scp - Linux User 07/2003, Seite 81 ff.
Mehr zu date und cal	Kalender auf der Kommandozeile - Linux User 07/2005, Seite 88 ff.
Rsync	http://samba.anu.edu.au/rsync
Rsnapshots	http://www.rsnapshot.org
Rsync	Synchroner Datenstrom - Linux User 04/2006, Seite 90 ff.
Rsync	Snapshot-Backup mit Rsync - Linux User 09/2004, Seite 72
Uhrzeit	Wer hat an der Uhr gedreht - Linux User 10/2005, Seite 61 ff.
Xdialog	http://xdialog.dyns.net
Prompt Color	https://wiki.archlinux.org/index.php/Color Bash Prompt#Return value visualisation
Sed	Workshop: Einführung in den Stram Editor Sed - Linux User 08/2015, Seite 68
Pacman	https://wiki.archlinux.de/title/pacman
Borg	Backup Programm https://borgbackup.readthedocs.io/en/stable/#

Tabelle 16.1: Verweise und Links