

FreeBSD für Umsteiger

Stand: 31. Juli 2004

Inhaltsverzeichnis

1. Über dieses Dokument	4
1.1 An wen richtet sich diese Einleitung?	4
1.2 Was kann ich von dieser Einleitung erwarten?	4
2. Weitergehende Informationen	5
2.1 Unix-Einsteiger	5
2.2 Manpages	5
2.2.1 Sektionen	5
2.2.2 Anzeigen von Manpages	6
2.2.3 Online-Manpages	7
2.3 Systemdokumentation	7
2.4 BSD-Foren	7
2.5 Mailinglisten	7
2.6 Internet-Suchmaschinen	7
3. Installation des Systems	8
3.1 Die Entwicklungszweige von FreeBSD	8
3.2 FreeBSD beziehen	9
3.3 Die Installation	9
4. Die ersten Schritte mit dem neuen System	13
4.1 Am System anmelden	13
4.2 Vom System abmelden	13
4.3 Das System anhalten	13
4.4 Anlegen von Benutzern	14
5. Grundlegende Administrationswerkzeuge	15
5.1 echo tut's meist auch	16
5.2 Der Easy Editor	16
6. Einrichtung des X-Servers	17
6.1 Grundlegende Konfiguration	17
6.2 Konfiguration anpassen	18

6.3 Testen des X-Servers	19
6.4 Der X-Server von X.org	20
7. KDE	20
7.1 KDE installieren	20
7.2 KDM starten	21
7.3 Und wo ist mein Shell-Prompt?	22
7.4 Wie werde ich jetzt root?	23
8. Netzwerk	23
8.1 Netzwerk konfigurieren	24
8.1.1 Hostname und IP-Adresse festlegen	24
8.1.2 Standard-Route festlegen	25
8.1.3 DNS-Server bekanntmachen	25
8.2 Zugriff auf NFS-Shares	26
8.3 Zugriff auf CIFS-Freigaben	27
8.4 Drucken auf einem anderen LPD-Host	27
9. Lokale Drucker einrichten	28
9.1 Druckerdatenbank vorbereiten	29
9.2 Nicht PostScript-fähige Drucker	29
9.2.1 GhostScript installieren	30
9.2.2 Filterskript erstellen	30
9.3 Den LPD starten	31
9.4 CUPS, die Alternative	31
10. Soundkarten einrichten	31
10.1 Creative SoundBlaster 128 PCI	32
10.2 Creative SoundBlaster 5.1 Live!	32
10.3 Intel onboard-Soundkarten	32
10.4 VIA onboard-Soundkarten	32
10.5 Creative SoundBlaster Audigy	33
10.6 Gerätedateien erstellen	34
11. Internet-Zugang	34
11.1 Zugangsdaten von T-Online	35
11.2 Analoge Modems	35
11.2.1 Analog-Modem an Nebenstelle	37
11.3 ISDN	37
11.4 DSL	37
11.5 Starten des PPP-Daemons	39

11.5.1	Automatisches starten	39
11.5.2	Manuelles starten	39
12.	Software verwalten	40
12.1	Packages von der Release-CD	40
12.1.1	Durchsuchen mit /stand/sysinstall	40
12.1.2	Schnelles Installieren mit pkg_add	40
12.2	Packages aus dem Internet beziehen	41
12.2.1	Zum Release zugehörige Packages	41
12.2.2	Aktuelle Packages	42
12.3	Das Portssystem	43
12.3.1	Zurechtfinden im Portssystem	43
12.3.2	Einen Port installieren	44
12.3.3	Portssystem aktualisieren	46
12.4	Welche Software ist installiert?	47
12.5	Software wieder loswerden	47
13.	Empfehlenswerte Software	48
13.1	Acrobat Reader	49
13.2	Bochs	49
13.3	Bourne Again Shell (Bash)	49
13.4	Deutsche KDE-Lokalisierung	49
13.5	GNU Ghostscript	49
13.6	GNU Ghostview	50
13.7	GIMP	50
13.8	Java	50
13.9	K Desktop Environment	50
13.10	Kopete	51
13.11	MPlayer	51
13.12	OpenOffice	51
13.13	RealPlayer	52
13.14	Vi Improved	52
13.15	XMMS	52
14.	Die FreeBSD-Verzeichnisstruktur	52
14.1	Root-Verzeichnis	53
14.2	Benutzerprogramme	53
14.3	Zum Booten benötigte Dateien	54
14.4	Gerätedateien	54
14.5	Konfigurationsdateien	54

14.5.1 Konfigurationsdateien des Grundsystems	54
14.6 Heimatverzeichnisse der Benutzer	56
14.7 Kernel-Module	57
14.8 Das Heimatverzeichnis von root	57
14.9 Administrationswerkzeuge	57
14.10 Temporäre Dateien	57
14.11 rc-Startscripts	57
14.12 Veränderbare Dateien	58

1. Über dieses Dokument

1.1 An wen richtet sich diese Einleitung?

Dieses Dokument richtet sich an Umsteiger, die von einem anderen Betriebssystem zu FreeBSD wechseln möchten. Einsteiger, die im Umgang mit einem PC keinerlei Erfahrung besitzen, werden es schwer haben. Im Vorteil sind dagegen Anwender, die bereits mit anderen Unix-Systemen oder mit Linux Erfahrungen gesammelt haben. Sie werden sich schnell mit FreeBSD zurechtfinden. Windows- und MacOS-Anwender werden sich erst umgewöhnen müssen.

1.2 Was kann ich von dieser Einleitung erwarten?

Ich werde die Schritte beschreiben, die ein Anwender durchführen muß, um FreeBSD als Desktop-System einsetzen zu können. So findet sich eine Anleitung, wie eine Soundkarte zur Mitarbeit bewegt werden kann, um Multimedia-Anwendungen ausführen zu können. Die Einrichtung eines Webservers werde ich dagegen sicher *nicht* beschreiben.

Die für einen Schritt notwendigen Programme werden nur insoweit erläutert, als dies zum Verständnis dieses Schritts notwendig ist. Eine tiefergehende Erläuterung würde den Rahmen dieser Einführung sprengen. Ich werde aber selbstverständlich beschreiben, wo der interessierte Anwender weiterführende Informationen finden kann.

2. Weitergehende Informationen

Diese Einleitung kann nicht alle Fragen beantworten und wird recht geradlinig vorgehen. Ziel ist es nur, Ihnen einen lauffähigen Desktop-Rechner unter FreeBSD bereitzustellen. Sofern Sie sich für einen Themenbereich näher interessieren, ein Kommando besser kennenlernen möchten oder allgemein Ihre Unix-Kenntnisse vertiefen möchten, benötigen Sie weiterführende Dokumentation.

2.1 Unix-Einsteiger

Diese Einleitung richtet sich ausdrücklich nicht an Unix-Einsteiger, die zuvor noch nie mit Unix oder Linux gearbeitet haben. Ich setze ein paar sehr wenige grundlegende Unix-Kenntnisse voraus.

Wenn Sie dagegen bisher nur mit Microsoft Windows oder Apple MacOS gearbeitet haben, benötigen Sie eventuell zuerst eine grundlegende Unix-Einführung. Suchen Sie mit der Suchmaschine Ihrer Wahl nach entsprechenden Dokumenten — es gibt sie zuhauf. Sie können sich z. B. die Informationen unter <ftp://andrsn.stanford.edu/pub/> ansehen. Wenn Sie ein gedrucktes Buch käuflich erwerben möchten, kann ich »Unix – Ein praktischer Einstieg« von Jerry Peek, Grace Todino & John Strang empfehlen, das im O'Reilly-Verlag herausgegeben wurde.

2.2 Manpages

Wenn Ihr Rechner erst einmal läuft, steht Ihnen schon einmal die primäre Informationsquelle zu FreeBSD zur Verfügung: Die Manpages.

2.2.1 Sektionen

Die Manpages sind in verschiedene thematische Sektionen aufgeteilt:

<i>Sektion</i>	<i>Inhalt</i>
1	Benutzerprogramme
2	Systemaufrufe und Fehlernummern
3	Bibliotheken für die Programmiersprache C
4	Beschreibungen zu Treibern
5	Beschreibt Dateiformate
7	Verschiedene Dokumente (z. B. Tipps zur Einrichtung einer Firewall, über die Organisation des Dateisystems unter FreeBSD, allgemeine Sicherheitstipps usw.)
8	Programme zur Systemadministration
9	Beschreibung von Kernelschnittstellen
a	Informationen zu allgemeinen Programmen

2.2.2 Anzeigen von Manpages

Zur Anzeige dient das Programm *man*. Um z. B. die Manpage zum Befehl *mkdir* anzuzeigen:

```
man mkdir
```

Jede Sektion enthält außerdem eine Manpage namens »*intro*«, die den Themenbereich der Sektion näher erläutert. Mit Eingabe von

```
man intro
```

wird die Manpage *intro* der 1. Sektion angezeigt. Mittels

```
man 7 intro
```

können Sie gezielt die Manpage *intro* der Sektion 7 anzeigen.

Um alle Manpages mit Namen *intro* aus allen Sektionen anzuzeigen, verwenden Sie:

```
man -a intro
```

Wir besprechen später noch, wann und wie diese Befehle eingegeben werden können.

2.2.3 Online-Manpages

Manpages können auch über eine HTML-Oberfläche gelesen werden. Besuchen Sie dazu die Website <http://www.freebsd.org/cgi/man.cgi>.

2.3 Systemdokumentation

Unter `/usr/share/doc/en/articles` befinden sich einige Dokumentationen (in englischer Sprache) zu umfangreicheren Themen, z. B. zur Einrichtung von Vinum, einem Software-RAID.

Im Verzeichnis `/usr/share/doc/en/books` liegen einige umfangreiche Bücher, wie etwa das offizielle »FreeBSD Handbook« (`/usr/share/doc/en/books/handbook`) sowie die »Frequently Asked Questions« (`/usr/share/doc/en/books/faq`), beide im HTML-Format. Beide Dokumente sind quasi Ihre Pflichtlektüre, wenn Sie nach Durcharbeiten dieser Einleitung ein laufendes System haben.

Es gibt außerdem Übersetzungen einiger dieser Werke. Sehen Sie z. B. in `/usr/share/doc/de` für deutsche Dokumentationen.

2.4 BSD-Foren

Auf <http://www.bsdforen.de> finden Sie ein deutsches Forum, das sich unter anderem mit FreeBSD beschäftigt. Wenn Sie ein Problem haben, verwenden Sie zuerst die Suchfunktion des Forums, um eventuelle Lösungen zu finden.

2.5 Mailinglisten

Auch eine deutschsprachige Mailingliste für die BSD-basierten Betriebssysteme gibt es. Sie finden dazu nähere Informationen auf <http://www.freebsd.de/maillinglists.html>. Dort gibt es ebenfalls eine Suchfunktion, mit der Sie das Listen-Archiv durchsuchen können.

2.6 Internet-Suchmaschinen

Es ist selten, daß Sie der erste sind, der auf ein bestimmtes Problem trifft. Geben Sie die aufgetretene Fehlermeldung in eine der vielen Internet-Suchmaschinen ein. Sie werden mit Sicherheit einige Ergebnisse finden, in denen Lösungen für Ihr Problem genannt werden.

Auch allgemeinere Fragen können, bei Wahl der richtigen Suchwörter, häufig bereits durch die Ergebnisse einer Suchmaschine beantwortet werden. Es gibt sehr viel gute Dokumentation zu FreeBSD, auch kreuz und quer im Internet verstreut. Wenn Sie beispielsweise bei der später beschriebenen Konfiguration des DSL-Zugangs auf Probleme stoßen, versuchen Sie einmal die Suchwörter »DSL mit FreeBSD«. Sie finden damit eine sehr ausführliche Anleitung zu diesem Thema.

3. Installation des Systems

Die Installation wird hier nur recht geradlinig beschrieben. Wenn Sie in einem oder mehreren Punkten abweichen, könnten einige der anderen Beschreibungen in diesem Dokument nicht mehr fehlerfrei ausgeführt werden.

3.1 Die Entwicklungszweige von FreeBSD

FreeBSD wird hauptsächlich in zwei Zweigen entwickelt.

Zunächst gibt es -Stable, das ausführlich getestet und für Produktionssysteme gedacht ist. In diesem Zweig werden nur kleinere, gründlich getestete Änderungen aufgenommen. Trotzdem kann es in seltenen Fällen vorkommen, daß auch unter -Stable etwas nicht wie erwartet funktioniert. Einsteiger sollten daher nicht sofort das aktuellste -Stable verwenden.

Daneben gibt es -Current. Dies ist der Entwicklerzweig, in den die neuesten Entwicklungen einfließen. Es ist nur für Leute gedacht, die sich mit dem System auskennen und mit unvorhergesehen auftretenden Fehlern umgehen können.

Zur Zeit ist FreeBSD 4 der -Stable-Zweig, FreeBSD 5 ist -Current.

Von Zeit zu Zeit werden von den verschiedenen Zweigen Releases erstellt. Ein Release ist eine ausführlich getestete Umgebung. Das Release eines -Stable-Zweiges ist die ideale Wahl für Einsteiger, Releases von -Current sind mit Vorsicht zu behandeln. Das beim Schreiben dieses Dokuments aktuelle Release des -Stable-Zweigs ist 4.10-Release, 5.2.1-Release das von -Current.

Wir befinden uns zur Zeit in der Umstellungsphase — mit dem nahenden Release von 5.3 soll FreeBSD 5 so ausführlich getestet sein, daß es zum -Stable-Zweig ernannt werden kann. Dann wird es mit FreeBSD 6 einen neuen -Current-Zweig geben. Einsteiger sollten zur Zeit trotzdem noch mit einem Release von FreeBSD 4 beginnen.

3.2 FreeBSD beziehen

Das FreeBSD-Team stellt ISO-Images bereit. Sie finden diese unter <ftp://ftp.de.freebsd.org/pub/FreeBSD/ISO-IMAGES-i386/>. Ich empfehle den Download von 4.10-Release. Sie finden im Verzeichnis 4.10 des FTP-Servers 4 Dateien. Sie benötigen im Prinzip nur die Datei *4.10-RELEASE-i386-disc1.iso*. Laden Sie diese herunter und brennen Sie diese auf eine CD (es handelt sich um ein Standard-ISO-Image, das von jedem Brennprogramm verarbeitet werden sollte).

Alternativ bietet die Lehmanns Fachbuchhandlung (<http://www.lob.de>) ein CD-Set an. Dieses Set enthält auch noch eine Reihe von Software-Paketen, die Sie sonst aus dem Internet herunterladen müssten. Für Anwender mit einer langsamen Internet-Anbindung könnte dies interessant sein.

Ob heruntergeladen oder von Lehmanns, ich werde diese CD zum Installieren von FreeBSD im weiteren »Install-CD« nennen — und wir werden sie auch nach der Installation noch brauchen.

3.3 Die Installation

Sie benötigen bei der hier vorgeschlagenen Methode etwa 8 GB unpartitionierten Speicherplatz auf Ihrer ersten Festplatte. Wenn die Festplatte zur Zeit vollständig mit anderen Partitionen belegt ist, müssen Sie diese erst verkleinern, um 8 GB freien Platz zu schaffen. Lesen Sie in der Dokumentation Ihres Betriebssystems nach, welche Schritte dazu durchgeführt werden müssen. FreeBSD benötigt außerdem eine *primäre* Partition. Sie dürfen also zur Zeit maximal drei primäre und erweiterte Partitionen auf der Festplatte angelegt haben.

Ideal wäre natürlich, wenn Sie mehrere Rechner haben und Sie einen allein für FreeBSD »opfern« können. Dann könnten Sie auch nicht mit anderen Betriebssystemen auf der gleichen Festplatte kollidieren. Der Rechner braucht nicht sonderlich schnell zu sein, alles ab etwa 400 MHz mit mindestens 128 MB RAM ist für den Anfang ausreichend.

Wenn die vorgenannten Voraussetzungen erfüllt sind, legen Sie die zuvor erstellte Install-CD in das Laufwerk Ihres Rechners ein und booten Sie davon.

Nach kurzer Zeit werden Sie gefragt, ob Sie den Kernel konfigurieren möchten. Dies können Sie mit Drücken der *Enter*-Taste direkt einmal überspringen.

Anschließend befinden wir uns im Hauptmenü. Wählen Sie dort den Menüpunkt »*Standard*« aus.

Sie befinden sich jetzt im *FDISK*. Hier können Sie ein *Slice* anlegen, das später die FreeBSD-Partitionen enthalten wird. Drücken Sie die Taste *C* und geben Sie als Größe *8192m* ein — dies entspricht 8 GB, was ich für die Installation empfehle. Den Partitionstyp belassen Sie *unbedingt* beim voreingestellten Wert. Wenn dies das einzige Slice auf der Festplatte ist, markieren Sie dieses durch drücken der *Pfeil-nach-unten*-Taste und kennzeichnen Sie es als aktiv (durch drücken der Taste *S*). Drücken Sie schließlich *Q*, um *FDISK* zu beenden.

Nun werden Sie gefragt, was Sie mit dem MBR anstellen wollen.

- Wählen Sie »*BootMgr*«, um den Boot-Manager von FreeBSD zu installieren. Sie können dann beim Booten mit den Funktionstasten das Betriebssystem auswählen, das Sie starten wollen.
- Mittels »*Standard*« wird einfach das als aktiv markierte Slice gebootet. Wenn Sie nur ein Slice auf Ihrer Festplatte anlegen, können Sie einfach diese Option wählen. Das Slice mußte dann aber unbedingt in *FDISK* als aktiv markiert worden sein.
- Und Sie können schließlich mit »*None*« keine Veränderungen vornehmen. Das ist der Fall, wenn Sie bereits einen anderen Boot-Manager auf Ihrem System verwenden.

Tip: FreeBSD wird einfach durch Laden des Codes im Bootblock seines Slices gebootet. Es verhält sich in dieser Hinsicht analog zu Betriebssystemen aus dem Hause Microsoft. Sie können FreeBSD also auf dem gleichen Wege wie MS-DOS oder Microsoft Windows von Ihrem Boot-Manager starten lassen.

Jetzt gelangen Sie in den *Disklabel Editor*, der in dem vorher erzeugten Slice die Partitionen erstellt. Hier zunächst ein Überblick der von mir vorgeschlagenen Partitionierung:

<i>Größe</i>	<i>Mountpoint</i>
512M	/
1024M	/home
1024M	/tmp
4096M	/usr
512M	/var

Gehen Sie nun für jede Zeile dieser Tabelle wie folgt vor: Drücken Sie zunächst die Taste *C*. Sie werden nach der Größe des Dateisystems gefragt, geben Sie dazu den Wert in der Spalte »*Größe*« (inklusive dem angehängten »*M*«) ein. Beantworten Sie die

nächste Frage durch auswählen des Punktes »FS«. Geben Sie dann als Mountpoint den zugehörigen Eintrag aus der Spalte »Mountpoint« an. Da wir zur Zeit mit einer amerikanischen Tastaturbelegung arbeiten, befindet sich der benötigte Schrägstrich praktischerweise direkt auf der *Bindestrich*-Taste, links der *rechten Shift*-Taste.

Wenn Sie alle Zeilen dieser Tabelle abgearbeitet haben, bleiben noch 1024 MB übrig. Drücken Sie nun wieder die Taste *C*, drücken Sie bei der Größe einfach *Enter* und wählen Sie im nächsten Menü den Punkt *Swap*. Damit erstellen Sie einen Auslagerungsbereich, der genau so groß wie der noch verbleibende freie Speicherplatz ist. Üblicherweise sollten 1024 MB Auslagerungsbereich mehr als ausreichend sein. Anschließend verlassen Sie auch den Disklabel Editor mit der Taste *Q*.

Nun wählen Sie die zu installierende Software, indem Sie den Cursor auf der Zeile »9 *X-User*« plazieren und *Enter* drücken. Die Frage, ob das Portssystem installiert werden soll, beantworten Sie ebenfalls mit »YES« (wir beschäftigen uns später noch ausführlich mit den Ports). Drücken Sie die *Tabulator*-Taste, um die Schaltfläche »OK« zu markieren, und fahren Sie mit *Enter* fort.

Als Installationsmedium können Sie dann »1 *CD/DVD*« auswählen. Die Installation der Software kann eine kurze Zeit in Anspruch nehmen.

Beantworten Sie die anschließende Frage nach der Konfiguration eines Netzwerkgeräts mit »No«.

Auch die Fragen betreffend Gateway, *inetd*, anonymem FTP-Zugang und dem NFS-Server können Sie mit »No« überspringen. Lediglich beim NFS-Client könnten Sie »Yes« wählen, wenn Sie später Zugriff auf NFS-Shares anderer Rechner in Ihrem Netzwerk herstellen wollen. Ansonsten ist auch hier »No« die richtige Wahl.

Die Frage nach dem Sicherheitsprofil ist auf einem Desktop-Rechner ebenfalls mit »No« zu erledigen.

Wählen Sie bei der Frage, ob Sie Ihre Konsole einstellen möchten, »Yes«. Im daraufhin erscheinenden Menü wählen Sie den Punkt »3 *Keymap*« und wählen anschließend die Tastaturliste »German ISO« aus. Fahren Sie mit »X *Exit*« fort.

Nun möchten Sie die Zeitzone konfigurieren, wählen Sie bei der entsprechenden Frage das voreingestellte »Yes«. Da Ihre BIOS-Uhr nach lokaler Zeit laufen sollte, wählen Sie anschließend »No«. Im nächsten Menü wählen Sie »8 *Europe*« und anschließend das Land, in dem der Rechner steht.

Die Frage nach der Linux-Binär-Kompatibilität sollten Sie mit »Yes« beantworten, da Sie vermutlich früher oder später auch Linux-Anwendungen unter FreeBSD ausführen wollen oder müssen. Sehen Sie im Kapitel 22 des FreeBSD Handbook nach, um nähere Informationen zu diesem Thema zu erhalten.

Nun die Frage nach der angeschlossenen Maus. Bei einer USB-Maus können Sie mit »No« antworten, da solche Mäuse automatisch erkannt werden. Für serielle oder PS/2-Mäuse wählen Sie »Yes«. Unter »3 Type« sollten Sie meist »1 Auto« wählen, außer, Sie haben eine der unter den anderen Punkten genannten Mäuse. Wählen Sie anschließend unter Punkt »4 Port« die entsprechende Schnittstelle, an der die Maus angeschlossen ist. Wählen Sie schließlich den Punkt »5 Flags«. Wenn Sie hier -3 eintragen, können Sie eine nicht vorhandene dritte Maustaste durch gleichzeitiges Drücken der anderen beiden Tasten emulieren. Unter Unix brauchen Sie recht häufig die mittlere Taste. Wenn Sie eine Maus mit einem Scroll-Rad besitzen, brauchen Sie dies nicht, weil die dritte Taste durch das Drücken (nicht Drehen!) des Rades erzeugt wird. Bei einer solchen Scroll-Maus tragen Sie vielmehr -z 4 ein, um das Rad später zum Bildlauf benutzen zu können.

Im nächsten Schritt wählen Sie auf jeden Fall »No«. Sie sollten keinesfalls Ihren X-Server während der Installation konfigurieren. Wenn Sie dabei etwas falsch machen würden und Ihr Rechner nicht mehr auf Eingaben reagiert, wäre die gesamte Installation hinüber. Wir konfigurieren den X-Server später noch viel komfortabler.

Die Packages möchten wir jetzt nicht durchsuchen, ebensowenig wollen wir einen Benutzer anlegen. Beide Fragen können mit »No« erledigt werden.

Wichtig ist aber auf jeden Fall das *root*-Paßwort. Geben Sie es zweimal hintereinander exakt gleich ein und vergessen Sie es *keinesfalls*. Sie brauchen dieses Paßwort später, um den Rechner zu administrieren. Ohne dieses Paßwort geht nicht mehr viel.

Beantworten Sie die nächste Frage noch mit »Yes«, um wieder zum Konfigurationsmenü zurückzukehren. Wählen Sie dort den Punkt »Distributions«, anschließend die Kategorie »src« und dort den Punkt »sys«. Damit installieren Sie die Quellen des Basissystems. Sie können diese später einmal brauchen, um einen eigenen Kernel zu konfigurieren. Aber auch für die Installation bestimmter Treiber, z. B. für eine Audigy-Soundkarte, werden diese Quellen benötigt. Wählen Sie anschließend zweimal nacheinander die Schaltflächen »OK«, um die Quellen zu installieren.

Das Konfigurationsmenü verlassen Sie über »X Exit«. Wählen Sie dann mit der *Tabulator*-Taste die Schaltfläche »Exit Install« und lassen Sie den Rechner neu booten. Achten Sie darauf, daß der Rechner anschließend von der Festplatte startet, nicht von der CD.

Das war es dann im Prinzip auch »schon« mit Ihrer vielleicht ersten FreeBSD-Installation. Nun steht die Konfiguration an.

4. Die ersten Schritte mit dem neuen System

Nun sollte sich FreeBSD auf Ihrem Rechner befinden und problemlos booten. Wenn der Bootvorgang abgeschlossen ist, sollten Sie eine Aufforderung wie die ähnliche sehen:

```
FreeBSD/i386 (rechner.domain.tld) (ttyv0)
login:
```

Bleibt natürlich die Frage, was wir damit jetzt anfangen können.

4.1 Am System anmelden

Durch Anzeige von *login:* teilt Ihnen das System mit, daß Sie sich nun mit einem Benutzernamen anmelden können. Geben Sie *root* ein und drücken Sie die *Enter*-Taste. Sie werden dann nach dem *root*-Paßwort gefragt, das Sie bei der Installation festgelegt hatten. Geben Sie das Paßwort ein (es wird während der Eingabe aber nicht auf dem Bildschirm dargestellt) und drücken Sie erneut die *Enter*-Taste.

Nun sollten Sie den Shell-Prompt angezeigt bekommen. Hier können Sie nun endlich Befehle eingeben.

4.2 Vom System abmelden

Wir führen nun unseren ersten Befehl aus. Geben Sie am Shell-Prompt

```
exit
```

ein und drücken Sie die *Enter*-Taste. Damit werden Sie vom System abgemeldet, es wird wieder der *login*-Prompt angezeigt. Melden Sie sich dann wieder als *root* an.

4.3 Das System anhalten

Schalten Sie ein laufendes System nie einfach ab. Sie müssen einen ordnungsgemäßen *Shutdown* durchführen. Um den Rechner anzuhalten, können Sie den Befehl

```
shutdown -h now
```

eingeben. Sie können den Computer ausschalten, wenn Sie die Meldung

```
The Operating System has halted.  
Please press any key to reboot...
```

sehen. Natürlich können Sie an dieser Stelle durch Drücken einer beliebigen Taste einen Neustart des Systems durchführen. Wenn Sie ohnehin vorhaben, den Computer neu zu booten, können Sie auch gleich den Befehl

```
shutdown -r now
```

verwenden. Wenn Sie weitere Hilfe zu diesem Befehl erhalten möchten, können Sie jetzt die entsprechende Manpage ansehen:

```
man shutdown
```

Wenn Sie mit den Cursor- oder den Bild-Tasten bis zum Ende der Manpage blättern, wird automatisch wieder das Shell-Prompt angezeigt. Sie können die Anzeige aber auch jederzeit durch Drücken der Taste *Q* beenden.

4.4 Anlegen von Benutzern

Sie arbeiten zur Zeit als Benutzer *root*. Dieser Benutzer hat die vollen Zugriffsrechte auf jeden Teil des Systems. Sie sollten nur unter diesem Benutzernamen arbeiten, wenn dies unbedingt nötig ist. Für Ihre tägliche Arbeit, wie z. B. das Schreiben von Briefen oder dem Surfen im Internet, brauchen Sie keinesfalls die Mächtigkeit des *root*. Aus Sicherheitsgründen sollten Sie sich einen Benutzernamen anlegen, der nur minimale Rechte im System besitzt.

Das Anlegen von Benutzern ist allerdings auch dem Benutzer *root* vorbehalten. Führen Sie daher als *root* folgenden Befehl aus, um den Benutzer *noob* zu erstellen:

```
pw useradd noob -u 500 -g wheel -s /bin/csh -m
```

Ändern Sie den rot markierten Benutzernamen *noob* entsprechend Ihren eigenen Vorstellungen ab. Sie sollten jedoch keine Leer- oder andere Sonderzeichen darin verwenden.

Die rot markierte *500* ist die Benutzer-ID, die diesem Benutzer zugeordnet werden soll. Jeder Benutzer sollte eine auf dem System eindeutige ID erhalten. Beginnen Sie mit *500*. Wenn Sie weitere Benutzer hinzufügen möchten, erhöhen Sie die ID jeweils auf den nächsten, noch nicht verwendeten Wert (*501*, *502* usw.).

Die anderen Angaben sind vorerst nicht interessant, Sie können diese natürlich jederzeit in der Manpage zum Befehl *pw* nachlesen.

Anschließend müssen Sie noch das Paßwort für den Benutzer *noob* festlegen:

```
passwd noob
```

Auch hier ersetzen Sie den rot markierten Text mit dem Benutzernamen, den Sie weiter oben tatsächlich verwendeten. Sie werden dann gebeten, das neue Paßwort für den Benutzer zweimal einzugeben. Das Paßwort wird nur geändert, wenn Sie es zweimal exakt gleich eingeben.

Anschließend können Sie sich mit *exit* abmelden. Versuchen Sie nun, sich am *login*-Prompt mit dem neuen Benutzernamen und dem zugehörigen Paßwort anzumelden. Die weiteren Ausführungen in diesem Dokument gehen davon aus, daß Sie sich korrekt unter Ihrem Benutzernamen anmelden können — Sie sollten dies also jetzt schon sicherstellen.

5. Grundlegende Administrationswerkzeuge

Eines der wichtigsten Werkzeuge eines Administrators ist der Text-Editor. Es gibt viele Text-Editoren für FreeBSD. Für den Anfang müssen Sie jedoch einen beherrschen, der zusammen mit FreeBSD ausgeliefert wird. Zum einen ist das der *vi*, viele Unix- und Linux-Anwender werden ihn zu schätzen wissen. Leider ist der Umgang mit ihm doch etwas gewöhnungsbedürftig. Wenn Sie den *vi* nicht beherrschen, können Sie den Easy Editor verwenden, ein wesentlich leichter handhabbarer Editor, der ebenfalls im Basissystem von FreeBSD enthalten ist.

5.1 *echo* tut's meist auch

Meist werden wir zur Konfiguration allerdings nicht zwingend einen Editor brauchen. Mit dem Befehl *echo* kann man beliebige Zeichen an eine Datei anhängen. Das ist meist ausreichend und so werde ich möglichst oft vom Befehl *echo* Gebrauch machen, da wir damit auch schneller als mit dem Editor fertig sind. Sie finden z. B. oft die Anweisung, folgendes einzugeben:

```
echo 'blafasel "blubb"  
noch mehr blubber' >> /etc/irgendwas.conf
```

Geben Sie also zunächst *echo*, ein Leerzeichen und dann das Apostroph-Zeichen ein. Dies ist das Zeichen, das Sie mit *Shift+#* erzeugen, nicht mit dem Akzent verwechseln! Danach tippen Sie *blafasel »blubb«* und drücken die *Enter*-Taste. Der Befehl wird noch nicht ausgeführt. Geben Sie *noch mehr blubber* ein und schließen Sie den Textteil mit einem weiteren Apostroph-Zeichen ab. Jetzt folgen zwei *>>*, damit wird der Text an eine Datei angehängt. In einigen Befehlen wird auch nur einzelnes *>* aufgeführt — damit wird dann der bestehende Inhalt einer Datei restlos überschrieben. Das ist also ein großer Unterschied, achten Sie darauf. Zum Abschluß geben Sie die Datei an, in die geschrieben werden soll, im Beispiel ist das die Datei */etc/irgendwas.conf*. Drücken Sie jetzt die *Enter*-Taste, um den Befehl tatsächlich auszuführen.

5.2 Der Easy Editor

Der Easy Editor macht seinem Namen alle Ehre — er ist wirklich einfach zu bedienen. Starten Sie den Editor mit Angabe eines Dateinamens:

```
ee /etc/rc.conf
```

Wenn die Datei bereits existiert, wird sie zur Bearbeitung in das Editor-Fenster geladen. Wenn die Datei noch nicht existiert, wird sie beim Abspeichern des Textes angelegt.

Die Bearbeitung einer Textdatei mit dem Easy Editor sollte keine Schwierigkeiten bereiten, sie sollte von anderen, einfachen Editoren bekannt sein. Wenn Sie den Editor beenden und die Änderungen an der Datei abspeichern wollen, drücken Sie die *Escape*-Taste. Es erscheint das »Hauptmenü«:

```

+-----+
| Hauptmenü |
| |
| a) Editor beenden |
| b) Hilfe |
| c) Dateioperationen |
| d) Bildschirm regenerieren |
| e) Einstellungen |
| f) Suche |
| g) Verschiedenes |
| |
| Escape zum Beenden |
+-----+

```

Der Cursor steht standardmäßig auf Punkt *a*). Sie können also einfach die *Enter*-Taste drücken, um diesen Punkt auszuwählen. Alternativ können Sie die Taste mit dem entsprechenden Buchstaben betätigen, hier also die *A*-Taste.

```

+-----+
| Ende-Menü |
| |
| a) Speichern |
| b) Verwerfen |
| |
| Escape zum Beenden |
+-----+

```

Drücken Sie im »Ende-Menü« wieder *Enter* oder *A*, um die Datei abzuspeichern und den Editor zu beenden.

Sie können ein Menü jederzeit mit der *Escape*-Taste beenden, wenn Sie wieder zurück zur Textbearbeitung kehren möchten.

6. Einrichtung des X-Servers

Zur Zeit lacht uns nur die textbasierte Systemkonsole entgegen, in der wir auch schon einige textbasierte Anwendungen ausgeführt haben. Wir möchten nun aber auch ein paar grafische Programme ausführen, um damit unsere tägliche Arbeit zu erledigen. Vielleicht sind Sie auch an Multimedia-Anwendungen interessiert. Kein Problem für FreeBSD.

Wir benötigen aber zunächst ein Programm, welches in der Lage ist, Grafikelemente auf dem Bildschirm darzustellen. Unter FreeBSD wird dazu der X-Server namens *XFree86* verwendet. Er wurde bei der Installation automatisch auf die Festplatte kopiert, wenn Sie die Installationsart »*X-User*« gewählt hatten.

6.1 Grundlegende Konfiguration

Wie fast jede Anwendung benötigt auch *XFree86* eine Konfigurationsdatei. Melden Sie sich als *root* an und führen Sie den Befehl

```
X -configure
```

aus, schon haben Sie eine grundlegende Konfiguration erzeugt. Sie wurde unter */root/XF86Config.new* abgelegt.

6.2 Konfiguration anpassen

Das war nicht sonderlich schwer. Wir werden jedoch noch zwei Anpassungen an der Konfiguration vornehmen müssen. Öffnen Sie die *XF86Config.new* dazu mit Ihrem Editor.

Zunächst sind Sie sicher an einem deutschen Tastatur-Layout interessiert. Suchen Sie daher den folgenden Abschnitt:

```
Section "InputDevice"
    Identifier "Keyboard0"
    Driver     "keyboard"
EndSection
```

Mit diesen Einstellungen wird der Standard, also ein englisches Tastatur-Layout, verwendet. Wir fügen noch ein paar Zeilen ein, so daß der Abschnitt anschließend wie folgt aussieht:

```
Section "InputDevice"
    Identifier "Keyboard0"
    Driver     "keyboard"
    Option     "XkbRules"      "xfree86"
    Option     "XkbModel"     "pc105"
    Option     "XkbLayout"    "de"
    Option     "XkbVariant"   "nodeadkeys"
EndSection
```

Dies geht davon aus, daß Sie eine Tastatur mit 105 Tasten (oft als »Win-Tastatur« beworben) und deutschsprachigem Layout im Einsatz haben.

Zum Ende der Datei finden Sie die *Section* »*Screen*«. Löschen Sie diese vollständig (also von der Zeile *Section* »*Screen*« bis zur Zeile *EndSection*). Fügen Sie dann folgende Zeilen ein:

```
Section "Screen"
    Identifier      "Screen0"
    Device          "Card0"
    Monitor         "Monitor0"
    DefaultDepth    16

    SubSection "Display"
        Depth       16
        Modes        "1024x768"
    EndSubSection
EndSection
```

In der Zeile *DefaultDepth* geben Sie die Farbtiefe an, mit welcher Sie den X-Server betreiben möchten. Im Beispiel sind dies 16 Bit (also 65.536 Farben). Den gleichen Wert müssen Sie ein paar Zeilen weiter unten in der Zeile *Depth* angeben.

In der Zeile *Modes* schließlich geben Sie die Auflösung an, mit welcher Sie den Bildschirm betreiben möchten. Sie können hier nur Standard-VESA-Auflösungen angeben, also z. B. *640x480*, *800x600*, *1024x768* (wie im Beispiel) oder *1600x1200*. Wenn Sie eine andere Auflösung wählen möchten, die nicht VESA-kompatibel ist, müssen Sie eine eigene Modeline erstellen. Damit können Sie sich später immer noch beschäftigen, für den Anfang sollte ein VESA-Modus genügen.

Speichern Sie die Konfigurationsdatei ab und kopieren Sie diese an den Ort, an dem sie von XFree86 erwartet wird:

```
cp /root/XF86Config.new /etc/X11/XF86Config
```

6.3 Testen des X-Servers

Nun können wir den X-Server testen. Rufen Sie dazu als *root* einfach den Befehl

```
X
```

auf. Wenn alles geklappt hat, sollten Sie nun auf dem Monitor ein gerastertes Hintergrundbild sehen. In der Mitte sehen Sie ein »X«, das ist der Mauszeiger. Bewegen Sie die Maus ein wenig hin und her. Achten Sie darauf, daß der Mauszeiger Ihren Bewegungen korrekt folgt. Wenn dies der Fall ist, haben Sie XFree86

erfolgreich konfiguriert. Sie können den X-Server jetzt mit *Strg+Alt+Backspace* beenden. Die Taste *Backspace* ist auch als »Rücklösch-Taste« bekannt, sie befindet sich direkt über der Enter-Taste.

6.4 Der X-Server von X.org

Mit der Version 4.4 hat XFree86 die Lizenzbedingungen geändert. Aufgrund dessen sind mehrere Linux-Distributionen dazu übergegangen, einen alternativen X-Server einzusetzen: X.org. Zur Zeit ist nicht absehbar, ob FreeBSD künftig weiterhin standardmäßig mit XFree86 ausgeliefert wird oder ob X.org den Vorzug erhält. Es ist bereits seit längerem in den Ports enthalten, kann also unter FreeBSD genutzt werden. Ich zeige später, wie Sie eine Anwendung aus dem Portssystem installieren können, möchte an dieser Stelle aber bereits auf die hervorragende Anleitung unter <http://www.bsdforen.de/showthread.php?t=5217> hinweisen. Für uns reicht XFree86 aber zur Zeit erstmal aus.

7. KDE

Wir haben jetzt also einen X-Server, der ein sich bewegendes »X« darstellen kann. Ein bißchen dürftig, um damit vernünftig arbeiten zu können. Wir brauchen also noch ein paar Programme für unseren X-Server.

Empfehlenswert für den Umsteiger ist KDE, das *K Desktop Environment*. Es ist auf den FreeBSD-Release-CDs enthalten, die Installation also nicht sonderlich kompliziert. Wir gehen die Einrichtung Schritt für Schritt durch.

7.1 KDE installieren

Auf der Installations-CD sind einige Software-Pakete enthalten. Natürlich nicht alle, die für FreeBSD verfügbar sind, diese würden bei weitem nicht auf eine einzige CD passen. Enthalten ist aber z. B. KDE, welches wir nun installieren möchten.

Später werden wir sehen, wie man Pakete schnell mittels *pkg_add* installiert. Damit können wir uns aber nur schlecht einen Überblick über die Pakete verschaffen, die auf der Installations-CD enthalten sind. Komfortabler ist es in diesem Fall, einen alten Bekannten rauszuholen: *sysinstall*. Sicher, um KDE zu installieren brauchen wir *sysinstall* nicht, wir wissen ja in diesem Falle schon, welches Paket wir haben wollen. Ich zeige an dieser Stelle aber absichtlich diesen Weg, damit Sie später wissen, wie Sie sich einen Überblick über die auf der Installations-CD gebotenen Pakete verschaffen können.

Legen Sie also nun Ihre Installations-CD ein und rufen Sie als *root* den Befehl

```
/stand/sysinstall
```

auf. Wählen Sie den Menüpunkt »*Configure*«. Aus dem dann erscheinenden Bildschirm wählen Sie den Punkt »*Packages*«. Sie werden dann nach dem Medium gefragt, auf dem sich die Pakete befinden. Bestätigen Sie einfach die Auswahl »*CD*«.

Anschließend sehen Sie eine Übersicht der vorhandenen Kategorien. Jede Kategorie enthält wiederum eine unterschiedliche Anzahl von Software-Paketen, die thematisch zu dieser Kategorie passen. Bewegen Sie den Cursor soweit nach unten, bis die Kategorie »*kde*« markiert ist und drücken Sie *Enter*. Sie sehen nun den Inhalt dieser Kategorie. Bewegen Sie den Cursor auf das Paket »*kde-3.2*« (die Versionsnummer kann differieren) und drücken Sie die *Leertaste*. Automatisch werden auch alle anderen Pakete in der Kategorie »*kde*« ausgewählt, da diese alle zum Betrieb von KDE benötigt werden, *sysinstall* achtet also auf eventuelle Abhängigkeiten zu anderen Paketen.

Drücken Sie dann *Enter*, um wieder zur Kategorie-Übersicht zurückzukehren. Wenn Sie möchten, können Sie nun in den anderen Kategorien stöbern. Wenn Sie vorher bereits mit Unix oder Linux gearbeitet haben, werden Sie hier mit Sicherheit das eine oder andere Ihnen bekannte Programm wiederfinden. Sie können es dann ebenfalls wie oben beschrieben zur Installation auswählen. Erwarten Sie aber nicht zu viel, es haben nicht alle erdenklichen Pakete auf der CD Platz gefunden.

Wenn Sie genug gestöbert haben, kehren Sie wieder in die Kategorie-Übersicht zurück. Wählen Sie durch Drücken der *Tabulator*-Taste die Schaltfläche »*Install*« aus. In einer kurzen Übersicht werden Ihnen anschließend noch einmal die Pakete angezeigt, die Sie zur Installation ausgewählt haben. Bestätigen Sie dies wieder mit der *Enter*-Taste, um die Pakete tatsächlich zu installieren. Dieser Vorgang kann ein wenig Zeit in Anspruch nehmen, bitte haben Sie Geduld.

Anschließend können Sie *sysinstall* wieder verlassen.

7.2 KDM starten

KDE enthält den KDM. Das ist ein grafischer Login-Manager. Er fragt Sie nach einem Benutzernamen und einem Paßwort und startet anschließend für diesen Benutzer die KDE-Oberfläche. Sie können den KDM beim Booten automatisch starten lassen. Sie machen den Rechner an, warten, bis der KDM automatisch

erscheint, geben Ihren Benutzernamen und das Paßwort ein und nach kurzer Zeit ist Ihr KDE-Desktop bereit. Toll, nicht? Leider funktioniert das nicht von alleine. Sie müssen FreeBSD sagen, daß es den KDM beim Booten starten soll.

Öffnen Sie als *root* die Datei */etc/ttys* in Ihren Editor. Suchen Sie die Zeile

```
ttyv8 "/usr/local/bin/xdm -nodaemon" xterm off secure
```

Ändern Sie die Zeile derart, daß sie wie folgt aussieht (die vorzunehmenden Änderungen habe ich deutlich **hervorgehoben**):

```
ttyv8 "/usr/local/bin/kdm -nodaemon" xterm on secure
```

Führen Sie nun den Befehl

```
kill -HUP 1
```

aus. Nach kurzer Zeit sollte sich der KDM auf Ihrem Bildschirm melden. Geben Sie Ihren Benutzernamen (bitte *nie* als *root* im KDM anmelden) und das Kennwort ein und drücken Sie die *Enter*-Taste.

Dann sollte Sie der KDE-Einrichtungs-Assistent begrüßen, der Ihnen beim ersten Einrichten von KDE hilft. Spielen Sie ein bißchen mit KDE rum. Wenn Sie genug haben, lesen Sie weiter.

7.3 Und wo ist mein Shell-Prompt?

Sicher fragen Sie sich jetzt, wo Sie die ganzen schönen Befehle, die ich Ihnen bisher gezeigt habe (und es folgen später noch viele weitere), eingeben können. Die Lösung: Öffnen Sie eine Terminal-Emulation. Klicken Sie dazu in der Leiste, die sich am unteren Bildschirmrand befindet, auf das Symbol  und es öffnet sich ein Terminal-Fenster. Sie können dann wie gewohnt Befehle ausführen. Schließen Sie das Fenster durch den Befehl

```
exit
```

oder indem Sie auf das *Schließen*-Symbol des Fensters klicken.

7.4 Wie werde ich jetzt *root*?

Da Sie nun als »normaler« Anwender ohne besondere Rechte angemeldet sind, werden Ihnen einige Aktionen, die *root* vorbehalten sind, verwehrt. Zur Administration des Rechners müssen Sie sich daher zuerst weitergehende Rechte verschaffen. Öffnen Sie dazu eine Terminal-Emulation, wie oben beschrieben, und verwenden Sie den Befehl

```
su -
```

Nach Eingabe des *root*-Paßworts sind Sie als eben dieser angemeldet und haben wieder volle Zugriffsrechte auf Ihren Rechner. Nach Abschluß der Administrations-Arbeiten beenden Sie diese *root*-Anmeldung wie gewohnt mit dem Befehl

```
exit
```

Danach erhalten Sie wieder ein Terminal zurück, in welchem Sie mit Ihren eingeschränkten Zugriffsrechten als Benutzer arbeiten können.

Wichtig: Unter FreeBSD können nur die Anwender *su* verwenden, die der Gruppe *wheel* zugeordnet sind. Dies ist der Fall, wenn Sie Ihren Benutzer so anlegen, wie dies in dieser Einleitung beschrieben wurde. Wenn Sie später weitere Benutzer hinzufügen, sollten Sie auf die korrekte Gruppenzugehörigkeit achten, sofern der neue Benutzer auch *su* benutzen können soll.

8. Netzwerk

Lesen Sie in diesem Abschnitt, wie Sie die Netzwerkkarte unter FreeBSD konfigurieren, um Zugriff auf Ihr Netzwerk zu erhalten. Außerdem erkläre ich kurz, wie Sie auf Ressourcen eines anderen Rechners zugreifen können.

Ich beschreibe nur, wie ein Client eingerichtet wird. Wenn Sie den Rechner als Netzwerkserverserver verwenden wollen, müssen Sie in der entsprechenden Literatur nachlesen. Im FreeBSD Handbook werden verschiedene Netzwerkdienste ausführlich besprochen — nutzen Sie das Handbook!

8.1 Netzwerk konfigurieren

Zunächst müssen Sie herausfinden, von welchem Kernel-Modul Ihre Netzwerkkarte unterstützt wird. Sehen Sie dazu die Boot-Meldungen durch, die Sie mit dem Programm *dmesg* anzeigen lassen können. Versuchen Sie zunächst folgenden Befehl:

```
dmesg | grep Ethernet
```

Das zeigt dann nur die Zeilen der Boot-Meldungen, die die Zeichenkette »*Ethernet*« enthalten. Die Ausgabe könnte wie folgt aussehen:

```
fxp0: <Intel Pro/100 Ethernet> port 0x2000-0x203f  
      mem 0xf4904000-0xf4904fff irq 11 at device 8.0 on pci  
fxp0: Ethernet address 00:21:0f:26:c1:02
```

Diese Netzwerkkarte wird also vom Treiber *fxp* unterstützt. Der Treiber numeriert alle Karten, für die er zuständig ist, durch. In diesem Falle ist nur eine *fxp*-Karte verbaut, sie hat die Nummer »0« erhalten. Merken Sie sich also in diesem Fall die Bezeichnung »*fxp0*«.

Wenn Sie eine recht günstige Karte haben, wird diese meist einen Realtek-Chipsatz besitzen. In diesem Falle wäre der Treiber *rl* zuständig, die erste Realtek-Netzwerkkarte besitzt dann den Namen »*rl0*«. Sehen Sie trotzdem immer die Boot-Meldungen durch, um den korrekten Namen der Netzwerkkarte zu ermitteln.

Werden Sie in den Boot-Meldungen nicht fündig, dann wird Ihre Netzwerkkarte vom Standard-Kernel nicht unterstützt. Konsultieren Sie das FreeBSD Handbook, um weitere Informationen zur dann nötigen Vorgehensweise zu erhalten.

8.1.1 Hostname und IP-Adresse festlegen

Die Konfiguration der Netzwerkkarte wird in der Datei */etc/rc.conf* gespeichert. Sehen Sie sich folgendes Beispiel an:

```
echo 'hostname="rechner.domain.tld"  
ifconfig_fxp0="inet 192.168.0.1"' >> /etc/rc.conf
```

Bevor Sie diesen Befehl ausführen, sollten Sie ihn an Ihre Bedürfnisse anpassen. Hierbei sollen Ihnen die folgenden Erläuterungen helfen:

Mit *hostname* setzen Sie den voll qualifizierten Hostnamen des Rechners.

Mittels *ifconfig fxp0* konfigurieren Sie die Netzwerkkarte *fxp0*. Diesen Namen müssen Sie, wie weiter oben beschrieben, vorher ermitteln. Hinter dem Schlüsselwort *inet* wird im Beispiel die IP-Adresse 192.168.0.1 festgelegt. Sie muß zum Rest Ihres Netzwerkes passen und darf nicht bereits von einem anderen Rechner des Netzwerkes verwendet werden. Die Netzwerkmaske wird automatisch von FreeBSD ermittelt, Sie müßten diese nur angeben, wenn Sie eine »nicht-standardgemäße« Netzwerkmaske verwenden wollen. Mit »nicht-standardgemäß« meine ich alles außer den Masken /8, /16 und /24. Sie brauchen die Angabe also im Prinzip nur, wenn Sie genau wissen, was Sie damit bezwecken wollen — für »normale« Fälle ist die von FreeBSD ermittelte Netzwerkmaske genau die richtige.

8.1.2 Standard-Route festlegen

Wenn Sie in Ihrem Netzwerk einen Router besitzen, über den Sie ins Internet gelangen können, führen Sie folgenden Befehl aus:

```
echo 'defaultrouter="192.168.0.99"' >> /etc/rc.conf
```

192.168.0.99 ist im Beispiel die IP-Adresse des Routers, der standardmäßig für alle Anfragen, deren Ziel außerhalb des lokalen Netzes liegt, verwendet wird.

8.1.3 DNS-Server bekanntmachen

Wenn in Ihrem Netzwerk außerdem ein DNS-Server zur Namensauflösung bereitsteht, können Sie diesen ebenfalls verwenden:

```
echo 'nameserver 192.168.0.99' >> /etc/resolv.conf
```

In diesem Beispiel wird davon ausgegangen, daß der oben als Router angegebene Rechner auch als DNS-Server fungieren kann. Prüfen Sie bitte, ob auch Ihr Router das kann, *bevor* Sie ihn hier zur Namensauflösung angeben.

Nach der Netzwerk-Konfiguration:

Wenn Sie alle Netzwerkeinstellungen vorgenommen haben, können Sie den Rechner jetzt neu starten. Dies ist die einfachste Möglichkeit, um die getroffenen Einstellungen zu übernehmen.

Anschließend sollten Sie mit den anderen Rechnern Ihres Netzwerkes kommunizieren können. Wenn Sie einen Router und einen DNS-Server angegeben haben, sollten Sie darüber jetzt sogar schon im Internet surfen können. Unter KDE dient der *Konqueror* u. a. als Webbrowser.

8.2 Zugriff auf NFS-Shares

Dies unterscheidet sich nicht grundlegend von anderen Unix- oder Linux-Systemen. Da ich derlei Vorkenntnisse voraussetze, will ich darauf auch nur ganz kurz eingehen.

Legen Sie zunächst den Mountpoint an, ganz nach Ihrer Wahl, z. B.:

```
mkdir -p /nfs/Musik
```

Nun mounten Sie das Share des NFS-Servers in das eben angelegte Verzeichnis:

```
mount -t nfs datenserver:/usr/local/MP3 /nfs/Musik
```

Über das Verzeichnis */nfs/Musik* können Sie jetzt auf das Verzeichnis */usr/local/MP3* des NFS-Servers *datenserver* zugreifen.

Wichtig: Unter FreeBSD kann grundsätzlich nur der Benutzer *root* den *mount*-Befehl verwenden. Das ist aus Sicherheitsgründen wünschenswert. Lesen Sie im FreeBSD Handbook, welche Vorkehrungen getroffen werden müssen, damit auch andere Benutzer Dateisysteme mounten können.

Sie können ein NFS-Dateisystem wie jedes andere wieder aushängen:

```
umount /nfs/Musik
```

Ansonsten erfolgt auch beim Herunterfahren des Rechners automatisch ein *umount*.

8.3 Zugriff auf CIFS-Freigaben

Wenn Sie noch nie etwas von CIFS gehört haben: Das ist der neue Name für SMB. Technisch hat sich nichts getan, lediglich der Name wurde von Microsoft geändert. So heißt auch die Implementierung, die FreeBSD schon im Basissystem mitbringt, weiterhin *smbfs*.

Mehr als tausend Worte sagt das folgende Beispiel. Nehmen wir an, Sie möchten die Freigabe namens *MUSIK* auf dem Rechner *WINDOSE* über den Mountpoint */smb/MP3* verfügbar machen (der Mountpoint muß auch hier vorher ggf. mit *mkdir* angelegt werden!). Ihr Benutzername auf *WINDOSE* sei *NOOB*. Führen Sie (als *root*) den folgenden Befehl aus:

```
mount -t smbfs //NOOB@WINDOSE/MUSIK /smb/MP3
```

Sie werden dann aufgefordert, das Paßwort von *NOOB* auf *WINDOSE* einzugeben. Auf die Freigabe kann anschließend über */smb/MP3* zugegriffen werden.

Sie können auch CIFS-Dateisysteme wie gewohnt mit *umount* wieder aushängen. Alternativ erfolgt dies auch hier automatisch beim Herunterfahren des Rechners.

8.4 Drucken auf einem anderen LPD-Host

Wenn Sie einen Drucker benutzen möchten, der an einem anderen Unix- oder Linux-Rechner (der Drucker-Host) angeschlossen ist, *und* auf dem anderen Rechner ein BSD LPD oder kompatibler Spooler verwendet wird, können Sie auf diesem wie folgt Ausdrucke fertigen. Wenn Sie einen Drucker *lokal* angeschlossen haben, lesen Sie bitte später in diesem Dokument über die Einrichtung lokaler Drucker.

Zunächst müssen Sie eine lokale Warteschlange anlegen. Diese wird in der */etc/printcap* definiert. Sie müssen angeben, an welchen Port welchen Rechners die Druckaufträge weiterverschickt werden sollen. Öffnen Sie die */etc/printcap* daher mit einem Editor und fügen Sie die folgenden Zeilen ein:

```
lp|fs800|Kyocera FS-800:\
    :lp=:\
    :rm=druckerhost:\
    :rp=fs800:\
    :sd=/var/spool/lpd/fs800:
```

Die erste Zeile ist der Name der lokalen Warteschlange, über diese Namen kann der Drucker vom lokalen Rechner aus angesprochen werden. Diese können, müssen aber nicht mit den Namen der Warteschlange auf dem Drucker-Host entsprechen. Es ist jedoch meist übersichtlicher, die gleichen Namen zu verwenden. Wenn dies der einzige Drucker ist, auf den von diesem Rechner aus Zugriff besteht, sollte die Warteschlange auch den Namen *lp* erhalten (siehe Beispiel). Das ist dann der Standard-Drucker, soweit vom Benutzer kein anderer explizit gewählt wird.

Hinter *rm* wird dann der Name des Drucker-Hosts angegeben, im Beispiel *druckerhost*.

Mittels *rp* teilen Sie dem LPD mit, wie die Warteschlange auf dem Drucker-Host heißt, in diesem Beispiel ist das *fs800*.

Schließlich wird hinter *sd* das Spool-Verzeichnis angegeben. Sie sollten für Drucker auf einem anderen Rechner grundsätzlich eigene Spool-Verzeichnisse anlegen. Wenn das Verzeichnis noch nicht existiert, holen Sie dies zuerst nach und setzen auch gleich die richtigen Zugriffsrechte darauf:

```
mkdir -p /var/spool/lpd/fs800
chmod 770 /var/spool/lpd/fs800
chown daemon:daemon /var/spool/lpd/fs800
```

Führen Sie als *root* den Befehl

```
lpd
```

aus, um den LPD sofort zu starten. Mittels

```
echo 'lpd_enable="YES"' >> /etc/rc.conf
```

wird er künftig auch bei jedem Bootvorgang automatisch gestartet.

9. Lokale Drucker einrichten

Im Basissystem von FreeBSD ist der LPD enthalten, mit dem Sie einen *lokalen* Drucker betreiben können. Wie Sie lediglich einen Drucker an einem anderen Rechner benutzen, wurde bereits vorher besprochen und Sie können diesen Abschnitt überspringen.

9.1 Druckerdatenbank vorbereiten

Der LPD verwendet eine Druckerdatenbank namens */etc/printcap*. In ihr stehen alle verfügbaren Drucker und deren Eigenschaften. Öffnen Sie die Datei in einem Editor und fügen Sie die folgenden Zeilen hinzu:

```
lp|fs800|Kyocera FS-800:\
    :sh:\
    :sd=/var/spool/lpd:\
    :lf=/var/log/lpd:\
    :lp=/dev/lpt0:\
    :if=/etc/lpfilter:
```

In der ersten Zeile werden die Namen der Warteschlange festgelegt. Der Name *lp* ist der Standarddrucker und sollte daher irgendwo definiert werden. *fs800* ist ein weiterer Name für die Warteschlange. Der letzte Name, *Kyocera FS-800*, ist eigentlich mehr eine Beschreibung für den Drucker.

Mittels *lp=/dev/lpt0* geben Sie an, daß der Drucker an der ersten parallelen Schnittstelle angeschlossen ist. Für den ersten USB-Drucker würden Sie hier statt dessen */dev/ulpt0* angeben.

Mittels *if=/etc/lpfilter* geben Sie an, daß ein Druckjob zuerst noch durch das angegebene Filterskript bearbeitet werden muß, bevor es zum Drucker geschickt werden kann. Wenn Sie einen PostScript-fähigen Drucker haben, brauchen Sie diese Zeile nicht.

9.2 Nicht PostScript-fähige Drucker

Anwendungen erzeugen ihre Druckjobs unter Unix üblicherweise als PostScript-Datei. Dies ist eine spezielle Programmiersprache, mit der eine Seite beschrieben werden kann. Hochwertige Drucker können PostScript meist direkt verarbeiten, der Druckjob kann also ohne Änderungen sofort an den Drucker gesendet werden. Günstigere Drucker verstehen jedoch häufig kein PostScript. Der Druckjob muß zuerst in etwas umgewandelt werden, das der Drucker versteht. Dazu verwenden wir ein Filterskript.

9.2.1 GhostScript installieren

Zunächst brauchen wir aber noch eine Anwendung, die in der Lage ist, PostScript-Code in die verschiedenen Druckersprachen zu übersetzen. Im Allgemeinen wird dazu GhostScript verwendet. Es ist auf der Install-CD enthalten. Diese Gelegenheit möchte ich auch gleich nutzen um zu zeigen, wie man ein Paket von der Install-CD mittels *pkg_add* installieren kann. Führen Sie dazu als *root* folgendes aus:

```
mount /cdrom
cd /cdrom/packages/All
pkg_add ghostscript-gnu-nox11*
```

Nach dem Mounten der CD wechseln Sie in das Verzeichnis */cdrom/packages/All*, in welchem sich alle Software-Pakete der CD befinden. Das ist wichtig, da *pkg_add* eventuell als Abhängigkeiten benötigte Pakete automatisch im aktuellen Verzeichnis sucht. Anschließend weisen Sie *pkg_add* an, das Paket, dessen Dateiname mit »*ghostscript-gnu-nox11*« beginnt, auf dem System zu installieren.

9.2.2 Filterskript erstellen

Legen Sie jetzt das Filterskript */etc/lpfilter* (oder der Name, den Sie in der *printcap* angegeben hatten) mit einem Editor an. Es kann wie folgt aussehen:

```
#!/bin/sh
/usr/local/bin/gs -q \
    -sDEVICE=ljet4 \
    -sPaperSize=a4 \
    -dNOPAUSE \
    -dBATCH \
    -sOutputFile=- \
    -
```

Hier wird also *gs* mit den zur Umwandlung benötigten Angaben aufgerufen. Sie müssen vermutlich nur das *DEVICE* ändern. Im Beispiel wird das Gerät *ljet4*, ein HP LaserJet 4 oder kompatibler Drucker, verwendet.

Wie finden Sie nun heraus, mit welchem Gerät Ihr Drucker angesprochen werden kann? Sehen Sie dazu auf der Webseite <http://www.ghostscript.com/doc/printer.htm> nach. Einige Drucker werden von mehreren Geräten unterstützt, eventuell mit unterschiedlichen Fähigkeiten. Lesen Sie daher nach Möglichkeit bis zum Ende der Liste und entscheiden Sie sich ggf. für das am besten geeignete Gerät.

Vergessen Sie abschließend nicht, das Filterskript ausführbar zu machen:

```
chmod a+x /etc/lpfilter
```

9.3 Den LPD starten

Führen Sie als *root* den Befehl

```
lpd
```

aus, um den LPD sofort zu starten. Geben Sie dann als *root*

```
echo 'lpd_enable="YES"' >> /etc/rc.conf
```

ein — damit wird der LPD künftig bei jedem Bootvorgang automatisch gestartet.

9.4 CUPS, die Alternative

Zum Ende des Themas möchte ich auch noch erwähnen, daß sich das »neue« Druckersystem für UNIX, genannt CUPS, ebenfalls auf der Install-CD befindet. Es wurde automatisch mit KDE installiert. Wenn Sie CUPS bevorzugen, können Sie es ebenso verwenden. Die Konfiguration werde ich jedoch aus Platzgründen hier nicht aufführen, sehen Sie dazu bitte in der CUPS-Dokumentation nach.

10. Soundkarten einrichten

Informieren Sie sich zunächst, ob und von welchem Modul Ihre Soundkarte unterstützt wird. Sehen Sie sich hierzu auf der Webseite <http://www.freebsd.org> um. Sie finden dort in der rechten Spalte unter »*Hardware Notes*« Informationen über die von einem bestimmten Release unterstützte Hardware. Suchen Sie in den *Hardware Notes* in der Rubrik »*Soundcards*« nach Ihrem Modell.

Wenn Sie den Namen des Moduls kennen, müssen Sie FreeBSD sagen, daß es genau dieses Modul beim Booten laden soll. Dies können wir über die Datei */boot/loader.conf* erledigen. Im folgenden zeige ich, wie Sie das für einige verbreitete Soundkarten bewerkstelligen können.

10.1 Creative SoundBlaster 128 PCI

Hierunter fallen auch Soundkarten, die zur SoundBlaster 128 kompatibel sind, z. B. die *SoundBlaster 4.1 Digital*. Führen Sie als *root* folgenden Befehl aus:

```
echo 'snd_pcm_load="YES"' >> /boot/loader.conf
```

Nach einem Neustart wird das Modul *pcm* geladen, welches die o.g. Soundkarten direkt unterstützt.

10.2 Creative SoundBlaster 5.1 Live!

Hierzu wird neben *pcm* noch das Modul *emu10k1* benötigt. Geben Sie als *root* folgenden Befehl ein:

```
echo 'snd_pcm_load="YES"  
snd_emu10k1_load="YES"' >> /boot/loader.conf
```

Nach dem »*YES*« in der ersten Zeile drücken Sie tatsächlich *Enter*. Der Befehl wird aber erst ausgeführt, wenn Sie nach der zweiten Zeile *Enter* drücken.

Auch hier sollten Sie einen Neustart durchführen, um das Modul zu laden.

10.3 Intel onboard-Soundkarten

Die auf vielen neueren Intel-Mainboards verwendete onboard-Soundkarte reicht Ihnen aus? Dann können Sie diese mit dem *ich*-Modul verwenden. Führen Sie als *root* folgendes aus (und denken Sie an den anschließenden Neustart):

```
echo 'snd_pcm_load="YES"  
snd_ich_load="YES"' >> /boot/loader.conf
```

10.4 VIA onboard-Soundkarten

So gut wie alle neueren Mainboards mit dem VIA-Chipsatz besitzen ebenfalls eine onboard-Soundkarte. Diese wird üblicherweise vom Modul *via8233* unterstützt. Geben Sie dazu den Befehl

```
echo 'snd_pcm_load="YES"  
snd_via8233_load="YES"' >> /boot/loader.conf
```

als *root* ein und booten Sie Ihren Rechner neu.

10.5 Creative SoundBlaster Audigy

Das war bisher recht langweilig. Hier aber noch eine etwas schwieriger einzurichtende Soundkarte, die *SoundBlaster Audigy* und die *Audigy 2*. Beide Karten verwenden den *emu10k2*-Chip, dieser wird zur Zeit noch nicht von FreeBSD-Stable unterstützt. Sie können sich aber von <http://chibis.persons.gfk.ru/audigy/> das Archiv namens *emu10kx-2004-07-18.tar.bz2* herunterladen. Das angegebene Datum kann variieren. Entpacken Sie die Datei daraufhin als *root*:

```
tar -xjf emu10kx*.bz2
```

Installieren Sie dann den Treiber wie folgt:

```
cd emu10kx-2004-07-18    (auch hier kann das Datum variieren)  
make  
make install
```

Damit haben Sie das Modul *emu10kx* übersetzt. Wir lassen es beim Booten durch Eingeben des folgenden Befehls künftig automatisch laden:

```
echo 'snd_pcm_load="YES"  
midi_load="NO"  
snd_emu10k1_load="NO"  
snd_emu10kx_load="YES"' >> /boot/loader.conf
```

Zusätzlich benötigen wir noch eine Gerätedatei, damit der Treiber mit dem Betriebssystem kommunizieren kann:

```
/sbin/mknod /dev/emu10kx0 c 201 0
```

Starten Sie jetzt ebenfalls den Rechner neu, um das Modul zu laden.

10.6 Gerätedateien erstellen

Wir sind noch nicht ganz fertig. Abschließend benötigen wir noch ein paar Gerätedateien, über welche die Multimedia-Anwendungen auf die Soundkarte zugreifen können. Erstellen Sie diese Dateien als *root* mit den Befehlen:

```
cd /dev
./MAKEDEV snd0
```

Wenn Sie mehrere Soundkarten verwenden (was unter FreeBSD keinerlei Probleme darstellt), müssen Sie für jede weitere Soundkarte entsprechende Gerätedateien erstellen:

```
./MAKEDEV snd1    (für die zweite Soundkarte)
./MAKEDEV snd2    (für die dritte Soundkarte)
./MAKEDEV snd3    (und wie viele Karten Sie Ihr Rechner auch immer haben mag)
```

11. Internet-Zugang

Als letzte »Hardware-Hürde« steht nun noch die Einrichtung des Internet-Zugangs aus. Wenn Sie über einen Router ins Internet gelangen, sollte dieses Thema bereits mit der Einrichtung des Netzwerks (weiter oben in diesem Dokument) erledigt sein. Wenn Sie das Modem (analog bzw. DSL) oder die ISDN-Karte jedoch direkt an Ihrem Rechner angeschlossen haben, müssen Sie erst dieses Kapitel durcharbeiten.

Zum Aufbau einer Internet-Verbindung verwende ich den Userland-PPP. Es gibt auch noch eine Kernel-Implementierung, sie soll uns aber nicht weiter interessieren. Die vorgestellten Konfigurationsdateien unterscheiden sich je nach dem, mit welchem Gerät (analoges Modem, ISDN-Karte oder DSL-Modem) die Internet-Verbindung aufgebaut wird.

Die Konfiguration wird in der Datei */etc/ppp/ppp.conf* gespeichert. Diese enthält standardmäßig Beispiel-Einträge. Sie können die Datei mit dem folgenden Befehl (als *root* ausgeführt) daher zuerst einmal leeren:

```
echo '' > /etc/ppp/ppp.conf
```

Dann öffnen Sie diese leere Datei, indem Sie als *root* Ihren Editor wie folgt aufrufen:

```
ee /etc/ppp/ppp.conf
```

Im Beispiel würde eine leere *ppp.conf* im Easy Editor geöffnet. Fügen Sie dann die entsprechende Konfiguration, wie nachfolgend gezeigt, ein.

11.1 Zugangsdaten von T-Online

In den nachfolgend vorgestellten Konfigurationsdateien müssen Sie u. a. einen Benutzernamen angeben. Bei T-Online ist die Zusammensetzung dieses Benutzernamens etwas komplizierter, daher möchte ich dies vorab erläutern.

Der Aufbau des Benutzernamens gestaltet sich wie folgt:

```
1111111111112222222222220001@t-online.de
```

- 111111111111** Die zwölfstellige Anschlußkennung.
- 222222222222** Die ebenfalls (meist) zwölfstellige T-Online-Nummer. Ältere Verträge haben eventuell Nummern mit weniger als zwölf Stellen. In einem solchen Falle müssen Sie der T-Online-Nummer noch eine Raute # anhängen.
- 0001** Die Mitbenutzer-Nummer. Diese ist in den allermeisten Fällen die »0001«.

Alle diese Angaben finden Sie in dem Faltblatt, das Ihrem T-Online-Vertrag beilagt. Und zum Schluß wird dann noch der String *@t-online.de* angehängt.

11.2 Analoge Modems

Die *ppp.conf* für den Zugang mit einem analogen Modem gestaltet sich wie folgt:

(Einige Zeilen beginnen mit Leerzeichen — dies ist wichtig und muß genau so übernommen werden!)

```

default:
  set log Phase Chat LCP IPCP CCP tun command
  ident user-ppp VERSION (built COMPILATIONDATE)
  set device /dev/cuaa0
  set timeout 300
  set speed 115200
  set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \
          \\\" AT OK-AT-OK ATE1Q0 OK \\dATDT\\T TIMEOUT 40 CONNECT"
  enable dns

Arcor:
  set phone "0192075"
  set authname arcor-ibc
  set authkey internet
  set login
  set timeout 0
  set ifaddr 10.0.0.1/0 10.0.0.2/0 255.255.255.255
  add default HISADDR

```

Zur Konfiguration folgende Erläuterungen:

Mit *set device* geben Sie die Schnittstelle an, über welche das Modem zu erreichen ist. */dev/cuaa0* bezeichnet unter FreeBSD die erste serielle Schnittstelle.

set timeout bestimmt die Wartezeit in Sekunden (im Beispiel also 5 Minuten), nach der die Leitung getrennt wird, wenn keine Daten darüber geschickt werden.

set speed bestimmt die Übertragungsgeschwindigkeit der seriellen Schnittstelle, moderne Modems sollten mit dem Wert von 115200 Baud keine Probleme haben.

set dial enthält die AT-Codes, die das Modem zum Wählen ausführen soll. Dieses Beispiel ist nur für Modems geeignet, die *nicht* an einer Nebenstelle betrieben werden. Dieser Fall wird nämlich weiter unten erläutert.

enable dns schreibt die DNS-Server, die Ihr Provider beim Verbindungsaufbau übermitteln sollte, automatisch in die Datei */etc/resolv.conf*, so daß Sie sich um die Namensauflösung nicht mehr weiter kümmern müssen.

Anschließend erstellen wir einen neuen Abschnitt namens *Arcor* — Sie können den Abschnitt nennen wie Sie wollen, Sie können sogar mehrere Abschnitte anlegen. Wählen Sie als Namen des Abschnitts etwas, das Sie sich behalten können.

set phone gibt die Telefonnummer an, die das Modem wählen soll. Im Falle des zum Beispiel herangezogenen Arcor Internet-By-Call-Tarifs ist das die *0192075*.

authname ist der Benutzername, den Ihr Provider vorgegeben hat. Beim Arcor-Zugang im Beispiel beträgt er immer »*arcor-ibc*«.

Schließlich müssen Sie hinter *authname* das Paßwort im Klartext angeben. Die Datei *ppp.conf* ist aus diesem Grunde standardmäßig auch nur für *root* lesbar.

Nach Durchführen der Änderungen speichern Sie die Datei ab.

11.2.1 Analog-Modem an Nebenstelle

Wenn Ihr Modem an einer Nebenstelle angeschlossen ist, müssen Sie meist zuerst durch Wählen einer 0 eine Amtsleitung anfordern. Damit der PPP dies tut, müssen Sie den AT-String in der Zeile *set dial* um die im folgenden *rot* markierten beiden Zeichen ergänzen:

```
set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \  
        \\\" AT OK-AT-OK ATE1Q0 OK \\dATDTOW\\T TIMEOUT 40 CONNECT"
```

Damit wird PPP angewiesen, zunächst eine 0 zu wählen und anschließend auf das Freizeichen zu warten. Erst dann wird die eigentliche Nummer gewählt.

11.3 ISDN

Obleich FreeBSD die benötigten Programme zum Aufbau einer ISDN-Verbindung bereits im Basissystem mitbringt, ist die Konfiguration doch eher ein Thema für fortgeschrittene Anwender. Ein ausführliches Howto, das die nötigen Schritte beschreibt, findet man <http://www.bsdforen.de/showthread.php?t=3467> oder auch im FreeBSD Handbook. Ich verweise insoweit auf diese Dokumente, da es den Rahmen dieser Einführung sprengen würde.

11.4 DSL

Die Konfiguration von DSL ist unter FreeBSD wesentlich einfacher als die der anderen beiden Zugangstechniken. Sie sieht einfach wie folgt aus:

(Auch hier der Hinweis, daß die Leerzeichen zu Beginn einiger Zeilen unbedingt erforderlich sind.)

```

default:
  set device PPPoE:fxp1
  set MTU 1492
  set MRU 1492
  set dial
  set timeout 300
  set crtscts off
  set speed sync
  accept lqr
  disable deflate
  disable pred1
  disable vjcomp
  disable acfcomp
  disable protocomp
  disable iface-alias
  enable dns
  set log phase
  set ifaddr 10.0.0.1/0 10.0.0.2/0 0.0.0.0 0.0.0.0
  add default HISADDR

T-Online:
  set login
  set authname 1111111111122222222222220001@t-online.de
  set authkey 55685422

```

Die Erläuterung im Einzelnen:

set device gibt die Netzwerkkarte an, über die das DSL-Modem angeschlossen ist. Im Beispiel ist das die zweite Intel-Netzwerkkarte namens *fxp1* (die erste Intel-Karte, *fxp0*, ist in diesem Beispiel an das interne Netzwerk angeschlossen). Ermitteln Sie den Namen ggfs. wieder über die Bootmeldungen, die Sie mit *dmesg* anzeigen lassen.

set timeout bestimmt die Wartezeit in Sekunden (im Beispiel also 5 Minuten), nach der die Leitung getrennt wird, wenn keine Daten darüber laufen.

enable dns schreibt die DNS-Server, die Ihr Provider beim Verbindungsaufbau übermitteln sollte, automatisch in die Datei */etc/resolv.conf*, so daß Sie sich um die Namensauflösung nicht mehr weiter kümmern müssen.

Anschließend erstellen wir einen neuen Abschnitt namens *T-Online* — Sie können den Abschnitt nennen wie Sie wollen, Sie können sogar mehrere Abschnitte anlegen. Wählen Sie als Namen des Abschnitts etwas, das Sie sich behalten können.

authname ist der Benutzername, den Ihr Provider vorgegeben hat. Beim T-Online-Zugang ist er etwas komplizierter zu ermitteln, aber das sollte nach den Informationen zu Anfang des Kapitels kein Problem mehr sein.

Schließlich müssen Sie hinter *authname* das Paßwort im Klartext angeben. Die Datei *ppp.conf* ist aus diesem Grunde standardmäßig auch nur für *root* lesbar.

Auch hier dürfen Sie das Abspeichern der Datei natürlich nicht vergessen.

11.5 Starten des PPP-Daemons

Nun müssen wir den PPP noch starten, damit er eine Internet-Verbindung aufbaut. Je nach dem, welchen Tarif Sie nutzen, sind hauptsächlich zwei Möglichkeiten für Sie interessant.

11.5.1 Automatisches starten

Wenn Sie eine Flatrate oder einen Volumentarif haben, möchten Sie PPP vermutlich automatisch beim Systemstart ausführen lassen. Sobald das erste Programm eine Anfrage ins Internet stellt, wird die Verbindung aufgebaut. Führen Sie dazu als *root* folgenden Befehl aus:

```
echo 'ppp_enable="YES"  
ppp_mode="auto"  
ppp_profile="T-Online"' >> /etc/rc.conf
```

Geben Sie in der letzten Zeile den Namen des Profils an. In meiner beispielhaften Konfiguration für den DSL-Zugang hatte ich das Profil »*T-Online*« genannt, genau dieses soll hier vom PPP auch verwendet werden.

Der Abbau erfolgt dann entweder durch den eingestellten Timeout oder durch herunterfahren des Rechners. Wenn Sie die Internet-Verbindung einmal von Hand trennen möchten, führen Sie als *root* den Befehl

```
killall -HUP ppp
```

aus — damit wird der PPP beendet, die Internet-Verbindung abgebaut.

11.5.2 Manuelles starten

Wenn Sie einen zeitbasierten Tarif haben, möchten Sie möglicherweise selbst genau kontrollieren, wann eine Internet-Verbindung auf- und abgebaut wird. Sie können dann den PPP durch Eingabe des Befehls

```
ppp -auto Arcor
```

als *root* starten (»Arcor« ist auch hier wieder der Name des zu verwendenden Profils). Sie trennen die Leitung dann, indem Sie den Befehl

```
killall -HUP ppp
```

ausführen.

[Einbinden in KDE]

12. Software verwalten

Wenn Sie die vorherigen Kapitel durchgearbeitet haben, sollten Sie jetzt einen laufenden FreeBSD-Desktop besitzen. Einige Anwendungen sind auch schon installiert, hauptsächlich allerdings solche, die mit KDE ausgeliefert werden. Daneben können Sie unter FreeBSD aber natürlich noch viele weitere Anwendungen installieren und nutzen. Im folgenden zeige ich verschiedene Wege, die man dazu beschreiten kann.

12.1 Packages von der Release-CD

Beide Punkte wurden in vorhergehenden Kapiteln schon ausführlich beschrieben. Hier der Vollständigkeit halber noch einmal eine kurze Auffrischung.

12.1.1 Durchsuchen mit */stand/sysinstall*

Legen Sie die Install-CD ein, starten Sie */stand/sysinstall*, wählen Sie nacheinander die Menüpunkte »Configure« und »Packages«. Sie sehen nun die verschiedenen Kategorien, aus denen Sie Software-Pakete zur Installation auswählen können.

12.1.2 Schnelles Installieren mit *pkg_add*

Sie können die Software-Pakete auch mit *pkg_add* einspielen. Das ist von Vorteil, wenn Sie bereits den Namen des zu installierenden Pakets kennen. Legen Sie dazu die Install-CD ein und führen Sie folgendes aus:

```
mount /cdrom
cd /cdrom/packages/All
pkg_add emacs*
```

Dies würde das Package, dessen Name mit *emacs* beginnt, installieren. Hinter diesem Paket verbirgt sich der Editor GNU Emacs, eine Alternative zum im Basissystem enthaltenen *vi*. Auch unter FreeBSD soll schließlich jeder den Editor verwenden können, den er bisher unter Unix/Linux eingesetzt hat.

Wenn Sie weitere Pakete unter einem anderen Pfadnamen gespeichert haben, die eventuell als Abhängigkeiten benötigt werden könnten (beispielsweise anderweitig heruntergeladene Pakete), können Sie *pkg_add* dies über die Umgebungsvariable *PKG_PATH* mitteilen:

```
setenv PKG_PATH "/tmp/Pakete:/usr/ports/packages" (in der Csh)
export PKG_PATH="/tmp/Pakete:/usr/ports/packages" (in der Bash, Zsh usw.)
```

Benötigt ein zu installierendes Paket ein anderes Paket, würde also nicht nur im aktuellen Verzeichnis danach gesucht, sondern auch in den Verzeichnissen */tmp/Pakete* und */usr/ports/packages*.

12.2 Packages aus dem Internet beziehen

Das Software-Angebot auf den CDs ist natürlich aus Platzgründen eher beschränkt. Sie können jedoch auch Pakete aus dem Internet herunterladen und mit *pkg_add* installieren.

12.2.1 Zum Release zugehörige Packages

Dies sind Packages, die zum Zeitpunkt der Erstellung eines Releases erzeugt wurden. Dadurch ist sichergestellt, daß sie optimal mit diesem Release funktionieren sollten. Leider ist auch auf dem FTP-Server von FreeBSD der Platz nicht unendlich — Pakete für ein bestimmtes Release werden meist nach ein bis zwei Jahren wieder vom Server genommen.

Wenn Sie beispielsweise das Multimedia-Programm *XMMS* als Paket installieren möchten, werden Sie es auf der Install-CD nicht finden. Geben Sie dann als *root*

```
pkg_add -r xmms
```

ein — mit dem Schalter `-r` weisen Sie `pkg_add` an, das Paket vom FreeBSD-Server herunter zu laden und anschließend zu installieren. Sollte XMMS weitere Pakete benötigen, die Sie noch nicht installiert haben, werden diese ebenfalls automatisch aus dem Internet gezogen und eingespielt.

Auch bei dieser Vorgehensweise können Sie die Variable `$PKG_PATH` setzen, der Zweck wurde oben schon beschrieben.

12.2.2 Aktuelle Packages

Da die zum Release zugehörigen Pakete auf einem bestimmten Versions-Stand stehen bleiben (der Stand zum Zeitpunkt des Releases), sind diese nach einiger Zeit natürlich veraltet. Wenn Sie unbedingt eine neuere Version haben möchten, können Sie immer aktuelle Pakete von der Adresse <ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/packages> herunterladen. Sie finden dort beispielsweise im Unterverzeichnis *multimedia* ein Paket namens `xmms-1.2.10.tgz` (der Versionsnummer kann sich natürlich im Laufe der Zeit ändern). Laden Sie diese Datei mittels

```
pftp ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/packages/multimedia/xmms-1.2.10.tgz
```

herunter. Die Installation erfolgt anschließend wie gewohnt mittels

```
pkg_add xmms-1.2.10.tgz
```

Wenn XMMS noch ein anderes Paket benötigt, das Sie zur Zeit nicht installiert haben, müssen Sie dieses separat herunterladen und zuerst installieren. Daneben sind Pakete auch unflexibel, da immer genau die Version eines abhängigen Pakets installiert sein muß, die beim Erstellen angegeben wurde. Eventuell installierte ältere Versionen werden nicht akzeptiert. Dies ist nervig.

Wenn Sie wissen, was Sie tun, können Sie `pkg_add` mit dem Schalter `-f` aufrufen. Dann werden zwar Warnungen ausgegeben, das Paket aber auf jeden Fall installiert. Es kann hinterher funktionieren — oder auch nicht. Das ist ein Nachteil der Packages. Verwenden Sie die vorgestellte Methode nur in Ausnahmefällen oder wenn Sie eine als funktionierend bekannte Installationsanleitung für ein bestimmtes Paket finden (das trifft z. B. auf OpenOffice oder KDE zu).

Meist erhält man mit dem Portssystem aber bessere Ergebnisse, wir schauen uns das gleich an.

12.3 Das Portssystem

Etwas umfangreicher will ich das Portssystem, eine Spezialität der freien BSD-Systeme, erläutern. Wenn Sie vorher mit Gentoo Linux gearbeitet haben, wird Ihnen dies bekannt vorkommen, da die dortigen »Portages« auf dem Portssystem basieren.

Das Portssystem stellt einen Mechanismus dar, mit dem Sie eine Software (im folgenden »Port« genannt) aus dem Quelltext kompilieren, installieren und auch wieder komplett deinstallieren können. Hierzu werden die Quelltexte eines Ports automatisch heruntergeladen, für FreeBSD benötigte Patches angewandt, der Quelltext übersetzt, die übersetzten Dateien in das System installiert und der Port beim Paketsystem registriert. Benötigt ein Port einen anderen Port, der bisher nicht installiert ist, so werden alle Schritte auch für den benötigten Port durchgeführt.

Dadurch, daß jede zu einem Port zugehörige Datei bei der Installation registriert wird, kann der Port später wieder restlos entfernt werden.

12.3.1 Zurechtfinden im Portssystem

Unter `/usr/ports` befindet sich eine schier unüberschaubare Verzeichnishierarchie. Zunächst befinden sich für verschiedene Kategorien (z. B. Editoren, Multimedia-Anwendungen oder Spiele) jeweils eigene Unterverzeichnisse in `/usr/ports`. In jedem dieser Unterverzeichnisse befindet sich wiederum für jeden Port, der zu dieser Kategorie gehört, ein weiteres Unterverzeichnis. Dieses Unterverzeichnis enthält einige Dateien, u. a. das *Makefile* und die Datei *pkg-descr*, die eine ausführliche Information zu diesem Port enthält.

Sehen Sie z. B. einmal in das Verzeichnis `/usr/ports/audio`. Sie finden dort Ports, die in irgendeiner Form Audiodaten verarbeiten. Im Unterverzeichnis `/usr/ports/audio/audacity` befindet sich beispielsweise der Port *Audacity*, ein Programm zum Bearbeiten von Audiodateien. In der Datei `/usr/ports/audio/audacity/pkg-descr` finden Sie dann eine ausführlichere Beschreibung sowie den URL zur Webseite des Projekts und die E-Mail-Adresse des Autors.

Sie können die Ports-Sammlung auch durchsuchen — führen Sie dazu im Verzeichnis `/usr/ports` den Befehl

```
make search name=audacity
```

aus. Hiermit suchen Sie nach allen Ports, die im Namen »Audacity« enthalten. Ausgegeben wird auch das Verzeichnis, in dem sich der Port befindet, eine kurze, einzeilige Information sowie eine Liste der Ports, die zum Kompilieren (»B-Deps«) sowie zum Ausführen (»R-Deps«) benötigt werden. Diese würden also, wenn Sie den Port übersetzen würden, ebenfalls installiert, soweit sie nicht bereits auf dem System vorhanden sind.

Nun wissen Sie also, wie Sie einen Port finden, wenn Sie seinen Namen kennen. Sie können aber auch einen allgemeineren Suchbegriff verwenden, wenn Sie nicht wissen, wie der Name des Ports lautet. Führen Sie dazu in `/usr/ports` den Befehl

```
make search key="mp3 editor"
```

aus (die Anführungszeichen brauchen Sie nur, wenn der Suchbegriff Leer- oder andere Sonderzeichen enthält). Hiermit werden solche Ports gefunden, die in ihrem Namen, dem Pfad zum Verzeichnis des Ports oder in seiner Beschreibung die Zeichenkette »mp3 editor« enthalten. Als Ergebnis erhalten Sie eine Liste der Programme, mit denen Sie MP3-Dateien bearbeiten können. Dies findet allerdings nicht das Programm Audacity, da es in seiner Beschreibung nicht die Zeichenkette »mp3 editor« enthält. Eventuell müssen Sie hier also mit den Suchbegriffen etwas experimentieren, um zu einem zufriedenstellenden Ergebnis zu kommen.

Es kann also auch hilfreich sein, einfach einmal eine Kategorie, für die man sich interessiert, komplett zu durchstöbern. Sehr komfortabel geht das über die Website <http://www.freshports.org/categories.php>, die zu jedem Port den Namen, eine kurze Beschreibung und von diesem Port benötigte andere Ports anzeigt.

12.3.2 Einen Port installieren

Als Beispiel für die Installation eines Ports möchte ich XMMS heranziehen. Dieses Programm spielt die gebräuchlichsten Audioformate (Wave, Audio-CDs, MP3, OggVorbis, MOD) ab. Es kann durch Plugins, die sich ebenfalls im Portssystem finden, um das Abspielen neuer Formate und anderer, nur erdenklichen Funktionen erweitert werden. Visualisierungs-Plugins stellen Audioströme grafisch dar, die Oberfläche kann durch Skins angepaßt werden — kurz: ein großartiges Spielzeug.

Wenn Sie unter `/usr/ports` mit

```
make search name=xmms
```

auf die Suche gehen, werden Sie mit Suchergebnissen erschlagen, da auch alle möglichen XMMS-Plugins aufgelistet werden. Für einen schnellen Überblick können Sie daher auch

```
whereis -s xmms
```

verwenden — dies liefert Ihnen als Ausgabe beispielsweise:

```
xmms: /usr/ports/chinese/xmms /usr/ports/multimedia/xmms  
      /usr/ports/russian/xmms
```

Hier sind also auch chinesisch- und russischsprachige Versionen im Angebot. Wir beschränken uns auf die Version `/usr/ports/multimedia/xmms`, die in verschiedene Sprachen mit lateinischem Alphabet (u. a. Englisch, Deutsch, Afrikaans, Galizisch usw.) übersetzt ist.

Wechseln Sie also nach `/usr/ports/multimedia/xmms` und führen Sie als *root* den Befehl

```
make install
```

aus. Dies lädt den Quellcode herunter, patcht ihn (wenn nötig), kompiliert ihn, installiert die kompilierten Dateien und registriert sie beim Paketverwaltungssystem. Das ist also der »Rundum-sorglos-Befehl«, der verschiedene *make*-Läufe impliziert und für Sie automatisch ausführt. Im folgenden dennoch die wichtigsten *make*-Läufe im Überblick. Sie können diese selbst eingeben, um die verschiedenen Schritte einer Installation genau zu kontrollieren und ggf. einzugreifen (wenn Sie wissen, was Sie tun, und sich mit dem Kompilieren von Software auskennen).

- | | |
|---------------------|---|
| <i>make fetch</i> | Lädt nur den Quellcode herunter, üblicherweise ein <i>gzip</i> -komprimiertes <i>tar</i> -Archiv, das in <code>/usr/ports/distfiles</code> abgelegt wird. |
| <i>make extract</i> | Entpackt den Quellcode. Dazu hat jeder Port in seinem Verzeichnis das Unterverzeichnis <i>work</i> , in welchem die entpackten Quellen abgelegt werden. |
| <i>make patch</i> | Wendet die Patches, die für die Ausführung unter FreeBSD notwendig sind, auf den entpackten Quellcode an. |

- make build* oder einfach nur *make*
Übersetzt den Quellcode und erzeugt ausführbare Dateien. Dies wird ebenfalls alles im bereits erwähnten Verzeichnis *work* durchgeführt.
- make install* Kopiert die übersetzten Dateien aus *work* in das System und registriert diese beim Paketverwaltungssystem.
- make package* Erstellt aus dem Port ein binäres Paket als *.tar.gz*, wie sie z. B. auf der Install-CD zu finden sind. Sie können dieses Paket dann mit *pkg_add* auf einem anderen Rechner installieren.

Jeder *make*-Lauf impliziert die vorhergehenden Schritte, soweit diese nicht bereits durchgeführt wurden. Das heißt, wenn Sie *make extract* angeben, wird zuerst ein *make fetch* ausgeführt, außer, die Quellcodes befinden sich bereits in */usr/ports/distfiles*. Ein *make package* impliziert dementsprechend auch ein *make install*. Tatsächlich existiert unter FreeBSD keine Möglichkeit, ein Binär-Paket zu erstellen, ohne den Port vorher in das System zu installieren.

Wenn Sie mit dem Übersetzen des Ports fertig sind, möchten Sie möglicherweise etwas Plattenspeicher freigegeben. Führen Sie

```
make clean
```

aus, um das Verzeichnis *work* zu leeren. Es werden auch alle Verzeichnisse *work* von abhängigen Ports geleert, soweit diese ebenfalls kompiliert wurden. Weiterhin enthält das Verzeichnis */usr/ports/distfiles* anschließend immer noch die heruntergeladenen Quellcodes der Software. Überlegen Sie sich, ob Sie diese noch einmal benötigen, meist nur dann, wenn Sie das Programm in dieser Version noch einmal kompilieren möchten. Ansonsten können Sie das Archiv mit dem Quellcode ebenfalls löschen, um den Speicherplatz freizugeben.

12.3.3 Portssystem aktualisieren

Auf Ihrer Festplatte befindet sich das Portssystem, das auf der Install-CD ausgeliefert wurde. Damit können Sie von den Ports genau die Versionen installieren, die zum Zeitpunkt des Erstellens der Install-CD aktuell waren. Software entwickelt sich aber weiter. Damit Sie eine neuere Version eines Ports installieren können, müssen Sie allerdings auch ein aktuelleres Portssystem installieren.

Dies ist im Prinzip kein großes Problem, da das Portssystem täglich aktualisiert wird und so immer die neuesten Versionen der Anwendungen anbieten kann. Sie müssen »nur« die neue Version des Portsystems herunterladen und installieren. Sie können sich jedoch Probleme einhandeln, wenn zwischenzeitlich neuere Versionen elementarer Systemwerkzeuge im Portssystem verfügbar sind und dann automatisch beim Übersetzen eines anderen Ports installiert werden könnten. Als Negativbeispiel soll hier die Programmiersprache *ruby* dienen, die von einigen Ports benötigt wird, aber bei der es manchmal nicht so leicht war, eine neue Version fehlerfrei zu installieren.

Bleiben Sie für den Anfang also bei der Version des Portsystems, die auf Ihrer Install-CD mitgeliefert wurde. Sie ist ideal auf das verwendete Release abgestimmt. Wenn Sie etwas mehr Übung mit den Ports haben, können Sie im Abschnitt 4.5 des FreeBSD Handbook alles über das Aktualisieren des Portsystems erfahren.

12.4 Welche Software ist installiert?

Wenn Sie sich einen Überblick über die installierte Software machen wollen, hilft Ihnen das Programm *pkg_info*. Rufen Sie es ohne weitere Angaben auf, werden alle installierten Pakete angezeigt. Auf einem Desktop-Rechner kann sich die Auflistung über mehrere Bildschirmseiten erstrecken, da hier meist recht viele Anwendungen installiert werden.

12.5 Software wieder loswerden

Es ist unerheblich, ob Sie eine Software mit *pkg_add* oder über das Portssystem installiert hatten. Der Weg, um eine Software zu deinstallieren, ist in beiden Fällen der gleiche.

Zuerst müssen Sie den *genauen* Namen des Pakets kennen, inklusive der Versionsnummer. Da Sie dies vermutlich nicht auswendig kennen, rufen Sie zunächst *pkg_info* auf, um an die nötigen Informationen heranzukommen. Verwenden Sie außerdem *grep*, um die Bildschirmausgabe nach einem bestimmten Suchbegriff zu filtern:

```
pkg_info | grep xmms
xmms-1.2.8_1    X Multimedia System - An audio player with a Winamp
```

Hier sagt uns *pkg_info* also, daß XMMS als Paket mit dem Namen *xmms-1.2.8_1* installiert ist. Wir löschen es wie folgt:

```
pkg_delete xmms-1.2.8_1
```

Sollte *pkg_delete* bemängeln, daß das zu deinstallierende Paket noch von anderen Paketen benötigt wird, sollten Sie gut überlegen, ob Sie es wirklich deinstallieren möchten. Eventuell könnten andere Pakete dann nicht mehr richtig funktionieren. Sie können die Deinstallation in einem solchen Falle mit dem Schalter *-f* erzwingen:

```
pkg_delete -f xmms-1.2.8_1
```

13. Empfehlenswerte Software

Das Portssystem enthält mehr als 10.000 Anwendungen — und täglich werden es mehr. Durchstöbern Sie das Portssystem ruhig nach für Sie interessanter Software, Sie werden sicher fündig. Für den schnellen Einstieg möchte ich aber ein paar oft im Desktop-Bereich eingesetzte Anwendungen kurz vorstellen, damit Sie nicht erst alle Ports durchsuchen müssen. Sie können diese Anwendungen natürlich auch mittels

```
pkg_add -r xmms
```

aus dem Internet als Pakete herunterladen, da das Übersetzen aus dem Quelltext auf langsamen Rechnern doch einige Zeit in Anspruch nehmen kann.

Bei einigen Anwendungen möchte ich außerdem interessante Optionen erwähnen, die beim Übersetzen verwendet werden können. Bei vielen Ports kann z. B. die Option *--enable-esd* benutzt werden, um die Unterstützung für das eSound-System zu aktivieren:

```
make --enable-esd install
```

Das funktioniert natürlich nur, wenn Sie die Software aus dem Quelltext übersetzen. Bei Paketen aus dem Internet müssen Sie mit den Optionen vorlieb nehmen, die beim damaligen Übersetzen des Pakets eingesetzt wurden.

Zur Verwendung der Software sei auch hier wieder auf die entsprechende Dokumentation verwiesen. Insbesondere können Sie, wie bereits besprochen, über die Datei *pkg-descr* im Verzeichnis des Ports die Adresse der Projekt-Homepage herausfinden und dort meist auf umfangreiche Dokumentation zugreifen.

13.1 Acrobat Reader

/usr/ports/print/acroread

Der Acrobat Reader von Adobe zum Anzeigen von PostScript- und PDF-Dateien.

Sie sollten das mit KDE mitgelieferte *KGHostview* bevorzugen, da beim Acrobat Reader der Bildlauf über das Mausrad nicht unterstützt wird und es diverse andere Probleme (bis hin zu Abstürzen der Software) gibt. Dieser Port wird daher im Prinzip nur benötigt, wenn Sie PDF-Dateien anzeigen lassen wollen, die *KGHostview* nicht richtig verarbeitet. Kann schon mal vorkommen, insbesondere Formularfelder in PDF-Dateien können zur Zeit nur mit dem Acrobat Reader ausgefüllt werden.

13.2 Bochs

/usr/ports/emulators/bochs

Emuliert einen kompletten Intel-PC innerhalb einer virtuellen Maschine.

--enable-all-optimizations

Aktiviert alle von den Entwicklern als stabil eingestuften Optimierungen.

--enable-vbe

Verbesserte Grafikemulation verwenden (bis zu 1024 · 768 Pixel mit 16 bit Farbtiefe können emuliert werden).

13.3 Bourne Again Shell (Bash)

/usr/ports/shells/bash2

Die vielen Linux-Umsteigern bekannte Shell in der Version 2.

13.4 Deutsche KDE-Lokalisierung

/usr/ports/german/kde3-i18n

Deutsche Lokalisierung für die KDE-Oberfläche.

13.5 GNU Ghostscript

/usr/ports/print/ghostscript-gnu-nox11

Wandelt PostScript-Dateien in eine der verschiedenen Druckersprachen um. Überwiegend für den Einsatz von nicht-PostScript-fähigen Druckern benötigt.

13.6 GNU Ghostview

/usr/ports/print/ghostview

Programm zum Anzeigen von PostScript- und PDF-Dateien. Wird u. a. von *KGhostview* benötigt, um PostScript- oder PDF-Dokumente auf dem Bildschirm darzustellen.

13.7 GIMP

/usr/ports/graphics/gimp

Umfangreiches Bildbearbeitungsprogramm.

13.8 Java

/usr/ports/java/jdk14

Java-Entwicklungsumgebung für FreeBSD. Wird z. B. vom Konqueror benutzt, um Java-Inhalte auf Webseiten darzustellen (nicht mit *JavaScript* verwechseln, das ist etwas ganz anderes). Außerdem gedacht zur Ausführung von Java-Anwendungen wie beispielsweise *LimeWire*.

/usr/ports/java/linux-sun-jdk14

Java-Entwicklungsumgebung für Linux. Sie brauchen diese nur, wenn Sie Java mit Linux-Programmen unter FreeBSD nutzen wollen. Linux-Programme können nämlich *keine* FreeBSD-Bibliotheken verwenden, da diese zu den Linux-Bibliotheken *nicht* kompatibel sind.

Die Java-Ports sind insofern etwas kompliziert, als daß *make* das Distfile nicht automatisch herunterladen darf. Suns Lizenzbedingungen verhindern dies. Wenn Sie *make* ausführen, erhalten Sie den Hinweis, wo Sie das Distfile herunterladen können. Laden Sie es herunter, verschieben Sie es nach */usr/ports/distfiles* und führen Sie *make* anschließend erneut aus.

13.9 K Desktop Environment

/usr/ports/x11/kde3

Grafische Desktop-Umgebung in der Version 3.x für FreeBSD.

Dies soll mehr der Information dienen. Als Einsteiger sollten Sie mit der KDE-Version arbeiten, die auf der Install-CD vorhanden ist. Das Aktualisieren von KDE auf die neueste Version ist nur etwas für fortgeschrittene Anwender. Und auch nur

die sollten die Binär-Pakete von <http://rabarber.fruitsalad.org/> verwenden, da dies nicht weniger kompliziert ist. Allerdings sparen einem diese Pakete sehr viel Zeit, da das Übersetzen von KDE aus dem Quelltext eine sehr langwierige Angelegenheit ist.

13.10 Kopete

/usr/ports/net/kopete

Ein Client für verschiedene Instant-Messaging-Netzwerke.

Kopete unterstützt die Protokolle Jabber, ICQ, AIM, MSN, Yahoo, IRC, Windows LANs, GaduGadu und SMS (Angaben laut Website, da ich natürlich unmöglich alle ausprobieren kann).

13.11 MPlayer

/usr/ports/multimedia/mplayer

Spielt verschiedenste Video-Formate ab (MPEG, AVI, WMV usw.).

--enable-esd

Ermöglicht die Ausgabe von Sounds über das eSound-System.

--disable-mencoder

Übersetzt den Port, ohne *mencoder* zu installieren. Es handelt sich dabei um ein Programm, mit dem Sie MPEG-Filme (z. B. DVDs, SVCDs) in andere Formate (z. B. DiVX) kodieren können. Wenn Sie das nicht vorhaben, können Sie mit dieser Option Zeit und Platz sparen.

13.12 OpenOffice

/usr/ports/german/openoffice-1.1

Eine Sammlung von Office-Programmen (Textverarbeitung, Tabellenkalkulation, Präsentationen, usw) mit deutschsprachiger Oberfläche.

Wenn Sie diese Software selbst übersetzen wollen, brauchen Sie *viel* Zeit und auch viel Speicherplatz, da es sich hierbei um einen der etwas größeren Ports handelt. Schneller geht es, wenn Sie sich ein bereits fertig übersetztes Binär-Paket herunterladen und dieses dann einfach mit *pkg_add* installieren. Fertige Pakete bekommen Sie unter der Adresse <http://projects.imp.ch/openoffice/>, auch Wörterbücher verschiedener Sprachen werden dort bereitgestellt. Diese Vorgehensweise wird von mir ausdrücklich empfohlen.

13.13 RealPlayer

/usr/ports/multimedia/linux-realplayer

Anwendung zum Abspielen von RealMedia-Streams.

Der Port erzeugt nach dem Installieren eine Fehlermeldung. Er versucht nämlich, ein grafisches Tool zu starten, um Sie ein bißchen über Ihre persönlichen Informationen auszuquetschen. Wenn Sie mit *su* zum Benutzer *root* gewechselt sind oder auf der Systemkonsole arbeiten, kann sich dieses Spionage-Tool nicht mit dem X-Server verbinden. Die Installation ist zu diesem Zeitpunkt trotzdem schon abgeschlossen, sodaß Sie sich über die Fehlermeldung keine weiteren Gedanken machen müssen. Datenschutz einfach mal anders.

13.14 Vi Improved

/usr/ports/editors/vim

Der Editor *vim*, eine verbesserte Version des im Basissystem vorhandenen *vi*.
Enthalten ist in diesem Port auch die grafische Version *gvim*.

/usr/ports/editors/vim-lite

Dieser Port enthält nur den *vim*, ohne die grafische Version *gvim*.

13.15 XMMS

/usr/ports/multimedia/xmms

Spielt verschiedenste Audio-Formate ab (Audio-CDs, MP3, OggVorbis usw.).

--enable-esd

Ermöglicht die Ausgabe von Sounds über das eSound-System.

14. Die FreeBSD-Verzeichnisstruktur

Ich beschreibe die meisten Verzeichnisse nur kurz, eine umfangreiche Liste finden Sie in der Manpage *hier*:

man hier

Lediglich auf die Konfigurationsdateien werde ich im entsprechenden Abschnitt etwas näher eingehen. Wenn Sie nähere Informationen benötigen, schauen Sie bitte in die entsprechende Manpage. Um beispielsweise die Manpage zum Programm *passwd* anzuzeigen:

```
man 1 passwd
```

Wie Sie weiter oben in diesem Dokument schon gelesen haben, gibt die Zahl die Sektion an, zu welcher die Manpage gehört. Sektion 1 enthält die Benutzerprogramme, wie z. B. *passwd*. Sie hätten die Angabe der Sektion auch weglassen können, dann wäre automatisch die zuerst gefundene Manpage angezeigt worden. Wenn Sie jetzt die Manpage zur Datei *passwd* anzeigen lassen wollen, müssen Sie explizit die Sektion angeben, sonst erhalten Sie wieder nur die Manpage zum Befehl *passwd*:

```
man 5 passwd
```

Dann sehen Sie die Manpage *passwd* aus Sektion 5 (=Konfigurationsdateien).

14.1 Root-Verzeichnis

Das Verzeichnis */* wird das Root- oder Wurzelverzeichnis genannt. Es ist die höchste Ebene der FreeBSD-Verzeichnishierarchie. Hier befindet sich auch die Datei *kernel*, der Betriebssystemkern. Ab FreeBSD 5.x wurde er allerdings nach */boot* verschoben.

14.2 Benutzerprogramme

Die Verzeichnisse */bin* und */usr/bin* enthalten Programme, die zum täglichen Arbeiten gebraucht werden. Außerdem finden Sie in */usr/local/bin* ausführbare Programme von Software, die nicht zum Grundsystem gehört, z. B. solche, die Sie aus dem Portssystem installiert haben.

14.3 Zum Booten benötigte Dateien

Im Verzeichnis */boot* finden sich die Dateien, die zum Booten von FreeBSD notwendig sind. Auch Dateien, die den Boot-Vorgang oder den Boot-Manager konfigurieren, sind hier zu finden. Sie sollten hier keine Änderungen vornehmen, wenn Sie nicht genau wissen, was Sie tun. Ansonsten könnte es passieren, daß FreeBSD nicht mehr booten kann. Hier befindet sich seit FreeBSD 5.x auch der Kernel.

14.4 Gerätedateien

Gerätedateien sind die Schnittstelle zwischen Kernel und Anwendungsprogrammen, um auf eine bestimmte Hardware zuzugreifen. Unter Unix wird fast alles durch eine Datei dargestellt, z. B. Tastatur, Maus, Festplatten, Disketten, CDs, Terminals usw.

14.5 Konfigurationsdateien

Unter FreeBSD können Konfigurationsdateien hauptsächlich in zwei Verzeichnissen vorgefunden werden:

/etc In diesem Verzeichnis befinden die Konfigurationsdateien, die zum Grundsystem von FreeBSD gehören.

/usr/local/etc

Hier befinden sich die Konfigurationsdateien vieler Programme, die nicht zum Grundsystem gehören. In der Regel sind das solche Programme, die Sie mittels *pkg_add* oder über das Portssystem installiert haben. Daneben können die Programme aber auch noch ganz andere Verzeichnisse zum Ablegen der Konfiguration wählen. Daher muß eventuell in der Programmdokumentation gesucht werden, wenn Sie die Konfigurationsdateien eines Programms nicht finden können.

Ich beschreibe nur die Konfigurationsdateien des Grundsystems, da die Liste ansonsten unendlich lange würde.

14.5.1 Konfigurationsdateien des Grundsystems

Im Verzeichnis */etc* finden Sie u. a. folgende Dateien:

crontab Systemweite Cron-Tabelle, in der immer wiederkehrende Aufgaben automatisiert werden können.

<i>fstab</i>	Liste der beim Systemstart einzuhängenden oder manuell einhängbaren Dateisysteme.
<i>group</i>	Die Gruppendatenbank, in der Benutzergruppen verwaltet werden.
<i>host.conf</i>	Konfiguriert die Reihenfolge, in der Hostnamen aufgelöst werden sollen.
<i>hosts</i>	Namensauflösung für kleine Netzwerke.
<i>hosts.allow</i>	Rechnerbasierte Zugriffsrechte auf Systemdienste.
<i>hosts.lpd</i>	Rechnerbasierte Zugriffsrechte auf das BSD-Druckersystem.
<i>inetd.conf</i>	Konfiguration des Internet Super Servers.
<i>login.conf</i>	Definiert Login-Klassen.
<i>master.passwd</i>	Die Paßwort-Datei. Enthält auch die verschlüsselten Paßwörter, sollte daher nur für <i>root</i> lesbar sein. Wenn Sie diese Datei bearbeiten möchten (und wissen, was Sie tun!), verwenden Sie dazu bitte den Befehl <i>vipw</i> .
<i>passwd</i>	Eine weitere Paßwort-Datei. Sie sollten diese Datei nie bearbeiten, da sie von <i>vipw</i> automatisch aus der <i>master.passwd</i> erzeugt wird. In der <i>passwd</i> sind einige Angaben, wie z. B. das verschlüsselte Paßwort, herausgefiltert, daher darf diese Datei auch von allen Benutzern gelesen werden.
<i>printcap</i>	Die Druckerdatenbank des BSD-Druckersystems.
<i>profile</i>	Enthält Befehle, die beim Anmelden jedes Benutzers von der Shell ausgeführt werden.
<i>pwd.db</i>	Die Paßwort-Datenbank im Berkley-Datenbank-Format. Das System bezieht seine Informationen ausschließlich aus dieser Datei. Sie wird automatisch von <i>vipw</i> angelegt und aktualisiert. Sie sollten diese Datei <i>niemals direkt</i> editieren.
<i>rc.conf</i>	Sicherlich eine der wichtigsten Konfigurationsdateien. Sie konfiguriert den Großteil des Boot-Vorgangs von FreeBSD, z. B. den Hostnamen, die IP-Adresse und den Start verschiedener Systemdienste. In <i>/etc/defaults/rc.conf</i> finden Sie eine Dokumentation aller Variablen, die in <i>/etc/rc.conf</i> verwendet werden können.

<i>resolv.conf</i>	Enthält die DNS-Server, die zur Namensauflösung verwendet werden sollen. Die Datei existiert standardmäßig nicht, Sie müssen sie erst anlegen, wenn Sie einen DNS-Server verwenden möchten.
<i>spwd.db</i>	Kopie der <i>pwd.db</i> , existiert nur zu Kompatibilitätszwecken mit älterer Software, welche die Paßwortdatenbank noch unter diesem Namen erwartet.
<i>sysctl.conf</i>	Enthält Systemvariablen, die beim Booten gesetzt werden sollen.
<i>syslog.conf</i>	Konfiguriert <i>syslog</i> , der Systemereignisse protokolliert.
<i>ttys</i>	Definition der Terminals, die von <i>init</i> gestartet werden.
<i>usbd.conf</i>	Legt Befehle fest, die beim Ein- oder Ausstecken eines bestimmten USB-Geräts ausgeführt werden sollen. Dies ist insbesondere interessant, wenn Sie eine USB-Maus mit einem Scroll-Rad besitzen. Sie können das Rad zum Bildlauf in KDE verwenden, wenn Sie in der <i>usbd.conf</i> die Zeile

```
attach "/usr/sbin/moused -p /dev/${DEVNAME}
-I /var/run/moused.${DEVNAME}.pid ;
/usr/sbin/vidcontrol -m on"
```

(alles auf einer einzigen Zeile, hier nur aus drucktechnischen Gründen auf drei Zeilen umgebrochen!)

wie folgt abändern:

```
attach "/usr/sbin/moused -p /dev/${DEVNAME} -z 4
-I /var/run/moused.${DEVNAME}.pid ;
/usr/sbin/vidcontrol -m on"
```

(einzufügender Text in **roter** Schriftfarbe dargestellt)

14.6 Heimatverzeichnisse der Benutzer

Im Verzeichnis */home* besitzt jeder Anwender sein persönliches Heimatverzeichnis. Dieses und die Verzeichnisse */tmp* und */var/tmp* sind normalerweise die einzigen Orte im Dateisystem, in denen ein Benutzer Schreibrechte besitzt.

14.7 Kernel-Module

Kernel-Module sind Code-Teile, die nach dem Booten des Betriebssystems zum Kernel hinzugeladen oder entladen werden können. Sie fügen dem Kernel zusätzliche Fähigkeiten hinzu. So gibt es z. B. Module für verschiedene Soundkarten. Wenn Sie ein solches Modul laden, kann der Kernel die jeweilige Soundkarte ansprechen. Sie finden die Module im Verzeichnis */modules*.

14.8 Das Heimatverzeichnis von *root*

Auch der Benutzer *root* hat ein Heimatverzeichnis — es findet sich unter */root* und sollte meist nicht weiter interessant sein.

14.9 Administrationswerkzeuge

Sie finden unter */sbin* und */usr/sbin* überwiegend Anwendungen, die nur vom Benutzer *root* ausgeführt werden können und der Administration des Rechners dienen. Administrationswerkzeuge von Dritt-Software (z. B. einem Port) werden unter */usr/local/sbin* abgelegt.

14.10 Temporäre Dateien

Unter */tmp* können von jedem Benutzer Dateien angelegt werden, die nur temporär benötigt werden. Hier speichert z. B. KDE Informationen über die aktuelle Sitzung.

14.11 rc-Startscripts

Wenn Sie von einem anderen Unix oder von Linux umsteigen, kennen Sie sicher die Möglichkeit, Shellscrippts beim Booten ausführen zu lassen. Hierfür gibt es unter FreeBSD das Verzeichnis */usr/local/etc/rc.d*. Der Dateinamen eines Scripts muß auf *.sh* enden und das Script muß ausführbar sein. Nur wenn diese beiden Voraussetzungen erfüllt sind, werden die Scripts am Ende des Bootvorgangs ausgeführt.

14.12 Veränderbare Dateien

Dateien, die sich häufig verändern, werden im Verzeichnis */var* abgelegt. Hier befinden sich z. B. Log-Dateien, Spool-Verzeichnisse des Druckersystems sowie eingegangene E-Mails.