mitp

Eric Amberg

Linux-Server

mit Debian 7 GNU/Linux

Das umfassende Praxis-Handbuch

Aktuell für die Version Debian 7 (Wheezy)

Eric Amberg

Linux Server mit Debian 7 GNU/Linux

Das umfassende Praxis-Handbuch

Aktuell für die Version Debian 7 (Wheezy)

mitp

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN 978-3-8266-8201-5
1. Auflage 2014

E-Mail: kundenbetreuung@hjr-verlag.de

Telefon: +49 6221/489-555
Telefax: +49 6221/489-410

www.mitp.de

Lektorat: Sabine Schulz
Sprachkorrektorat: Petra Heubach-Erdmann
Satz: III-satz, Husby, www.drei-satz.de
Coverbild: © sergios – fotolia.de

Inhaltsverzeichnis

Einleitung 23

Teil 1 Allgemeine Systemadministration 35

1 **Woher bekomme ich Debian-Linux?** 37
1.1 Die Quellen von Debian-Linux 37
1.2 Download per FTP oder HTTP 39
1.3 MD5- und SHA1-Prüfsummen 42
1.4 Download per BitTorrent 45
1.5 Download mittels jigdo 47
1.6 Download per Netzinstallation 54
1.7 Zusammenfassung und Weiterführendes 56

2 **Debian installieren** 57
2.1 Hardware-Voraussetzungen 57
2.2 Installation des Debian-Grundsystems 59
2.2.1 Booten von CD oder DVD 59
2.2.2 Sprach- und Ländereinstellungen 60
2.2.3 Automatische Hardware-Erkennung 61
2.2.4 Netzwerkkonfiguration 62
2.2.5 Host- und Domainname 65
2.2.6 Benutzer und Passwörter einrichten 66
2.2.7 Partitionierung 67
2.2.8 Weitere Installationsschritte 75
2.3 Experteninstallation 83
2.3.1 Bootoptionen 83
2.3.2 Experteninstallation 86
2.3.3 Menüpunkte der manuellen Partitionierung 89
2.3.4 Logical Volume Manager 92
2.3.5 RAID 94
2.4 Zusammenfassung und Weiterführendes 101

3 **Debian-Paketmanagement** 103
3.1 dpkg – das Basistool 103
3.1.1 dpkg-Optionen 104
3.1.2 Workshop: Pakete mit dpkg verwalten 105
3.2 Die APT-Tools 111
3.2.1 Das »motivierende Einstiegsbeispiel« 111
3.2.2 Einführung in die APT-Tools 113

3.2.3 Grundfunktionen von apt-get ... 114
3.2.4 Definition der Paketquellen ... 115
3.2.5 Welche Quellen eintragen? ... 117
3.2.6 Erweiterte Funktionen von apt-get ... 117
3.2.7 Upgrade von Squeeze auf Wheezy ... 118
3.2.8 aptitude – das Frontend zu apt-get ... 120
3.2.9 apt-cache ... 125
3.3 Softwareauswahl mit Tasksel ... 128
3.4 Multiarch-Support ... 129
3.5 Weiterführende Informationen und Backgrounds ... 130
3.5.1 Wie organisiert dpkg seine Daten? ... 130
3.5.2 Der Aufbau eines Debian-Pakets ... 131
3.5.3 debconf ... 132
3.5.4 Installation von Software mittels Tarballs ... 133
3.6 Zusammenfassung und Weiterführendes ... 134

4 Das Debian-System – Grundlagen ... 137
4.1 Die Konsole ... 137
4.2 Herunterfahren und Neustarten des Systems ... 138
4.3 Basisbefehle zur Navigation ... 139
4.3.1 Aktuelles Verzeichnis anzeigen lassen ... 139
4.3.2 Inhalt eines Verzeichnisses anzeigen lassen ... 140
4.3.3 In ein anderes Verzeichnis wechseln ... 140
4.3.4 Pfadangaben ... 141
4.4 Die Struktur des Dateisystems ... 141
4.5 Dateioperationen ... 145
4.5.1 Dateien und Verzeichnisse erstellen ... 146
4.5.2 Textdateien bearbeiten mit nano ... 146
4.5.3 vim – ein Crashkurs ... 148
4.5.4 Textdateien betrachten ... 151
4.5.5 Kopieren von Dateien und Verzeichnissen ... 152
4.5.6 Verschieben und Umbenennen ... 153
4.5.7 Löschen von Dateien und Verzeichnissen ... 153
4.5.8 Eine Verknüpfung erstellen ... 154
4.5.9 Eine Übung zum Vertiefen ... 156
4.6 Man-Pages – Hilfe zur Selbsthilfe ... 157
4.6.1 Die Man-Pages nutzen ... 157
4.6.2 whatis und apropos ... 159
4.6.3 info – die neuen Man-Pages ... 160
4.7 Zusammenfassung und Weiterführendes ... 160

5 Einbinden von Dateisystemen ... 163
5.1 mount und umount ... 163
5.2 Die virtuellen Dateisysteme ... 165
5.2.1 udev – Dynamische Geräteverwaltung ... 166
5.2.2 USB-Geräte ... 169

5.3 Die Datei /etc/fstab ... 170
5.4 udev, HAL und D-Bus ... 174
5.5 Zusammenfassung und Weiterführendes ... 177

6 Der Linux-Systemstart ... 179
6.1 GRUB – Der Linux-Bootloader ... 179
6.1.1 GRUB 2 ... 181
6.1.2 UEFI ... 182
6.1.3 GUID Partition Table (GPT) ... 183
6.2 System V versus systemd ... 183
6.2.1 Das Konzept der Runlevels ... 184
6.2.2 Die Organisation des Systemstarts ... 184
6.2.3 Die Runlevel-Verzeichnisse ... 186
6.3 Die Verwaltung der Dienste ... 188
6.4 Einrichten der Links in den Runlevel-Verzeichnissen ... 189
6.4.1 rcconf ... 189
6.4.2 update-rc.d ... 190
6.4.3 Workshop – Anpassen der Runlevels ... 191
6.4.4 Grundlagen zu systemd ... 193
6.5 Zusammenfassung und Weiterführendes ... 193

7 Benutzerverwaltung ... 195
7.1 Einen Benutzer anlegen ... 195
7.2 Die Datei /etc/passwd ... 196
7.3 Benutzer modifizieren ... 198
7.4 Einen Benutzer löschen ... 198
7.5 Gruppen erstellen, zuweisen und löschen ... 199
7.6 Die Datei /etc/group ... 199
7.7 Informationen über einen Benutzer abfragen ... 200
7.8 Passwörter vergeben ... 201
7.9 Die Datei /etc/shadow ... 202
7.10 Kennwortrichtlinien ... 203
7.11 Einen neuen Benutzer mit su testen ... 204
7.12 Workshop: Einrichten von Benutzern ... 204

8 Rechteverwaltung ... 209
8.1 Das Linux-Rechtesystem ... 209
8.2 Unterschiede zwischen Verzeichnissen und Dateien ... 211
8.3 Eigentümer und Gruppe festlegen ... 212
8.4 Rechte vergeben mit chmod und umask ... 213
8.5 Besondere Rechte ... 216
8.6 Ein Übungsszenario ... 217
8.7 Access Control Lists (ACLs) ... 220
8.7.1 ACLs aktivieren ... 220
8.7.2 Wie funktionieren ACLs? ... 221
8.7.3 Probleme bei der Nutzung von ACLs ... 226
8.7.4 Wo werden ACLs sinnvoll eingesetzt? ... 227

8.8 Quotas – Einschränkungen des Speicherplatzes für Benutzer ... 227
8.8.1 Quota-Unterstützung aktivieren ... 228
8.8.2 Quotas festlegen ... 229
8.8.3 Quotas kontrollieren ... 230
8.9 Zusammenfassung und Weiterführendes ... 231

9 Einführung in die Bash ... 233
9.1 Bash oder Dash? ... 233
9.2 Was macht eigentlich eine Shell? ... 234
9.3 Die Kommandoeingabe ... 235
9.4 Verschachtelte Shells ... 236
9.5 Aliasse ... 238
9.6 Die Bash-Konfigurationsdateien ... 239
9.7 Ein- und Ausgabeumleitungen ... 241
9.8 Pipes ... 243
9.9 Die Ausgabe eines Befehls mit tee teilen ... 243
9.10 Befehle verketten ... 244
9.11 Patterns (Jokerzeichen) ... 245
9.12 Sonderzeichen und Maskierung ... 246
9.13 Kommandosubstitution ... 248
9.14 Shellvariablen ... 249
9.14.1 Shellvariablen vs. Umgebungsvariablen ... 250
9.14.2 Workshop: Shell- und Umgebungsvariablen ... 250
9.14.3 Shell- und Umgebungsvariablen anzeigen ... 251
9.14.4 PATH – Die Pfadfinder-Variable ... 252
9.14.5 PS1 – Der Prompt ... 253
9.14.6 Weitere wichtige Umgebungsvariablen ... 254
9.15 Zusammenfassung und Weiterführendes ... 255

10 Wichtige Befehle zur Systemadministration ... 257
10.1 Dateien und Verzeichnisse suchen ... 257
10.1.1 find ... 258
10.1.2 locate ... 259
10.2 grep und die Regular Expressions ... 260
10.2.1 grep ... 260
10.2.2 Regular Expressions ... 262
10.3 sed – Manipulation von Textdateien ... 264
10.3.1 Die ersten Schritte mit sed ... 264
10.3.2 Adressen ... 265
10.3.3 Weiterführende Anwendungsbeispiele ... 266
10.4 Awk – Auswertung von Textdateien ... 267
10.4.1 Einführung ... 268
10.4.2 Mehrzeilige Awk-Skripte ... 270
10.5 Komprimierung von Dateien ... 272
10.5.1 compress ... 272
10.5.2 gzip und gunzip ... 273
10.5.3 bzip2 und bunzip2 ... 273

10.6 Der Midnight-Commander ... 274
10.6.1 Grundfunktionen ... 274
10.6.2 Dateien ansehen ... 276
10.6.3 Dateien bearbeiten ... 277
10.6.4 Die Befehlszeile ... 277
10.6.5 Das Menü ... 277
10.7 Weitere nützliche Befehle ... 278
10.7.1 wc – Ausgabezeilen zählen ... 278
10.7.2 cat – Textdateien vollständig ausgeben ... 279
10.7.3 Ordnung schaffen mit sort ... 279
10.7.4 Datum und Uhrzeit mit date ... 280
10.7.5 Identifikation – whoami, id und who ... 281
10.8 Zusammenfassung und Weiterführendes ... 282

11 System- und Festplattenmanagement ... 283
11.1 Systemstatus – CPU, RAM, Prozesse ... 283
11.1.1 vmstat – RAM, Swap und CPU ... 283
11.1.2 top – die Top-Ten-Liste ... 285
11.1.3 free – verfügbarer Arbeitsspeicher ... 286
11.1.4 uptime – Zeit seit dem Booten ... 286
11.1.5 uname – Systembezeichnung und -version ... 287
11.2 Prozessverwaltung ... 287
11.2.1 ps – Die Prozessliste ... 287
11.2.2 pstree – Mutter, Vater, Kind ... 288
11.2.3 kill – Prozesse »umbringen« ... 289
11.2.4 killall – alle gleichartigen Prozesse beenden ... 289
11.3 Festplattenmanagement – Grundlagen ... 290
11.3.1 fdisk – den Kuchen aufteilen ... 290
11.3.2 Formatierung ... 293
11.3.3 Einbinden in das Dateisystem ... 295
11.4 LVM – der Logical Volume Manager ... 296
11.5 Debugging und Troubleshooting ... 302
11.5.1 fsck – Wenn's mal nicht so läuft ... 302
11.5.2 du – Wer braucht welchen Platz? ... 303
11.6 df – Wie viel Platz habe ich noch? ... 305
11.7 Zusammenfassung und Weiterführendes ... 305

12 Zeitlich gesteuerte Backups ... 307
12.1 Wozu eigentlich Backups? ... 307
12.2 RAID versus Backup ... 308
12.3 Backup-Medien ... 308
12.3.1 Auswahl des geeigneten Mediums ... 308
12.3.2 Zugriff auf die Backup-Medien ... 309
12.4 Backup-Strategien ... 312
12.4.1 Das Generationenprinzip ... 312
12.4.2 Sicherungsarten ... 312
12.4.3 Die richtige Strategie entwickeln ... 313

12.5 Welche Daten sind zu sichern? ... 315
12.6 Die Sicherungswerkzeuge ... 316
12.6.1 dump und restore ... 316
12.6.2 Dateien archivieren mit tar ... 317
12.6.3 cpio – eine Alternative zu tar ... 321
12.6.4 Rohdaten sichern mit dd ... 322
12.6.5 AMANDA – Netzwerk-Backups ... 323
12.7 Zeitlich gesteuerte Aufträge mit cron ... 323
12.7.1 Der cron-Daemon ... 323
12.7.2 cron-Jobs für Benutzer ... 325
12.7.3 Einen cron-Job mit crontab erstellen ... 326
12.7.4 Zeitlich gesteuerte Sicherungen einrichten ... 326
12.8 Zusammenfassung und Weiterführendes ... 327

13 Einführung in die Shellskript-Programmierung ... 329
13.1 Was sind Shellskripte eigentlich? ... 329
13.2 Ein Skript zum Erstellen von Skripten ... 330
13.3 Variablen ... 332
13.4 Bedingte Verzweigungen – wenn, dann ... 334
13.5 Schleifen – wiederholte Ausführung ... 336
13.6 Parameter beim Skriptstart übergeben ... 338
13.7 Zeichenketten ausschneiden ... 342
13.8 Listen – Die for-Schleife ... 343
13.9 Fälle unterscheiden mit case ... 345
13.10 Zustände abfragen mit test ... 346
13.11 Analyse eines Init-Skripts ... 347
13.12 Zusammenfassung und Weiterführendes ... 351

14 Protokollierung ... 353
14.1 Zeitsynchronisation mit NTP ... 354
14.2 Der Syslog-Daemon ... 357
14.2.1 Die Herkunftsarten (facilities) ... 357
14.2.2 Die Prioritäten (priorities) ... 358
14.3 syslog.conf ... 358
14.4 Remote Logging ... 361
14.5 logger – syslog für eigene Skripte ... 362
14.6 syslog-ng ... 363
14.7 Rotation der Logdateien ... 365
14.8 Analyse der Logdaten ... 368
14.8.1 Manuelle Analyse ... 369
14.8.2 Automatisierte Analyse ... 369
14.8.3 Logsurfer und SEC ... 369
14.8.4 Logtool ... 370
14.8.5 Webalizer ... 370
14.9 Zusammenfassung und Weiterführendes ... 371

15 Den Kernel anpassen ... 373
15.1 Monolithische versus modulare Kernel ... 373

15.2 Distributions- und Original-Kernel ... 374
15.2.1 Die Kernel-Versionen ... 374
15.2.2 Der Original-Kernel ... 375
15.2.3 Distributions-Kernel ... 376
15.2.4 Die Kernel-Module ... 376
15.3 Einen Distributionskernel einbinden ... 379
15.3.1 Den neuen Kernel installieren ... 379
15.3.2 Wie ist der Kernel-Start organisiert? ... 381
15.4 Workshop: Den eigenen Kernel kompilieren ... 382
15.4.1 Den aktuellen Kernel herunterladen ... 382
15.4.2 Den Kernel konfigurieren ... 384
15.4.3 Variante 1 – Debian-Kernel-Paket erstellen ... 387
15.4.4 Variante 2 – Kernel manuell erstellen ... 388
15.5 Zusammenfassung und Weiterführendes ... 390

16 Das X-Window-System ... 391
16.1 Was ist eigentlich X Window? ... 392
16.2 Wie funktioniert X? ... 393
16.3 X Window mit Desktop-Umgebung installieren ... 394
16.4 Einführung in die Bedienung von GNOME 3 ... 396
16.4.1 Die Funktionsleisten (Panels) ... 397
16.4.2 Das Menü Anwendungen ... 399
16.4.3 Das GNOME-Tweak-Tool ... 401
16.4.4 Das Menü Systemwerkzeuge ... 402
16.4.5 Workshop: GNOME nutzen ... 403
16.4.6 Auf der traditionellen Konsole arbeiten ... 414
16.5 Troubleshooting ... 415
16.6 Start des X-Servers ... 416
16.6.1 Den X-Server mit startx starten ... 416
16.6.2 Den X-Server mit Display-Manager starten ... 417
16.7 X Window im Netzwerk ... 421
16.8 Zusammenfassung und Weiterführendes ... 422

17 Netzwerkgrundlagen und TCP/IP ... 423
17.1 Netzwerkgrundlagen ... 424
17.1.1 LAN, MAN, WAN, GAN, Internet ... 424
17.1.2 Ethernet, WLAN, ISDN, DSL, Standleitungen ... 424
17.1.3 Kabel, Stecker und Spezifikationen ... 426
17.1.4 Repeater, Hubs, Switches und Router ... 427
17.2 Die Schichtenmodelle ... 428
17.2.1 ISO-OSI-Schichtenmodell ... 428
17.2.2 Das TCP/IP-Referenzmodell ... 429
17.3 Was ist eigentlich ein Protokoll? ... 430
17.4 Das Internet Protokoll ... 431
17.4.1 IP-Adresse und Subnetzmaske ... 431
17.4.2 Netzadressen und Broadcasts ... 433
17.4.3 Netzklassen, NAT und private Netzbereiche ... 433

17.5 Bridges, Router und Gateways 440
17.5.1 Bridges 440
17.5.2 Router, Next Hop und Standard-Gateways 440
17.5.3 Gateways 442
17.6 ARP 442
17.7 TCP und UDP 443
17.7.1 TCP 443
17.7.2 UDP 444
17.7.3 Ports 444
17.8 ICMP 445
17.9 Die Anwendungsprotokolle 447
17.9.1 DNS 447
17.9.2 NetBIOS, SMB und WINS 447
17.9.3 DHCP 448
17.9.4 WWW 448
17.9.5 FTP 448
17.9.6 E-Mail 449
17.10 Zusammenfassung und Weiterführendes 449

18 Netzwerkkonfiguration 451
18.1 Bevor wir anfangen: Das Szenario 451
18.2 Die Netzwerkkarte 453
18.3 Eine IP-Adresse festlegen 453
18.3.1 IP-Adresse festlegen mittels ifconfig 454
18.3.2 Die IP-Adresse mit iproute2 festlegen 454
18.3.3 Konfiguration über /etc/network/interfaces 455
18.3.4 Die neue Konfiguration aktivieren 456
18.3.5 Konfiguration über DHCP 456
18.4 Standard-Gateway und statische Routen 457
18.4.1 Das Standard-Gateway mit route festlegen 457
18.4.2 Das Standard-Gateway per ip festlegen 457
18.4.3 Der Weg über /etc/network/interfaces 457
18.4.4 Statische Routen definieren 458
18.5 Namensauflösung konfigurieren 459
18.5.1 Die Datei /etc/hosts 459
18.5.2 Konfiguration des DNS-Clients 460
18.5.3 Der Hostname 461
18.6 Zusammenfassung und Weiterführendes 461

19 Fehlersuche im Netzwerk 463
19.1 Netzwerktools 463
19.1.1 ping 463
19.1.2 ifconfig 465
19.1.3 traceroute 466
19.1.4 netstat 467
19.1.5 telnet 469

19.1.6 nslookup . . . 470
19.1.7 tcpdump . . . 471
19.2 Wireshark . . . 474
19.3 Lösungsstrategie . . . 477
19.4 Zusammenfassung und Weiterführendes . . . 478

20 Fernwartung mit SSH . . . 479
20.1 Wie funktioniert SSH? . . . 479
20.2 Konfiguration des SSH-Dienstes . . . 480
20.3 Der SSH-Client . . . 482
20.3.1 Der SSH-Client von OpenSSH . . . 482
20.3.2 PuTTY – SSH unter Windows . . . 484
20.4 SCP und SFTP . . . 486
20.4.1 SCP . . . 486
20.4.2 SFTP . . . 487
20.4.3 WinSCP . . . 488
20.5 Anwendungen durch SSH tunneln . . . 490
20.5.1 X11 durch SSH tunneln . . . 490
20.5.2 Andere Applikationen durch SSH tunneln . . . 490
20.6 Zusammenfassung und Weiterführendes . . . 493

Teil 2 Der Backoffice-Server . . . 495

21 DHCP – dynamische Zuweisung der IP-Konfiguration . . . 499
21.1 Das Szenario . . . 499
21.2 Was kann DHCP? . . . 500
21.3 Wie funktioniert DHCP? . . . 500
21.4 Installation des DHCP-Servers . . . 502
21.5 Konfiguration des DHCP-Servers . . . 504
21.5.1 Workshop: DHCP-Grundkonfiguration . . . 504
21.5.2 Fortgeschrittene DHCP-Konfiguration . . . 510
21.6 Der DHCP-Relay-Agent . . . 512
21.7 Dynamische DNS-Aktualisierung . . . 514
21.8 Übung: DHCP im Szenario-Netzwerk . . . 514
21.9 Zusammenfassung und Weiterführendes . . . 516

22 NFS – Dateiübertragung zwischen Linux-Computern . . . 517
22.1 Das Szenario . . . 517
22.2 NFS-Grundlagen . . . 518
22.3 NFS installieren . . . 518
22.4 Konfiguration von NFS . . . 519
22.4.1 Workshop: Grundkonfiguration von NFS . . . 519
22.4.2 Fortgeschrittene NFSv3-Konfiguration . . . 523
22.5 NFSv4 . . . 526
22.6 Übung: NFS im Szenario-Netzwerk . . . 526
22.7 Zusammenfassung und Weiterführendes . . . 527

23 Drucken im Netzwerk ... 529
23.1 Das Szenario ... 529
23.2 Drucksysteme unter Linux ... 530
23.3 Installation von CUPS ... 530
23.4 Konfiguration von CUPS ... 533
23.5 Den Drucker nutzen ... 544
23.6 Drucken im (Linux-)Netzwerk ... 544
23.7 Zusammenfassung und Weiterführendes ... 545

24 Samba Teil I – Grundlagen des Windows-Servers ... 547
24.1 Grundlagen: NetBIOS und SMB ... 548
24.1.1 NetBIOS ... 548
24.1.2 SMB und CIFS ... 549
24.1.3 NetBIOS-Namensdienst und WINS ... 549
24.1.4 Arbeitsgruppen und Domänen ... 550
24.1.5 Die Netzwerkumgebung ... 551
24.2 Installation des Samba-Servers ... 552
24.3 Grundkonfiguration des Samba-Servers ... 553
24.3.1 Ein erster Blick auf die Konfiguration ... 553
24.3.2 Workshop: Einfache Verzeichnisfreigaben ... 554
24.3.3 Workshop: Sicherheit auf Benutzerebene und Einbinden der Home-Verzeichnisse ... 560
24.3.4 Workshop – Netzwerkdrucker mit Samba ... 570

25 Samba Teil II – Erweiterte Samba-Konfiguration ... 575
25.1 Das Domänenkonzept von Windows ... 575
25.2 Das Szenario ... 577
25.3 Workshop: Samba als Domänen-Controller ... 577
25.3.1 Computer-Konten einrichten ... 578
25.3.2 Einen Domänen-Administrator erstellen ... 578
25.3.3 Logon- und Profil-Verzeichnisse ... 578
25.3.4 Samba als PDC konfigurieren ... 579
25.3.5 Windows in die Domäne bringen ... 582
25.3.6 Das Anmeldeskript ... 586
25.3.7 Servergespeicherte Profile ... 587
25.4 Tipps zur Samba-Administration ... 588
25.4.1 Einen Mitgliedserver erstellen ... 588
25.4.2 Zugriffsberechtigungen auf Freigabeebene ... 589
25.4.3 Gast-Account einrichten ... 589
25.4.4 CD- und DVD-Laufwerke freigeben ... 590
25.5 Übung: Eine Domäne im Architekturbüro ... 590
25.6 Samba-Administration mittels SWAT ... 590
25.6.1 inetd, der Super-Daemon ... 591
25.6.2 xinetd – der Nachfolger von inetd ... 592
25.6.3 SWAT starten ... 592
25.6.4 SWAT mit SSL ... 594
25.7 Samba 4 und Active Directory ... 595
25.8 Zusammenfassung und Weiterführendes ... 596

26 Apache Teil I – Aufbau eines Intranets ... 597
26.1 Das Szenario ... 598
26.2 Grundlagen der Webkommunikation ... 598
26.3 Der Apache Webserver ... 599
26.4 Apache installieren ... 600
26.5 Grundkonfiguration von Apache ... 602
26.5.1 Die Apache-Konfigurationsstruktur ... 602
26.5.2 Globale Parameter in apache2.conf ... 604
26.5.3 Workshop – eine erste Website ... 606
26.6 Fehlercodes und Statusmeldungen ... 610
26.7 Kontexte ... 611
26.7.1 <Directory>-Kontext ... 612
26.7.2 <Location>-Kontext ... 612
26.7.3 <IfModule>-Kontext ... 612
26.8 Die Direktiven innerhalb der Kontexte ... 613
26.8.1 Options ... 613
26.8.2 Deny, Allow und Order ... 615
26.8.3 AllowOverride und .htaccess ... 615
26.9 Das Szenario – wie geht es weiter? ... 616

27 Datenbanken mit MySQL ... 617
27.1 Das Szenario ... 618
27.2 Datenbank-Grundlagen ... 618
27.2.1 Relationale Datenbank-Management-Systeme ... 618
27.2.2 Was beinhaltet eine Datenbank? ... 619
27.2.3 Das Entity Relationship Model ... 620
27.2.4 Die dritte Normalform und Objekttabellen ... 621
27.2.5 Primär- und Fremdschlüssel ... 622
27.2.6 Verbindungstabellen ... 623
27.2.7 Aufbau der Datenbank ... 623
27.3 Installation von MySQL ... 624
27.4 SQL ... 626
27.5 Workshop: Erstellen einer Datenbank ... 626
27.5.1 Die Datenbank kreieren ... 627
27.5.2 Datentypen ... 627
27.5.3 Erstellen der Tabellen ... 628
27.5.4 Verändern der Datenbankstruktur ... 630
27.6 Die Beispieldatenbank ... 631
27.7 Workshop: Datensätze einfügen und abändern ... 633
27.7.1 INSERT ... 633
27.7.2 LOAD DATA INFILE ... 634
27.7.3 UPDATE ... 634
27.7.4 DELETE ... 635
27.8 Workshop: Abfragen mit SELECT ... 635
27.8.1 Einfache Abfragen ... 635
27.8.2 Komplexe Abfragen ... 636
27.9 Weiterführende SELECT-Optionen ... 637

27.9.1 Spalten-Aliasse ... 637
27.9.2 Logische und arithmetische Ausdrücke ... 637
27.9.3 Funktionen ... 638
27.9.4 Gruppierung ... 638
27.10 Datenbankadministration ... 639
27.10.1 Das root-Passwort ändern ... 639
27.10.2 Benutzer einrichten und löschen ... 639
27.10.3 Benutzerrechte verwalten ... 640
27.10.4 Datensicherung ... 641
27.11 Und wie geht es weiter? ... 642

28 Dynamische Webseiten mit PHP ... 643
28.1 Einführung in PHP ... 643
28.2 PHP für Apache aktivieren ... 644
28.3 Das erste PHP-Skript ... 645
28.4 Workshop: Datenbankabfragen mittels PHP ... 645
28.4.1 Ziele und Vorgehensweise ... 646
28.4.2 Die Startseite ... 646
28.4.3 Die Bibliotheksdateien ... 648
28.4.4 Die Mitarbeiterliste ... 651
28.4.5 Die Mitarbeitersuche ... 652
28.4.6 Eingabe eines neuen Mitarbeiters ... 654
28.5 Alternative: CMS ... 655
28.6 phpMyAdmin ... 655
28.7 Zusammenfassung und Weiterführendes ... 657

Teil 3 Der Root-Server ... 659

29 Apache Teil II – Der Webserver im Internet-Einsatz ... 663
29.1 Virtuelle Hosts ... 663
29.1.1 Wie funktionieren virtuelle Hosts? ... 664
29.1.2 Workshop: Virtuelle Hosts einrichten ... 664
29.2 HTTPS ... 668
29.2.1 Wie funktioniert SSL? ... 668
29.2.2 Workshop: Den Apache SSL-fähig machen ... 673
29.3 Serverüberwachung ... 681
29.3.1 Überwachung von CPU, HD, RAM etc. ... 682
29.3.2 Webalizer ... 683
29.4 Bandbreite einsparen ... 685
29.4.1 Die Funktionsweise von mod_deflate ... 685
29.4.2 Das Modul mod_deflate einbinden ... 686
29.5 Zusammenfassung und Weiterführendes ... 686

30 DNS – Namensauflösung im Internet ... 687
30.1 Das Lab ... 688

30.2 Das Szenario . . . 688
30.3 Einführung in das Domain Name System . . . 689
30.3.1 Domains und Domainnamen . . . 689
30.3.2 Zonen und DNS-Servertypen . . . 691
30.3.3 Die DNS-Namensauflösung . . . 693
30.4 Installation von BIND9 . . . 694
30.5 Den DNS-Server mit rndc administrieren . . . 695
30.5.1 rndc konfigurieren . . . 695
30.5.2 rndc nutzen . . . 696
30.6 Workshop: Die DNS-Clients nutzen . . . 697
30.6.1 nslookup . . . 697
30.6.2 dig . . . 702
30.6.3 host . . . 704
30.6.4 /etc/resolv.conf . . . 704
30.7 Workshop: Die erste Zone einrichten . . . 706
30.7.1 Die Dateistruktur von BIND9 unter Debian . . . 707
30.7.2 Einrichten der Zonendatei . . . 707
30.7.3 Die Zone definieren und einbinden . . . 710
30.7.4 Die neue Zone testen . . . 711
30.8 Workshop: Eine reverse Zone erstellen . . . 712
30.8.1 Einrichten der Zonendatei . . . 712
30.8.2 Die Zone definieren und einbinden . . . 713
30.8.3 Die neue Zone testen . . . 713
30.9 Einen sekundären Server aufsetzen . . . 714
30.10 DNS-Sicherheit . . . 715
30.10.1 Den Zonentransfer auf bestimmte Slaves beschränken . . . 716
30.10.2 Die Authentizität des Masters sicherstellen . . . 716
30.10.3 Weitere Sicherheitseinstellungen . . . 719
30.10.4 DNSSEC . . . 720
30.11 Workshop: DDNS . . . 721
30.11.1 Vorbereitungen und Voraussetzungen . . . 721
30.11.2 Den DHCP-Server konfigurieren . . . 722
30.11.3 Den DNS-Server konfigurieren . . . 723
30.11.4 Den Client konfigurieren . . . 724
30.11.5 Die Arbeit der Serverdienste kontrollieren . . . 724
30.12 Zusammenfassung und Weiterführendes . . . 725

31 Lokaler E-Mail-Server mit Content-Filter . . . 727
31.1 Das Szenario . . . 728
31.2 Das Lab . . . 728
31.3 Grundlagen der E-Mail-Kommunikation . . . 729
31.3.1 SMTP – Das Mail-Protokoll . . . 729
31.3.2 Der Weg einer E-Mail . . . 729
31.3.3 POP3 . . . 731
31.3.4 IMAP4 . . . 731
31.3.5 ESMTP . . . 732

31.3.6 Unix to Unix Copy (UUCP) . . . 732
31.3.7 ... und was ist MAPI? . . . 732
31.3.8 Weitere wichtige E-Mail-Konzepte . . . 732
31.4 Installation von Postfix . . . 734
31.5 Wie funktioniert Postfix? . . . 736
31.5.1 Stoppen und Starten von Postfix . . . 736
31.5.2 Lokale Mails . . . 736
31.5.3 Mails aus dem Netzwerk . . . 737
31.5.4 Der Queue-Manager . . . 738
31.5.5 Weiterleitung der Mail . . . 738
31.5.6 ... und wo sind die Dateien und Verzeichnisse von Postfix? . . . 738
31.6 Workshop: Ein interner Mail-Server mit Postfachabholung . . . 740
31.6.1 Grundkonfiguration des Mail-Servers . . . 741
31.6.2 E-Mail-Benutzer einrichten . . . 742
31.6.3 Den E-Mail-Client konfigurieren und testen . . . 742
31.6.4 Die Postfächer überprüfen . . . 745
31.6.5 Mails vom Provider abholen . . . 746
31.6.6 Einen POP3-Server einrichten . . . 747
31.7 Workshop: Content-Filter einrichten . . . 749
31.7.1 ClamAV auf dem Mail-Server einrichten . . . 749
31.7.2 AMaViS einbinden . . . 751
31.7.3 Spamschutz mit SpamAssassin . . . 758
31.7.4 Unerwünschte Dateitypen blockieren . . . 759
31.8 Weitere Schritte . . . 761
31.8.1 Hinter den Kulissen von AMaViS . . . 761
31.8.2 Tuning-Parameter von AMaViS . . . 761
31.8.3 Das Verhalten von AMaViS anpassen . . . 762
31.8.4 Das Quarantäneverhalten anpassen . . . 763
31.8.5 Grundüberlegungen zu SpamAssassin . . . 763
31.8.6 Wie arbeitet SpamAssassin? . . . 764
31.9 Zusammenfassung und Weiterführendes . . . 765

32 Internet-Mail-Server mit SMTP-Authentication . . . 767
32.0.1 Das Szenario . . . 768
32.1 Das Lab . . . 768
32.2 Administration der Mail-Queues . . . 768
32.2.1 Der Weg einer Mail durch die Mail-Queues . . . 768
32.2.2 Administrationstools . . . 769
32.3 Mappings und Lookup-Tables . . . 770
32.3.1 /etc/aliases . . . 771
32.3.2 postmap . . . 771
32.3.3 Einbinden der Mapping-Tabellen . . . 771
32.3.4 Einfache Listen in Dateien auslagern . . . 772
32.3.5 Canonical-Adressen . . . 772
32.3.6 Relocated-Adressen . . . 772

32.4 Mailbox-Formate ... 773
32.4.1 mbox ... 773
32.4.2 maildir ... 773
32.5 Mehrere Domains verwalten ... 774
32.5.1 Gemeinsame Domains mit lokalen Benutzern ... 774
32.5.2 Getrennte Domains mit lokalen Benutzern ... 775
32.5.3 Getrennte Domains mit virtuellen Accounts ... 776
32.6 Workshop: Virtuelle Domains und POP3/IMAP-Server ... 778
32.6.1 Virtuelle Domains mit lokalen Benutzern ... 778
32.6.2 Virtuelle Domains mit virtuellen Mailboxen ... 786
32.7 Workshop: SMTP-Authentication mit Cyrus SASL ... 790
32.7.1 Was ist SASL? ... 790
32.7.2 Die Authentifikationsmethode ... 790
32.7.3 Das Authentifikationssystem ... 791
32.7.4 Cyrus-SASL installieren ... 791
32.7.5 Grundüberlegungen zur Authentifikation ... 792
32.7.6 SMTP-Auth mit sasldb ... 793
32.7.7 SMTP-Auth mit lokalen Benutzern ... 796
32.8 Mail-System mit MySQL-Backend ... 798
32.9 Einen Webmailer einrichten ... 798
32.10 Zusammenfassung und Weiterführendes ... 801

33 FTP – Dateiübertragung im Internet ... 803
33.1 Szenario ... 804
33.2 Das Lab ... 804
33.3 Wie funktioniert FTP? ... 804
33.3.1 Aktives FTP ... 804
33.3.2 Passives FTP ... 805
33.3.3 Binary- und ASCII-Modus ... 805
33.3.4 Anonymous FTP ... 806
33.4 Installation von ProFTPD ... 806
33.5 Grundkonfiguration von ProFTPD ... 807
33.6 Workshop: Eine FTP-Sitzung ... 809
33.7 Workshop: Erweiterte Konfiguration ... 813
33.7.1 Ein FTP-Home-Verzeichnis zuweisen ... 813
33.7.2 Sitzungen begrenzen ... 815
33.7.3 Ein Nur-Lesen-Verzeichnis einbinden ... 815
33.8 Anonymous-FTP ... 816
33.9 Virtuelle Benutzer ... 818
33.10 Virtuelle FTP-Hosts ... 819
33.11 Zusammenfassung und Weiterführendes ... 820

34 iptables als Personal-Firewall ... 821
34.1 Das Lab ... 822
34.2 Firewall-Grundlagen ... 822
34.2.1 Die Paketfilter-Firewall ... 822

34.2.2 Die Stateful-Inspection-Firewall ... 822
34.2.3 Die Application-Layer-Firewall ... 822
34.2.4 Die Personal-Firewall ... 823
34.2.5 Die Netzwerk-Firewall ... 823
34.2.6 Netfilter/iptables ... 823
34.3 Wie funktionert iptables? ... 824
34.3.1 Tables – die Tabellen ... 824
34.3.2 Chains – die Regelketten ... 824
34.3.3 Rules – die Filterregeln ... 824
34.3.4 Policies – die Richtlinien ... 825
34.4 Workshop: Ein Firewall-Skript erstellen ... 825
34.4.1 Das Skript vorbereiten ... 825
34.4.2 Globale Operationen ... 826
34.4.3 Lokale Kommunikation ... 827
34.4.4 Targets ... 828
34.4.5 Stateful Inspection ... 828
34.4.6 Selbst erstellte Chains und Logging ... 829
34.4.7 Verschiedene Dienste freigeben ... 830
34.4.8 FTP – der »Firewall-Killer« ... 833
34.4.9 Das Firewall-Skript ... 835
34.5 Firewall Builder – Frontend zu iptables ... 836
34.6 Zusammenfassung und Weiterführendes ... 837

Teil 4 Linux als Gateway ... 839

35 Linux als Router ... 843
35.1 Wie funktioniert Routing? ... 843
35.2 Statisches und dynamisches Routing ... 846
35.2.1 Statische Routen eintragen ... 846
35.2.2 Das Standardgateway ... 847
35.2.3 Dynamisches Routing ... 848
35.3 Einen Router einrichten und konfigurieren ... 849
35.4 Der Weg ins Internet ... 850
35.4.1 Ein vorhandener ISDN/DSL-Router ... 850
35.4.2 Einen DSL-Anschluss einrichten ... 851
35.4.3 Einen DNS-Caching-Server einrichten ... 857
35.4.4 IP-Masquerading einrichten ... 857
35.5 Zusammenfassung und Weiterführendes ... 859

36 iptables als Netzwerk-Firewall ... 861
36.1 Das Szenario ... 862
36.2 Wozu eigentlich eine DMZ? ... 862
36.3 Aufbau der Laborumgebung ... 863

36.4 iptables als Netzwerk-Firewall . . . 864
36.5 Aufbau des DMZ-Servers für die Laborumgebung . . . 864
36.6 Grundgerüst des Firewall-Skripts . . . 865
36.6.1 Vorarbeiten . . . 865
36.6.2 Chains vorbereiten . . . 867
36.6.3 Stateful Inspection . . . 867
36.6.4 Loopback-Kommunikation . . . 868
36.6.5 Antispoofing-Regeln . . . 868
36.6.6 Das Grundgerüst zusammengefasst . . . 869
36.7 Das Firewall-Regelwerk – normale Regeln . . . 871
36.7.1 NetBIOS und RPC . . . 871
36.7.2 Kommunikation von und zum Gateway . . . 871
36.7.3 Web, FTP, Mail, POP3 und SSH . . . 872
36.7.4 DNS – Namensauflösungen . . . 873
36.8 SNAT, DNAT und MASQUERADING . . . 873
36.8.1 SNAT und Masquerading . . . 873
36.8.2 DNAT . . . 875
36.8.3 NAT mit iptables . . . 876
36.9 Die letzte Regel . . . 877
36.10 Das Firewall-Skript im Ganzen . . . 878
36.11 DynDNS – immer über den eigenen Namen erreichbar . . . 880
36.11.1 Bei einem DynDNS-Provider anmelden . . . 880
36.11.2 Den DynDNS-Client installieren . . . 883
36.12 Zusammenfassung und Weiterführendes . . . 885

37 Squid-Proxyserver . . . 887
37.1 Die Laborumgebung . . . 887
37.2 Das Szenario . . . 888
37.3 Wie arbeitet ein Proxy? . . . 889
37.3.1 Proxy-Grundlagen . . . 889
37.3.2 Warum einen Proxy einsetzen? . . . 890
37.3.3 Squid – der HTTP- und FTP-Proxyserver . . . 890
37.4 Squid installieren . . . 890
37.5 Grundkonfiguration von Squid . . . 891
37.5.1 Aufbau der Datei squid.conf . . . 893
37.5.2 Portbindung und andere Netzwerkoptionen . . . 894
37.5.3 Parents einrichten . . . 895
37.5.4 Den Cache konfigurieren . . . 896
37.5.5 Cache-Verzeichnisse und Logging . . . 897
37.5.6 FTP- und DNS-Einstellungen . . . 899
37.6 Zugriffssteuerung via Access-Lists . . . 899
37.7 Authentifizierung . . . 900
37.8 URL-Filter mit Squid . . . 902
37.9 Zusammenfassung und Weiterführendes . . . 903

Teil 5 Server-Security ... 905

38 Das Serversystem härten ... 911
38.1 Installation des Betriebssystems und der Dienste ... 911
38.1.1 Partitionierung ... 911
38.1.2 Das Dateisystem ... 912
38.1.3 Installation des Grundsystems ... 912
38.1.4 Weitere Maßnahmen bei der Installation ... 913
38.1.5 Dienste reduzieren ... 914
38.2 Nach der Installation ... 914
38.2.1 Das System updaten ... 915
38.2.2 LILO und GRUB sichern ... 915
38.2.3 BIOS-Einstellungen ... 916
38.2.4 Der physische Standort ... 917
38.3 Dienste absichern ... 917
38.3.1 SSH ... 918
38.3.2 Apache Webserver ... 918
38.3.3 Squid ... 922
38.3.4 FTP ... 922
38.3.5 DNS-Server ... 923
38.4 Weitere Maßnahmen ... 924

39 Einbruchserkennung mit Intrusion-Detection-Systemen ... 925
39.1 Wie funktioniert ein IDS? ... 925
39.1.1 Host-Instrusion-Detection-Systeme (HIDS) ... 926
39.1.2 Network-Intrusion-Detection-Systeme (NIDS) ... 927
39.1.3 Intrusion-Prevention-Systeme (IPS) ... 928
39.1.4 Vor- und Nachteile von ID/IP-Systemen ... 928
39.2 Tripwire ... 929
39.2.1 Tripwire installieren ... 929
39.2.2 Die Tripwire-Dateien ... 932
39.2.3 Die Konfigurationsdatei twcfg.txt ... 933
39.2.4 Die Policy-Datei twpol.txt ... 934
39.2.5 Praxistipps zur Konfiguration ... 937
39.2.6 Tripwire initialisieren ... 937
39.2.7 Eine Integritätsprüfung durchführen ... 938
39.2.8 Die Tripwire-Datenbank aktualisieren ... 939
39.3 Snort ... 940
39.3.1 Wie funktioniert Snort? ... 940
39.3.2 Snort installieren ... 941
39.3.3 Snort-Konfiguration – Ein Überblick ... 942
39.3.4 Snort testen ... 946
39.3.5 Snort updaten ... 946
39.4 Zusammenfassung und Weiterführendes ... 948

40 Desaster Recovery ... 949

Stichwortverzeichnis ... 959

Einleitung

Herzlich Willkommen! Mit diesem Buch halten Sie einen Lehrgang über die System- und Netzwerkadministration von Linux-Servern im Allgemeinen und Debian GNU/Linux im Besonderen in den Händen. Zwar stelle ich die Administration eines Linux-Servers am Beispiel von Debian GNU/Linux vor, jedoch ist nur der geringste Teil des Buches Debian-spezifisch. Auch wenn Sie eine andere Linux-Distribution, wie zum Beispiel OpenSUSE oder Fedora, nutzen, werden Sie großen Nutzen aus diesem Buch ziehen können!

Die meisten Bestandteile eines Linux-Systems, wie zum Beispiel der Kernel, die Shell, die Linux-Befehle und die Serverdienste, sind distributionsübergreifend vorhanden. Die Entwickler einer Distribution passen nur bestimmte, im Grunde recht überschaubare Details im System an. So sind zum Beispiel die Startskripte teilweise unterschiedlich aufgebaut. Während die eine Distribution in der Voreinstellung in den Runlevel 2 bootet, ist dies bei anderen Distributionen Runlevel 3. Natürlich gibt es weitere Änderungen und jede Distribution bringt ihre eigenen Konfigurationstools mit. Wenn man jedoch von diesen Details abstrahiert, bleibt ein Linux-System ein Linux-System – egal, ob es den Namen Debian, OpenSUSE oder Fedora trägt! Dies gilt umso mehr im Serverbereich, da der Administrator hier ohnehin sein individuelles, auf die Erfordernisse abgestimmtes, Serversystem aufbauen wird.

Daher liegt einer der Schwerpunkte dieses Buches darauf, die Hintergründe und Funktionsweisen der Systeme zu durchleuchten – wer sein System kennt, kann die Administrationsaufgaben wesentlich effektiver bewältigen als jemand, der nur über oberflächliches Wissen verfügt.

Für wen ist dieses Buch geeignet?

Das Buch ist genau das richtige für Sie, wenn Sie als Poweruser bereits ein wenig Erfahrung in der Betreuung von PCs und eventuell sogar kleinen Serversystemen haben – egal ob Windows oder Linux – und sich nun systematisch in die System- und Netzwerkadministration von Linux-Servern (vor allem unter Debian GNU/Linux) einarbeiten möchten. Es ist in erster Linie zum »Mitmachen« konzipiert. Viele Bücher über das Thema Linux-Systemadministration handeln die Themen inhaltlich ab, überlassen es aber der Eigeninitiative des Lesers, die angebotenen Inhalte in der einen oder anderen Form in die Praxis umzusetzen.

Aus meiner Erfahrung als Kursleiter und Dozent für Fachinformatiker für Systemintegration weiß ich jedoch, dass es für den Lernenden oftmals schwierig ist, einen praxisnahen Übungsansatz zu finden. Und so bleibt der angelesene Stoff nicht lange hängen und verflüchtigt sich schnell wieder.

Dieses Problem versuche ich in diesem Buch dadurch zu lösen, dass ich Ihnen in vielen Kapiteln Workshops mit Schritt-für-Schritt-Anleitungen und zum Teil weitergehenden

Übungen anbiete. Ich kann Sie nur immer wieder ermutigen, mir nichts zu glauben, bis Sie es nicht selbst nachgeprüft haben – nur wenn Sie tatsächlich mit dem Serversystem arbeiten, werden Sie Ihren Server kennen lernen.

Dabei erstellen wir gemeinsam ein Szenario, in dem Sie der Administrator bzw. die Administratorin des expandierenden *Architekturbüros Windschief* sind. Je nach Situation ergeben sich immer neue Herausforderungen, denen Sie sich als Administrator gegenübersehen. Dabei werden Sie die verschiedensten Serverdienste aufsetzen und konfigurieren, um – je nach Grundszenario – einen kompletten Linux-Server aufzubauen.

Der Aufbau dieses Buches

Ziel des Buches ist es, Ihnen die notwendigen Grundlagen zur Administration eines (Debian-)Linux-Servers in unterschiedlichen Umgebungen zu verschaffen. Dazu ist das Buch in fünf Teile gegliedert, die jeweils unterschiedliche Aspekte bzw. Anwendungsbereiche eines Servers beleuchten. Die Inhalte der Teilbereiche werden weiter unten erläutert. Neben dem allgemeinen (ersten) Teil habe ich für Sie drei typische Szenarien entworfen, die Ihnen in dieser oder abgewandelter Form in der Praxis begegnen könnten:

- Backoffice-Server
- Root-Server
- Linux als Gateway

Ich habe versucht, möglichst lebendige Szenarien im Rahmen des bereits eingangs erwähnten *Architekturbüros Windschief* zu entwickeln, um Ihnen eine Identifikation mit den gestellten Administrationsaufgaben zu erleichtern. Sie werden Ihren Server gemäß den sich ändernden Anforderungen schrittweise aufbauen und erweitern.

Last but not least werden wir im fünften Teil unseren Fokus auf die Sicherheit unseres Servers legen, da dies ein elementarer Bestandteil der Serveradministration ist.

> Wir arbeiten in der neuen Auflage mit *Wheezy* (Debian 7), der zurzeit aktuellen Version von Debian.

Das Buch ist als Lehrbuch konzipiert, die einzelnen Kapitel bauen also an einzelnen Stellen aufeinander auf. Arbeiten Sie das Buch von Anfang bis Ende durch, werden Sie einen sehr guten Einblick in die Arbeit als Administrator eines Linux-Servers bekommen haben. Andererseits sind die einzelnen Themenbereiche klar voneinander abgegrenzt, so dass Sie dieses Buch auch später als Nachschlagewerk verwenden können, zumal wir in den Kapiteln auch über den Tellerrand hinausschauen werden, um zu sehen, was uns das eine oder andere Programm über die Anforderungen unseres Szenarios hinaus noch bieten kann.

Lassen Sie uns einen Blick auf die Inhalte der fünf Teilbereiche des Buches werfen.

Teil 1 – Allgemeine Systemadministration

Wir werden klassisch starten: Zunächst lernen Sie, Ihr Debian-System zu installieren. Sollten Sie eine andere Distribution nutzen, werden Sie an dieser Stelle abweichende Installationsschritte durchführen müssen. Anschließend werden wir uns mit dem Paketmanage-

ment von Debian beschäftigen – ein sehr leistungsfähiges, aber gewöhnungsbedürftiges Konzept.

Sie lernen das (Debian-)Linux-System aus der Administratorsicht kennen – wo befindet sich was, wie sind die Runlevel organisiert usw. Wir schauen hier auch auf Gemeinsamkeiten und Unterschiede zu anderen Distributionen.

Anschließend steigen wir in die Benutzerverwaltung ein. Diesem Bereich kommt auf einem Server eine große Bedeutung zu, da naturgemäß mehrere User auf unseren Server zugreifen und normalerweise die Systemaccounts zur Authentifizierung und Zugriffsberechtigung verwendet werden.

Sie lernen wichtige Befehle zur Systemadministration kennen, damit Sie in der Lage sind, Ihr System zu beherrschen. Darüber hinaus werden wir einige Grundlagen zur Shellskript-Programmierung schaffen, damit Sie Routine-Aufgaben auch automatisieren können. Fast alle Prozeduren auf einem Linux-System sind durch Shellskripte realisiert. Diese Shellskripte greifen auf genau die Programme zurück, die Sie vorher als äußerst nützliche Administrationstools kennen gelernt haben.

Haben Sie diese Grundlagen gemeistert, wird es Zeit, in das Herz Ihres Servers zu schauen: den Kernel. Sie werden lernen, den Kernel unseren persönlichen Bedürfnissen anzupassen, ihn zu »customizen«. Das ist anspruchsvoll, aber unter bestimmten Bedingungen sehr nützlich. Da es sich hier sozusagen um eine »Herzoperation« handelt, ist äußerste Konzentration und Genauigkeit vonnöten – nichts, was man mal eben nebenbei erledigen sollte.

Leider läuft nicht immer alles nach Plan – für diesen Fall ist das Logfile Ihr bester Freund. Wird es normalerweise stiefmütterlich behandelt, werden wir es hegen und pflegen. Sie werden die Logfiles Ihres Systems immer wieder konsultieren, um Fehler zu finden und sich über den Status des Systems zu informieren.

In diesem Zusammenhang ist eine passende Backup-Strategie absolut essenziell. Im Falle eines Desasters, das im Allgemeinen immer dann auftaucht, wenn man es am wenigsten gebrauchen kann, können Sie kalt lächelnd Ihre Daten wiederherstellen.

Ein X-Window-System ist für einen Server in der Regel nicht notwendig und unter Umständen sogar nicht erwünscht. Andererseits kann es die Administration deutlich vereinfachen. Wir schauen uns an, was das X-Window-System für uns tun kann.

Anschließend geht es um die Netzwerkgrundlagen. Sie lernen TCP/IP etwas genauer kennen und werden Ihren Server fit für die Netzwerkkommunikation machen. Außerdem werden Sie ab diesem Zeitpunkt in der Lage sein, Ihren Server »remote«, also aus der Ferne zu administrieren – SSH macht es möglich. Jetzt sind Sie bereit für das große Abenteuer!

Teil 2 – Der Backoffice-Server

Unser erstes Szenario führt uns in das lokale Netz des *Achitekturbüros Windschief*. Hier soll ein Server das bisherige Peer-to-Peer-Konzept (alle Workstations sind gleichberechtigt) ablösen. Wir werden also Schritt für Schritt typische Dienste in einem solchen Umfeld einführen.

Zunächst überlassen wir die Client-Netzwerkkonfiguration dem DHCP-Server. Dies schafft dem Admin – das sind Sie – mehr Luft für andere Dinge, da er nicht jeden Client einzeln konfigurieren muss.

Danach üben wir tanzen – Samba, um genau zu sein: Dieser Dienst leistet erstklassige Arbeit bei der Integration von Linux und Windows, da er nicht nur einen Datei- und Druckserver für Windows-Clients bereitstellt, sondern darüber hinaus auch noch als Domänencontroller Chef einer Windows-Domäne werden kann.

Ein Intranet muss her! Dabei hilft uns der »Indianer« – sprich: der Apache Webserver. Wir werden ein einfaches Intranet aufbauen, um die Grundfunktionen des Apache kennen zu lernen. Im nächsten Teil werden wir den Webserver ein bisschen gründlicher unter die Lupe nehmen.

Was in einem Unternehmen nicht fehlen darf, ist eine Datenbank – genauer: ein relationales Datenbank-Management-System. Unsere Wahl fällt auf MySQL, weil es einfach zu bedienen und weit verbreitet ist. Außerdem ist es erste Wahl für Webdatenbanken. Damit können wir das Backend für unser Intranet schaffen.

Um das Backend (MySQL) mit dem Frontend (Apache) zu verbinden, benötigen wir einen Vermittler. Hier bietet sich PHP als serverseitige Skriptsprache an, da PHP eine sehr ausgereifte Schnittstelle zu MySQL bereitstellt.

Teil 3 – Der Root-Server

Inzwischen sind sie für fast jeden Geldbeutel erschwinglich und werden immer beliebter: die Root-Server. Dabei handelt es sich um einen dedizierten Server, der im Rechenzentrum eines Providers installiert wird. Sie als »Mieter« dieses Servers haben alleinigen Root-Zugriff und können diesen genau so konfigurieren, wie Sie einen Rechner bei sich zu Hause oder in Ihrem Unternehmen einrichten könnten. Die Administration erfolgt über SSH, also konsolenbasiert. Sie haben oftmals auch die Möglichkeit, den Server über eine Weboberfläche zu verwalten, aber das werden wir nicht weiter betrachten, da sich die interessanten Aspekte hier Blackbox-artig hinter den Menüs und Dialogfenstern verstecken.

Mit einem solchen Server werden Sie selbst zum Provider mit allen Rechten und Pflichten: Sie können verschiedene Domains bzw. Websites hosten, sind für die DNS-Einträge verantwortlich und müssen den E-Mail-Verkehr abwickeln. Eine spannende und sehr anspruchsvolle Aufgabe, vor allem, wenn Sie zahlende Kunden haben.

Da darf nichts schiefgehen, da hier unter Umständen SLAs (Service Level Agreements) greifen, die Sie mit Ihren Kunden vereinbart haben – zum Beispiel 99,5-prozentige Verfügbarkeit, Datenwiederherstellung innerhalb von vier Stunden o. Ä. Aber selbst wenn nicht, dürfte die Verfügbarkeit der Internetpräsenz eine sehr wichtige Rolle spielen, so dass die Ausfallzeit generell so gering wie möglich zu halten ist – Experimente sollten Sie also lieber auf einem Testsystem zu Hause machen und Änderungen am Produktivsystem erst vornehmen, wenn Sie alles getestet haben.

Wir werden uns in diesem Abschnitt mit der Konfiguration des Apache für mehrere Domains beschäftigen. Außerdem gehe ich auf weitere Funktionalitäten wie zum Beispiel die Unterstützung für verschiedene Zusatzfunktionen (unter anderem SSL/TLS) ein und wir werden unseren Webserver tunen, um auch höherem Datenverkehr gerecht zu werden.

Außerdem werden wir einen DNS-Server aufbauen, der die gehosteten Domains verwaltet. Mein ehemaliger Chef sagte einmal zu mir: »Bei Einstellungsgesprächen frage ich die Kandidaten immer über DNS aus – wer DNS versteht, versteht das Internet!« DNS ist ein anspruchsvolles und essenzielles Konzept, und wir werden es durchleuchten.

Dagegen ist die wichtigste Anwendung im Internet noch immer E-Mail. Die Konfiguration eines E-Mail-Servers gehört zu den anspruchsvollsten Aufgaben eines Netzwerkadministrators. Wir werden Postfix statt den von Debian standardmäßig installierten Exim-Mailserver nutzen, da Postfix verbreiteter und besser dokumentiert ist. Leistungsfähig und relativ einfach zu konfigurieren sind sie beide.

Außer den bunten Bilderchen und E-Mail benötigen Sie allerdings oft auch noch FTP, zum Beispiel um die Übertragung der Dokumente einer Webpräsenz vom Client auf den Server zu ermöglichen. Außerdem ist FTP bis heute der Standard in Sachen Datenübertragung.

Last but not least müssen Sie Ihren Server absichern. Wir werden wichtige Sicherheitsaspekte schon im Rahmen der einzelnen Serverdienste untersuchen, jedoch gibt es auch globale Pflichtmaßnahmen – eine davon lernen Sie hier kennen: die Firewall. Mit `iptables` liefert Linux eine sehr brauchbare Paketfilter- bzw. Stateful Inspection-Firewall. An dieser Stelle werden wir zunächst eine Personal-Firewall aufsetzen, während ich im nächsten Teil auf `iptables` als Netzwerk-Firewall eingehen werde.

Teil 4 – Linux als Gateway

In letzter Zeit sind DSL-Router sogar mit WLAN-Funktionalität so preisgünstig geworden, dass viele dazu übergehen, eine vormals installierte Linux-Lösung abzulösen. Dennoch hat die Linux-Gateway-Lösung noch immer viele Vorteile, die ich in diesem Teil beleuchten werde.

Sie werden lernen, wie Sie aus Linux einen Router machen, eine Netzwerk-Firewall mit `iptables` aufbauen und einen DNS-Caching-Server installieren. Darüber hinaus werden wir uns `Squid` – den bekanntesten Proxy-Server für Linux – ansehen und für unser Netzwerk nutzbar machen.

Haben Sie keine feste IP-Adresse – was der Häufigkeitsfall sein dürfte –, möchten aber trotzdem immer erreichbar sein, bietet sich DynDNS an. Damit ist es möglich, einen festen Domainnamen (zum Beispiel `hansjuergen.dyndns.org`) jeweils auf die aktuelle, dynamisch vom Provider zugewiesene Adresse Ihres Routers auflösen zu lassen.

Teil 5 – Server-Security

Sie haben bis zu diesem Zeitpunkt schon einiges über Sicherheit gelernt, da wir die wichtigsten spezifischen Sicherheitsmaßnahmen der einzelnen Dienste bereits an Ort und Stelle untersucht haben. Jedoch gibt es darüber hinaus noch einige allgemeine wichtige Maßnahmen, die Sie treffen sollten. Diese beleuchte ich in diesem Abschnitt.

Dazu gehört die Härtung Ihres Linux-Servers. Darunter versteht man geeignete Maßnahmen, um das System so unangreifbar wie möglich und andererseits so zugänglich wie nötig zu machen – eigentlich ein eigenes Buch.

Mit `Tripwire` können Sie erkennen, ob sich jemand an Ihren Systemdateien zu schaffen gemacht hat. Hierbei handelt es sich um ein Host Intrusion Detection-System (HIDS). Wir werden `Tripwire` installieren und konfigurieren.

Des Weiteren werden wir eine »Netzwerkkamera« in Form eines NIDS (Network Intrusion Detection-System) installieren. Ein NIDS kann als solches keine Angriffe verhindern, aber

Warnungen ausgeben, wenn ein Angriff – oder die Vorbereitung eines solchen – erkannt wird. Dafür nutzen wir `Snort`, das Schwein mit der Riesennase.

Bevor wir zum Ende kommen, möchte ich mit Ihnen noch einen sehr wichtigen Aspekt besprechen: Desaster-Recovery. Der Standardfall sieht in etwa folgendermaßen aus: Eine kritische Komponente versagt ihren Dienst, keiner weiß so richtig, was zu tun ist, und alles rennt hektisch durcheinander. Die so genannte *Downtime* (die Zeit, in der der Server nicht verfügbar ist) wird dadurch unnötig verlängert.

Mit einem guten Notfallplan können Sie einem solchen Fall jedoch relativ entspannt entgegensehen. Der Plan ermöglicht es Ihnen, im Notfall schnell und effizient zu reagieren und die Downtime auf ein Minimum zu beschränken.

Konventionen in diesem Buch

Wo vorhanden, habe ich versucht, mich an Standards in der Darstellung zu halten. Leider ist dies nicht überall möglich. In diesem Buch gelten folgende Darstellungsregeln:

Die Syntax eines Befehls sieht folgendermaßen aus:

```
Befehl <Pflichtangabe> [<Freiwillige Optionen oder Parameter>]
```

In jedem Fall werden Pflichtangaben in spitze, freiwillige Angaben in eckige Klammern gefasst. Ersetzungen, also Angaben, die Sie einsetzen müssen, werden ebenfalls generell in spitze Klammern gefasst, im Befehl oder im Konfigurationsparameter entfallen die Klammern generell, wenn nicht anders angegeben.

Befehle und Eingaben sind **`fett`** gedruckt, Ausgaben von Befehlen und Dateilistings `nicht fett`. Spezielle Begriffe sind – je nach Kontext – *kursiv* oder in »Anführungszeichen« gesetzt.

Für Passwörter habe ich pauschal immer sechs fettgedruckte Asteriske (******) gesetzt. Ihr Passwort sollte tunlichst länger sein, mindestens acht oder besser zwölf Zeichen! Bei Linux-Programmen sehen Sie in der Regel die Passworteingabe nicht auf dem Bildschirm, weder maskiert noch überhaupt als Zeichen. Die Passworteingabe wird jedoch immer regulär mit einem ↵ abgeschlossen.

Zum Thema Kommentieren: Bei Linux ergibt sich häufig die Notwendigkeit, in einer Konfigurationsdatei eine Zeile aktiv oder inaktiv zu setzen. Dies geschieht durch das Entfernen bzw. Hinzufügen von Kommentarzeichen (meistens #) als erstes Zeichen der Zeile. Da es keine einheitliche Meinung zur Aussage von Ein- und Auskommentieren gibt, habe ich in den entsprechenden Fällen immer hinzugeschrieben, um welchen Schritt es sich handelt.

Warum Debian GNU/Linux?

Ich hatte Ihnen eingangs erläutert, warum Sie auch großen Nutzen aus diesem Buch ziehen können, wenn Sie eine andere Linux-Distribution als Debian GNU/Linux verwenden möchten (oder müssen). Dennoch kann ich Ihnen Debian GNU/Linux nur wärmstens ans Herz legen, wenn Sie eine zuverlässige und stabile Serverplattform benötigen. Doch gibt es noch andere Gründe, die für Debian sprechen, wie Sie im Folgenden lesen können.

Es stimmt! Debian ist nicht so wie andere Linux-Distributionen! Der kleine, aber wichtige Unterschied liegt im Detail:

- Debian beinhaltet ausschließlich freie Software und steht als Ganzes unter der GPL (GNU Public License) – andere Distributionen beinhalten häufig kommerzielle Software-Pakete. Einerseits ist dies schade, weil man auf einige schöne Prograämmchen verzichtet hat, andererseits bleiben Sie als Anwender in jedem Fall lizenztechnisch (im Rahmen der GPL) auf der sicheren Seite. Lesen Sie weiter unten, was es mit der GPL auf sich hat.
- Debian bietet in der Standardversion (stable) nicht die neueste, sondern stets ausgereifte und ausführlich getestete Software. Das liegt an der Philosophie der Debian-Entwickler, ein möglichst stabiles Betriebssystem bereitzustellen. Dagegen bieten andere Distributionen wie SuSE (bzw. OpenSUSE) und Red Hat (bzw. Fedora), deren Versionsrad sich inzwischen immer schneller dreht, immer die neuesten Versionen einer Software. Mit dem Resultat, dass diese Systeme sich in manchen Situationen in puncto Stabilität – etwas ketzerisch formuliert – langsam hinter Windows einreihen müssen ... und das mag schon etwas heißen. Im Übrigen ist es auch Debian-Nutzern möglich, sich immer die neueste, schönste und schnellste Software zu besorgen. Das ist aber nur unter ganz bestimmten Umständen sinnvoll. Ich gehe weiter unten darauf ein.
- Debian als Projekt ist nicht kommerziell – das heißt, es wird ausschließlich von engagierten Linux-Programmierern in deren Freizeit erstellt und weiterentwickelt. Diese erhalten dafür kein Geld – allenfalls einen warmen Händedruck. Vielen Dank an dieser Stelle an die vielen idealistischen Programmierer und Betreuer, die ihre Freizeit dafür opfern, dieses professionelle Betriebssystem zu pflegen!
- Debian nutzt ein eigenes Paket-Management-System namens `dpkg`. Obgleich es vielleicht das leistungsfähigste System ist, benötigt ein Einsteiger doch etwas Eingewöhnungszeit, da es sich vom mehr verbreiteten RPM-System unterscheidet – ich werde versuchen, diese Eingewöhnungszeit so kurz wie möglich zu halten und Ihnen zu zeigen, wie Sie hocheffizient damit arbeiten können. Kennen Sie es erst einmal, werden Sie es lieben!
- Debian ist an einigen Stellen nicht wirklich bequem in der Konfiguration. Aber andere Linux-Distributionen sind dies auch nicht, wenn man ein wenig mehr als die Standardkonfiguration möchte, da man dann auch Hand anlegen muss, anstatt die schicken Frontends wie zum Beispiel YaST von SuSE zu nutzen. Wer sein System per Hand konfiguriert, weiß, was läuft! Das ist einer der Gründe, warum Experten oftmals auf Debian schwören: Ein Debian-System fordert anfangs zwar etwas mehr Eingewöhnungszeit, aber wenn es einmal »funzt« (funktioniert), dann wissen Sie auch, warum – Sie haben es nämlich selbst und höchst eigenhändig konfiguriert ...

Um den Kritikern den Wind aus den Segeln zu nehmen: Nein, Debian ist kein optimales Einsteigersystem, da es vom Anwender oft ein wenig mehr Know-how abverlangt, um ein lauffähiges System zu konfigurieren! Aber mit der entsprechenden Hilfestellung ist die Einarbeitung in Debian problemlos möglich. Dieses Buch wird Ihnen dabei helfen.

Im Übrigen handelt das Buch von Linux bzw. Debian GNU/Linux als Serversystem. Somit gehe ich natürlich von einem gewissen Grundwissen aus, da kaum ein Einsteiger gleich mit der Konfiguration eines Servers beginnen wird. Trotzdem werde ich versuchen, Ihnen alle nötigen Informationen zukommen zu lassen – die »Basics« als kurze Zusammenfassung, alles andere mehr oder weniger ausführlich.

Genug davon. Schauen wir uns also die Besonderheiten von Debian GNU/Linux einmal etwas genauer an. Oben war die Rede von einer GPL. Diese sagt im Kern Folgendes aus:

Statt der üblichen Einschränkungen einer Lizenz gewährt die GPL (GNU Public License) vier Freiheiten:

- Das Programm darf für jeden (auch kommerziellen) Zweck genutzt werden.
- Das Programm darf beliebig oft kopiert und kostenlos verteilt werden. Auf Anfrage muss der Quellcode dem Empfänger zur Verfügung gestellt werden.
- Das Programm darf beliebig verändert und angepasst werden, um den eigenen Bedürfnissen gerecht zu werden.
- Die geänderte Version darf ebenfalls kostenlos weitergegeben oder aber kommerziell vertrieben werden, immer unter der Maßgabe von Punkt 2.

Ein Großteil der Software in gängigen Linux-Distributionen steht unter der GPL. Allerdings erlaubt Debian im Gegensatz zu vielen anderen Distributionen keine Ausnahme von der GPL. Andere nützliche Software wie zum Beispiel Adobe Reader steht nicht unter der GPL. Diese müssen Sie sich aus anderen Quellen besorgen.

Die Releases und Versionen von Debian

Während es bei anderen Linux-Distributionen nur eine Versionsnummer gibt, ist Debian etwas anders aufgebaut. Daran muss man sich erst einmal gewöhnen.

Debian-Releases

Es gibt grundsätzlich drei aktuelle Versionen von Debian:

stable: Die aktuelle, offizielle Debian-Version. Hierbei handelt es sich durchweg um Software, die in umfangreichen Praxistests ihre Stabilität und Zuverlässigkeit unter Beweis gestellt hat. Diese Tests laufen nicht im Labor einer Softwareschmiede ab, sondern im täglichen Betrieb tausender Debian-User und -Entwickler. Sie können sicher sein, dass dieses Release über einen sehr hohen Reifegrad verfügt. Der Nachteil: Die Software ist teilweise veraltet. Dies wirkt sich allerdings nur dann aus, wenn Sie Features einer neueren Version nutzen möchten, die in der älteren, distributionseigenen Version noch nicht vorhanden ist. Im Serverbereich ist das eher selten.

testing: Diese, für Workstations vielleicht beliebteste, Debian-Distribution enthält recht aktuelle Pakete, die zwar schon intensiven Tests unterzogen, aber noch nicht in das Stable-Release übernommen wurden. Hier vereinen sich Aktualität der Programme und Stabilität auf hohem Niveau. Allerdings mit Abstrichen auf beiden Seiten: Sie werden vereinzelt noch über Software-Probleme stolpern und haben andererseits noch immer nicht die absolut neueste Software. Für einen Server stellt die »stable«-Version oft die bessere Variante dar.

unstable: Hier finden Sie endlich die absolut neuesten Versionen aller Programme und Pakete – allerdings befinden sich diese in der Regel noch im Entwicklungsstadium. Sie sollten Pakete aus diesem Bereich nicht in einer kritischen Produktionsumgebung verwenden, weil Ihnen hier niemand für Stabilität und Zuverlässigkeit garantieren wird – aus gutem Grund! Mit anderen Worten: Sollten Sie einmal die schönste, neueste und tollste Version