Das FreeBSD-Handbuch

Das FreeBSD-Handbuch

Version: 48232
2016-02-21 von bhd.

Zusammenfassung

Willkommen bei FreeBSD! Dieses Handbuch beschreibt die Installation und den täglichen Umgang mit *FreeBSD 8.4-RELEASE*, *9.3-RELEASE* und *FreeBSD 10.3-RELEASE*. Das Handbuch ist *jederzeit unter Bearbeitung* und das Ergebnis der Arbeit vieler Einzelpersonen. Dies kann dazu führen, dass bestimmte Bereiche nicht mehr aktuell sind und auf den neuesten Stand gebracht werden müssen. Bei Unklarheiten empfiehlt es sich daher stets, die englische Originalversion des Handbuchs zu lesen.

Wenn Sie bei der Übersetzung des Handbuchs mithelfen möchten, senden Sie bitte eine E-Mail an die Mailingliste 'FreeBSD German Documentation Project' <de-bsd-translators@de.FreeBSD.org>.

Die aktuelle Version des Handbuchs ist immer auf dem FreeBSD-Webserver verfügbar und kann in verschiedenen Formaten und in komprimierter Form vom FreeBSD FTP-Server oder einem der vielen Spiegel heruntergeladen werden (ältere Versionen finden Sie hingegen unter `http://docs.FreeBSD.org/doc/`). Vielleicht möchten Sie das Handbuch aber auch durchsuchen.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Liste der Beispiele

Vorwort

Über dieses Buch

Der erste Teil dieses Buchs führt FreeBSD-Einsteiger durch den Installationsprozess und stellt leicht verständlich Konzepte und Konventionen vor, die UNIX® zu Grunde liegen. Sie müssen nur neugierig sein und bereitwillig neue Konzepte aufnehmen, wenn diese vorgestellt werden, um diesen Teil durchzuarbeiten.

Wenn Sie den ersten Teil bewältigt haben, bietet der umfangreichere zweite Teil eine verständliche Darstellung vieler Themen, die für FreeBSD-Administratoren relevant sind. Wenn Kapitel auf anderen Kapiteln aufbauen, wird das in der Übersicht am Anfang eines Kapitels erläutert.

Weitere Informationsquellen entnehmen Sie bitte Anhang B, *Bibliografie*.

Änderungen gegenüber der dritten Auflage

Die aktuelle Auflage des Handbuchs ist das Ergebnis der engagierten Arbeit Hunderter Mitarbeiter des FreeBSD Documentation Projects in den vergangenen 10 Jahren. Die wichtigsten Änderungen dieser Auflage gegenüber der dritten Auflage von 2004 sind:

- Kapitel 25, *DTrace* informiert Sie über die mächtigen Funktionen zur Leistungsmessung, die dieses Werkzeug bietet.
- Kapitel 21, *Dateisystemunterstützung* enthält Informationen über die Unterstützung nicht-nativer Dateisysteme in FreeBSD, wie beispielsweise ZFS von Sun™.
- Kapitel 17, *Security Event Auditing* informiert über die neuen Auditing-Fähigkeitenvon FreeBSD.
- Kapitel 22, *Virtualisierung* enthält Informationen zur Installation von FreeBSD in verschiedenen Virtualisierungs-Programmen.
- Kapitel 3, *FreeBSD 9.X (und neuer) installieren* wurde hinzugefügt, um die Installation von FreeBSD mit dem neuen Installationswerkzeug, bsdinstall, zu dokumentieren.

Änderungen gegenüber der zweiten Auflage (2004)

Die dritte Auflage des Handbuchs war das Ergebnis der über zwei Jahre dauernden engagierten Arbeit des FreeBSD Documentation Projects. Die gedruckte Ausgabe war derart umfangreich, dass es notwendig wurde, sie in zwei Bände aufzuteilen. Die wichtigsten Änderungen dieser Auflage waren:

- Kapitel 12, *Konfiguration und Tuning* enthält neue Abschnitte über ACPI, Energie- und Ressourcenverwaltung und das Werkzeug `cron`.
- Kapitel 14, *Sicherheit* erläutert nun Virtual Private Networks (VPNs), Zugriffskontrolllisten (ACLs) und Sicherheitshinweise.
- Kapitel 16, *Verbindliche Zugriffskontrolle* ist ein neues Kapitel, das vorgeschriebene Zugriffskontrollen vorstellt und erklärt, wie FreeBSD-Systeme mit MACs abgesichert werden können.
- Kapitel 18, *Speichermedien* enthält neue Informationen über USB-Speichergeräte, Dateisystem-Snapshots, Quotas, Datei- und Netzwerk-basierte Dateisysteme sowie verschlüsselte Partitionen.
- Zum Kapitel 27, *PPP* wurde ein Abschnitt über Fehlersuche hinzugefügt.

- Kapitel 28, *Elektronische Post (E-Mail)* wurde um Abschnitte über alternative Transport-Agenten (MTAs), SMTP-Authentifizierung, UUCP, fetchmail, procmail und weitere Themen erweitert.
- Kapitel 29, *Netzwerkserver* ist ein weiteres neues Kapitel dieser Auflage. Das Kapitel beschreibt, wie der Apache HTTP-Server, ftpd und ein Samba-Server für Microsoft® Windows®-Clients eingerichtet werden. Einige Abschnitte aus dem Kapitel 31, *Weiterführende Netzwerkthemen* befinden sich nun, wegen des thematischen Zusammenhangs, in diesem Kapitel.
- Das Kapitel 31, *Weiterführende Netzwerkthemen* beschreibt nun den Einsatz von Bluetooth®-Geräten unter FreeBSD und das Einrichten von drahtlosen Netzwerken sowie ATM-Netzwerken.
- Neu hinzugefügt wurde ein Glossar, das die im Buch verwendeten technischen Ausdrücke definiert.
- Das Erscheinungsbild der Tabellen und Abbildungen im Buch wurde verbessert.

Änderungen gegenüber der ersten Auflage (2001)

Die zweite Auflage ist das Ergebnis der engagierten Arbeit der Mitglieder des FreeBSD Documentation Projects über zwei Jahre. Die wichtigsten Änderungen gegenüber der ersten Auflage sind:

- Ein Index wurde erstellt.
- Alle ASCII-Darstellungen wurden durch Grafiken ersetzt.
- Jedes Kapitel wird durch eine Übersicht eingeleitet, die den Inhalt des Kapitels zusammenfasst und die Voraussetzungen für ein erfolgreiches Durcharbeiten des Kapitels darstellt.
- Der Inhalt wurde in die logischen Abschnitte „Erste Schritte“, „Systemadministration“ und „Anhänge“ unterteilt.
- Kapitel 2, *FreeBSD 8.X (und älter) installieren* wurde komplett neu geschrieben und mit Abbildungen versehen, die Einsteigern das Verständnis des Texts erleichtern.
- Kapitel 4, *Grundlagen des UNIX Betriebssystems* wurde um den Abschnitt „Dämonen, Signale und Stoppen von Prozessen“ erweitert.
- Das Kapitel 5, *Installieren von Anwendungen: Pakete und Ports* behandelt nun auch Pakete.
- Kapitel 6, *Das X-Window-System* wurde neu geschrieben. Der Schwerpunkt liegt auf modernen Benutzeroberflächen wie KDE und GNOME unter XFree86™ 4.X.
- Das Kapitel 13, *FreeBSDs Bootvorgang* wurde erweitert.
- Kapitel 18, *Speichermedien* ist aus den beiden Kapiteln „Laufwerke“ und „Sicherungen“ entstanden. Die in den beiden Kapiteln diskutierten Themen sind so leichter zu verstehen. Hinzugekommen ist ein Abschnitt über Software- und Hardware-RAID.
- Das Kapitel 26, *Serielle Datenübertragung* wurde reorganisiert und auf FreeBSD 4.X/5.X angepasst.
- Das Kapitel 27, *PPP* wurde aktualisiert.
- Kapitel 31, *Weiterführende Netzwerkthemen* wurde um viele neue Abschnitte erweitert.
- Kapitel 28, *Elektronische Post (E-Mail)* wurde um einen Abschnitt über die Konfiguration von Sendmail erweitert.
- Kapitel 11, *Linux-Binärkompatibilität* behandelt zusätzlich die Installation von Oracle® und SAP® R/3®.
- Neu hinzugekommen sind:

- Kapitel 12, *Konfiguration und Tuning*.
- Kapitel 8, *Multimedia*.

Gliederung

Dieses Buch ist in fünf Abschnitte unterteilt. Der erste Abschnitt, *Erste Schritte*, behandelt die Installation und die Grundlagen von FreeBSD. Dieser Abschnitt sollte in der vorgegebenen Reihenfolge durchgearbeitet werden, schon Bekanntes darf aber übersprungen werden. Der zweite Abschnitt, *Oft benutzte Funktionen*, behandelt häufig benutzte Funktionen von FreeBSD. Dieser Abschnitt sowie alle nachfolgenden Abschnitte können in beliebiger Reihenfolge gelesen werden. Jeder Abschnitt beginnt mit einer kurzen Übersicht, die das Thema des Abschnitts und das nötige Vorwissen erläutert. Die Übersichten helfen dem Leser, interessante Kapitel zu finden und erleichtern das Stöbern im Handbuch. Der dritte Abschnitt, *Systemadministration*, behandelt die Administration eines FreeBSD-Systems. Der vierte Abschnitt, *Netzwerke*, bespricht Netzwerke und Netzwerkdienste. Der fünfte Abschnitt enthält Anhänge und Verweise auf weitere Informationen.

Kapitel 1, Einleitung
: Dieses Kapitel macht Einsteiger mit FreeBSD vertraut. Es behandelt die Geschichte, die Ziele und das Entwicklungsmodell des FreeBSD-Projekts.

Kapitel 3, FreeBSD 9.X (und neuer) installieren
: Beschreibt den Ablauf der Installation von FreeBSD 9.*x* und neuere mittels bsdinstall.

Kapitel 2, FreeBSD 8.X (und älter) installieren
: Beschreibt den Ablauf der Installation von FreeBSD 8.*x* und früher mittels sysinstall. Spezielle Installationsmethoden, wie die Installation mit einer seriellen Konsole, werden ebenfalls behandelt.

Kapitel 4, Grundlagen des UNIX Betriebssystems
: Erläutert die elementaren Kommandos und Funktionen von FreeBSD. Wenn Sie schon mit Linux® oder einem anderen UNIX® System vertraut sind, können Sie dieses Kapitel überspringen.

Kapitel 5, Installieren von Anwendungen: Pakete und Ports
: Zeigt wie mit der innovativen Ports-Sammlung oder mit Paketen Software von Fremdherstellern installiert wird.

Kapitel 6, Das X-Window-System
: Beschreibt allgemein das X Window System und geht speziell auf X11 unter FreeBSD ein. Weiterhin werden grafische Benutzeroberflächen wie KDE und GNOME behandelt.

Kapitel 7, Desktop-Anwendungen
: Enthält eine Aufstellung verbreiteter Anwendungen wie Browser, Büroanwendungen und Office-Pakete und beschreibt wie diese Anwendungen installiert werden.

Kapitel 8, Multimedia
: Erklärt, wie Sie auf Ihrem System Musik und Videos abspielen können. Beispielhaft werden auch Anwendungen aus dem Multimedia-Bereich beleuchtet.

Kapitel 9, Konfiguration des FreeBSD-Kernels
: Erklärt, warum Sie einen angepassten Kernel erzeugen sollten und gibt ausführliche Anweisungen wie Sie einen angepassten Kernel konfigurieren, bauen und installieren.

Kapitel 10, Drucken
: Beschreibt, wie Sie Drucker unter FreeBSD verwalten. Diskutiert werden Deckblätter, das Einrichten eines Druckers und ein Abrechnungssystem für ausgedruckte Seiten.

Kapitel 11, Linux-Binärkompatibilität
: Beschreibt die binäre Kompatibilität zu Linux®. Weiterhin werden ausführliche Installationsanleitungen für Oracle®, SAP® R/3® und Mathematica gegeben.

Kapitel 27, PPP
: Erklärt wie Sie mit PPP, SLIP oder PPP über Ethernet ein FreeBSD-System mit einem entfernten System verbinden.

Kapitel 28, Elektronische Post (E-Mail)
: Erläutert die verschiedenen Bestandteile eines E-Mail Servers und zeigt einfache Konfigurationen für sendmail, dem meist genutzten E-Mail-Server.

Kapitel 29, Netzwerkserver
: Bietet ausführliche Informationen und Beispielkonfigurationen, die es Ihnen ermöglichen, Ihren FreeBSD-Rechner als *Network File System Server*, *Domain Name Server*, *Network Information Server*, oder als Zeitsynchronisationsserver einzurichten.

Kapitel 30, Firewalls
: Erklärt die Philosophie hinter softwarebasierten Firewalls und bietet ausführliche Informationen zur Konfiguration der verschiedenen, für FreeBSD verfügbaren Firewalls.

Kapitel 31, Weiterführende Netzwerkthemen
: Behandelt viele Netzwerkthemen, beispielsweise das Verfügbarmachen einer Internetverbindung für andere Rechner eines LANs, Routing, drahtlose Netzwerke, Bluetooth®, IPv6, ATM und andere mehr.

Anhang A, Bezugsquellen für FreeBSD
: Enthält eine Aufstellung der Quellen von denen Sie FreeBSD beziehen können: CD-ROM, DVD sowie Internet-Sites.

Anhang B, Bibliografie
: Dieses Buch behandelt viele Themen und kann nicht alle Fragen erschöpfend beantworten. Die Bibliografie enthält weiterführende Bücher, die im Text zitiert werden.

Anhang C, Ressourcen im Internet
: Enthält eine Aufstellung der Foren, die FreeBSD Benutzern für Fragen und Diskussionen zur Verfügung stehen.

Anhang D, OpenPGP-Schlüssel
: Enthält PGP-Fingerabdrücke von etlichen FreeBSD Entwicklern.

Konventionen in diesem Buch

Damit der Text einheitlich erscheint und leicht zu lesen ist, werden im ganzen Buch die nachstehenden Konventionen beachtet:

Typographie

Kursiv
: Für Dateinamen, URLs, betonte Teile eines Satzes und das erste Vorkommen eines Fachbegriffs wird ein *kursiver* Zeichensatz benutzt.

`Fixschrift`
: Fehlermeldungen, Kommandos, Umgebungsvariablen, Namen von Ports, Hostnamen, Benutzernamen, Gruppennamen, Gerätenamen, Variablen und Code-Ausschnitte werden in einer `Fixschrift` dargestellt.

Fett
: Fett kennzeichnet Anwendungen, Kommandozeilen und Tastensymbole.

Benutzereingaben

Tasten werden fett dargestellt, um sie von dem umgebenden Text abzuheben. Tasten, die gleichzeitig gedrückt werden müssen, werden durch ein + zwischen den einzelnen Tasten dargestellt:

Ctrl+Alt+Del

Im gezeigten Beispiel soll der Benutzer die Tasten Ctrl, Alt und Del gleichzeitig drücken.

Tasten, die nacheinander gedrückt werden müssen, sind durch Kommas getrennt:

Ctrl+X, Ctrl+S

Das letzte Beispiel bedeutet, dass die Tasten Ctrl und X gleichzeitig betätigt werden und danach die Tasten Ctrl und S gleichzeitig gedrückt werden müssen.

Beispiele

Beispiele, die durch `C:\>` eingeleitet werden, zeigen ein MS-DOS® Kommando. Wenn nichts Anderes angezeigt wird, können diese Kommandos unter neuen Versionen von Microsoft® Windows® auch in einem DOS-Fenster ausgeführt werden.

```
E:\> tools\fdimage floppies\kern.flp A:
```

Beispiele, die mit **#** beginnen, müssen unter FreeBSD mit Superuser-Rechten ausgeführt werden. Dazu melden Sie sich entweder als `root` an oder Sie wechseln von Ihrem normalen Account mit su(1) zu dem Benutzer `root`.

```
# dd if=kern.flp of=/dev/fd0
```

Beispiele, die mit `%` anfangen, werden unter einem normalen Benutzer-Account ausgeführt. Sofern nichts Anderes angezeigt wird, verwenden die Beispiele die Syntax der C-Shell.

```
% top
```

Danksagung

Dieses Buch ist aus Beiträgen von vielen Leuten aus allen Teilen der Welt entstanden. Alle eingegangen Beiträge, zum Beispiel Korrekturen oder vollständige Kapitel, waren wertvoll.

Einige Firmen haben dieses Buch dadurch unterstützt, dass Sie Autoren in Vollzeit beschäftigt und die Veröffentlichung des Buchs finanziert haben. Besonders BSDi (das später von Wind River Systems übernommen wurde) beschäftigte Mitglieder des FreeBSD Documentation Projects, um dieses Buch zu erstellen. Dadurch wurde die erste (englische) gedruckte Auflage im März 2000 möglich (ISBN 1-57176-241-8). Wind River Systems bezahlte dann weitere Autoren, die die zum Drucken nötige Infrastruktur verbesserten und zusätzliche Kapitel beisteuerten. Das Ergebnis dieser Arbeit ist die zweite (englische) Auflage vom November 2001 (ISBN 1-57176-303-1). Zwischen 2003 und 2004 bezahlte FreeBSD Mall, Inc mehrere Mitarbeiter für die Vorbereitung der gedruckten dritten Auflage.

Teil I. Erste Schritte

Dieser Teil des FreeBSD-Handbuchs richtet sich an Benutzer und Administratoren für die FreeBSD neu ist. Diese Kapitel

- geben Ihnen eine Einführung in FreeBSD,
- geleiten Sie durch den Installationsprozess,
- erklären Ihnen die Grundlagen von UNIX® Systemen,
- zeigen Ihnen, wie Sie die Fülle der erhältlichen Anwendungen Dritter installieren und
- führen Sie in X, der Benutzeroberfläche von UNIX® Systemen ein. Es wird gezeigt, wie Sie den Desktop konfigurieren, um effektiver arbeiten zu können.

Wir haben uns bemüht, Referenzen auf weiter vorne liegende Textteile auf ein Minimum zu beschränken, so dass Sie diesen Teil des Handbuchs ohne viel Blättern durcharbeiten können.

Inhaltsverzeichnis

Kapitel 1. Einleitung

Restrukturiert, umorganisiert und Abschnitte neu geschrieben von Jim Mock.

1.1. Überblick

Herzlichen Dank für Ihr Interesse an FreeBSD! Das folgende Kapitel behandelt verschiedene Aspekte des FreeBSD Projekts wie dessen geschichtliche Entwicklung, seine Ziele oder das Entwicklungsmodell.

Nach dem Durcharbeiten des Kapitels wissen Sie über folgende Punkte Bescheid:

- Wo FreeBSD im Vergleich zu anderen Betriebssystemen steht
- Die Geschichte des FreeBSD Projekts
- Die Ziele des FreeBSD Projekts
- Die Grundlagen des FreeBSD-Open-Source-Entwicklungsmodells
- Und natürlich woher der Name „FreeBSD“ kommt.

1.2. Willkommen zu FreeBSD

FreeBSD ist ein auf 4.4BSD-Lite basierendes Betriebssystem für Intel (x86 und Itanium®), AMD64 und Sun UltraSPARC® Rechner. An Portierungen zu anderen Architekturen wird derzeit gearbeitet. Mehr zur Geschichte von FreeBSD erfahren Sie in die Geschichte von FreeBSD oder aus den aktuellen Release-Informationen. Falls Sie das FreeBSD Projekt unterstützen wollen (z.B. mit Quellcode, Hardware- oder Geldspenden), lesen Sie den FreeBSD unterstützen Artikel.

1.2.1. Was kann FreeBSD?

FreeBSD hat zahlreiche bemerkenswerte Eigenschaften. Um nur einige zu nennen:

- *Präemptives Multitasking* mit dynamischer Prioritätsanpassung zum reibungslosen und ausgeglichenen Teilen der Systemressourcen zwischen Anwendungen und Anwendern, selbst unter schwerster Last.
- *Mehrbenutzerbetrieb* erlaubt es, viele FreeBSD-Anwender gleichzeitig am System mit verschiedenen Aufgaben arbeiten zu lassen. Beispielsweise können Geräte wie Drucker oder Bandlaufwerke, die sich nur schwerlich unter allen Anwendern des Systems oder im Netzwerk teilen lassen, durch setzen von Beschränkungen auf Benutzer oder Gruppen wichtige Systemressourcen vor Überbeanspruchung geschützt werden.
- Starke*TCP/IP-Netzwerkfähigkeit* mit Unterstützung von Industriestandards wie SCTP, DHCP, NFS, NIS, PPP, SLIP, IPsec und IPv6. Das bedeutet, Ihr FreeBSD-System kann in einfachster Weise mit anderen Systemen interagieren. Zudem kann es als Server-System im Unternehmen wichtige Aufgaben übernehmen, beispielsweise als NFS- oder E-Mail-Server oder es kann Ihren Betrieb durch HTTP- und FTP-Server beziehungsweise durch Routing und Firewalling Internetfähig machen.
- *Speicherschutz* stellt sicher, dass Anwendungen (oder Anwender) sich nicht gegenseitig stören. Stürzt eine Anwendung ab, hat das keine Auswirkung auf andere Prozesse.
- Der Industriestandard *X-Window-System* (X11R7) als Industriestandard bietet eine grafische Benutzeroberfläche (GUI). Minimale Voraussetzung zur Verwendung ist lediglich eine Grafikkarte und ein Bildschirm, die beide den VGA-Modus unterstützen.
- *Binärkompatibilität* mit vielen auf anderen Betriebssystemen erstellten Programmen wie Linux, SCO, SVR4, BSDI und NetBSD.

- Tausende zusätzliche *leicht zu portierende* Anwendungen sind über die FreeBSD *Ports* und *Paket*-Sammlung verfügbar. Warum mühselig im Netz nach Software suchen, wenn diese bereits vorhanden ist?
- Tausende zusätzliche *leicht zu portierende* Anwendungen sind über das Internet zu beziehen. FreeBSD ist Quellcode-kompatibel mit den meisten kommerziellen UNIX® Systemen. Daher bedürfen Anwendungen häufig nur geringer oder gar keiner Anpassung, um auf einem FreeBSD-System zu kompilieren.
- Seitenweise anforderbarer *virtueller Speicher* und „merged VM/buffer cache“ -Entwurf bedient effektiv den großen Speicherhunger mancher Anwendungen bei gleichzeitigem Aufrechterhalten der Bedienbarkeit des Systems für weitere Benutzer.
- *SMP*-Unterstützung für Systeme mit mehreren CPUs.
- Ein voller Satz von *C* und *C++* Entwicklungswerkzeugen. Viele zusätzliche Programmiersprachen für höhere Wissenschaft und Entwicklung sind in der Ports- und Packages-Sammlung verfügbar.
- *Quellcode* für das gesamte System bedeutet größtmögliche Kontrolle über Ihre Umgebung. Warum sollte man sich durch proprietäre Lösungen knebeln und sich auf Gedeih und Verderb der Gnade eines Herstellers ausliefern, wenn man doch ein wahrhaft offenes System haben kann?
- Umfangreiche *Online-Dokumentation*.
- *und viele weitere!*

FreeBSD basiert auf dem 4.4BSD-Lite-Release der Computer Systems Research Group (CSRG) der Universität vn Kalifornien in Berkeley und führt die namhafte Tradition der Entwicklung von BSD-Systemen fort. Zusätzlich zu der herausragenden Arbeit der CSRG hat das FreeBSD Projekt tausende weitere Arbeitsstunden investiert, um das System zu verfeinern und maximale Leistung und Zuverlässigkeit bei Alltagslast zu bieten. Während viele kommerzielle Riesen Probleme damit haben, PC-Betriebssysteme mit derartigen Funktionen, Leistungspotential und Zuverlässigkeit anzubieten, kann FreeBSD damit schon jetzt aufwarten!

Die Anwendungsmöglichkeiten von FreeBSD werden nur durch Ihre Vorstellungskraft begrenzt. Von Software-Entwicklung bis zu Produktionsautomatisierung, von Lagerverwaltung über Abweichungskorrektur bei Satelliten; Falls etwas mit kommerziellen UNIX® Produkten machbar ist, dann ist es höchstwahrscheinlich auch mit FreeBSD möglich. FreeBSD profitiert stark von tausenden hochwertigen Anwendungen aus wissenschaftlichen Instituten und Universitäten in aller Welt. Häufig sind diese für wenig Geld oder sogar kostenlos zu bekommen. Kommerzielle Anwendungen sind ebenso verfügbar und es werden täglich mehr.

Durch den freien Zugang zum Quellcode von FreeBSD ist es in unvergleichbarer Weise möglich, das System für spezielle Anwendungen oder Projekte anzupassen. Dies ist mit den meisten kommerziellen Betriebssystemen einfach nicht möglich. Beispiele für Anwendungen, die unter FreeBSD laufen, sind:

- *Internet-Dienste:* Die robuste TCP/IP-Implementierung in FreeBSD macht es zu einer idealen Plattform für verschiedenste Internet-Dienste, wie zum Beispiel:
  - HTTP-Server (Standard oder mit SSL-Verschlüsselung)
  - IPv4- und IPv6-Routing
  - Firewall NAT („IP-Masquerading“)-Gateways
  - FTP-Server
  - E-Mail-Server
  - Und mehr...
- *Bildung:* Sind Sie Informatikstudent oder Student eines verwandten Studiengangs? Die praktischen Einblicke in FreeBSD sind die beste Möglichkeit etwas über Betriebssysteme, Rechnerarchitektur und Netzwerke zu lernen. Einige frei erhältliche CAD-, mathematische und grafische Anwendungen sind sehr nützlich, gerade für diejenigen, deren Hauptinteresse in einem Computer darin besteht, *andere* Arbeit zu erledigen!

- *Forschung:* Mit dem frei verfügbaren Quellcode für das gesamte System bildet FreeBSD ein exzellentes Studienobjekt in der Disziplin der Betriebssysteme, wie auch in anderen Zweigen der Informatik. Es ist beispielsweise denkbar, das räumlich getrennte Gruppen gemeinsam an einer Idee oder Entwicklung arbeiten. Das Konzept der freien Verfügbarkeit und -nutzung von FreeBSD ermöglicht so die freie Verwendung, ohne sich gross Gedanken über Lizenzbedingungen zu machen oder aufgrund von Beschränkungen evtl. in einem offenen Forum bestimmte Dinge nicht diskutieren zu dürfen.

- *Netzwerkfähigkeit:* Brauchen Sie einen neuen Router? Oder einen Name-Server (DNS)? Eine Firewall zum Schutze Ihres Intranets vor Fremdzugriff? FreeBSD macht aus dem in der Ecke verstaubenden 386- oder 486-PC im Handumdrehen einen leistungsfähigen Router mit anspruchsvollen Paketfilter-Funktionen.

- *Embedded:* FreeBSD ist eine exzellente Plattform, um auf embedded Systemen aufzubauen. Mit der Unterstützung für die ARM®-, MIPS®- und PowerPC®-Plattformen, verbunden mit dem robusten Netzwerkstack, aktuellen Neuerungen und der freizügigen BSD-Lizenz stellt FreeBSD eine ausgezeichnete Basis für embedded Router, Firewalls und andere Geräte dar.

- *Desktop:* FreeBSD ist eine gute Wahl für kostengünstige X-Terminals mit dem frei verfügbaren X11-Server. FreeBSD bietet die Auswahl aus vielen Open Source Desktop Umgebungen, dazu gehören auch die GNOME und KDE GUIs. FreeBSD kann sogar „plattenlos“ booten, was einzelne Workstations sogar noch günstiger macht und die Verwaltung erleichtert.

- *Software-Entwicklung:* Das Standard-FreeBSD-System wird mit einem kompletten Satz an Entwicklungswerkzeugen bereitgestellt, unter anderem einem vollständigen C/C++-Compiler und -Debugger. Entwicklungswerkzeugen. Viele zusätzliche Programmiersprachen für Wissenschaft und Entwicklung sind aus der Ports- und Packages-Sammlung zu haben.

FreeBSD ist sowohl in Form von Quellcode als auch in Binärform auf CD-ROM, DVD und über Anonymus FTP erhältlich. Lesen Sie dazu Anhang A, *Bezugsquellen für FreeBSD*, um weitere Informationen zum Bezug von FreeBSD zu erhalten.

1.2.2. Wer verwendet FreeBSD?

FreeBSDs fortgeschrittene Eigenschaften, bewährte Sicherheit und vorhersehbare Release-Zyklen, genauso wie seine tolerante Lizenz haben dazu geführt, dass es als Plattform zum Aufbau vieler kommerzieller und quelloffener Geräte und Produkte verwendet wird, dazu gehören ein paar der weltgrössten IT-Unternehmen:

- Apache - Die Apache Software Foundation lässt den Grossteil seiner der Öffentlichkeit zugänglichen Infrastruktur, inklusive des möglicherweise grössten SVN-Repositories der Welt mit über 1,4 Millionen Commits, auf FreeBSD laufen.

- Apple - OS X verwendet viel von FreeBSDs eigenem Netzwerkstack, virtuellem Dateisystem und den Benutzerumgebungskomponenten für sein eigenes System. Apple iOS nutzt ebenso Elemente, die es von FreeBSD übernommen hat

- Cisco - IronPort Network Sicherheits- und Anti-Spam-Appliance verwendet einen modifizierten FreeBSD-Kernel.

- Citrix - Die NetScaler Reihe von Sicherheits-Appliances bietet auf den Schichten 4-7 Load-Balancing, Content Caching, Anwendungsfirewall, gesichertes VPN und mobilen Cloud-Netzwerkzugriff, gepaart mit der Mächtigkeit der FreeBSD-Shell.

- Dell KACE - Die KACE Systemmanagement-Appliances nutzen FreeBSD wegen seiner Zuverlässigkeit, Skalierbarkeit und Gemeinschaft, welche deren zukünftige Weiterentwicklung fördert.

- Experts Exchange - Alle öffentlich zugänglichen Webserver werden von FreeBSD betrieben und machen starken Gebrauch von Jails, ohne den Überhang von Virtualisierung, um Entwicklungs- und Testumgebung voneinander zu isolieren.

- Isilon - Isilons Unternehmens-Speicherappliances basieren auf FreeBSD. Die extrem liberale FreeBSD-Lizenz erlaubt Isilon ihr intellektuelles Eigentum durch den gesamten Kernel zu integrieren und kann sich so auf das Erstellen ihres Produktes und nicht des Betriebssystems fokussieren.

- iXsystems - Die TrueNAS-Linie von vereinheitlichten Speicherappliances beruht auf FreeBSD. Zusätzlich zu deren kommerziellen Produkten, managed iXsystems auch noch die beiden Open Source Projekte PC-BSD und FreeNAS.

- Juniper - Das JunOS Betriebssystem, welches alle Juniper Netzwerkgeräte (inklusive Router, Switche, Sicherheits- und Netzwerkappliances) antreibt, verwendet FreeBSD Juniper ist einer der vielen Hersteller, welcher das symbolische Verhältnis zwischen dem Projekt und dem Hersteller von kommerziellen Produkten darstellt. Verbesserungen, die Juniper entwickelt hat, werden ebenso in FreeBSD aufgenommen, um die Komplexität der Integration neuer Eigenschaften von FreeBSD zurück in zukünftige JunOS Versionen zu vereinfachen.

- McAfee - SecurOS, die Basis von McAfee Enterprise-Firewallprodukten inkl. Sidewinder basiert auf FreeBSD.

- NetApp - Die Data ONTAP GX Reihe von Speicherappliances basieren auf FreeBSD. Zusätzlich hat NetApp viele Neuheiten beigesteuert, inklusive des neuen BSD-lizensierten Hypervisors bhyve.

- Netflix - Die OpenConnect-Appliance, die Netflix verwendet, um Filme zu seinen Kunden zu streamen basiert auf FreeBSD. Netflix hat weitreichende Beiträge zum Quellcode von FreeBSD beigetragen und arbeitet daran, ein möglichst geringes Delta zur normalen Version beizubehalten. Netflix OpenConnect-Appliances sind für mehr als 32% vom gesamten Internetverkehr in Nordamerika verantwortlich.

- Sandvine - Sandvine nutzt FreeBSD as die Basis für deren Echtzeit Hochgeschwindigkeits-Netzwerkplattform, welche den Kern deren intelligenter Netzwerkpolicy-Kontrollprodukte darstellt.

- Sony - Sowohl die PlayStation 3 und PlayStation 4 Spielekonsolen verwenden modifizierte Versionen von FreeBSD.

- Sophos - Das Sophos Email-Appliance Produkt basiert auf einem abgesicherten FreeBSD und scannt eingehende E-Mail auf Spam und Viren, während es gleichzeitig ausgehende Mail auf Schadsoftware und versehentlichen Versand von vertraulichen Informationen überwacht.

- Spectra Logic - Die nTier Reihe von archivspeicherfähigen Appliances nutzt FreeBSD und OpenZFS.

- The Weather Channel - Die IntelliStar Appliance, welche am Kopfende eines jeden Kabelversorgers installiert ist und für das Einspeisen von lokalen Wettervorhersagen in das Kabelfernsehprogramm verantwortlich ist, läuft auf FreeBSD.

- Verisign - Verisign ist für den Betrieb der .com und .net Root-Domainregistries genauso verantwortlich wie für die dazugehörige DNS-Infrastruktur. Sie verlassen sich auf einen Reihe von verschiedenen Netzwerkbetriebssystemen inklusive FreeBSD, um zu gewährleisten, dass es keine gemeinsame Fehlerstelle in deren Infrastruktur gibt.

- WhatsApp - Als WhatsApp eine Plattform benötigte, die in der Lage ist, mehr als 1 Million gleichzeitiger TCP-Verbindungen pro Server abzuarbeiten, entschied man sich für FreeBSD. Anschliessend fuhren Sie damit fort, auf 2,5 Millionen Verbindungen pro Server hochzuskalieren.

- Wheel Systems - Die FUDO Sicherheitsappliance erlaubt es Unternehmen, Vertragspartner und Administratoren, die anderen Systemen arbeiten durchführen, zu überwachen, zu kontrollieren, aufzuzeichnen und zu begutachten. Dies basiert auf all den besten Sicherheitseigenschaften von FreeBSD, inklusive ZFS, GELI, Capsicum, HAST und auditdistd.

FreeBSD hat ebenfalls eine Reihe von verwandten Open Source Projekten hervorgebracht:

- BSD Router - Einen FreeBSD-basierten Ersatz für grosse Unternehmensrouter, der entwickelt wurde, um auf Standard PC-Hardware zu laufen.

- FreeNAS - Ein eigens dafür entworfenes FreeBSD für den Zweck als Netzwerk-Dateiserver Appliance zu fungieren. Es enthält eine Python-basierte Webschnittstelle, um das Management von sowohl UFS- als auch ZFS-Systemen zu vereinfachen. Enthalten sind NFS, SMB/CIFS, AFP, FTP und iSCSI. Ebenfalls enthalten ist ein erweiterteres Plugin-System basierend auf FreeBSD-Jails.
- GhostBSD - Eine auf den Desktop-Einsatz orientierte Distribution von FreeBSD, welche mit einer Gnome-Desktop-Umgebung ausgeliefert wird.
- mfsBSD - Eine Sammlung von Werkzeugen zum Erstellen von FreeBSD-Systemimages, welches ausschliesslich im Hauptspeicher läuft.
- NAS4Free - Eine Dateiserverdistribution basierend auf FreeBSD mit einer von PHP-getriebenen Webschnittstelle.
- PC-BSD - Eine massgeschneiderte Version von FreeBSD, die sich an Desktop-Benutzern mit graphischen Oberflächenwerkzeugen orientiert, um die Mächtigkeit von FreeBSD allen Benutzern zur Verfügung zu stellen. Entwickelt wurde sie mit dem Ziel, den Übergang von Windows- und OS X-Benutzern zu erleichtern.
- pfSense - Eine Firewalldistribution basierend auf FreeBSD mit eine grossen Menge von Fähigkeiten und ausgedehnter IPv6-Unterstützung.
- m0n0wall - Eine abgespeckt Version von FreeBSD, die zusammen mit einem Webserver und PHP ausgeliefert wird. Entwickelt als eine eingebettete Firewall-Appliance mit einem Verbrauch von weniger als 12 MB.
- ZRouter - Eine Open Source Firmware-Alternative für eingebettete Geräte, die auf FreeBSD basiert. Entwickelt wurde sie, um die proprietäre Firmware von Standard-Routern zu ersetzen.

FreeBSD wird auch dazu eingesetzt, um einige der grössten Webseiten des Internets zu betreiben. Dazu gehören:

- Yahoo!
- Yandex
- Rambler
- Sina
- Pair Networks
- Sony Japan
- Netcraft
- Netflix
- NetEase
- Weathernews
- TELEHOUSE America

und viele weitere. Wikipedia pflegt eine Liste von Produkten, die auf FreeBSD basieren.

1.3. Über das FreeBSD Projekt

Der folgende Abschnitt bietet einige Hintergrundinformationen zum FreeBSD Projekt, einschließlich einem kurzen geschichtlichen Abriss, den Projektzielen und dem Entwicklungsmodell.

1.3.1. Kurzer geschichtlicher Abriss zu FreeBSD

Das FreeBSD Projekt erblickte das Licht der Welt Anfang 1993 teils als Auswuchs des „Unofficial 386BSD Patchkit“ unter der Regie der letzten drei Koordinatoren des Patchkits: Nate Williams, Rod Grimes und Jordan Hubbard.

Das ursprüngliche Ziel war es, einen zwischenzeitlichen Abzug von 386BSD zu erstellen, um ein paar Probleme zu beseitigen, die das Patchkit-Verfahren nicht lösen konnte. Der frühe Arbeitstitel für das Projekt war „386BSD 0.5“ oder „386BSD Interim“ als Referenz darauf.

386BSD war das Betriebssystem von Bill Jolitz, welches bis zu diesem Zeitpunkt heftig unter fast einjähriger Vernachlässigung litt. Als das Patchkit mit jedem Tag anschwoll und unhandlicher wurde, entschied man sich, Bill Jolitz zu helfen, indem ein übergangsweise „bereinigter“ Abzug zur Verfügung gestellt wurde. Diese Pläne wurden durchkreuzt, als Bill Jolitz plötzlich seine Zustimmung zu diesem Projekt zurückzog, ohne einen Hinweis darauf, was stattdessen geschehen sollte.

Das Trio entschied, dass das Ziel sich weiterhin lohnen würde, selbst ohne die Unterstützung von Bill und so wurde entschieden, den Namen FreeBSD zu verwenden, der von David Greenman geprägt wurde. Die anfänglichen Ziele wurden festgelegt, nachdem man sich mit den momentanen Benutzern des Systems besprach und abzusehen war, dass das Projekt die Chance hatte, Realität zu werden, kontaktierte Jordan Walnut Creek CDROM mit dem Vorhaben, FreeBSDs Verteilung auch auf diejenigen auszuweiten, die noch keinen Internetzugang besaßen. Walnut Creek CDROM unterstützte nicht nur die Idee durch die Verbreitung von FreeBSD auf CD, sondern ging auch so weit dass es dem Projekt eine Maschine mit schneller Internetverbindung zur Verfügung stellte, um damit zu arbeiten. Ohne den von Walnut Creek bisher nie dagewesenen Grad von Vertrauen in ein, zur damaligen Zeit, komplett unbekanntes Projekt, wäre es unwahrscheinlich, dass FreeBSD so weit gekommen wäre, wie es heute ist.

Die erste auf CD-ROM (und netzweit) verfügbare Veröffentlichung war FreeBSD 1.0 im Dezember 1993. Diese basierte auf dem Band der 4.3BSD-Lite („Net/2“) der Universität von Kalifornien in Berkeley. Viele Teile stammten aus 386BSD und von der Free Software Foundation. Gemessen am ersten Angebot, war das ein ziemlicher Erfolg und Sie ließen dem das extrem erfolgreiche FreeBSD 1.1 im Mai 1994 folgen.

Zu dieser Zeit formierten sich unerwartete Gewitterwolken am Horizont, als Novell und die Universität von Kalifornien in Berkeley (UCB) ihren langen Rechtsstreit über den rechtlichen Status des Berkeley Net/2-Bandes mit einem Vergleich beilegten. Eine Bedingung dieser Einigung war es, dass die UCB große Teile des Net/2-Quellcodes als „belastet“ zugestehen musste, und dass diese Besitz von Novell sind, welches den Code selbst einige Zeit vorher von AT&T bezogen hatte. Im Gegenzug bekam die UCB den „Segen“ von Novell, dass sich das 4.4BSD-Lite-Release bei seiner endgültigen Veröffentlichung als unbelastet bezeichnen darf. Alle Net/2-Benutzer sollten auf das neue Release wechseln. Das betraf auch FreeBSD. Dem Projekt wurde eine Frist bis Ende Juli 1994 eingeräumt, das auf Net/2-basierende Produkt nicht mehr zu vertreiben. Unter den Bedingungen dieser Übereinkunft war es dem Projekt noch erlaubt ein letztes Release vor diesem festgesetzten Zeitpunkt herauszugeben. Das war FreeBSD 1.1.5.1.

FreeBSD machte sich dann an die beschwerliche Aufgabe, sich Stück für Stück aus einem neuen und ziemlich unvollständigen Satz von 4.4BSD-Lite-Teilen, wieder aufzubauen. Die „Lite“ -Veröffentlichungen waren deswegen leicht, weil Berkeleys CSRG große Code-Teile, die für ein start- und lauffähiges System gebraucht wurden, aufgrund diverser rechtlicher Anforderungen entfernen musste und weil die 4.4-Portierung für Intel-Rechner extrem unvollständig war. Das Projekt hat bis November 1994 gebraucht diesen Übergang zu vollziehen, was dann zu dem im Netz veröffentlichten FreeBSD 2.0 und zur CD-ROM-Version (im späten Dezember) führte. Obwohl FreeBSD gerade die ersten Hürden genommen hatte, war dieses Release ein maßgeblicher Erfolg. Diesem folgte im Juni 1995 das robustere und einfacher zu installierende FreeBSD 2.0.5.

Seit dieser Zeit hat FreeBSD eine Reihe von Releases veröffentlicht, die jedes mal die Stabilität, Geschwindigkeit und Menge an verfügbaren Eigenschaften der vorherigen Version verbessert.

Momentan werden langfristige Entwicklungsprojekte im 10.X-CURRENT (Trunk)-Zweig durchgeführt, und Abzüge (Snapshots) der Releases von 10.X werden regelmässig auf den Snapshot-Servern zur Verfügung gestellt.

1.3.2. Ziele des FreeBSD-Projekts

Beigetragen von Jordan Hubbard.

Das FreeBSD Projekt stellt Software her, die ohne Einschränkungen für beliebige Zwecke eingesetzt werden kann. Viele von uns haben beträchtlich in Quellcode und das Projekt investiert und hätten sicher nichts dagegen, hin und wieder ein wenig finanziellen Ausgleich dafür zu bekommen. Aber in keinem Fall bestehen wir darauf. Wir glau-

ben unsere erste und wichtigste „Mission“ ist es, Software für jeden Interessierten und zu jedem Zweck zur Verfügung zu stellen, damit die Software größtmögliche Verbreitung erlangt und größtmöglichen Nutzen stiftet. Das ist, glaube ich, eines der grundlegenden Ziele freier Software, welche wir mit größter Begeisterung unterstützen.

Der Code in unserem Quellbaum, der unter die General Public License (GPL) oder die Library General Public License (LGPL) fällt, stellt geringfügig mehr Bedingungen. Das aber vielmehr im Sinne von eingefordertem Zugriff, als das übliche Gegenteil der Beschränkungen. Aufgrund zusätzlicher Abhängigkeiten, die sich durch die Verwendung von GPL-Software bei kommerziellem Gebrauch ergeben, bevorzugen wir daher Software unter dem transparenteren BSD-Copyright, wo immer es angebracht ist.

1.3.3. Das FreeBSD-Entwicklungsmodell

Beigetragen von Satoshi Asami.

Die Entwicklung von FreeBSD ist ein offener und flexibler Prozess, der durch den Beitrag von buchstäblich tausenden Leuten rund um die Welt ermöglicht wird, wie an der Liste der Beitragenden ersehen können. Die vielen Entwickler können aufgrund der Entwicklungs-Infrastruktur von FreeBSD über das Internet zusammenarbeiten. Wir suchen ständig nach neuen Entwicklern, Ideen und jenen, die sich in das Projekt tiefer einbringen wollen. Nehmen Sie einfach auf der Mailingliste FreeBSD technical discussions Kontakt mit uns auf. Die Mailingliste FreeBSD announcements steht für wichtige Ankündigungen, die alle FreeBSD-Benutzer betreffen, zur Verfügung.

Unabhängig davon ob Sie alleine oder mit anderen eng zusammen arbeiten, enthält die folgende Aufstellung nützliche Informationen über das FreeBSD Projekt und dessen Entwicklungsabläufe.

Die SVN-Repositories
: Der Hauptquellbaum von FreeBSD wurde über viele Jahre ausschließlich mit CVS (Concurrent-Versions-System) gepflegt, einem frei erhältlichen Versionskontrollsystem, welches mit FreeBSD geliefert wird. Im Juni 2008 begann das FreeBSD Project mit dem Umstieg auf SVN (Subversion). Dieser Schritt wurde notwendig, weil durch technische Einschränkungen von CVS aufgrund des rapide wachsenden Quellcodebaumes und dem Umfang der bereits gespeichterten Revisisionsinformationen an dessen Grenzen zu stoßen begann. Die Repositories des Dokumentationsprojekts und die Ports-Sammlung wurden ebenfalls von CVS zu SVN im Mai und Juli 2012 umgezogen. Lesen Sie dazu Synchronisation der Quellen für weitere Informationen zum Beziehen der FreeBSD `src/` Repository und Die Ports-Sammlung verwenden für Details zum Beziehen der FreeBSD Ports-Sammlung.

Die Committer-Liste
: Die *Committer* sind diejenigen Leute, welche *schreibenden* Zugriff auf den Subversion-Baum besitzen und berechtigt sind, Änderungen an den FreeBSD-Quellen (der Begriff „Committer“ stammt aus dem Versionskontrollbefehl `commit` , der dazu verwendet wird, Änderungen in das Repository zu bringen). Die beste Möglichkeit, Beiträge zur Bewertung der Committer-Liste einzusenden, ist das send-pr(1)-Kommando. Falls etwas an diesem System nicht zu funktionieren scheint, dann können Sie diese ebenso durch eine E-Mail an „FreeBSD committers“ erreichen.

The FreeBSD core team
: Die *FreeBSD core team* ist mit dem Vorstand vergleichbar, wenn das FreeBSD Projekt ein Unternehmen wäre. Die Hauptaufgabe des Core Teams ist es sicherzustellen, dass sich das Projekt als Ganzes in einem guten Zustand befindet und sich in die richtige Richtung bewegt. Das Einladen von engagierten und verantwortungsvollen Entwicklern zu dem Zweck, sich der Gruppe von Committern anzuschliessen, ist eine der Funktionen des Core Teams, genauso wie neue Mitglieder des Core Teams zu rekrutieren, wenn andere ausscheiden. Das aktuelle Core Team wurde aus einer Menge von Kandidaten aus dem Kreis der Committer im Juli 2012 gewählt. Wahlen werden alle zwei Jahre abgehalten.

Anmerkung

Wie die meisten Entwickler auch, sind die Mitglieder des Core Teams Freiwillige, wenn es um die Entwicklung von FreeBSD geht und erhalten keinerlei finanziellen Vorteil aus

> dem Projekt, deshalb sollte „Verpflichtung“ nicht fehlverstanden werden mit „garantierter Unterstützung“. Die „Vorstands“-Analogie oben ist nicht sehr akkurat und kann vielleicht besser damit umschrieben werden, dass diese Leute ihr Leben für FreeBSD gegen jedwede Vernunft geopfert haben.

Aussenstehende Beitragende
: Schliesslich stellt die grösste, aber nichtsdestotrotz wichtigste Gruppe von Entwicklern die der Benutzer selbst dar, die stetig Rückmeldungen und Fehlerbehebungen liefert. Der hauptsächliche Weg mit FreeBSDs nichtzentralisierter Entwicklung Kontakt zu halten, ist, die FreeBSD technical discussions Mailingliste zu abonnieren, auf der solche Dinge diskutiert werden. Lesen Sie dazu Anhang C, *Ressourcen im Internet* für weitere Informationen über die verschiedenen FreeBSD-Mailinglisten.

  Liste der Beitragenden ist eine, die lang ist und stetig wächst, also warum nicht FreeBSD beitreten und noch heute etwas zurückgeben?

  Code ist nicht die einzige Art, zu dem Projekt etwas beizutragen. Für eine ausführlichere Liste von Dingen die getan werden müssen, lesen Sie auf der FreeBSD Projektwebseite.

Zusammenfassend ist unser Entwicklungsmodell als eine lose Menge von konzentrischen Kreisen organisiert. Das zentralisierte Modell ist mit der Praktikabilität der *Anwender* von FreeBSD entworfen worden, die mit der einfachen Art einhergeht, eine zentrale Basis für den Code zu haben und keine potentiellen Beiträge auszuschliessen! Unser Ansporn ist es, ein stabiles Betriebssystem mit einer grossen Menge von kohärenten Anwendungsprogrammen, welches die Benutzer einfach installieren und verwenden können - dieses Modell funktioniert darin sehr gut, dieses Ziel zu erreichen.

Alles was wir von denen verlangen, die uns als FreeBSD-Entwickler beitreten ist, etwas von der gleichen Hingabe an den Erfolg, die seine momentanen Gemeinschaft inne hat, zu besitzen.

1.3.4. Programme von Drittherstellern

Zusätzlich zur Basisdistribution bietet FreeBSD eine Sammlung von portierter Software mit tausenden der am meisten nachgefragten Programme an. Als diese Zeilen geschrieben wurden, gab es über 24,000 Ports! Die Liste der Ports reicht von HTTP-Servern, zu Spielen, Sprachen, Editoren und so ziemlich alles, was dazwischen liegt. Die gesamte Port-Sammlung ist geschätzt 500 MB gross. Um einen Port zu übersetzen, wechseln Sie einfach in das Verzeichnis des Programms, das sie installieren möchten und geben `make install` ein und das System erledigt den Rest. Die gesamte Originaldistribution für jeden Port, den Sie bauen wird dynamisch heruntergeladen, so dass sie nur genügend Plattenplatz zum bauen des Ports, den sie haben möchten, zur Verfügung stellen müssen. Fast jeder Port ist auch als vorkompiliertes„Paket“, das über das folgende einfache Kommando (`pkg_add`) für diejenigen, die keine kompilierten Port aus den Quellen wünschen. Weitere Informationen zu Ports und Paketen finden Sie in Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*.

1.3.5. Zusätzliche Dokumentation

Alle momentanen FreeBSD Versionen bieten eine Option im Installer (entweder sysinstall(8) oder bsdinstall(8)), um zusätzliche Dokumentation unter `/usr/local/share/doc/freebsd` während des initialen Systemsetups zu installieren. Dokumentation kann auch zu einem späteren Zeitpunkt über Pakete installiert werden, wie es Abschnitt 24.4.6.2, „Verwendung von Dokumentations-Paketen“ beschreibt. Sie können ebenso die lokal installierten Anleitungen mit jedem HTML-fähigen Browser lesen, indem Sie die folgende URL verwenden:

Das FreeBSD Handbuch
: `/usr/local/share/doc/freebsd/handbook/index.html`

Die FreeBSD FAQ
: `/usr/local/share/doc/freebsd/faq/index.html`

Genauso erhalten Sie auch die Master (und am häufigsten aktualisierten) Kopien von `http://www.FreeBSD.org/` .

Kapitel 2. FreeBSD 8.X (und älter) installieren

Überarbeitet und teilweise neu geschrieben von Jim Mock.
Der Gang durch sysinstall und alle Bildschirmabzüge von Randy Pratt.
Übersetzt von Martin Heinen und Johann Kois.

2.1. Übersicht

FreeBSD wird mit dem textorientierten und einfach zu benutzendem Installationsprogramm installiert. Beginnend mit FreeBSD 9.0-RELEASE handelt es sich dabei um das Programm bsdinstall. Ältere FreeBSD-Versionen verwenden hingegen nach wie vor sysinstall für die Installation. Dieses Kapitel beschreibt die Installation von FreeBSD über sysinstall. Der Einsatz von bsdinstall wird hingegen in Kapitel 3, *FreeBSD 9.X (und neuer) installieren* besprochen.

Dieses Kapitel behandelt folgende Punkte:

- Das Erzeugen von FreeBSD-Startdisketten.
- Wie FreeBSD Platten anspricht und aufteilt.
- Wie sysinstall ausgeführt wird.
- Die Menüs von sysinstall und die erforderlichen Eingaben in den Menüs.

Bevor Sie dieses Kapitel lesen, sollten Sie

- Die Hardware-Notes der FreeBSD-Release, die Sie installieren wollen, lesen und sicherstellen, dass Ihre Hardware unterstützt wird.

Anmerkung

Diese Installationsanleitung gilt für Rechner mit i386™-Architektur (PC-kompatible Rechner). Abweichende Anweisungen für andere Plattformen werden, falls notwendig, gegeben. Obwohl diese Anleitung so aktuell wie möglich ist, kann das Installationsverfahren von dem hier gezeigten geringfügig abweichen. Legen Sie bitte daher diese Anleitung nicht wortwörtlich aus, sondern lassen Sie sich von diesem Kapitel durch den Installationsprozess leiten.

2.2. Hardware-Anforderungen

2.2.1. Minimalkonfiguration

Die zur Installation von FreeBSD erforderliche Minimalkonfiguration hängt von der zu installierenden FreeBSD-Version sowie von der Hardware-Architektur ab.

Informationen zur jeweiligen Minimalkonfiguration finden Sie in den folgenden Abschnitten dieses Kapitels. Je nachdem, wie Sie FreeBSD installieren, benötigen Sie eventuell auch ein Diskettenlaufwerk, ein unterstütztes CD-ROM-Laufwerk, oder auch eine Netzwerkkarte. Abschnitt 2.3.7, „Das Startmedium vorbereiten“ des Handbuchs enthält weitere Informationen zu den verschiedenen Installationsarten.

2.2.1.1. Die FreeBSD/i386- und FreeBSD/pc98-Architekturen

Sowohl FreeBSD/i386 als auch FreeBSD/pc98 benötigen jeweils mindestens einen 486-Prozessor sowie mindestens 24 MB RAM. Außerdem benötigen Sie für eine Minimalinstallation mindestens 150 MB freien Platz auf Ihrer Festplatte.

Anmerkung

In den meisten derartigen Konfigurationen ist es besser, für mehr RAM und mehr Plattenplatz zu sorgen, statt einen schnelleren Prozessor einzubauen.

2.2.1.2. Die FreeBSD/amd64-Architektur

Es gibt zwei Klassen von Prozessoren, auf denen Sie FreeBSD/amd64 ausführen können. Die erste Klasse bilden die AMD64-Prozessoren (zu denen AMD Athlon 64-, AMD Athlon 64-FX-, oder AMD Opteron-Prozessoren gehören).

Die zweite Klasse von Prozessoren, auf denen Sie diese FreeBSD/amd64 einsetzen können, ist die Intel® EM64T-Architektur. Prozessoren dieser Klasse sind beispielsweise Intel® Core™ 2 Duo-, Quad-, und Extreme-Prozessoren sowie die Intel® Xeon™-Prozessorreihen 3000, 5000, und 7000.

Wenn Sie einen auf dem Chipsatz nVidia nForce3 Pro-150 basierenden Rechner haben, *müssen* Sie im BIOS das IO-APIC deaktivieren. Erlaubt ihr BIOS dies nicht, müssen Sie stattdessen ACPI deaktivieren. Der Grund dafür sind Fehler im Pro-150-Chipsatz, die bis jetzt nicht behoben werden konnten.

2.2.1.3. Die FreeBSD/sparc64-Architektur

Um FreeBSD/sparc64 zu installieren, benötigen Sie eine unterstützte Plattform (lesen Sie dazu auch Abschnitt 2.2.2, „Unterstützte Hardware“ des Handbuchs).

Sie benötigen außerdem eine seperate Festplatte, wenn Sie FreeBSD/sparc64 installieren wollen, da es derzeit leider noch nicht möglich ist, die Platte mit einem weiteren Betriebssystem zu teilen.

2.2.2. Unterstützte Hardware

Die Hardware-Notes, die mit jedem FreeBSD-Release ausgeliefert werden, enthalten eine Liste lauffähiger Hardware. Die Hardware-Notes befinden sich üblicherweise in der Datei `HARDWARE.TXT` im Wurzelverzeichnis der Distribution (CD-ROM oder FTP). Sie können die Hardware-Notes außerdem im Dokumentationsmenü von sysinstall oder auf der Webseite Release Information lesen.

2.3. Vor der Installation

2.3.1. Erstellen Sie eine Geräteliste

Bevor Sie FreeBSD installieren, erfassen Sie die Komponenten Ihres Rechners. Die FreeBSD-Installation wird die Komponenten (Festplatten, Netzwerkkarten, CD-ROM-Laufwerke) zusammen mit der Modellbezeichnung und des Herstellers anzeigen. FreeBSD wird auch versuchen, die richtige Konfiguration der Geräte zu ermitteln. Dazu gehören die benutzten Interrupts (IRQ) und IO-Ports. Wegen der Unwägbarkeiten von PC-Hardware kann die Konfiguration der Geräte allerdings fehlschlagen. In diesem Fall müssen Sie die von FreeBSD ermittelte Konfiguration korrigieren.

Wenn Sie schon ein anderes Betriebssystem, wie Windows® oder Linux installiert haben, können Sie die Hardware-Konfiguration mit den Mitteln dieses Betriebssystems bestimmen. Wenn Sie nicht sicher sind, welche Einstellungen eine Erweiterungskarte besitzt, sehen Sie auf der Karte selbst nach. Manchmal sind die Einstellungen

dort aufgedruckt. Gebräuchliche IRQs sind `3`, `5` und `7`. Die Adressen von IO-Ports werden normalerweise hexadezimal, zum Beispiel `0x330`, angegeben.

Halten Sie die Gerätekonfiguration vor der Installation in einer Tabelle wie der nachstehenden fest:

Tabelle 2.1. Gerätekonfiguration

Gerät	IRQ	IO-Ports	Anmerkung
erste Festplatte	-	-	40 GB, Seagate, erster IDE-Master
CD-ROM	-	-	erster IDE-Slave
zweite Festplatte	-	-	20 GB, IBM, zweiter IDE-Master
erster IDE-Controller	14	0x1f0	
Netzwerkkarte	-	-	Intel® 10/100
Modem	-	-	3Com® 56K Faxmodem, an COM1
...			

Nachdem Sie wissen, über welche Hardware Ihr Rechner verfügt, müssen Sie diese Informationen mit den Hardwareanforderungen der zu installierenden FreeBSD-Version abgleichen.

2.3.2. Sichern Sie Ihre Daten

Wenn der Rechner, auf dem Sie FreeBSD installieren wollen, wichtige Daten enthält, sichern Sie bitte diese Daten. Prüfen Sie auch, dass Sie die Daten aus der Sicherung wiederherstellen können, bevor Sie FreeBSD installieren. Die FreeBSD-Installation fragt zwar nach, bevor Sie Daten auf Ihre Festplatte schreibt, Ihre Daten sind allerdings unwiderruflich verloren, wenn der Installationsvorgang einmal angelaufen ist.

2.3.3. Den Installationsort von FreeBSD festlegen

Wenn Sie die gesamte Festplatte für FreeBSD verwenden wollen, müssen Sie sich an dieser Stelle keine weiteren Gedanken machen – lesen Sie bitte im nächsten Abschnitt weiter.

Wenn Sie allerdings FreeBSD neben anderen Betriebssystemen betreiben wollen, müssen Sie wissen, wie Daten auf einer Festplatte abgelegt werden und welche Auswirkungen dies hat.

2.3.3.1. Platteneinteilung von FreeBSD/i386-Systemen

Eine PC-Festplatte wird in einzelne Bereiche unterteilt, die *Partitionen* heißen. FreeBSD verwendet intern ebenfalls Partitionen. Um Verwechslungen und Unklarheiten zu vermeiden, werden diese Plattenbereiche unter FreeBSD als Slices bezeichnet. So verwendet beispielsweise das Werkzeug `fdisk` den Begriff Slices, um sich auf PC-Partitionen zu beziehen. Auf einer PC-Festplatte können maximal vier Partitionen, die *primäre Partitionen* genannt werden, angelegt werden. Eine *erweiterte Partition* hebt diese Beschränkung auf. Eine Festplatte kann nur eine erweiterte Partition enthalten, die wiederum weitere so genannte *logische Partitionen* enthalten kann.

Jede Partition besitzt eine *Partitions-ID* – eine Zahl, die den Typ der Partition festlegt. FreeBSD-Partitionen tragen die Partitions-ID `165`.

Üblicherweise kennzeichnen Betriebssysteme Partitionen in einer besonderen Art und Weise. Beispielsweise werden jeder primären und logischen Partition unter MS-DOS® und dem verwandten Windows® Laufwerksbuchstaben beginnend mit `C:` zugewiesen.

FreeBSD muss auf einer primären Partition installiert werden. In dieser Partition hält FreeBSD alle Daten einschließlich der Dateien, die Sie anlegen. Verfügt das System über mehrere Festplatten, können Sie auf allen oder einigen Platten eine FreeBSD-Partition einrichten. Zur Installation von FreeBSD benötigen Sie eine freie Partition: Dies kann eine extra für die Installation eingerichtete Partition sein oder eine existierende Partition, die nicht mehr benötigte Daten enthält.

Wenn auf allen Platten bereits sämtliche Partitionen benutzt werden, müssen Sie eine der Partitionen für FreeBSD frei machen. Benutzen Sie dazu die Werkzeuge des eingesetzten Betriebssystems (`fdisk` unter MS-DOS® oder Windows®).

Verfügt das System über eine freie Partition, benutzen Sie diese Partition. Es kann allerdings sein, dass Sie eine oder mehrere der vorhandenen Partitionen vorher verkleinern müssen.

Eine minimale FreeBSD-Installation benötigt nur 100 MB Plattenplatz. Diese Installation ist allerdings *sehr* begrenzt und lässt wenig Platz für Ihre eigenen Dateien. Realistischer sind 250 MB für FreeBSD ohne graphische Benutzeroberfläche und 350 MB für FreeBSD mit einer graphischen Benutzeroberfläche. Sie benötigen weiteren Platz für die Installation zusätzlicher Software.

Um die Partitionen zu verkleinern, können Sie beispielsweise das kommerzielle PartitionMagic® oder das freie GParted benutzen. Sowohl GParted als auch PartitionMagic® können auch NTFS-Partitionen verändern. GParted ist auf vielen Linux-Live-CDs, beispielsweise der SystemRescueCD, verfügbar.

Bei der Veränderung von Microsoft® Vista-Partionen kommt es manchmal zu Problemen. In einem solchen Fall ist es von Vorteil, wenn Sie eine Vista-Installations-CD zur Verfügung haben. Wie bei jeder Änderung an Ihrer Festplatte sollten Sie auch hier zuerst ein aktuelles Backup anlegen.

Warnung

Der falsche Gebrauch dieser Werkzeuge kann Daten auf der Festplatte löschen. Vor dem Einsatz dieser Werkzeuge stellen Sie bitte sicher, dass Sie frische, funktionierende Datensicherungen besitzen.

Beispiel 2.1. Eine bestehende Partition verwenden

Nehmen wir an, Sie haben einen Rechner mit einer 4 GB Festplatte, auf der schon eine Version von Windows® installiert ist. Weiterhin haben Sie die Platte in zwei Laufwerke `C:` und `D:` unterteilt, die jeweils 2 GB groß sind. Auf `C:` wird 1 GB benutzt und 0,5 GB von Laufwerk `D:` werden benutzt.

Sie haben also eine Festplatte mit zwei Partitionen und könnten alle Daten von Laufwerk `D:` auf das Laufwerk `C:` kopieren. Damit wäre die zweite Partition für FreeBSD frei.

Beispiel 2.2. Eine bestehende Partition verkleinern

Nehmen wir an, Sie haben einen Rechner mit einer 4 GB Festplatte auf der schon eine Version von Windows® installiert ist. Während der Installation von Windows® haben sie eine große Partition `C:` angelegt, die 4 GB groß ist. Von den 4 GB werden 1,5 GB benutzt und Sie wollen 2 GB für FreeBSD verwenden.

Sie haben zwei Möglichkeiten, FreeBSD zu installieren:

1. Sichern Sie die Daten der Windows®-Partition und installieren Sie Windows® erneut auf einer 2 GB großen Partition.

2. Verkleinern Sie die Windows®-Partition mit einem der oben aufgeführten Werkzeuge.

2.3.4. Netzwerkparameter ermitteln

Wird während der Installation ein Netzwerk benötigt (weil Sie über FTP oder von einem NFS-Server installieren wollen), müssen Sie die Konfiguration des Netzwerks kennen. Während der Installation werden Netzwerkparameter abgefragt, damit sich FreeBSD mit dem Netzwerk verbinden und die Installation abschließen kann.

2.3.4.1. Verbindung über Ethernet oder ein Kabel/DSL-Modem

Wenn Sie sich mit einem Ethernet verbinden oder eine Internet-Verbindung mit einem Ethernet-Adapter über Kabel oder DSL herstellen, benötigen Sie die nachstehenden Daten:

1. IP-Adresse
2. IP-Adresse des Default-Gateways
3. Hostname
4. IP-Adressen der DNS-Server
5. Subnetzmaske

Wenn Sie die Daten nicht besitzen, fragen Sie bitte Ihren Systemadministrator oder Ihren Service-Provider. Können die Daten über *DHCP* bezogen werden, merken Sie sich diese Tatsache.

2.3.4.2. Verbindung über ein Modem

Auch wenn Sie sich mit einem normalen Modem bei einem ISP einwählen, können Sie FreeBSD aus dem Internet installieren. Die Installation über ein Modem dauert nur sehr lange.

Sie benötigen die nachstehenden Daten:

1. Die Telefonnummer des ISPs.
2. Die COM-Schnittstelle, an der das Modem angeschlossen ist.
3. Den Benutzernamen und das Passwort für Ihr Konto.

2.3.5. Lesen Sie die FreeBSD-Errata

Auch wenn das FreeBSD-Project bemüht ist, ein Release so stabil wie möglich herzustellen, treten ab und an Fehler auf. In seltenen Fällen betrifft ein Fehler die Installations-Prozedur. Die Fehler und deren Behebungen werden in den FreeBSD-Errata festgehalten. Lesen Sie bitte die Errata, bevor Sie FreeBSD installieren, damit Sie nicht in frisch entdeckte Probleme laufen.

Dokumentation zu jedem Release, inklusive der Errata zu jedem Release, finden Sie im Release-Bereich des FreeBSD Webauftritts.

2.3.6. Die Installationsdateien beschaffen

FreeBSD kann von Dateien aus irgendeiner der nachstehenden Quellen installiert werden:

- von einer CD-ROM oder einer DVD
- von einem USB-Stick
- von einer MS-DOS®-Partition auf demselben Rechner
- von einem SCSI- oder QIC-Bandlaufwerk

- von Disketten
- von einem FTP-Server, wenn erforderlich auch durch eine Firewall oder durch einen HTTP-Proxy
- von einem NFS-Server
- über eine feste serielle oder eine feste parallele Verbindung

Wenn Sie eine FreeBSD-CD oder FreeBSD-DVD gekauft haben, besitzen Sie schon alles, was Sie zur Installation benötigen. Lesen Sie bitte im nächsten Abschnitt (Abschnitt 2.3.7, „Das Startmedium vorbereiten“) weiter.

Wenn Sie sich die FreeBSD-Installationsdateien noch nicht besorgt haben, lesen Sie bitte zuerst den Abschnitt 2.13, „Eigene Installationsmedien herstellen“. Dort werden die notwendigen Vorbereitungen für eine Installation von den eben genannten Medien beschrieben. Wenn Sie den Abschnitt durchgearbeitet haben, lesen Sie bitte in Abschnitt 2.3.7, „Das Startmedium vorbereiten“ weiter.

2.3.7. Das Startmedium vorbereiten

Um FreeBSD zu installieren, müssen Sie Ihren Rechner mit einem speziellen Startmedium hochfahren, das die Installationsroutine startet. Sie können das Installationsprogramm nicht unter einem anderen Betriebssystem ausführen. Ein Rechner startet normalerweise das auf der Festplatte installierte Betriebssystem, er kann aber auch von Disketten gestartet werden. Aktuelle Rechner können in der Regel auch von einer CD-ROM oder von einem USB-Stick starten.

Tipp

Wenn Sie eine FreeBSD CD-ROM oder DVD besitzen (gekauft oder selbst erstellt) und Ihr Rechner von CD-ROM oder DVD starten kann (üblicherweise können Sie das mit der BIOS-Option `Boot Order` einstellen), können Sie diesen Abschnitt überspringen. Eine FreeBSD CD-ROM oder DVD lässt sich direkt starten; Sie können damit FreeBSD ohne weitere Vorbereitungen installieren.

Um einen bootbaren USB-Stick zu erstellen, gehen Sie wie folgt vor:

1. Das Speicherabbild für den USB-Stick herunterladen

   Das Speicherabbild finden Sie auf dem FreeBSD-FTP-Server `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/arch/ISO-IMAGES/version/FreeBSD-version-RELEASE-arch-memstick.img` (oder einem Spiegelserver) im Verzeichnis `ISO-IMAGES/`. Ersetzen Sie *`arch`* und *`version`* durch die von Ihnen verwendete Architektur und die FreeBSD-Version, die Sie einsetzen wollen. Für FreeBSD/i386 10.3-RELEASE finden Sie das Speicherabbild für den USB-Stick beispielsweise unter folgenden Link: `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/ISO-IMAGES/10.3/FreeBSD-10.3-RELEASE-i386-memstick.img` .

   Das benötigte Speicherabbild hat den Dateityp `.img`. Das Verzeichnis `ISO-IMAGES/` enthält verschiedene Speicherabbilder. Sie müssen also (basierend auf der zu installierenden FreeBSD-Version und/oder Ihrer Hardware) das für Sie passende Speicherabbild herunterladen.

   Wichtig

   Sichern Sie Ihre Daten, bevor Sie fortfahren, da im nächsten Schritt alle auf dem USB-Stick befindlichen Daten *gelöscht* werden.

2. Den USB-Stick vorbereiten

Prozedur 2.1. Den USB-Stick unter FreeBSD vorbereiten

Warnung

Das Beispiel im nächsten Schritt verwendet `/dev/da0` als die Gerätedatei, über die Sie den USB-Stick ansprechen. Achten Sie besonders darauf, dass Sie die richtige Gerätedatei verwenden, da Sie ansonsten unbeabsichtigt Daten löschen könnten.

- Das Image mit dd(1) auf den Stick schreiben

  Bei der `.img`-Datei handelt es sich *nicht* um eine normale Datei, die Sie einfach auf den Stick kopieren können. Vielmehr handelt es sich dabei um ein Image des kompletten Dateisystems, das Sie mit dd(1) direkt auf den USB-Stick schreiben müssen:

  ```
  # dd if=FreeBSD-10.3-RELEASE-i386-memstick.img of=/dev/da0 bs=64k
  ```

  Wird dabei die Fehlermeldung `Operation not permitted` angezeigt, stellen Sie bitte sicher, dass das Zielgerät nicht verwendet, manuell eingehängt oder von einem Systemprogramm automatisch eingehängt wurde. Dann versuchen Sie es erneut.

Prozedur 2.2. Den USB-Stick unter Windows® vorbereiten

Warnung

Stellen Sie unbedingt sicher, dass Sie im folgenden Schritt den korrekten Laufwerkbuchstaben für Ihren USB-Stick angeben, da Sie ansonsten unbeabsichtigt Daten löschen könnten.

1. Image Writer für Windows herunterladen

   Image Writer für Windows ist ein frei verfügbares Programm, mit dem Sie ein Image auf einen USB-Stick schreiben können. Laden Sie das Programm von `https://launchpad.net/win32-image-writer/` herunter und entpacken Sie es in einen Ordner auf Ihrer Festplatte.

2. Das Image mit Image Writer auf den Stick schreiben

   Klicken Sie doppelt auf das Symbol Win32DiskImager, um das Programm zu starten. Vergewissern Sie sich, dass es sich bei dem unter `Device` angezeigten Laufwerk um Ihren USB-Stick handelt. Danach klicken Sie auf das Ordnersymbol und wählen die zuvor heruntergeladene Image-Datei aus. Klicken Sie auf Save, um die Image-Datei zu laden. Nachdem Sie alle Eingaben nochmals geprüft haben, müssen Sie noch sicherstellen, dass kein anderes Programm auf den USB-Stick zugreift. Danach klicken Sie auf den Button Write, um das Image auf den USB-Stick zu schreiben.

Um Startdisketten zu erzeugen, benutzen Sie die nachstehende Anleitung:

1. Abbilder der Startdisketten besorgen

Wichtig

Beachten Sie, dass ab FreeBSD 8.*X* Startdisketten nicht mehr unterstützt werden. Lesen Sie bitte weiter oben in diesem Kapitel, wie Sie FreeBSD von einer CD-ROM, einer DVD oder einem USB-Stick installieren können.

Die Abbilder der Startdisketten befinden sich auf dem Installationsmedium im Verzeichnis `floppies/`; sie können auch aus dem Internet heruntergeladen werden: `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/arch/version-RELEASE/floppies/` . Ersetzen Sie *`arch`* und *`version`* durch die passende Architektur und die passende Version. Beispielsweise stehen die Startdisketten von FreeBSD/i386 9.3-RELEASE unter `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/9.3-RELEASE/floppies/` .

Die Abbilder besitzen die Dateinamenerweiterung `.flp`. Im Verzeichnis `floppies/` befinden sich verschiedene Abbilder; welches Sie benutzen, hängt von der zu installierenden FreeBSD-Version und in einigen Fällen vom Zielrechner ab. In den meisten Fällen werden Sie vier Disketten benötigen: `boot.flp`, `kern1.flp` , `kern2.flp` sowie `kern3.flp` . Lesen Sie bitte die Datei `README.TXT` im Verzeichnis `floppies/`, sie enthält aktuelle Informationen zu den Abbildern.

Wichtig

Wenn Sie die Abbilder aus dem Internet herunterladen, benutzen Sie bitte den *Binärmodus* des FTP-Programms. Einige Web-Browser verwenden den *Textmodus* (oder *ASCII*-Modus), was dazu führt, dass sich die erstellten Disketten nicht starten lassen.

2. Die Disketten vorbereiten

Pro Abbild benötigen Sie eine Diskette. Es ist wichtig, dass die verwendeten Disketten fehlerfrei sind. Sie können dies sicherstellen, indem Sie die Disketten selbst formatieren, verlassen Sie sich bitte nicht auf vorformatierte Disketten. Das Formatierprogramm von Windows® zeigt fehlerhafte Blöcke nicht an, es markiert die Blöcke einfach als fehlerhaft und ignoriert sie dann. Benutzen Sie neue Disketten, wenn Sie diese Installationsart verwenden.

Wichtig

Wenn Sie FreeBSD installieren und das Installationsprogramm abstürzt, einfriert oder sich merkwürdig verhält, sind oft fehlerbehaftete Disketten die Ursache. Schreiben Sie die Abbilder auf neue Disketten und versuchen Sie, noch mal zu installieren.

3. Die Abbilder auf Disketten schreiben

Die `.flp`-Dateien sind *keine* normalen Dateien, die Sie auf eine Diskette kopieren. Sie können die Abbilder *nicht* von einem Laufwerk auf ein anderes Laufwerk kopieren. Die Abbilder werden mit einem speziellen Werkzeug direkt auf die Diskette geschrieben.

Wenn Sie die Startdisketten unter MS-DOS® oder Windows® erstellen, können Sie das mitgelieferte Werkzeug `fdimage` verwenden.

Wenn Sie die Abbilder auf der CD-ROM verwenden und das CD-ROM-Laufwerk den Laufwerksbuchstaben `E:` besitzt, führen Sie den nachstehenden Befehl aus:

```
E:\> tools\fdimage floppies\boot.flp A:
```

Führen Sie das Kommando für jede `.flp`-Datei aus. Wechseln Sie bitte jedes Mal die Diskette und beschriften Sie die Diskette mit dem Namen der kopierten Datei. Falls Sie die Abbilder an anderer Stelle liegen haben, passen Sie bitte die Kommandozeile an. Wenn Sie keine CD-ROM besitzen, können Sie `fdimage` aus dem Verzeichnis `tools` des FreeBSD-FTP-Servers herunterladen.

Wenn Sie Startdisketten auf einem UNIX® System (zum Beispiel einem anderen FreeBSD System) erstellen, schreiben Sie die Abbilder mit dem Befehl dd(1) direkt auf die Disketten. Auf einem FreeBSD-System lautet die Kommandozeile:

```
# dd if=boot.flp of=/dev/fd0
```

Unter FreeBSD spricht `/dev/fd0` das erste Diskettenlaufwerk an (das Laufwerk `A:`), `/dev/fd1` spricht das Laufwerk `B:` an. Andere UNIX® Varianten verwenden unter Umständen andere Gerätenamen, die in der Dokumentation des jeweiligen Systems beschrieben sind.

Nun ist alles für die FreeBSD-Installation vorbereitet.

2.4. Die Installation starten

Wichtig

Die Installationsprozedur lässt die Daten auf Ihren Laufwerken solange unverändert bis die nachstehende Meldung erscheint:

```
Last Chance: Are you SURE you want continue the installation?

If you're running this on a disk with data you wish to save then WE
STRONGLY ENCOURAGE YOU TO MAKE PROPER BACKUPS before proceeding!

We can take no responsibility for lost disk contents!
```

Vor dieser Meldung kann die Installationsprozedur jederzeit abgebrochen werden, ohne die Daten auf der Festplatte zu verändern. Wenn Sie meinen, etwas falsch konfiguriert zu haben, können Sie vor diesem Zeitpunkt einfach den Rechner ausschalten.

2.4.1. Der Systemstart

2.4.1.1. Systemstart von i386™-Systemen

1. Schalten Sie zunächst Ihren Rechner aus.

2. Schalten Sie den Rechner ein. Während des Starts sollte angezeigt werden, wie Sie das Systemeinstellungsmenü (oder BIOS) erreichen. Meist drücken Sie dazu die Tasten F2, F10, Del oder Alt+S. Benutzen Sie die angezeigte Tastenkombination. Viele Rechner zeigen beim Systemstart eine Grafik an. Typischerweise können Sie die Grafik mit der Taste Esc entfernen und so die angezeigten Meldungen lesen.

3. Suchen Sie Option, die einstellt von welchem Gerät der Rechner startet. Normalerweise wird die Option `Boot Order` genannt und zeigt eine Geräteliste, beispielsweise `Floppy`, `CD-ROM`, `First Hard Disk` an.

   Wenn Sie von einer CD-ROM starten, stellen Sie sicher, das das CD-ROM-Laufwerk ausgewählt ist. Starten Sie hingegen von einem USB-Stick oder von einer Startdiskette, wählen Sie ebenfalls den entsprechenden

Eintrag aus. Wenn Sie nicht sicher sind, lesen Sie bitte im Handbuch des Rechners oder im Handbuch der Systemplatine nach.

Stellen Sie das gewünschte Startmedium ein und sichern Sie die Einstellungen. Der Rechner sollte dann neu starten.

4. Wenn Sie (wie in Abschnitt 2.3.7, „Das Startmedium vorbereiten“ beschrieben) ein bootbaren USB-Stick vorbereitet haben, stöpseln Sie diesen bitte ein, bevor Sie Ihren Rechner einschalten.

   Wenn Sie den Rechner von einer CD-ROM starten, legen Sie die CD-ROM so früh wie möglich in das Laufwerk ein.

   ### Anmerkung

   Bis einschließlich FreeBSD 7.3 kann FreeBSD auch von einer Startdiskette aus installiert werden. Eine Anleitung hierzu finden Sie in Abschnitt 2.3.7, „Das Startmedium vorbereiten“). Legen Sie die erste Diskette (diese enthält das Abbild `boot.flp`) in das Diskettenlaufwerk ein und starten Sie den Rechner.

   Wenn Ihr Rechner wieder normal startet und das existierende Betriebssystem lädt, kann das folgende Ursachen haben:

   1. Das Startmedium (Diskette, CD-ROM) ist nicht schnell genug eingelegt worden. Belassen Sie das Startmedium im Laufwerk und starten Sie Ihren Rechner neu.

   2. Die BIOS-Einstellungen sind falsch vorgenommen worden. Wiederholen Sie diesen Schritt, bis Sie die richtige Einstellung gefunden haben.

   3. Das verwendete BIOS kann nicht von dem gewünschten Medium starten.

5. FreeBSD startet jetzt. Wenn Sie von einer CD-ROM starten, sehen Sie die folgenden Meldungen (Versionsangaben entfernt):

```
Booting from CD-Rom...
645MB medium detected
CD Loader 1.2

Building the boot loader arguments
Looking up /BOOT/LOADER... Found
Relocating the loader and the BTX
Starting the BTX loader

BTX loader 1.00 BTX version is 1.02
Consoles: internal video/keyboard
BIOS CD is cd0
BIOS drive C: is disk0
BIOS drive D: is disk1
BIOS 636kB/261056kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1

Loading /boot/defaults/loader.conf
/boot/kernel/kernel text=0x64daa0 data=0xa4e80+0xa9e40 syms=[0x4+0x6cac0+0x4+0x88e9d]
\
```

   Wenn Sie mit Startdisketten hochfahren, sehen Sie folgende Meldungen (Versionsangaben entfernt):

```
Booting from Floppy...
Uncompressing ... done
```

```
BTX loader 1.00  BTX version is 1.01
Console: internal video/keyboard
BIOS drive A: is disk0
BIOS drive C: is disk1
BIOS 639kB/261120kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1

Loading /boot/defaults/loader.conf
/kernel text=0x277391 data=0x3268c+0x332a8 |

Insert disk labelled "Kernel floppy 1" and press any key...
```

Folgen Sie der Anweisung und entfernen Sie die `boot.flp`-Diskette, anschließend legen Sie die `kern1.flp` -Diskette ein und drücken Enter. Starten Sie das System mit der ersten Diskette und legen Sie, wenn Sie dazu aufgefordert werden, die anderen Disketten ein.

6. Unabhängig davon, ob Sie von Disketten oder von CD-ROM gestartet haben, erscheint danach das FreeBSD Bootloader-Menü:

Abbildung 2.1. FreeBSD Boot Loader Menu

Warten Sie entweder zehn Sekunden oder drücken Sie Enter.

2.4.1.2. Systemstart bei SPARC64®-Systemen

Die meisten SPARC64®-Systeme sind so konfiguriert, dass sie automatisch von der Festplatte starten. Um FreeBSD auf einem solchen System zu installieren, müssen Sie das System aber über das Netzwerk oder von einer CD-ROM starten. Daher müssen Sie den Bootprozess unterbrechen und das System über das PROM (OpenFirmware) starten.

Dazu starten Sie Ihr System neu und warten, bis die Startmeldung erscheint. Der genaue Wortlaut hängt vom eingesetzten Modell ab, die Nachricht sollte aber ähnlich der folgenden aussehen:

```
Sun Blade 100 (UltraSPARC-IIe), Keyboard Present
Copyright 1998-2001 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.2, 128 MB memory installed, Serial #51090132.
Ethernet address 0:3:ba:b:92:d4, Host ID: 830b92d4.
```

Um den Startvorgang zu unterbrechen, drücken Sie nun die Tastenkombination L1+A oder Stop+A. Verwenden Sie eine serielle Verbindung, senden Sie das Signal `BREAK` über die serielle Konsole (etwa durch die Eingabe von `~#` in den Programmen tip(1) oder cu(1)). In beiden Fällen landen Sie anschließend am PROM-Prompt:

```
ok     ❶
```

```
ok {0} ❷
```

❶ Der auf Einprozessorsystemen verwendete Prompt.
❷ Der Prompt auf Mehrprozessorsystemen. Die Zahl steht dabei für die Anzahl der vorhandenen Prozessoren.

Nun legen Sie Ihre CD-ROM in das Laufwerk ein und geben am PROM-Prompt `boot cdrom` ein. Danach startet Ihr System von der eingelegten CD-ROM.

2.4.2. Die Geräteerkennung prüfen

Die letzten paar Hundert Zeilen der Bildschirmausgabe werden gesichert und können geprüft werden.

Um sich den Bildschirmpuffer anzusehen, drücken Sie die Taste Scroll-Lock. Im Puffer können Sie mit den Pfeiltasten oder den Tasten PageUp und PageDown blättern. Um zur normalen Bildschirmausgabe zurückzukehren, drücken Sie nochmals die Taste Scroll-Lock.

Prüfen Sie mit diesem Verfahren nun die Ausgaben der Geräteerkennung. Sie werden einen Text ähnlich wie in Abbildung 2.2, „Ausgabe der Geräteerkennung“ sehen. Die genauen Ausgaben sind abhängig von den in Ihrem System installierten Geräten.

```
avail memory = 253050880 (247120K bytes)
Preloaded elf kernel "kernel" at 0xc0817000.
Preloaded mfs_root "/mfsroot" at 0xc0817084.
md0: Preloaded image </mfsroot> 4423680 bytes at 0xc03ddcd4

md1: Malloc disk
Using $PIR table, 4 entries at 0xc00fde60
npx0: <math processor> on motherboard
npx0: INT 16 interface
pcib0: <Host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcib1:<VIA 82C598MVP (Apollo MVP3) PCI-PCI (AGP) bridge> at device 1.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <Matrox MGA G200 AGP graphics accelerator> at 0.0 irq 11
isab0: <VIA 82C586 PCI-ISA bridge> at device 7.0 on pci0
isa0: <iSA bus> on isab0
atapci0: <VIA 82C586 ATA33 controller> port 0xe000-0xe00f at device 7.1 on pci0
ata0: at 0x1f0 irq 14 on atapci0
ata1: at 0x170 irq 15 on atapci0
uhci0 <VIA 83C572 USB controller> port 0xe400-0xe41f irq 10 at device 7.2 on pci
0
usb0: <VIA 83572 USB controller> on uhci0
usb0: USB revision 1.0
uhub0: VIA UHCI root hub, class 9/0, rev 1.00/1.00, addr1
uhub0: 2 ports with 2 removable, self powered
pci0: <unknown card> (vendor=0x1106, dev=0x3040) at 7.3
dc0: <ADMtek AN985 10/100BaseTX> port 0xe800-0xe8ff mem 0xdb000000-0xeb0003ff ir
q 11 at device 8.0 on pci0
dc0: Ethernet address: 00:04:5a:74:6b:b5
miibus0: <MII bus> on dc0
ukphy0: <Generic IEEE 802.3u media interface> on miibus0
ukphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
ed0: <NE2000 PCI Ethernet (RealTek 8029)> port 0xec00-0xec1f irq 9 at device 10.
0 on pci0
ed0 address 52:54:05:de:73:1b, type NE2000 (16 bit)
isa0: too many dependant configs (8)
isa0: unexpected small tag 14
orm0: <Option ROM> at iomem 0xc0000-0xc7fff on isa0
fdc0: <NEC 72065B or clone> at port 0x3f0-0x3f5,0x3f7 irq 6 drq2 on isa0
fdc0: FIFO enabled, 8 bytes threshold
fd0: <1440-KB 3.5” drive> on fdc0 drive 0
atkbdc0: <Keyboard controller (i8042)> at port 0x60,0x64 on isa0
atkbd0: <AT Keyboard> flags 0x1 irq1 on atkbdc0
kbd0 at atkbd0
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: model Generic PS/@ mouse, device ID 0
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
sio0 at port 0x3f8-0x3ff irq 4 flags 0x10 on isa0
sio0: type 16550A
sio1 at port 0x2f8-0x2ff irq 3 on isa0
sio1: type 16550A
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
pppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/15 bytes threshold
plip0: <PLIP network interface> on ppbus0
ad0: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata0-master UDMA33
acd0: CD-RW <LITE-ON LTR-1210B> at ata1-slave PIO4
Mounting root from ufs:/dev/md0c
/stand/sysinstall running as init on vty0
```

Abbildung 2.2. Ausgabe der Geräteerkennung

Prüfen Sie die Ausgabe der Geräteerkennung sorgfältig und stellen Sie sicher, dass FreeBSD alle erwarteten Geräte gefunden hat. Wenn ein Gerät nicht gefunden wurde, wird es nicht angezeigt. Ist dies bei Ihnen der Fall, müssen Sie

einen angepassten Kernel erstellen, da das betroffene Gerät (beispielsweise eine Soundkarte) in diesem Fall vom `GENERIC` -Kernel nicht unterstützt wird.

Sie gelangen im nächsten Schritt in ein Menü, in dem Sie über die Cursortasten das Land, in dem Sie sich befinden, auswählen können (Abbildung 2.3, „Ihr Land auswählen“). Durch die Bestätigung mit der Enter-Taste wird automatisch das von Ihnen gewählte Land sowie die dazu passende Tastaturbelegung gewählt.

Abbildung 2.3. Ihr Land auswählen

Haben Sie als Land United States gewählt, wird automatisch die amerikanische Standardtastatur verwendet. Haben Sie hingegen ein anderes Land angegeben, erscheint das folgende Menü, in dem Sie Ihre Tastaturbelegung auswählen können (bestätigen Sie Ihre Auswahl mit der Enter-Taste).

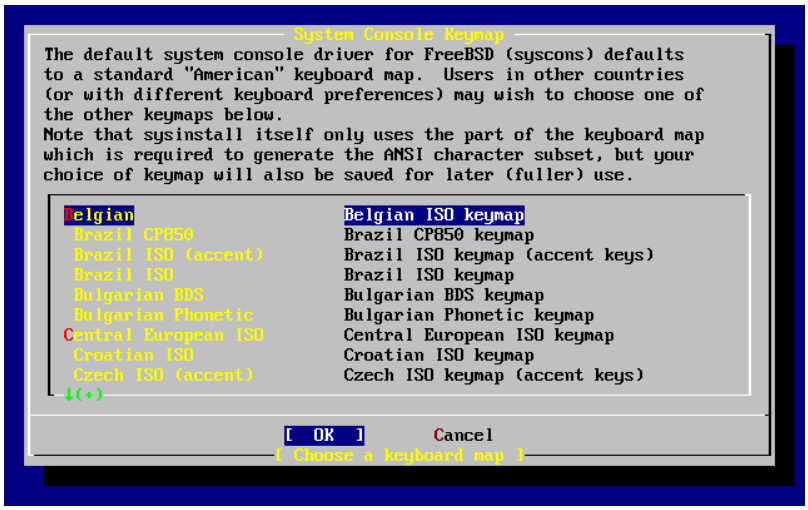


Abbildung 2.4. Die Tastaturbelegung auswählen

Nachdem Sie das Land ausgewählt haben, erscheint das Hauptmenü von sysinstall.

2.5. Das Werkzeug sysinstall

Zum Installieren von FreeBSD stellt das FreeBSD-Project das Werkzeug sysinstall zur Verfügung. Das Werkzeug arbeitet textorientiert und bietet eine Reihe von Menüs und Bildschirmen, um den Installationsprozess zu konfigurieren und zu steuern.

Die Menüs von sysinstall werden mit Tasten wie den Pfeiltasten, Enter, Tab oder Space bedient. Eine ausführliche Beschreibung der Tastenbelegung ist in der Gebrauchsanweisung von sysinstall enthalten.

Die Gebrauchsanweisung können Sie lesen, indem Sie den Menüpunkt Usage auswählen. Stellen Sie sicher, dass die Schaltfläche [Select], wie in Abbildung 2.5, „Die Gebrauchsanweisung von sysinstall auswählen“ gezeigt, aktiviert ist und drücken Sie die Taste Enter.

Es erscheinen Anweisungen wie das Menüsystem zu bedienen ist. Wenn Sie diese gelesen haben, drücken Sie Enter, um in das Hauptmenü zurückzukehren.

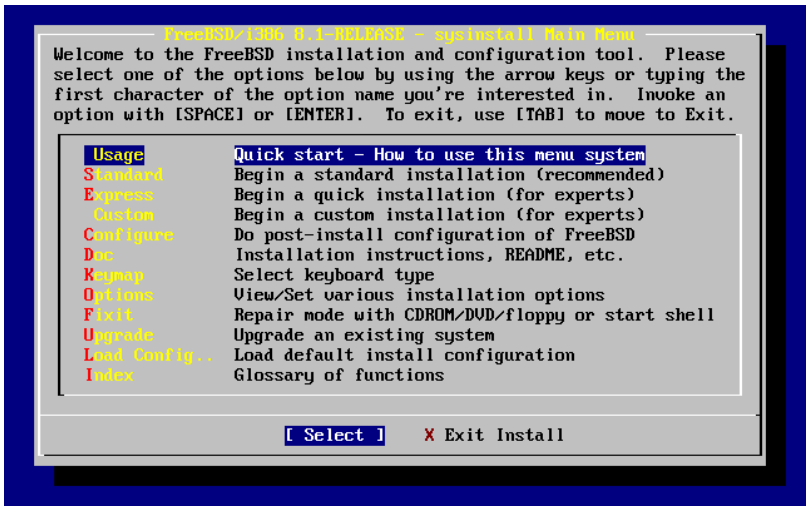


Abbildung 2.5. Die Gebrauchsanweisung von sysinstall auswählen

2.5.1. Die Dokumentation abrufen

Aus dem Hauptmenü wählen Sie mit den Pfeiltasten Doc aus und drücken Enter.

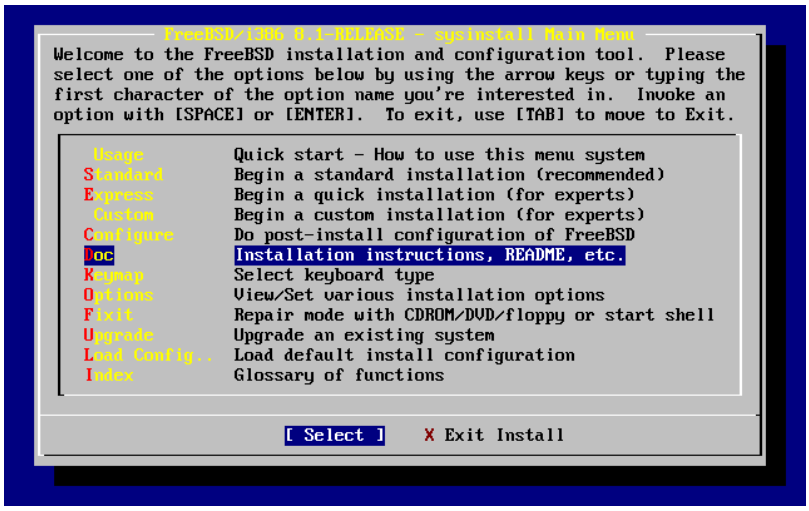


Abbildung 2.6. Die Dokumentation abrufen

Es wird das Dokumentationsmenü angezeigt.

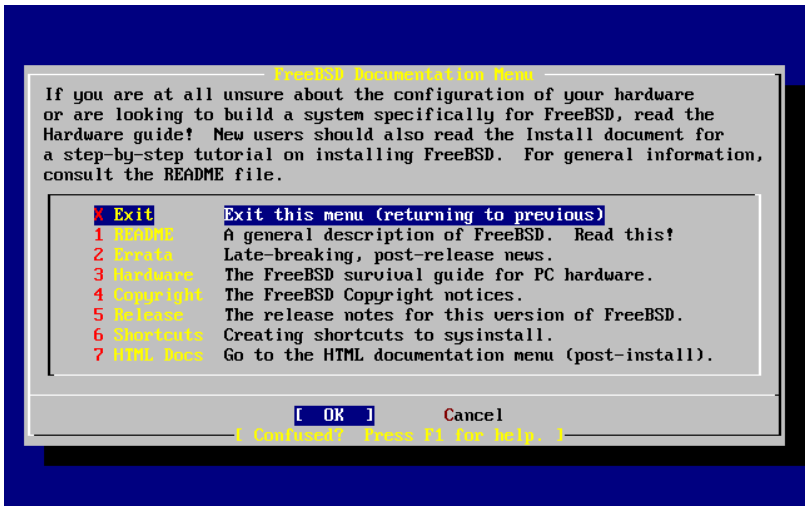


Abbildung 2.7. Das Dokumentationsmenü von sysinstall

Lesen Sie bitte unbedingt die mitgelieferte Dokumentation.

Um ein Dokument zu lesen, wählen Sie das Dokument mit den Pfeiltasten aus und drücken Enter. Wenn Sie das Dokument gelesen haben, kommen Sie mit der Taste Enter in das Dokumentationsmenü zurück.

Um in das Hauptmenü zurückzukommen, wählen Sie mit den Pfeiltasten Exit aus und drücken die Taste Enter.

2.5.2. Die Tastaturbelegung ändern

Um die Tastaturbelegung zu ändern, wählen Sie den Menüpunkt Keymap und drücken Enter. Dies ist nur erforderlich wenn Sie eine nicht standard-konforme Tastatur oder eine andere als eine amerikanische Tastatur einsetzen.

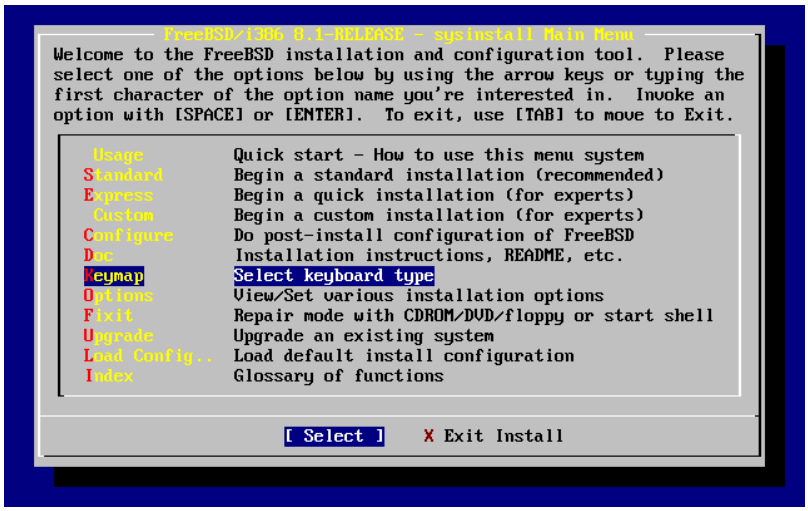


Abbildung 2.8. Das Hauptmenü von sysinstall

Eine andere Tastaturbelegung können Sie mit den Pfeiltasten markieren und der Taste Space auswählen. Wenn Sie die Taste Space nochmals drücken wird die Auswahl aufgehoben. Haben Sie eine Tastaturbelegung ausgewählt, markieren Sie mit den Pfeiltasten [OK] und drücken Sie Enter.

Der Bildschirmabzug zeigt nur einen der verfügbaren Belegungen an. Mit der Taste Tab markieren Sie die Schaltfläche [Cancel], die mit der Vorgabe-Belegung wieder in das Hauptmenü zurückführt.

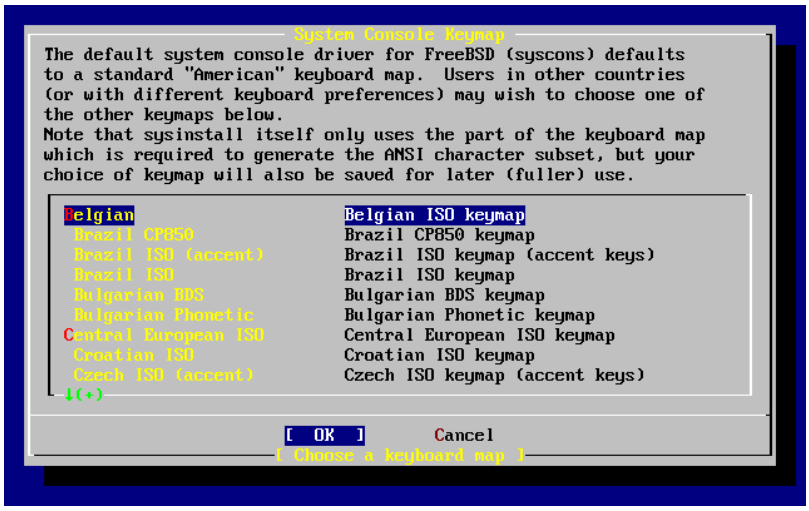


Abbildung 2.9. Sysinstall Keymap Menu

2.5.3. Installationsoptionen einstellen

Wählen Sie Options aus und rücken die Taste Enter.

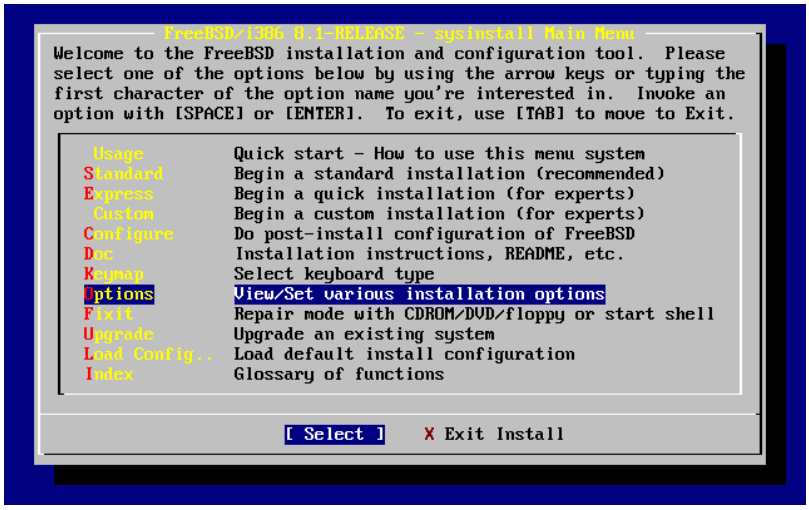


Abbildung 2.10. Das Hauptmenü von sysinstall

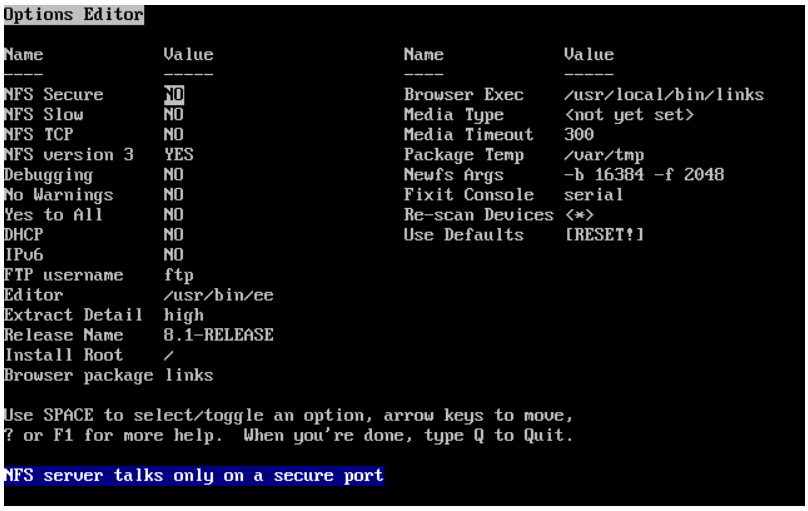


Abbildung 2.11. Optionen von sysinstall

Für die meisten Benutzer sind die voreingestellten Werte völlig ausreichend und brauchen daher nicht geändert werden. Der Name des Releases variiert mit der zu installierenden Version von FreeBSD.

Eine Beschreibung der ausgewählten Option erscheint blau hervorgehoben am unteren Ende des Bildschirms. Mit der Option Use Defaults können Sie alle Optionen auf die Vorgabewerte zurückstellen.

Wenn Sie die Hilfeseite zu den verschiedenen Optionen lesen wollen, drücken Sie die Taste F1.

Die Taste Q führt in das Hauptmenü zurück.

2.5.4. Eine Standard-Installation starten

Die Standard-Installation sollte von allen UNIX®- oder FreeBSD-Anfängern benutzt werden. Markieren Sie mit den Pfeiltasten Standard und drücken Sie Enter, um die Installation zu starten.

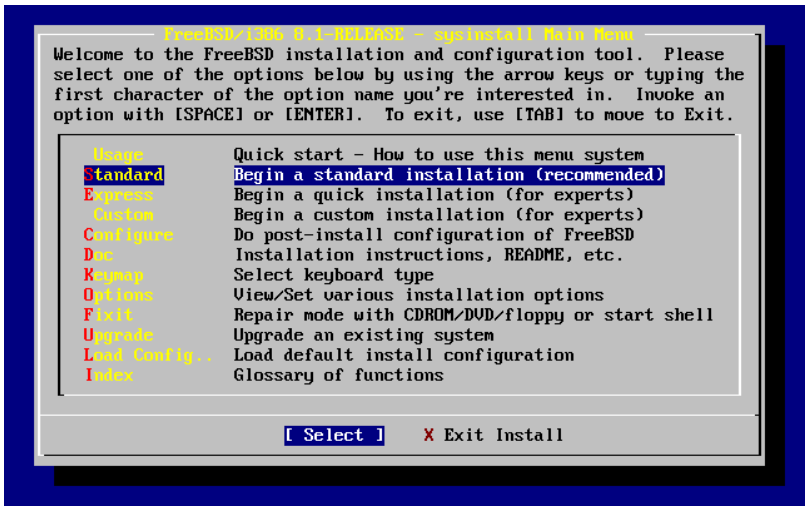


Abbildung 2.12. Die Standard-Installation starten

2.6. Plattenplatz für FreeBSD bereitstellen

Ihre erste Aufgabe ist, FreeBSD Plattenplatz bereitzustellen und den Plattenplatz für sysinstall kenntlich zu machen (*label*). Sie müssen daher wissen, wie FreeBSD mit Platten umgeht.

2.6.1. Nummerierung der Laufwerke im BIOS

Bevor Sie FreeBSD installieren und konfigurieren, sollten Sie Sie einen wichtigen Punkt beachten. Dies gilt insbesondere dann, wenn Sie mehrere Festplatten besitzen.

In einem PC, der unter einem vom BIOS abhängigen Betriebssystem, wie MS-DOS® oder Microsoft® Windows® läuft, kann das BIOS die normale Reihenfolge der Laufwerke verändern und das Betriebssystem beachtet diese Änderung. Mit dieser Funktion kann der Rechner von einem anderen Laufwerk als dem so genannten „primären Laufwerk“ gestartet werden. Die Funktion ist sehr zweckmäßig für Benutzer, die Datensicherungen auf einer zweiten Platte erstellen und dafür Werkzeuge wie Ghost oder xcopy einsetzen. Wenn die erste Platte ausfällt, von einem Virus befallen wird oder durch einen Fehler des Betriebssystems verunstaltet wird, können die Platten im BIOS logisch getauscht werden. Es sieht so aus, als wären die Laufwerke, ohne Öffnen des Gehäuses getauscht worden.

Teurere Systeme mit SCSI-Controllern haben oft BIOS-Erweiterungen, mit denen die Reihenfolge von bis zu sieben SCSI-Platten in ähnlicher Weise verändert werden kann.

Ein Benutzer, der es gewohnt ist, diese BIOS-Funktionen zu benutzen, mag überrascht sein, dass FreeBSD sich nicht wie erwartet verhält. FreeBSD verwendet das BIOS nicht und weiß daher nichts von der logischen Plattenordnung im BIOS. Dies kann zu sehr verwirrenden Situationen führen, insbesondere wenn die Platten identische Geometrien besitzen und Kopien voneinander sind.

Vor der Installation von FreeBSD sollte im BIOS die normale Nummerierung der Laufwerke eingestellt und so belassen werden. Ist es nötig, die Reihenfolge der Laufwerke zu verändern, so sollte das immer auf dem schweren Weg, also durch Öffnen des Gehäuses und Verändern der Jumper und Kabel, erfolgen.

Von Bills und Freds ungewöhnlichen Abenteuern

Bill macht aus einer älteren Wintel Kiste ein neues FreeBSD-System für Fred. Auf einer SCSI-Platte, die er mit der SCSI-ID 0 konfiguriert, installiert Bill FreeBSD.

Nachdem Fred das System einige Tage benutzt hat, bemerkt er, dass die ältere SCSI-Platte viele Fehler meldet und beschwert sich bei Bill.

Nach einigen Tagen entschließt sich Bill, die Sache in die Hand zu nehmen. Er schnappt sich eine identische SCSI-Platte aus dem Lager im Hinterzimmer und baut diese, nachdem Sie einen Oberflächenscan überstanden hat, mit der SCSI-ID 4 ein. Anschließend kopiert er die Daten von der Platte mit der SCSI-ID 0 auf die Platte mit der SCSI-ID 4. Da die neue Platte zufriedenstellend läuft, stellt Bill im SCSI-BIOS die Reihenfolge der Platten so um, dass das System von der neuen Platte startet. Nach einem problemlosen Start von FreeBSD läuft das System und Fred ist zufrieden.

Nach einiger Zeit haben Bill und Fred Lust auf ein weiteres Abenteuer – Sie wollen das System auf eine neue FreeBSD-Version aktualisieren. Bill ersetzt die angeschlagene Platte mit der SCSI-ID 0 durch eine gleiche Platte aus dem Lager. Auf der ausgetauschten Platte installiert er problemlos mithilfe von Freds Startdisketten die neue Version von FreeBSD.

Fred braucht ein paar Tage, um die neue FreeBSD-Version zu testen und entscheidet, dass Sie für den produktiven Einsatz geeignet ist. Nun müssen die Daten von der alten Platte (mit der SCSI-ID 4) kopiert werden. Fred hängt dazu die alte Platte ein und stellt bestürzt fest, dass alle Daten verschwunden sind.

Wo sind die Daten hin?

Bill kopierte die Daten von der Platte mit der SCSI-ID 0 auf die Platte mit der SCSI-ID 4. Als Bill die Startreihenfolge im SCSI-BIOS änderte, führte er sich nur selbst an der Nase herum. FreeBSD lief weiterhin auf der Platte mit der SCSI-ID 0. Durch die Änderung der Startreihenfolge wurde nur ein Teil des Boot- und Loader-Codes von der Platte mit der SCSI-ID 4 geladen. Die Kernel-Treiber von FreeBSD ignorieren die BIOS-Einstellungen und benutzen die normale Nummerierung. Das System lief also weiterhin auf der Platte mit der SCSI-ID 0 und alle Daten von Fred befanden sich auf dieser Platte. Es schien nur so, als würde das System auf der Platte mit der SCSI-ID 4 laufen.

Wir sind erleichtert zu bemerken, dass keine Daten verloren gingen oder verändert wurden. Die alte Platte wurde im Müll wiedergefunden und Freds Daten konnten wiederhergestellt werden (Bill weiß jetzt, dass er noch viel zu lernen hat).

Obwohl in diesem Beispiel SCSI-Platten verwendet wurden, gelten die Konzepte gleichermaßen für IDE-Platten.

2.6.2. Slices mit Fdisk erstellen

Anmerkung

Zu diesem Zeitpunkt werden noch keine Änderungen auf die Festplatte ausgeschrieben. Sie können daher sysinstall jederzeit verlassen, und erneut beginnen, wenn Sie denken, einen Fehler gemacht zu haben. Sie können sysinstall über die Menüs verlassen, die Taste U drücken oder die Option Undo wählen. Wenn Sie einmal nicht wissen, wie Sie ein Menü verlassen, können Sie den Rechner auch einfach ausschalten.

Nachdem Sie in sysinstall die Standard-Installation ausgewählt haben, werden Sie folgende Meldung sehen:

```
Message
```

```
In the next menu, you will need to set up a DOS-style ("fdisk")
partitioning scheme for your hard disk. If you simply wish to devote
all disk space to FreeBSD (overwriting anything else that might be on
the disk(s) selected) then use the (A)ll command to select the default
partitioning scheme followed by a (Q)uit. If you wish to allocate only
free space to FreeBSD, move to a partition marked "unused" and use the
(C)reate command.
                                [  OK  -]

                        [ Press enter or space -]
```

Drücken Sie, wie angegeben, Enter. Im nächsten Bildschirm werden alle Festplatten angezeigt, die der Kernel während der Geräteerkennung gefunden hat. Abbildung 2.13, „Ein Laufwerk für Fdisk aussuchen“ zeigt ein Beispiel von einem System mit zwei IDE-Platten, die als `ad0` und `ad2` erkannt wurden.

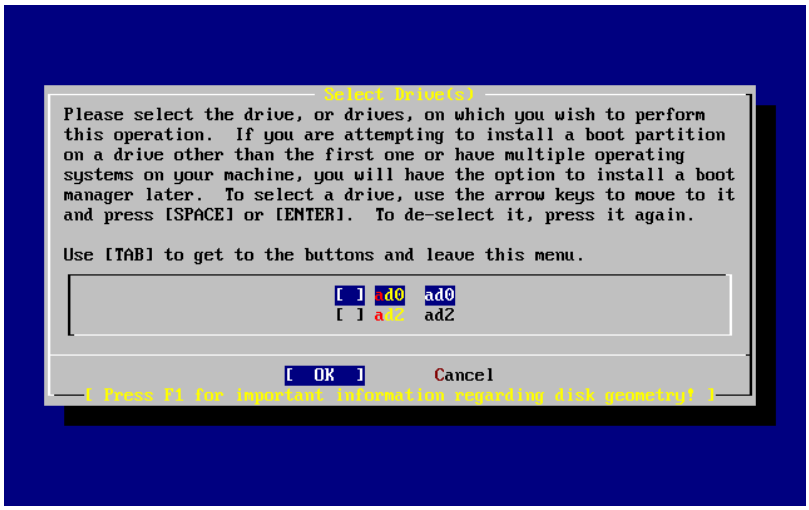


Abbildung 2.13. Ein Laufwerk für Fdisk aussuchen

Sie fragen sich vielleicht, warum `ad1` nicht angezeigt wird. Wurde die Platte vielleicht nicht erkannt?

Stellen Sie sich ein System mit zwei IDE-Platten vor. Eine Platte ist als Master am ersten Controller, die andere als Master am zweiten Controller angeschlossen. Wenn FreeBSD die Platten in der Reihenfolge, in der sie gefunden werden, nummerieren würde, hießen die Platten `ad0` und `ad1` und alles würde funktionieren.

Wenn Sie nun am ersten IDE-Controller eine dritte Platte als Slave anschließen würden, wäre diese Platte `ad1`. Die vorher `ad1` genannte Platte würde nun `ad2` heißen. Dateisysteme werden auf Geräten wie `ad1s1a` angelegt. Daher könnte es passieren, dass auf einmal Dateisysteme nicht mehr gefunden werden und Sie FreeBSD umkonfigurieren müssten.

Um diese Probleme zu umgehen, kann der Kernel so eingestellt werden, dass er Platten nach ihrem Anschlussort anstelle der gefundenen Reihenfolge benennt. Nach diesem Schema ist die Master-Platte am zweiten IDE-Controller *immer* `ad2`, auch wenn es die Geräte `ad0` oder `ad1` gar nicht gibt.

Dieses Verhalten ist in FreeBSD voreingestellt und der Grund warum im Beispiel die Geräte `ad0` und `ad2` angezeigt werden. Der Rechner, von dem die gezeigte Ausgabe stammt, hatte zwei IDE-Platten, die beide als Master konfiguriert waren, und keine Slave-Platten.

Wählen Sie die Platte aus, auf die Sie FreeBSD installieren wollen und drücken Sie [OK]. Anschließend startet Fdisk und zeigt einen Bildschirm wie den in Abbildung 2.14, „Typischer Fdisk-Bildschirm vor dem Editieren“.

Der Bildschirm von Fdisk ist in drei Abschnitte unterteilt.

Der erste Abschnitt umfasst die ersten beiden Zeilen der Anzeige. Er enthält Einzelheiten über die aktuell ausgewählte Platte, unter anderem den FreeBSD-Gerätenamen, die Plattengeometrie und die Kapazität der Platte.

Der zweite Abschnitt zeigt die auf der Platte befindlichen Slices. Angezeigt wird der Anfang und das Ende der Slice, die Größe der Slice, der FreeBSD-Gerätename, eine Beschreibung und der Subtyp. Im Beispiel sehen Sie zwei unbenutzte Slices, die durch die Plattenbelegung auf PCs entstehen. Weiterhin sehen Sie eine große FAT-Slice, die ziemlich sicher unter MS-DOS®/Windows® als Laufwerk `C:` auftaucht und eine erweiterte Slice, die unter MS-DOS®/Windows® weitere Laufwerke enthalten kann.

Im dritten Abschnitt sind die Kommandos von Fdisk zusammengefasst.

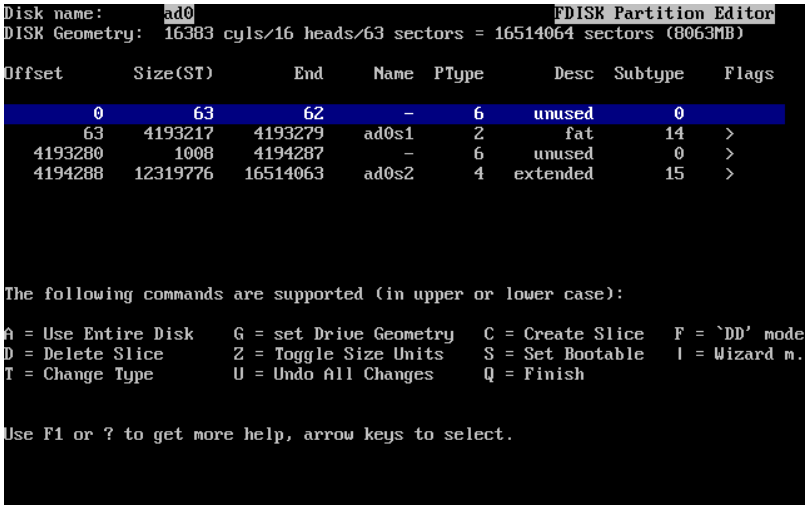


Abbildung 2.14. Typischer Fdisk-Bildschirm vor dem Editieren

Die nächsten Schritte hängen von der beabsichtigten Einteilung der Festplatte ab.

Wenn Sie die gesamte Festplatte für FreeBSD verwenden wollen, drücken Sie die Taste A (entspricht dem Menüpunkt Use Entire Disk). Später im Installationsverlauf müssen Sie diese Auswahl bestätigen, danach werden alle bisherigen Daten von der Festplatte gelöscht. Diese Auswahl löscht vorher vorhandene Slices und ersetzt sie durch einen kleinen unbenutzten Bereich (der wieder durch das PC-Design bedingt ist) und eine große Slice für FreeBSD. Wählen Sie dann die neu erstellte Slice mit den Pfeiltasten aus und drücken Sie die Taste S, um die Slice als startfähig (bootbar) zu markieren. Abbildung 2.15, „Eine Partition über die gesamte Platte“ zeigt den Bildschirm zu diesem Zeitpunkt. Beachten Sie das `A` in der Spalte `Flags`. Dies zeigt an, dass die Slice *aktiv* ist und das System von dieser Slice starten wird.

Um Platz für FreeBSD zu schaffen, können Sie auch bestehende Slices löschen. Markieren Sie dazu die Slice mit den Pfeiltasten und drücken Sie die Taste D. Danach legen Sie eine neue Slice mit der Taste C an. Sie werden nach der Größe der zu erstellenden Slice gefragt; der Vorgabewert entspricht der größten Slice, die angelegt werden kann (entspricht entweder dem größten freien Bereich auf der Festplatte oder der ganzen Festplatte).

Wenn Sie schon Platz für FreeBSD geschaffen haben (beispielsweise mit PartitionMagic®), können Sie eine neue Slice direkt mit der Taste C anlegen. Sie werden wieder nach der Größe der anzulegenden Slice gefragt.

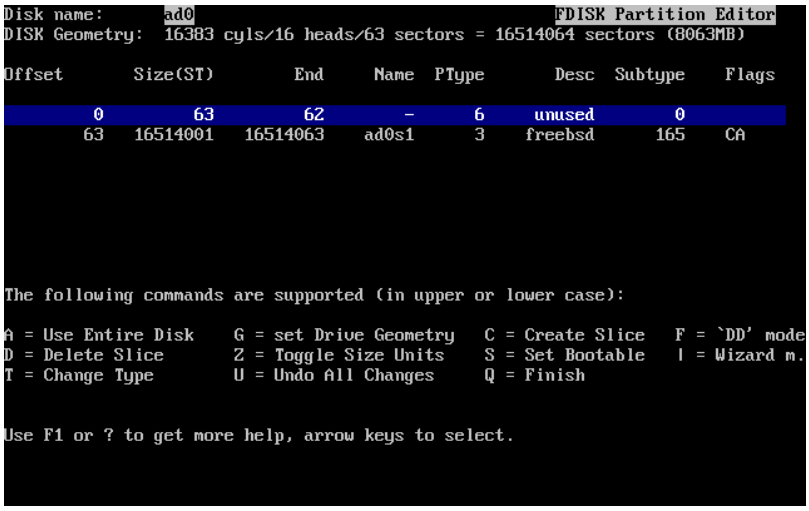


Abbildung 2.15. Eine Partition über die gesamte Platte

Drücken Sie die Taste Q, wenn Sie fertig sind. Sysinstall merkt sich die Änderungen, schreibt sie aber noch nicht auf die Festplatte.

2.6.3. Einen Boot-Manager installieren

Sie können nun einen Boot-Manager installieren. Unter folgenden Umständen sollten Sie den FreeBSD-Boot-Manager installieren:

- Das System besitzt mehr als ein Laufwerk und FreeBSD ist auf einem anderen Laufwerk als dem ersten Laufwerk installiert.
- FreeBSD teilt sich das Laufwerk mit einem anderen Betriebssystem. Beim Systemstart wollen Sie auswählen, welches Betriebssystem gestartet wird.

Wird der Rechner ausschließlich mit FreeBSD betrieben und FreeBSD ist auf dem ersten Laufwerk installiert, dann genügt der Standard-Boot-Manager. Wenn Sie einen anderen Boot-Manager benutzen, der FreeBSD starten kann, wählen Sie bitte None aus.

Nachdem Sie die Auswahl getroffen haben, drücken Sie die Taste Enter.

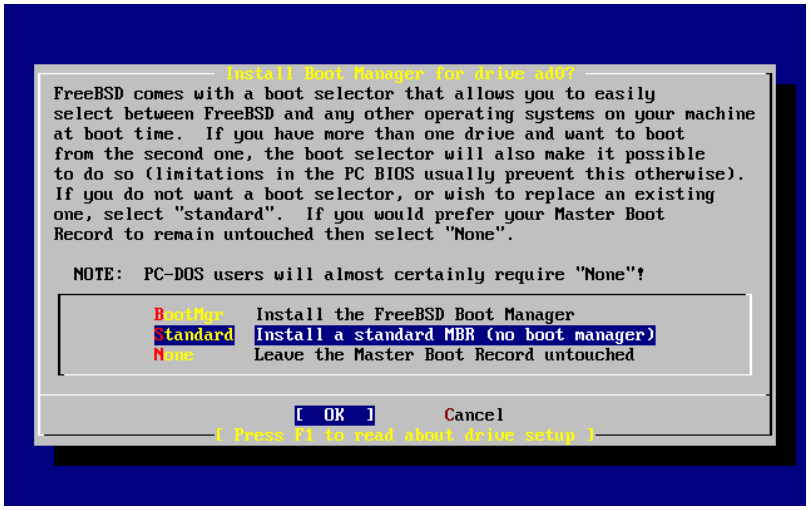
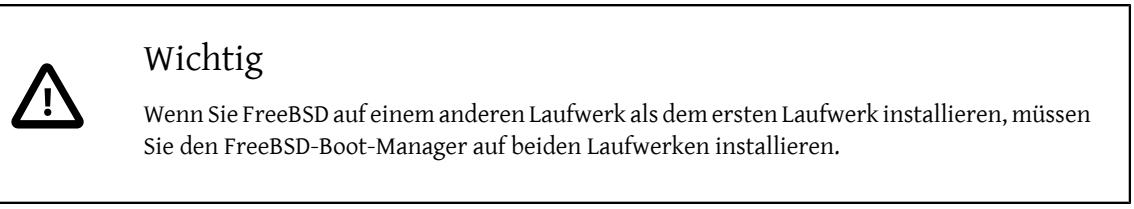


Abbildung 2.16. Sysinstall Boot-Manager-Menü

In der Hilfe, die Sie mit der Taste F1 aufrufen, werden Probleme beschrieben, die entstehen können, wenn sich zwei Betriebssysteme ein Laufwerk teilen.

2.6.4. Slices auf einem anderen Laufwerk anlegen

Wenn das System mehr als ein Laufwerk besitzt, kehrt die Installationsprozedur nach der Auswahl des Boot-Managers zum Bildschirm `Select Drives` zurück. Sie können hier ein anderes Laufwerk auswählen und auf diesem Laufwerk mit Fdisk weitere Slices anlegen.



Wichtig

Wenn Sie FreeBSD auf einem anderen Laufwerk als dem ersten Laufwerk installieren, müssen Sie den FreeBSD-Boot-Manager auf beiden Laufwerken installieren.

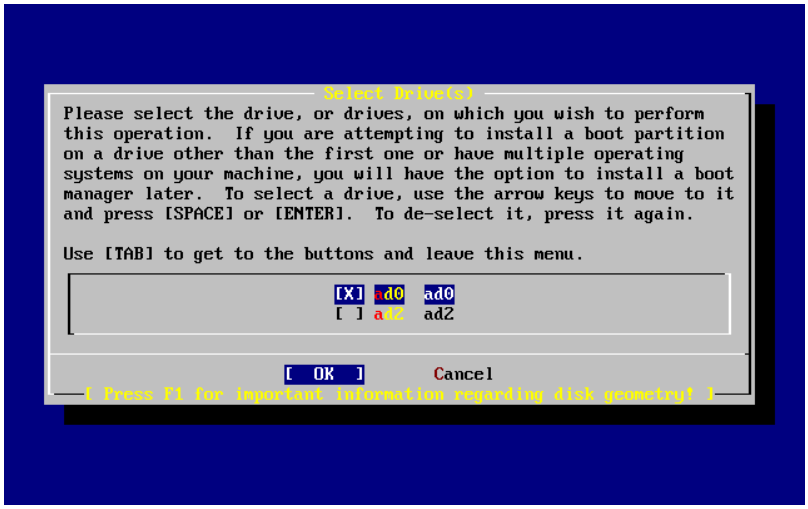


Abbildung 2.17. Die Laufwerksauswahl verlassen

Die Taste Tab wechselt zwischen dem zuletzt ausgewählten Laufwerk und den Schaltflächen [OK] und [Cancel].

Drücken Sie einmal die Taste Tab, um [OK] auszuwählen und drücken Sie anschließend Enter um die Installation weiterzuführen.

2.6.5. Partitionen mit Bsdlabel anlegen

In jeder angelegten Slice müssen Sie Partitionen anlegen. Die Partitionen werden mit Buchstaben von `a` bis `h` gekennzeichnet. Die Buchstaben `b`, `c` und `d` haben eine besondere Bedeutung, die Sie beachten sollten.

Einige Anwendungen profitieren von einer besonderen Aufteilung der Partitionen, insbesondere wenn das System mehr als ein Laufwerk besitzt. Bei der ersten FreeBSD-Installation sollten Sie sich allerdings nicht zu viele Gedanken über die Partitionen machen. Wichtiger ist, dass Sie FreeBSD installieren und benutzen. Wenn Sie mehr Erfahrung mit FreeBSD gesammelt haben, können Sie FreeBSD jederzeit mit anderen Partitionen installieren.

Das folgende Schema legt vier Partitionen an: Eine Partition für den Auslagerungsbereich (*swap space*) und drei Partitionen für Dateisysteme.

Tabelle 2.2. Partitionen auf dem ersten Laufwerk

Partition	Dateisystem	Größe	Beschreibung
`a`	`/`	1 GB	Das Root-Dateisystem. Jedes andere Dateisystem wird irgendwo unterhalb von diesem Dateisystem eingehangen. 1 GB ist eine

Partition	Dateisystem	Größe	Beschreibung
			vernünftige Größe für dieses Dateisystem. Sie werden hier wenig Daten speichern und FreeBSD benötigt ungefähr 128 MB Platz auf diesem Dateisystem. Der Rest ist für temporäre Daten und die Reserve, falls künftige Versionen von FreeBSD mehr Platz in `/` benötigen.
`b`	N/A	2-3 x RAM	Der Auslagerungsbereich befindet sich auf der `b`-Partition. Es ist schon fast eine Kunst, die Größe des Auslagerungsbereichs richtig zu bestimmen. Eine gute Daumenregel ist, den Auslagerungsbereich zwei bis dreimal größer als den Hauptspeicher (RAM) anzulegen. Sie sollten mindestens 64 MB für den Auslagerungsbereich vorsehen. Wenn das System also weniger als 32 MB Hauptspeicher besitzt, richten Sie einen 64 MB großen Auslagerungsbereich ein. Besitzt das System mehr als ein Laufwerk, können Sie auf jedem Laufwerk Auslagerungsbereiche anlegen. Da FreeBSD alle Auslagerungsbereiche benutzt, wird der Vorgang des Auslagerns durch mehrere Bereiche beschleunigt. Berechnen Sie in diesem Fall die Größe des benötigten Auslagerungsbereichs, beispielsweise 128 MB, und teilen Sie die Größe durch die Anzahl der Laufwerke. Dies gibt die Größe des Auslagerungsbereichs auf jedem Laufwerk. Mit zwei Platten ergibt das in diesem Beispiel 64 MB Auslagerungsbereich pro Platte.
`e`	`/var`	512 MB bis 4096 MB	Das Verzeichnis `/var` enthält Dateien, die sich dauernd ändern (Protokolldateien und Dateien für Verwaltungszwecke) und auf die im Normalbetrieb oft zugegriffen wird. Liegen diese Dateien in einem gesonderten Dateisystem, kann FreeBSD den Zugriff auf die Dateien optimieren, ohne den Zugriff auf Dateien mit einem anderen Zugriffmuster zu stören.
`f`	`/usr`	Der Rest des Laufwerks (mindestens 8 GB)	Alle anderen Dateien werden normalerweise im Verzeichnis `/usr` oder einem Unterverzeichnis von `/usr` abgelegt.

Warnung

Die eben genannten Werte dienen nur als Beispiel und sollten nur von erfahrenen Benutzern editiert werden. Wir empfehlen Ihnen, die vom Partitionseditor vorgeschlagene Aufteilung (`Auto Defaults`) zu verwenden.

Wenn Sie FreeBSD auf mehr als einem Laufwerk installieren, müssen Sie noch weitere Partitionen in den Slices auf den anderen Laufwerken anlegen. Am einfachsten legen Sie pro Laufwerk zwei Partitionen an: eine für den Auslagerungsbereich und eine andere für ein Dateisystem.

Tabelle 2.3. Partitionen auf weiteren Laufwerken

Partition	Dateisystem	Größe	Beschreibung
`b`	-	-	Wie schon besprochen, können Sie den Auslagerungsbereich auf mehrere Platten verteilen. Auch wenn die `a`-Partition frei ist, sollte der Aus-

Partition	Dateisystem	Größe	Beschreibung
			lagerungsbereich entsprechend der Konvention auf der `b`-Partition angelegt werden.
`e`	/disk*n*	Der Rest des Laufwerks	Der Rest der Platte wird von einer großen Partition eingenommen. Sie könnten für diese Partition die `a`-Partition anstelle der `e`-Partition benutzen. Allerdings ist die `a`-Partition per Konvention für das Root-Dateisystem (`/`) reserviert. Sie brauchen die Konvention nicht zu beachten, da aber sysinstall die Konvention beachtet, ist die Installation sauberer, wenn Sie das auch tun. Sie können das Dateisystem irgendwo einhängen. Das Beispiel schlägt die Verzeichnisse `/diskn` vor, wobei *n* die Laufwerke nummeriert. Sie können ein anderes Schema verwenden, wenn Sie möchten.

Wenn Sie die Aufteilung der Partitionen festgelegt haben, können Sie die Partitionen mit sysinstall anlegen. Es erscheint die nachstehende Meldung:

```
                              Message
Now, you need to create BSD partitions inside of the fdisk
partition(s) just created. If you have a reasonable amount of disk
space (1GB or more) and don't have any special requirements, simply
use the (A)uto command to allocate space automatically. If you have
more specific needs or just don't care for the layout chosen by
(A)uto, press F1 for more information on manual layout.

                                [  OK  -]
                          [ Press enter or space -]
```

Drücken Sie Enter, um den FreeBSD-Partitionseditor, der Disklabel heißt, zu starten.

Abbildung 2.18, „Sysinstall Disklabel-Editor“ zeigt den Einstiegsbildschirm von Disklabel. Der Bildschirm ist in drei Bereiche geteilt.

Die ersten Zeilen zeigen den Namen des Laufwerks, das Sie gerade bearbeiten und die Slice, die die erstellten Partitionen enthält (Disklabel spricht hier von Partitionen anstatt von Slices). Der freie Platz einer Slice, der noch keiner Partition zugeordnet ist, wird ebenfalls angezeigt.

In der Mitte des Bildschirms werden die angelegten Partitionen, der Name des Dateisystems, das sich in der Partition befindet, dessen Größe und die Optionen zum Erstellen des Dateisystems angezeigt.

Das untere Drittel des Bildschirms zeigt die in Disklabel gültigen Tastenkombinationen.

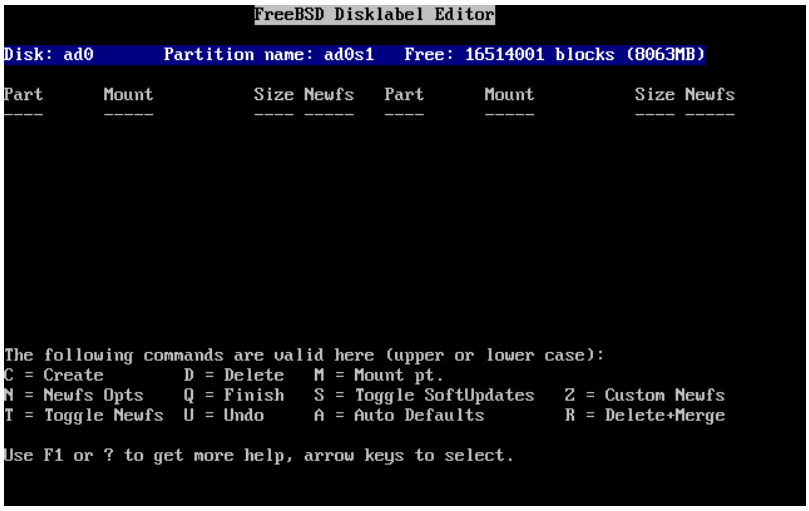


Abbildung 2.18. Sysinstall Disklabel-Editor

Disklabel kann für Sie automatisch Partitionen mit vorgegebenen Größen erstellen (diese Standardgrößen werden durch einen internen Partitionierungsalgorithmus ermittelt, der auf der Plattengröße beruht). Probieren Sie das bitte jetzt aus und drücken Sie die Taste A. Der Bildschirm sieht danach ähnlich wie in Abbildung 2.19, „Sysinstall Disklabel-Editor mit automatischen Vorgaben“ aus. Abhängig von der Größe des Laufwerks können die Vorgabewerte richtig oder falsch sein. Da Sie die Vorgaben nicht akzeptieren müssen, spielt das keine Rolle.

Anmerkung

FreeBSD legt das Verzeichnis `/tmp` in einer eigenen Partition an. Dies verhindert, dass sich die Root-Partition mit temporären Dateien füllt.

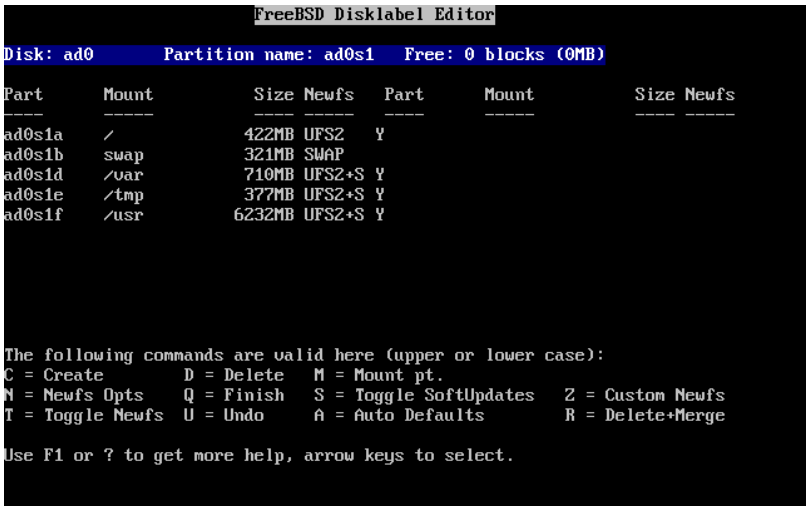


Abbildung 2.19. Sysinstall Disklabel-Editor mit automatischen Vorgaben

Wollen Sie die vorgegebenen Partitionen nicht verwenden und durch eigene ersetzen, markieren Sie mit den Pfeiltasten die erste Partition und drücken Sie die Taste D, um die Partition zu löschen. Wiederholen Sie dies für alle vorgegebenen Partitionen.

Um die erste Partition (a), die als / eingehangen wird, zu erstellen, drücken Sie die Taste C. Stellen Sie dabei sicher, dass die richtige Slice im oberen Teil des Bildschirms markiert ist. Wie in Abbildung 2.20, „Die Größe einer Partition

festlegen“, erscheint ein Fenster, in dem Sie die Größe der Partition angeben müssen. Sie können die Größe in Blöcken oder einer Zahl gefolgt von `M` für Megabyte, `G` für Gigabyte oder `C` für Zylinder angeben.

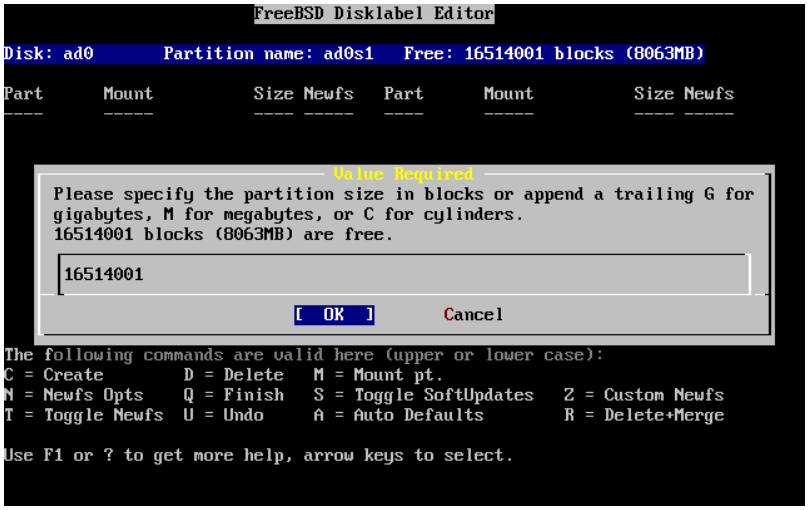


Abbildung 2.20. Die Größe einer Partition festlegen

Die vorgegebene Größe erstellt eine Partition, die den Rest der Slice ausfüllt. Wenn Sie die Größen aus dem früheren Beispiel verwenden, löschen Sie die vorgeschlagene Größe mit der Taste Backspace und tragen Sie **`512M`** ein, wie in Abbildung 2.21, „Die Größe einer Partition ändern“ gezeigt. Drücken Sie anschließend [OK].

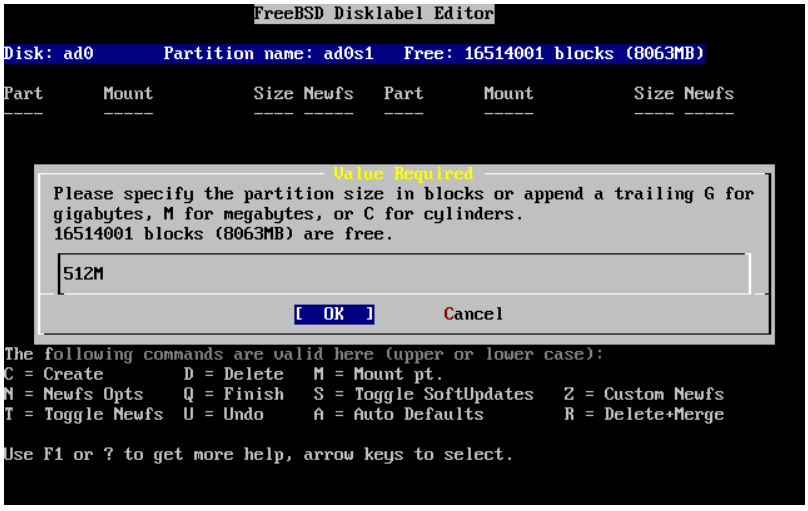


Abbildung 2.21. Die Größe einer Partition ändern

Nachdem Sie die Größe der Partition festgelegt haben, werden Sie gefragt, ob die Partition ein Dateisystem oder einen Auslagerungsbereich enthalten soll (siehe Abbildung 2.22, „Den Partitionstyp festlegen“). Die erste Partition enthält ein Dateisystem, wählen Sie FS aus und drücken Sie die Taste Enter.

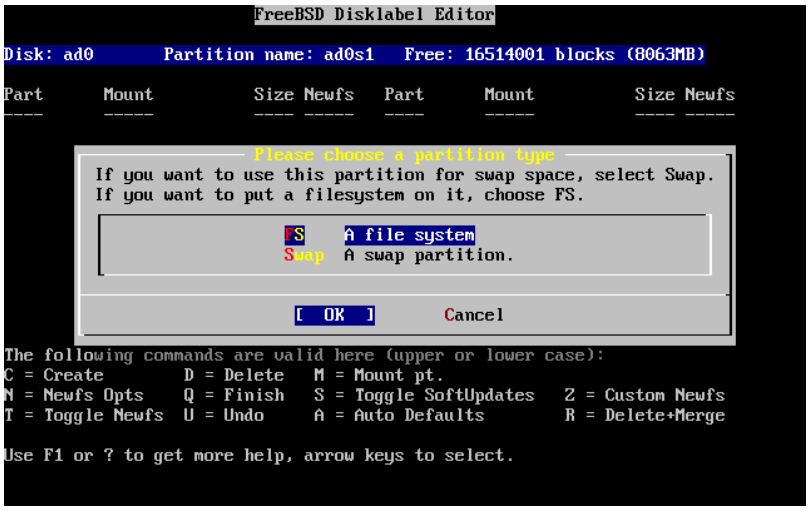


Abbildung 2.22. Den Partitionstyp festlegen

Abschließend müssen Sie, weil Sie ein Dateisystem erstellen, angeben, wo das Dateisystem eingehangen wird. Die Eingabe ist in Abbildung 2.23, „Den Mountpoint festlegen“ dargestellt. Das Root-Dateisystem wird in `/` eingehangen, geben Sie daher `/` ein und drücken Sie die Taste Enter.

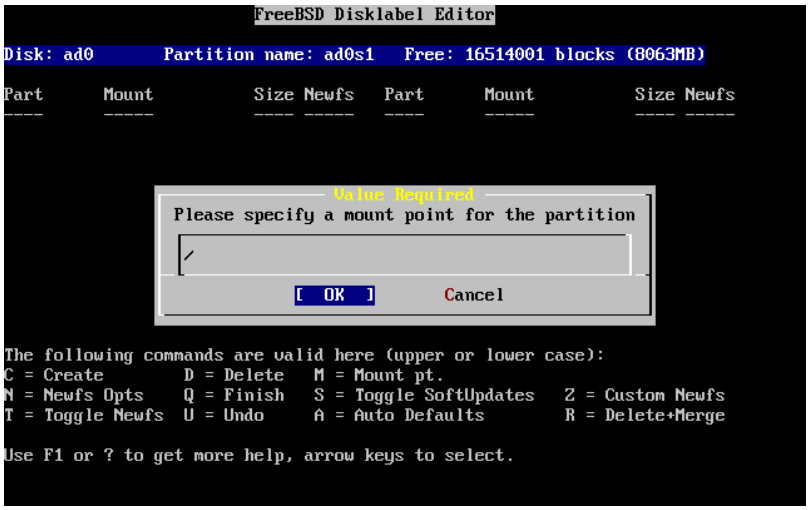


Abbildung 2.23. Den Mountpoint festlegen

Auf dem Bildschirm wird jetzt die neu angelegte Partition angezeigt. Wiederholen Sie diese Prozedur für die restlichen Partitionen. Beim Anlegen des Auslagerungsbereichs werden Sie nicht nach einem Mountpoint gefragt, da ein Auslagerungsbereich nie eingehangen wird. Wenn Sie die letzte Partition anlegen, `/usr`, können Sie die vorgeschlagene Größe stehen lassen. Das Dateisystem wird dann den Rest der Slice einnehmen.

Der letzte Bildschirm von Disklabel sieht wie in Abbildung 2.24, „Sysinstall Disklabel-Editor“ aus (Ihre Werte werden von den gezeigten Werten abweichen). Drücken Sie die Taste Q, um Disklabel zu verlassen.

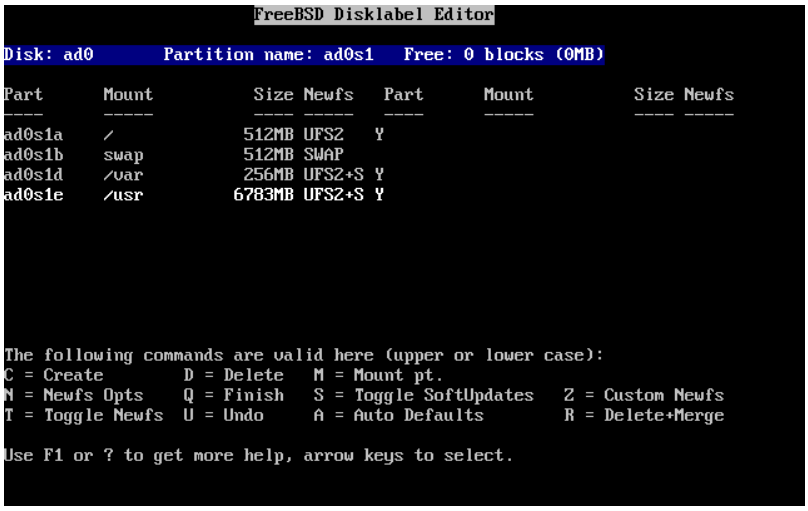


Abbildung 2.24. Sysinstall Disklabel-Editor

2.7. Den Installationsumfang bestimmen

2.7.1. Die Distribution auswählen

Welche Software Sie installieren, hängt hauptsächlich vom Zweck des Rechners und dem zur Verfügung stehenden Plattenplatz ab. Die vorgegebenen Distributionen reichen von der minimalen Installation bis hin zu einer kompletten Installation. Anfänger sollten eine der vorgegebenen Distributionen auswählen, erfahrene Benutzer können die zu installierende Distribution anpassen.

Die Taste F1 führt zu einem Hilfebildschirm, der die Distributionen und deren Inhalte beschreibt. Drücken Sie Enter, um die Hilfe zu verlassen und zur Auswahl der Distribution zurückzukehren.

Wenn Sie eine graphische Benutzeroberfläche installieren wollen, müssen Sie die Konfiguration des X-Servers und die Auswahl der Benutzeroberfläche nach erfolgreicher Installation durchführen. Die Installation und Konfiguration des X-Servers wird in Kapitel 6, *Das X-Window-System* besprochen.

Wenn Sie einen angepassten Kernel erstellen wollen, wählen Sie eine Distribution aus, die den Quellcode (*source code*) enthält. Warum und wie Sie einen angepassten Kernel erstellen, erfahren Sie in Kapitel 9, *Konfiguration des FreeBSD-Kernels*.

Natürlich ist das flexibelste System das, auf dem alles installiert ist. Wenn das System über ausreichend Plattenplatz verfügt, wählen Sie mit den Pfeiltasten die Option All aus (siehe Abbildung 2.25, „Die Distribution auswählen“) und drücken die Taste Enter. Wenn Sie Bedenken haben, dass der Plattenplatz nicht ausreicht, wählen Sie eine Distribution, die weniger Software enthält. Machen Sie sich keine unnötigen Sorgen um die richtige Distribution, ausgelassene Distribution können später nachinstalliert werden.

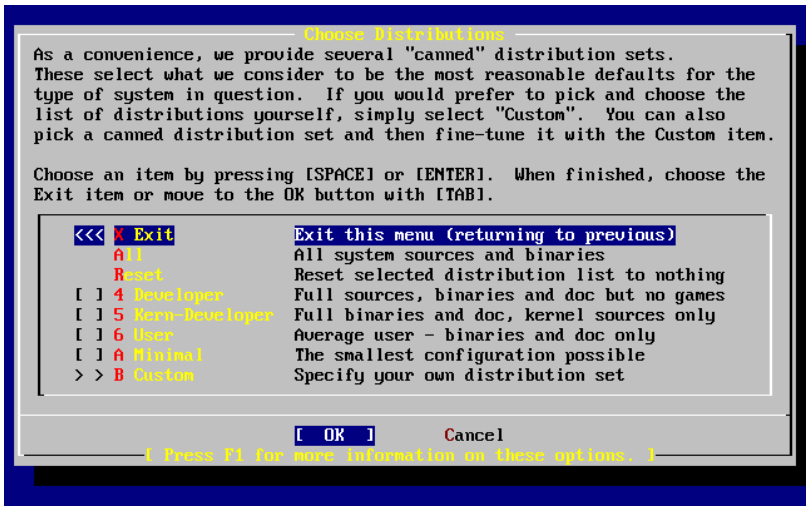


Abbildung 2.25. Die Distribution auswählen

2.7.2. Die Ports-Sammlung installieren

Nach der Auswahl der Distribution haben Sie Gelegenheit, die FreeBSD-Ports-Sammlung zu installieren. Mit der Ports-Sammlung lässt sich Software Dritter auf einfache Art und Weise installieren. Der Quellcode der zu installierenden Software ist nicht in der Ports-Sammlung enthalten. Stattdessen enthält die Ports-Sammlung Dateien, die den Installationsprozess (herunterladen, übersetzen und installieren) automatisieren. Die Ports-Sammlung wird in Kapitel 5, *Installieren von Anwendungen: Pakete und Ports* besprochen.

Der Installationsprozess prüft nicht, ob ausreichend Platz für die Ports-Sammlung vorhanden ist. Wählen Sie die Ports-Sammlung bitte nur aus, wenn das System über ausreichenden Platz verfügt. In FreeBSD 10.3 nimmt die Ports-Sammlung ungefähr 500 MB Plattenplatz in Anspruch. Neuere Versionen von FreeBSD benötigen mit Sicherheit noch mehr Platz.

```
                        User Confirmation Requested
Would you like to install the FreeBSD ports collection?

This will give you ready access to over 24,000 ported software packages,
at a cost of around 500 MB of disk space when "clean" and possibly much
more than that if a lot of the distribution tarballs are loaded
(unless you have the extra CDs from a FreeBSD CD/DVD distribution
available and can mount it on /cdrom, in which case this is far less
of a problem).

The ports collection is a very valuable resource and well worth having
on your /usr partition, so it is advisable to say Yes to this option.

For more information on the ports collection & the latest ports,
visit:
    http://www.FreeBSD.org/ports

                              [ Yes -]     No
```

Wählen Sie mit den Pfeiltasten [Yes] aus, um die Ports-Sammlung zu installieren. Wählen Sie [No] aus, um die Ports-Sammlung auszulassen. Drücken Sie danach die Taste Enter, es erscheint wieder das Distributionsmenü.

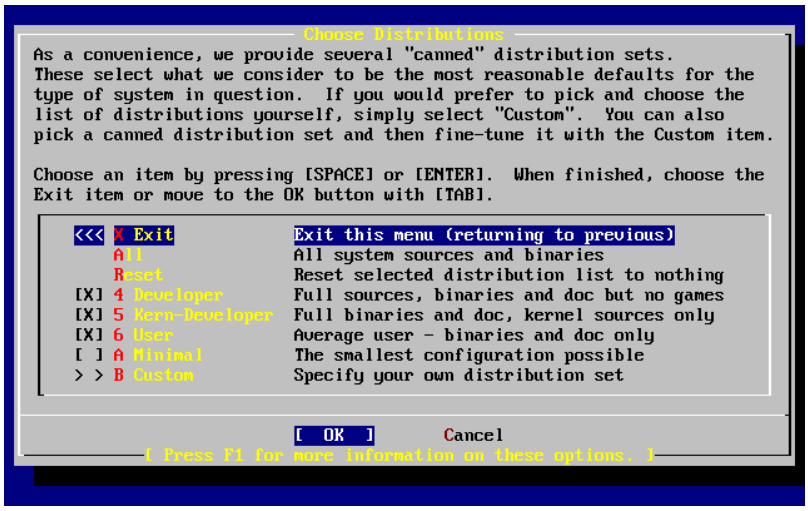


Abbildung 2.26. Die Distributionen bestätigen

Wenn Sie mit den ausgewählten Optionen zufrieden sind, wählen Sie mit den Pfeiltasten Exit aus (stellen Sie sicher, dass [OK] aktiv ist) und drücken Sie die Taste Enter.

2.8. Das Installationsmedium auswählen

Wenn Sie von einer CD-ROM oder einer DVD installieren, wählen Sie bitte Install from a FreeBSD CD/DVD aus. Stellen Sie sicher, dass [OK] aktiv ist und drücken Sie dann die Taste Enter, um mit der Installation fortzufahren.

Wenn Sie ein anderes Installationsmedium benutzen, wählen Sie die passende Option aus und folgen den angezeigten Anweisungen.

Die Hilfeseiten über Installationsmedien erreichen Sie mit der Taste F1. Drücken Sie Enter, um zur Auswahl des Installationsmediums zurückzukehren.

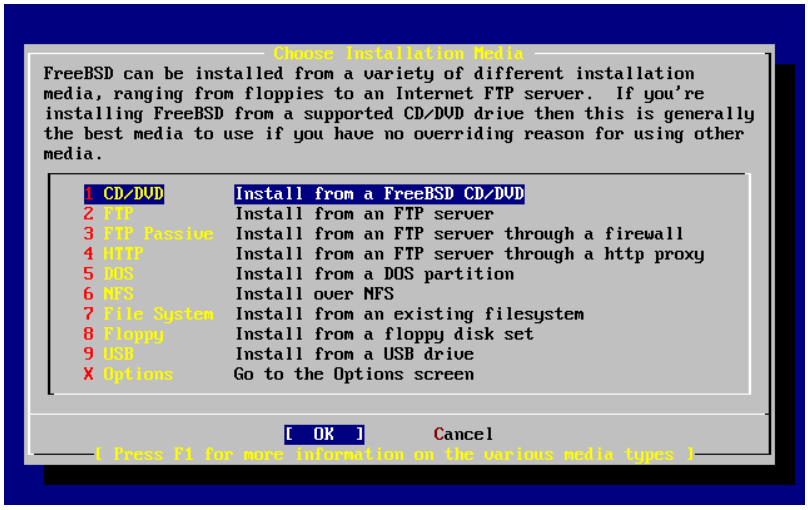


Abbildung 2.27. Das Installationsmedium auswählen

FTP-Installationsmodi

Sie können zwischen drei FTP-Installationsmodi wählen: Active-FTP, Passive-FTP oder über einen HTTP-Proxy.

FTP Active: Install from an FTP server
: Diese Option führt alle FTP-Operationen im Active-Mode aus. Dieser Modus funktioniert nicht durch Firewalls, er funktioniert aber mit alten FTP-Servern, die den Passive-Mode nicht beherrschen. Wenn die Verbindung im Passive-Mode (das ist die Vorgabe) hängt, versuchen Sie den Active-Mode.

FTP Passive: Install from an FTP server through a firewall
: Mit dieser Option benutzt sysinstall Passive-Mode für alle FTP-Operationen. In diesem Modus funktionieren Verbindungen durch Firewalls, die einkommende Pakete auf beliebigen TCP-Ports blockieren.

FTP via a HTTP proxy: Install from an FTP server through a http proxy
: Diese Option weist sysinstall an, alle FTP-Operationen mit HTTP über einen Proxy (wie ein Web-Browser) durchzuführen. Der Proxy leitet die Anfragen an den richtigen FTP-Server weiter. Mit dieser Option passieren Sie eine Firewall, die FTP-Verbindungen verbietet, aber einen HTTP-Proxy anbietet. Neben dem FTP-Server müssen Sie in diesem Fall den Proxy-Server angeben.

Bei einem FTP-Proxy-Server müssen Sie normalerweise den Ziel-FTP-Server als Teil des Benutzernamens hinter dem Klammeraffen („@“) angeben. Der Proxy-Server übernimmt die Kommunikation mit dem Ziel-FTP-Server. Nehmen wir an, Sie wollen von `ftp.FreeBSD.org` über den FTP-Proxy `foo.example.com` auf Port `1234` installieren.

Wählen Sie das Menü Options aus und setzen Sie dort den FTP-Benutzernamen (*username*) auf `ftp@ftp.FreeBSD.org`. Als Passwort geben Sie bitte Ihre E-Mail-Adresse an. Setzen Sie das Installationsmedium auf Active-FTP oder Passive-FTP, je nachdem welchen Modus der Proxy-Server unterstützt. Für die URL geben Sie `ftp://foo.example.com:1234/pub/FreeBSD` an.

Der Proxy-Server `foo.example.com` leitet Zugriffe auf das Verzeichnis `/pub/FreeBSD` an den Server `ftp.FreeBSD.org` weiter. Daher können `foo.example.com` als FTP-Server angeben.

2.9. Die Installation festschreiben

Wenn Sie wünschen, kann die Installation nun beginnen. Dies ist die letzte Gelegenheit, die Installation abzubrechen und Änderungen auf der Festplatte zu vermeiden.

```
                      User Confirmation Requested
Last Chance! Are you SURE you want to continue the installation?

If you're running this on a disk with data you wish to save then WE
STRONGLY ENCOURAGE YOU TO MAKE PROPER BACKUPS before proceeding!

We can take no responsibility for lost disk contents!

                             [ Yes -]    No
```

Wählen Sie [Yes] aus und drücken Sie Enter, um weiter zu machen.

Die Installationsdauer hängt von den ausgewählten Distributionen, dem Installationsmedium und der Geschwindigkeit des Rechners ab. Während der Installation wird der Fortgang mit Statusmeldungen angezeigt.

Die Installation ist beendet, wenn die folgende Meldung erscheint:

```
                                Message

Congratulations! You now have FreeBSD installed on your system.

We will now move on to the final configuration questions.
For any option you do not wish to configure, simply select No.

If you wish to re-enter this utility after the system is up, you may
do so by typing: /usr/sbin/sysinstall.

                                  [ OK -]

                       [  Press enter or space  -]
```

Drücken Sie die Taste Enter, um die Nacharbeiten durchzuführen.

Wenn Sie [No] auswählen und Enter drücken wird die Installation abgebrochen und das System wird nicht verändert. Die nachstehende Meldung wird angezeigt:

```
                                Message
Installation complete with some errors.  You may wish to scroll
through the debugging messages on VTY1 with the scroll-lock feature.
You can also choose "No" at the next prompt and go back into the
installation menus to retry whichever operations have failed.

                                  [ OK -]
```

Die Meldung wird angezeigt, weil nichts installiert wurde. Drücken Sie Enter, um in das Hauptmenü zurückzukehren. Dort können Sie die Installationsprozedur verlassen.

2.10. Arbeiten nach der Installation

Nach einer erfolgreichen Installation wird das System konfiguriert. Sie können das System direkt konfigurieren oder nach einem Neustart. Nach einem Neustart rufen Sie `sysinstall` auf und wählen den Menüpunkt Configure.

2.10.1. Netzwerkkonfiguration

Wenn Sie schon PPP für eine FTP-Installation konfiguriert haben, erscheint dieser Bildschirm nicht. Sie können die Konfiguration später in sysinstall vornehmen.

Netzwerke und die Konfiguration von FreeBSD als Gateway oder Router werden eingehend im Kapitel Weiterführende Netzwerkthemen behandelt.

```
                      User Confirmation Requested
   Would you like to configure any Ethernet or PPP network devices?

                              [ Yes -]   No
```

Wenn Sie eine Netzwerkkarte konfigurieren wollen, wählen Sie [Yes] aus und drücken Sie die Taste Enter. Wählen Sie [No], um die Netzwerkkonfiguration zu überspringen.

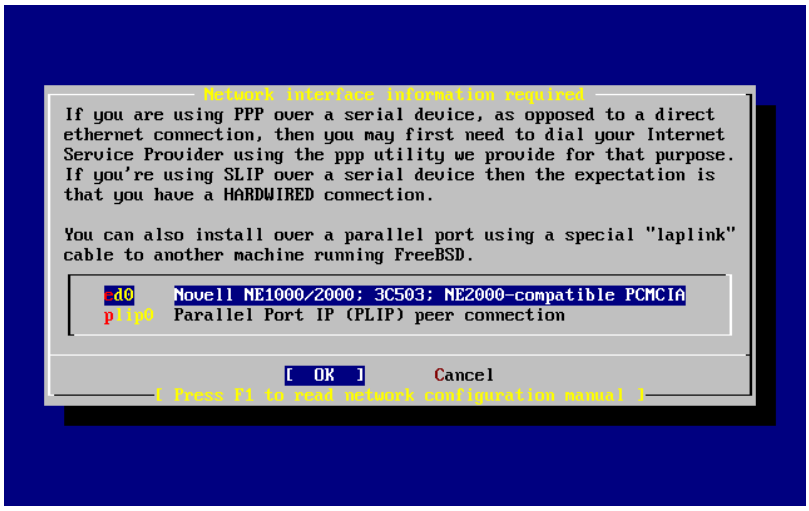


Abbildung 2.28. Eine Netzwerkkarte auswählen

Wählen Sie die zu konfigurierende Karte mit den Pfeiltasten aus und drücken Sie die Taste Enter.

```
                    User Confirmation Requested
      Do you want to try IPv6 configuration of the interface?

                              Yes    [ No -]
```

Für das gezeigte Installationsbeispiel genügte das momentan verwendete Internet-Protokoll (IPv4). Daher wurde mit den Pfeiltasten [No] ausgewählt und mit der Taste Enter bestätigt.

Wenn Sie durch einen RA-Server mit einem IPv6-Netzwerk verbunden sind, wählen Sie bitte [Yes] und drücken die Taste Enter. Die Suche nach den RA-Servern dauert einige Sekunden.

```
                            User Confirmation Requested
       Do you want to try DHCP configuration of the interface?

                              Yes    [ No -]
```

Falls Sie das Dynamic Host Configuration Protocol (DHCP) nicht verwenden, wählen Sie [No] aus und drücken Sie Enter.

Wenn Sie [Yes] auswählen, wird das Programm dhclient ausgeführt und bei Erfolg die Netzwerkkarte konfiguriert. Mehr über DHCP können Sie in Abschnitt 29.6, „Dynamic Host Configuration Protocol (DHCP)“ nachlesen.

Der nächste Bildschirmabzug zeigt die Netzwerkkonfiguration eines Systems, das Gateway für das lokale Netz ist.

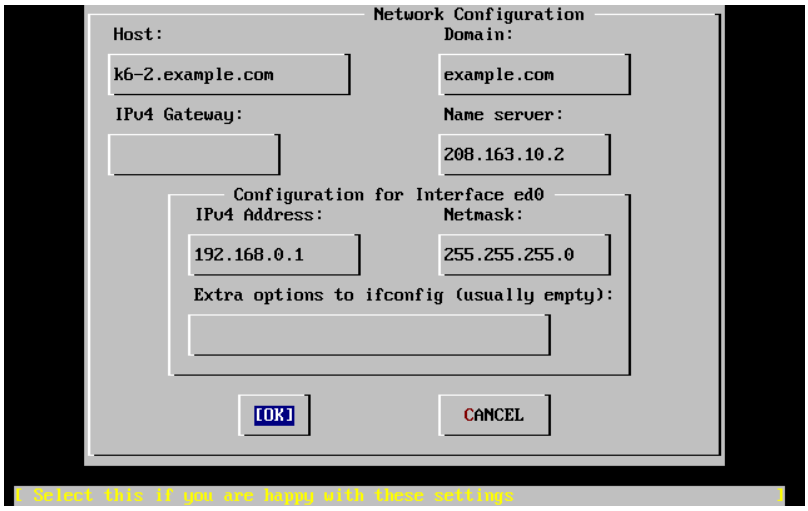


Abbildung 2.29. Die Netzwerkkarte ed0 konfigurieren

Tragen Sie in die Felder, die Sie mit der Taste Tab auswählen können, die richtige Konfiguration ein.

Host
: Der vollständige Rechnername (*fully-qualified hostname*), wie in diesem Beispiel `k6-2.example.com`.

Domain
: Der Domain-Name, in dem sich der Rechner befindet. Im Beispiel ist das `example.com`.

IPv4 Gateway
: Die IP-Adresse des Rechners, der Pakete an entfernte Netze weiterleitet. Sie müssen dieses Feld ausfüllen, wenn der sich der Rechner in einem Netzwerk befindet. *Lassen Sie das Feld leer*, wenn der Rechner der Gateway in das Internet ist. Der IPv4-Gateway wird auch *default gateway* oder *default route* genannt.

Name server
: Die IP-Adresse des lokalen DNS-Servers. Im Beispiel gibt es keinen lokalen DNS-Server, daher wurde der DNS-Server des Providers (`208.163.10.2`) benutzt.

IPv4 address
: Die IP-Adresse der Netzwerkkarte (`192.168.0.1`).

Netmask (Netzmaske)
: Im Beispiel werden Adressen aus einem Klasse C Netz (`192.168.0.0` bis `192.168.0.255`) benutzt. Standardmäßig besitzt ein Klasse C Netz die Netzmaske `255.255.255.0` .

Extra options to ifconfig (Optionen für ifconfig)
: Zusätzliche Optionen für den Befehl `ifconfig`, die spezifisch für die verwendete Netzwerkkarte sind. Im Beispiel sind keine Optionen angegeben.

Wenn Sie alle Werte eingegeben haben, wählen Sie mit Tab [OK] aus und drücken Sie Enter.

```
                  User Confirmation Requested
       Would you like to bring the ed0 interface up right now?

                           [ Yes -]   No
```

Wenn Sie [Yes] auswählen und Enter drücken, wird die Netzwerkkonfiguration aktiviert. Allerdings bringt dies zu diesem Zeitpunkt nicht viel, da der Rechner noch neu gestartet werden muss.

2.10.2. Gateway einrichten

```
                        User Confirmation Requested
```

```
       Do you want this machine to function as a network gateway?

                                [ Yes -]    No
```

Wählen Sie [Yes], wenn der Rechner ein Gateway für ein lokales Netz ist und Pakete an andere Netze weiterleitet. Wenn der Rechner ein normaler Netzknoten ist, wählen Sie [No] aus. Bestätigen Sie die auswahl mit der Taste Enter.

2.10.3. IP-Dienste einrichten

```
                        User Confirmation Requested
Do you want to configure inetd and the network services that it provides?

                                 Yes   [ No -]
```

Wenn [No] ausgewählt wird, werden Dienste wie telnetd nicht aktiviert. Benutzer können sich dann von entfernten Rechnern nicht mit telnet an dieser Maschine anmelden. Lokale Benutzer können aber auf entfernte Rechner mit telnet zugreifen.

Die Dienste können Sie nach der Installation aktivieren, indem Sie die Datei `/etc/inetd.conf` editieren. Dies wird in Abschnitt 29.2.1, „Überblick“ beschrieben.

Wenn Sie jetzt weitere Dienste aktivieren möchten, wählen Sie [Yes] aus. Es erscheint die nachstehende Rückfrage:

```
                        User Confirmation Requested
The Internet Super Server (inetd) allows a number of simple Internet
services to be enabled, including finger, ftp and telnetd.  Enabling
these services may increase risk of security problems by increasing
the exposure of your system.

With this in mind, do you wish to enable inetd?

                               [ Yes -]   No
```

Bestätigen Sie die Rückfrage mit [Yes].

```
                        User Confirmation Requested
inetd(8) relies on its configuration file, /etc/inetd.conf, to determine
which of its Internet services will be available.  The default FreeBSD
inetd.conf(5) leaves all services disabled by default, so they must be
specifically enabled in the configuration file before they will
function, even once inetd(8) is enabled.  Note that services for
IPv6 must be separately enabled from IPv4 services.

Select [Yes] now to invoke an editor on /etc/inetd.conf, or [No] to
use the current settings.

                               [ Yes -]   No
```

Wenn Sie [Yes] auswählen, können Sie Dienste aktivieren, in dem Sie das Zeichen # am Zeilenanfang entfernen.

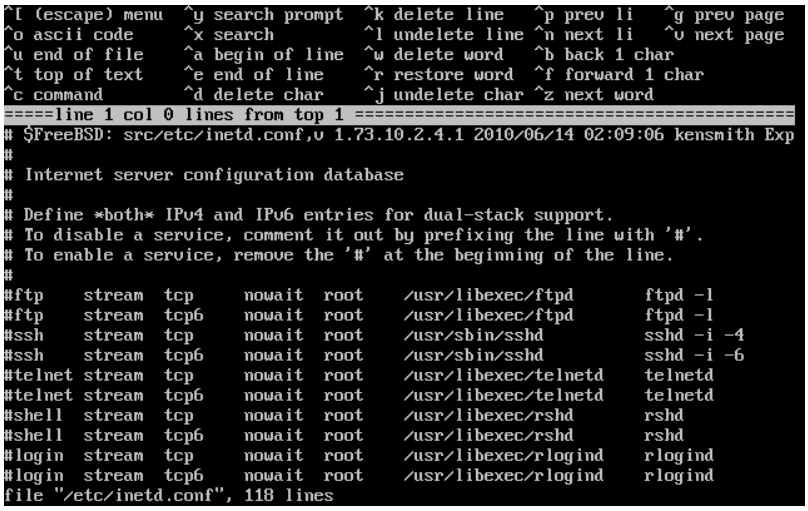
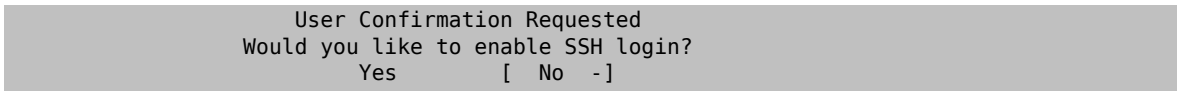


Abbildung 2.30. `inetd.conf` editieren

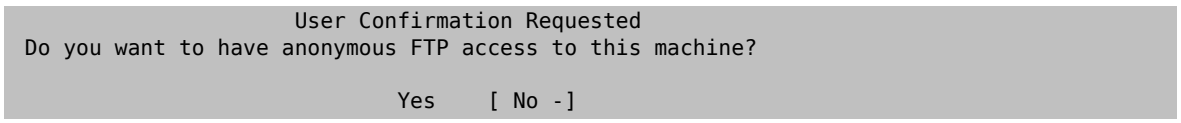
Wenn Sie die gewünschten Dienste aktiviert haben, drücken Sie die Taste Esc. Es erscheint ein Menü, in dem Sie die Änderungen abspeichern und den Editor verlassen können.

2.10.4. SSH aktivieren



Durch die Auswahl von [Yes], wird sshd(8), der OpenSSH-Daemon aktiviert. Danach ist es möglich, sich über eine verschlüsselte Verbindung auf Ihrem System anzumelden. Weitere Informationen über OpenSSH finden Sie in Abschnitt 14.10, „OpenSSH“ des FreeBSD-Handbuchs.

2.10.5. Anonymous-FTP



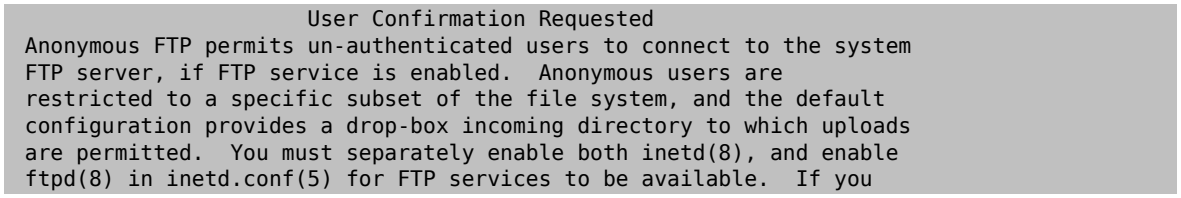
2.10.5.1. Anonymous-FTP verbieten

Wenn Sie die vorgegebene Auswahl [No] mit der Taste Enter bestätigen, können Benutzer, die ein Konto und ein Passwort auf dem System besitzen, immer noch mit FTP auf das System zugreifen.

2.10.5.2. Anonymous-FTP erlauben

Wenn Sie Anonymous-FTP erlauben, darf jeder auf Ihr System zugreifen. Bedenken Sie die Folgen für die Systemsicherheit (siehe Kapitel 14, *Sicherheit*) bevor Sie diese Option aktivieren.

Um Anonymous-FTP zu aktivieren, wählen Sie mit den Pfeiltasten [Yes] aus und drücken Sie die Taste Enter. Es erscheint folgende Meldung:



```
did not do so earlier, you will have the opportunity to enable inetd(8)
again later.

If you want the server to be read-only you should leave the upload
directory option empty and add the -r command-line option to ftpd(8)
in inetd.conf(5)

Do you wish to continue configuring anonymous FTP?

                          [ Yes -]         No
```

Diese Nachricht informiert Sie darüber, dass der FTP-Dienst auch in der Datei `/etc/inetd.conf` aktiviert werden muss, wenn Sie anonyme FTP-Verbindungen erlauben wollen (lesen Sie dazu auch Abschnitt 2.10.3, „IP-Dienste einrichten“ des FreeBSD-Handbuchs). Wählen Sie [Yes] und drücken Sie Enter, um fortzufahren. Danach erscheint der folgende Bildschirm:

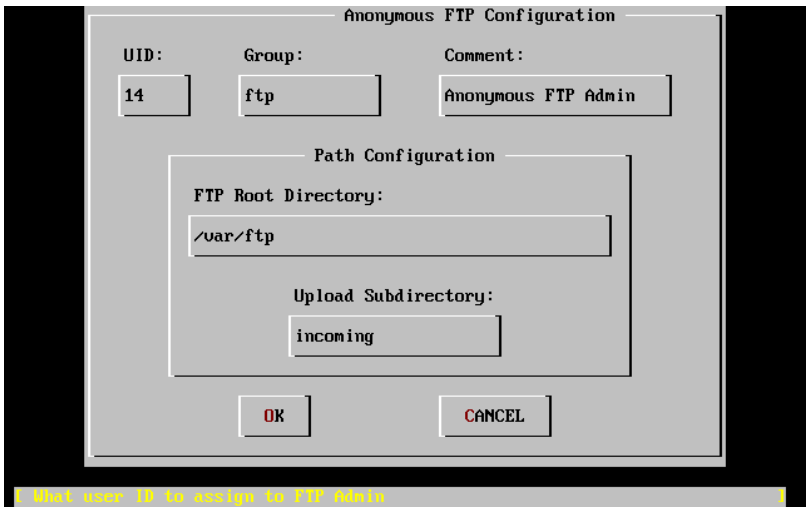


Abbildung 2.31. Anonymous-FTP konfigurieren

Mit der Taste Tab wechseln Sie zwischen den Feldern, in die Sie die benötigten Informationen eingeben.

UID
: Die User-ID, die dem anonymen FTP-Benutzer zugewiesen werden soll. Alle hochgeladenen Dateien werden diesem User-ID gehören.

Group
: Die Gruppe, zu der der anonyme FTP-Benutzer gehören soll.

Comment
: Eine Beschreibung dieses Benutzers in der Datei `/etc/passwd` .

FTP Root Directory
: Ort, an dem Dateien für anonymen FTP-Zugang bereitgestellt werden sollen.

Upload Subdirectory
: Das Verzeichnis, in dem von einem anonymen FTP-Benutzer hochgeladene Dateien gespeichert werden.

Das FTP-Wurzelverzeichnis wird per Voreinstellung in `/var` angelegt. Wenn in `/var` zu wenig Platz vorhanden ist, können Sie das FTP-Wurzelverzeichnis beispielsweise nach `/usr/ftp` verlegen.

Wenn Sie mit den Einstellungen zufrieden sind, drücken Sie die Taste Enter.

```
                        User Confirmation Requested
```

```
          Create a welcome message file for anonymous FTP users?

                                  [ Yes -]    No
```

Wenn Sie [Yes] auswählen und mit Enter bestätigen, können Sie die Begrüßungsmeldung des FTP-Servers in einem Editor ändern.

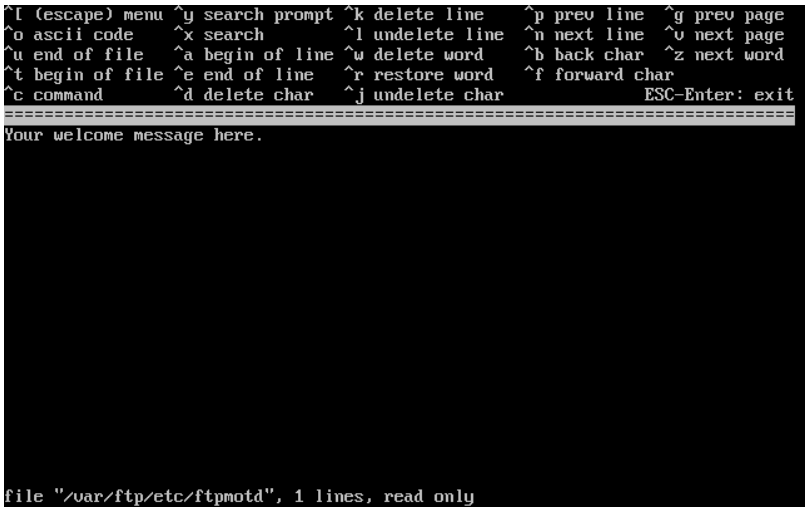


Abbildung 2.32. Begrüßungsmeldung des FTP-Servers editieren

Der Editor, in dem Sie sich befinden, heißt ee. Folgen Sie den Anweisungen, um die Meldung zu editieren. Sie können die Meldung auch später in einem Editor Ihrer Wahl editieren. Merken Sie sich dazu den Dateinamen, der im Editor unten angezeigt wird.

Wenn Sie die Taste Esc drücken, erscheint ein Menü, in dem a) leave editor vorgewählt ist. Drücken Sie die Taste Enter, um den Editor zu verlassen. Falls Sie Änderungen vorgenommen haben, bestätigen Sie die Änderungen nochmals mit Enter.

2.10.6. Network-File-System einrichten

Mit dem Network-File-System (NFS) können Sie über ein Netzwerk auf Dateien zugreifen. Ein Rechner kann NFS-Server, NFS-Client oder beides sein. NFS wird in Abschnitt 29.3, „Network File System (NFS)“ besprochen.

2.10.6.1. NFS-Server einrichten

```
                        User Confirmation Requested
 Do you want to configure this machine as an NFS server?

                                 Yes    [ No -]
```

Wenn Sie keinen NFS-Server benötigen, wählen Sie [No] aus und bestätigen Sie mit Enter.

Wenn Sie [Yes] auswählen, erscheint der Hinweis, dass die Datei `exports` angelegt werden muss.

```
                                Message
Operating as an NFS server means that you must first configure an
/etc/exports file to indicate which hosts are allowed certain kinds of
access to your local filesystems.
Press [Enter] now to invoke an editor on /etc/exports
                                [ OK -]
```

Drücken Sie Enter und es wird ein Editor gestartet, in dem Sie die Datei `exports` editieren können.

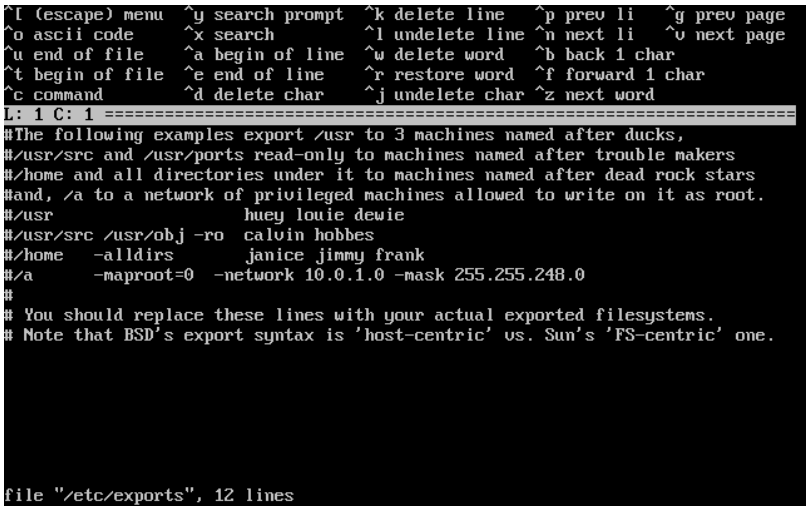


Abbildung 2.33. `exports` editieren

Folgen Sie den Anweisungen, um Dateisysteme zu exportieren. Sie können die Datei auch später in einem Editor Ihrer Wahl editieren. Merken Sie sich dazu den Dateinamen, der im Editor unten angezeigt wird.

Drücken Sie die Taste Esc und es erscheint ein Menü, in dem a) leave editor vorgewählt ist. Drücken Sie die Taste Enter, um den Editor zu verlassen.

2.10.6.2. NFS-Client einrichten

Mit einem NFS-Client können Sie auf NFS-Server zugreifen.

```
                      User Confirmation Requested
 Do you want to configure this machine as an NFS client?

                                Yes   [ No -]
```

Wählen Sie entweder [Yes] oder [No] aus und drücken Sie Enter.

2.10.7. Die Systemkonsole einrichten

Sie können verschiedene Merkmale der Systemkonsole anpassen.

```
                     User Confirmation Requested
       Would you like to customize your system console settings?

                                [ Yes -]  No
```

Wenn Sie die Merkmale der Systemkonsole anpassen wollen, wählen Sie [Yes] aus und drücken Sie die Taste Enter.

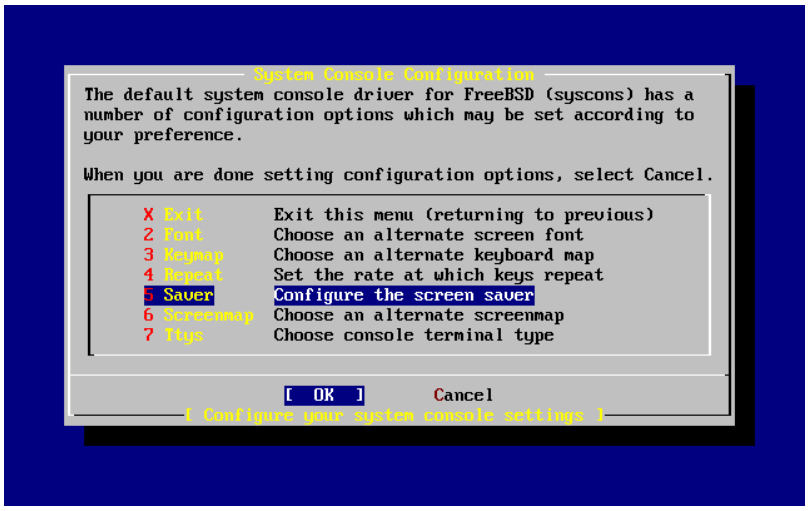


Abbildung 2.34. Merkmale der Systemkonsole

Oft wird ein Bildschirmschoner auf der Konsole aktiviert. Wälen Sie mit den Pfeiltasten Saver aus und drücken Sie die Taste Enter.

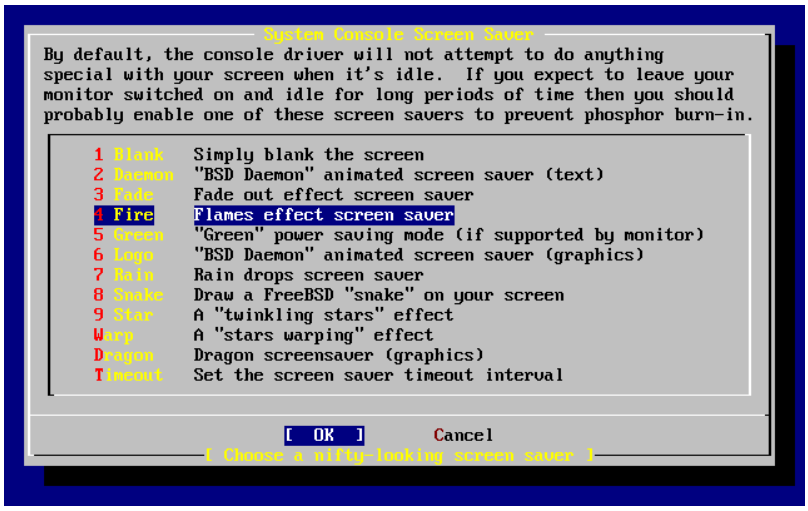


Abbildung 2.35. Bildschirmschoner auswählen

Wählen Sie den gewünschten Bildschirmschoner mit den Pfeiltasten aus und drücken Sie Enter. Das Konfigurationsmenü der Systemkonsole erscheint wieder.

In der Voreinstellung wird der Bildschirmschoner nach 300 Sekunden aktiviert. Um diese Zeitspanne zu ändern, wählen Sie wieder Saver aus. Mit den Pfeiltasten wählen Sie dann Timeout aus und drücken Enter. Es erscheint ein Eingabefenster:

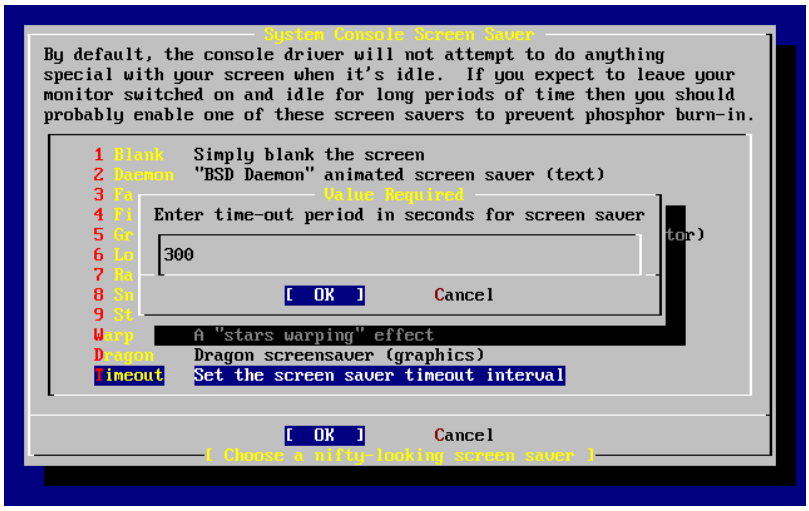


Abbildung 2.36. Den Bildschirmschoner einstellen

Ändern Sie die Zeitspanne und wählen Sie [OK] aus. Mit Enter kehren Sie in das Konfigurationsmenü der Systemkonsole zurück.

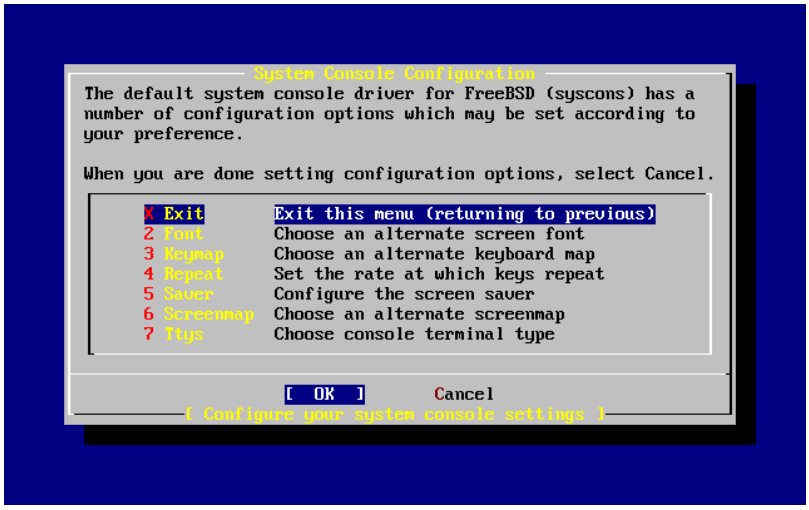


Abbildung 2.37. Die Konfiguration der Systemkonsole verlassen

Um die Nacharbeiten fortzuführen, wählen Sie Exit aus und drücken Sie Enter.

2.10.8. Die Zeitzone einstellen

Wenn Sie die Zeitzone richtig einstellen, kann Ihr Rechner automatisch regional bedingte Zeitumstellungen ausführen und andere von der Zeitzone abhängige Funktionen handhaben.

Das folgende Beispiel gilt für den Osten der USA. Ihre Auswahl hängt vom geographischen Standort Ihres Rechners ab.

```
                    User Confirmation Requested
         Would you like to set this machine's time zone now?

                              [ Yes -]   No
```

Um die Zeitzone einzustellen, wählen Sie [Yes] und drücken Enter.

```
                    User Confirmation Requested
```

```
Is this machine's CMOS clock set to UTC? If it is set to local time
or you don't know, please choose NO here!

                              Yes    [ No -]
```

Je nachdem ob die Systemzeit die Zeitzone UTC verwendet, wählen Sie [Yes] oder [No] aus. Bestätigen Sie die Auswahl mit der Taste Enter.

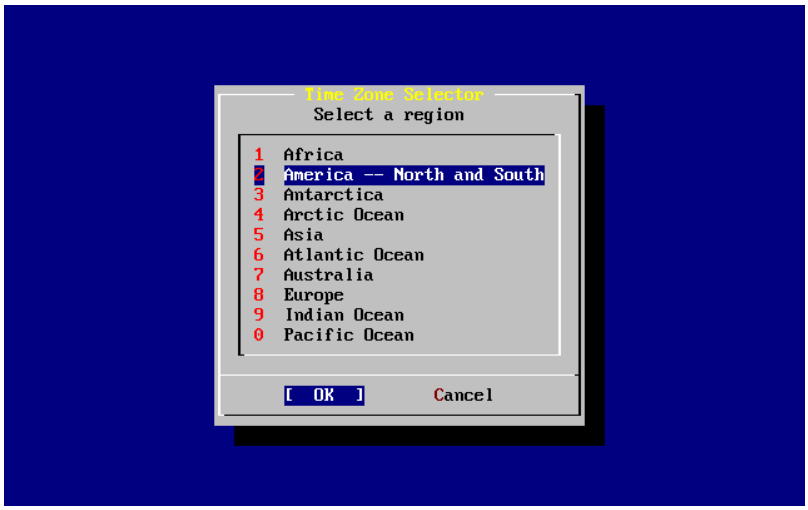


Abbildung 2.38. Das Gebiet auswählen

Wählen Sie mit den Pfeiltasten das richtige Gebiet aus und drücken Sie Enter.

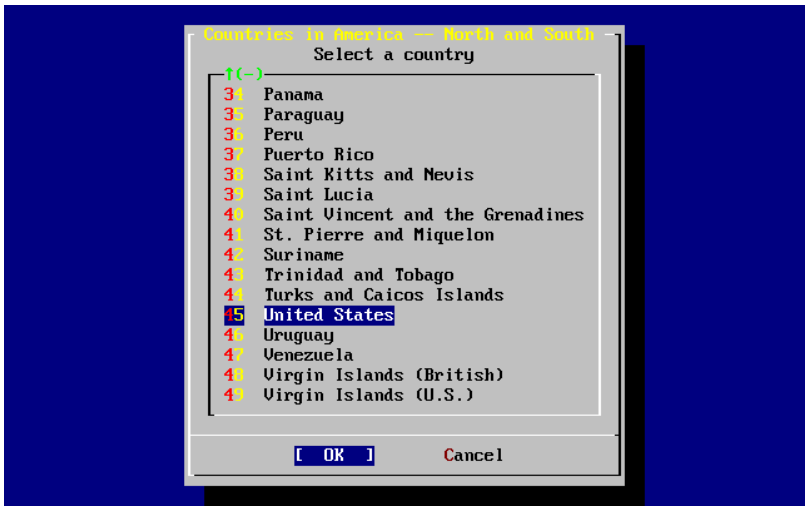


Abbildung 2.39. Das Land auswählen

Wählen Sie mit den Pfeiltasten das richtige Land aus und drücken Sie Enter.

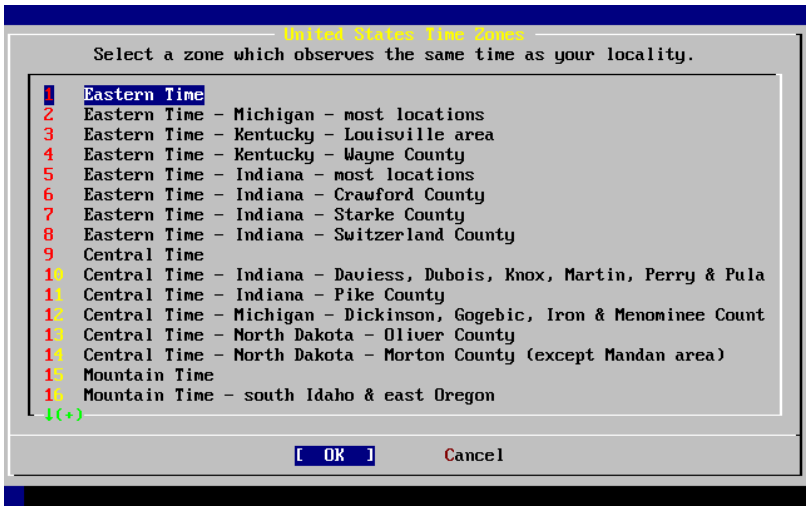


Abbildung 2.40. Die Zeitzone auswählen

Wählen Sie mit den Pfeiltasten die richtige Zeitzone aus drücken Sie Enter.

```
                        Confirmation
          Does the abbreviation 'EDT' look reasonable?

                        [ Yes -]   No
```

Wenn die angezeigte Abkürzung der Zeitzone richtig ist, bestätigen Sie diese mit der Taste Enter.

2.10.9. Linux-Kompatibilität

Anmerkung

Die folgenden Anweisungen sind nur für FreeBSD 7.X gültig. Installieren Sie eine FreeBSD 8.X-Version, wird der folgende Bildschirm nicht angezeigt.

```
                  User Confirmation Requested
        Would you like to enable Linux binary compatibility?

                        [ Yes -]   No
```

Wenn Sie [Yes] auswählen und Enter drücken, können Sie Linux-Software auf FreeBSD laufen lassen. Später wird dazu die notwendige Software installiert.

Wenn Sie über FTP installieren, müssen Sie mit dem Internet verbunden sein. Einige FTP-Server bieten nicht alle verfügbare Software an. Es kann sein, dass die nötige Software für die Linux-Kompatibilität nicht installiert werden kann, dies können Sie später jedoch nachholen.

2.10.10. Die Maus konfigurieren

Mit einer 3-Tasten-Maus können Sie Texte auf der Konsole und in Programmen markieren und einfügen (*cut and paste*). Wenn Sie eine 2-Tasten-Maus besitzen, können Sie eine 3-Tasten-Maus emulieren. Lesen Sie dazu nach der Installation die Hilfeseite moused(8). Das folgende Beispiel zeigt die Konfiguration einer nicht-USB-Maus (PS/2 oder serielle Maus):

```
                  User Confirmation Requested
        Does this system have a PS/2, serial, or bus mouse?
```

```
                              [ Yes -]    No
```

Wählen Sie [Yes] für eine PS/2-, eine serielle oder eine Bus-Maus. Haben Sie hingegen eine USB-Maus, wählen Sie [No]. Danach drücken Sie Enter.

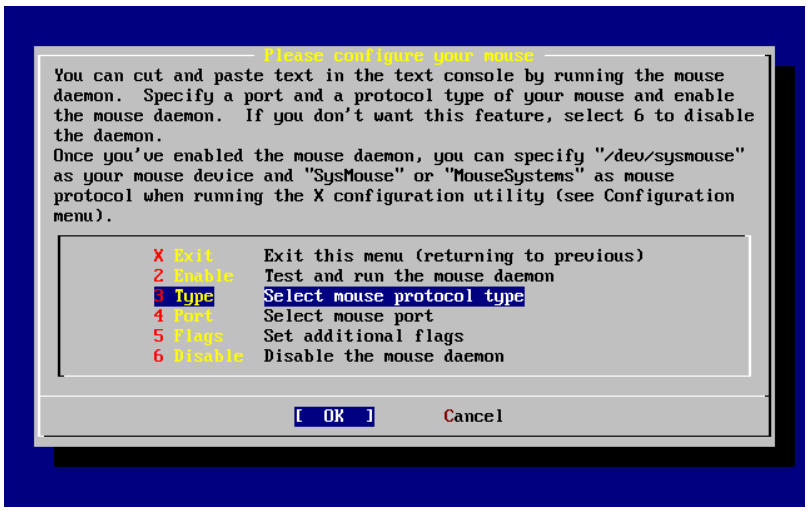


Abbildung 2.41. Das Mausprotokoll festlegen

Markieren Sie mit den Pfeiltasten Type und drücken Sie press Enter.

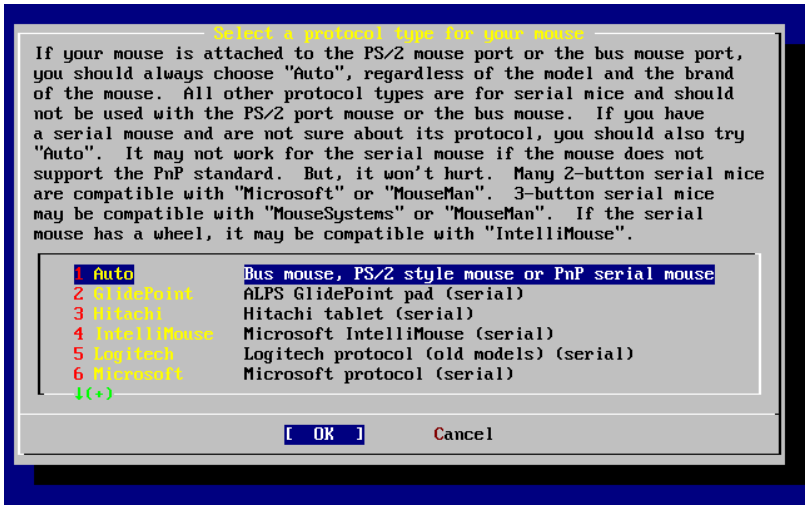


Abbildung 2.42. Das Mausprotokoll festlegen

Im Beispiel wurde eine PS/2-Maus verwendet, sodass die Vorgabe Auto passend war. Sie können das Protokoll mit den Pfeiltasten ändern. Stellen Sie sicher, dass [OK] aktiviert ist und verlassen Sie das Menü mit der Taste Enter.

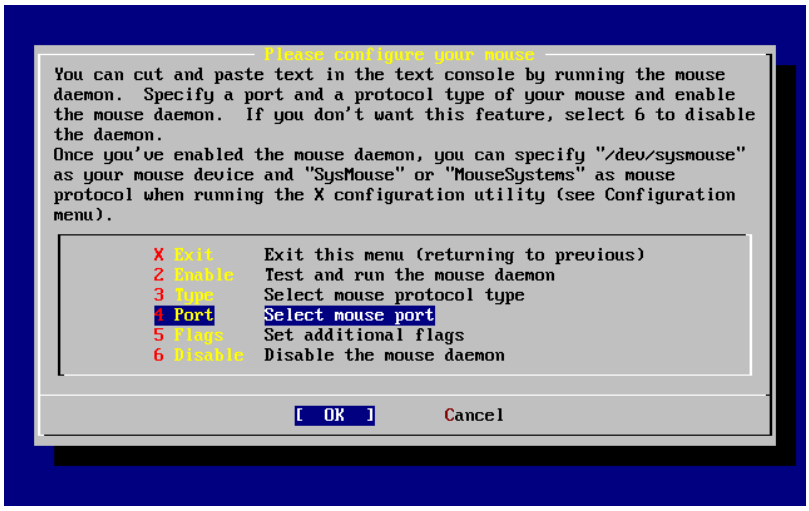


Abbildung 2.43. Den Mausport einstellen

Wählen Sie mit den Pfeiltasten Port und drücken Sie die Taste Enter.

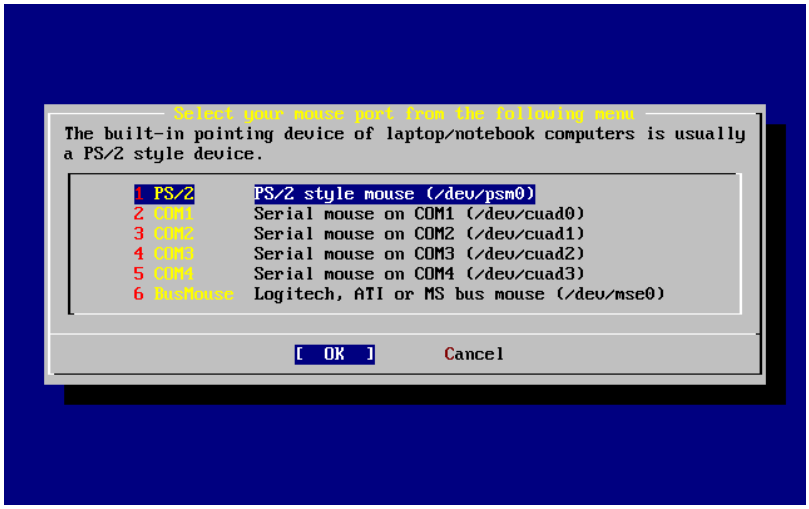


Abbildung 2.44. Den Mausport einstellen

Im Beispiel wurde eine PS/2-Maus verwendet, sodass die Vorgabe PS/2 richtig war. Sie können den Port mit den Pfeiltasten ändern. Bestätigen Sie die Auswahl mit der Taste Enter.

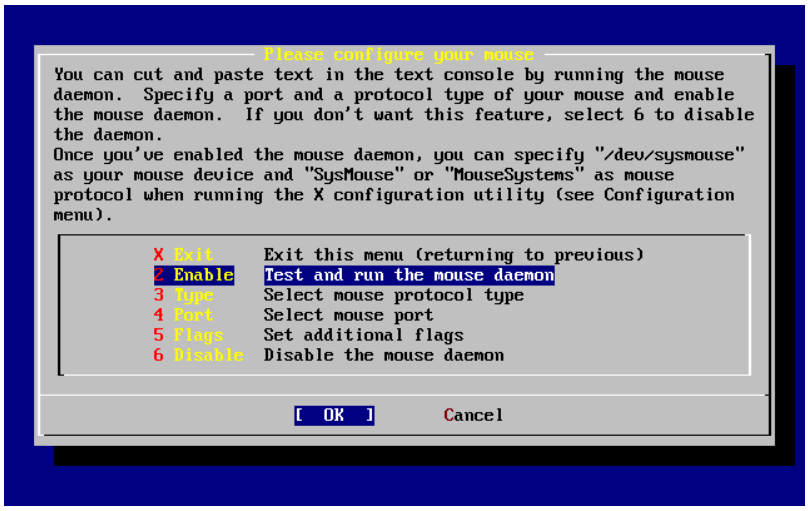


Abbildung 2.45. Den Mouse-Daemon aktivieren

Wählen Sie nun mit den Pfeiltasten Enable aus und drücken Sie die Taste Enter, um den Mouse-Daemon zu aktivieren und zu testen.

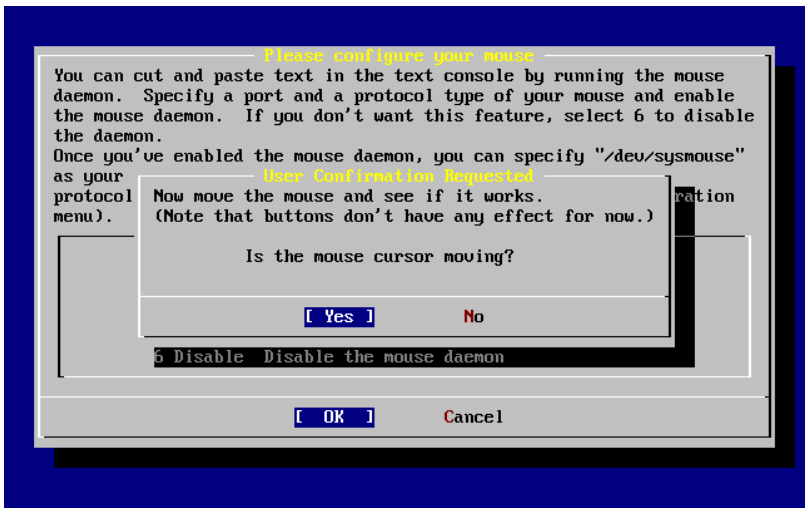


Abbildung 2.46. Den Mouse-Daemon testen

Bewegen Sie die Maus hin und her und prüfen Sie, dass sich der Mauszeiger entsprechend bewegt. Wenn alles in Ordnung ist, wählen Sie [Yes] aus und drücken Sie Enter. Wenn sich die Maus nicht richtig verhält, wurde sie nicht korrekt konfiguriert. Wählen Sie in diesem Fall [No] und versuchen Sie, die Einstellungen zu korrigieren.

Um mit den Nacharbeiten fortzufahren, wählen Sie mit den Pfeiltasten Exit aus und drücken Sie Enter.

2.10.11. Pakete installieren

Pakete (*packages*) sind schon übersetzte Programme und sind ein zweckmäßiger Weg, Programme zu installieren.

Beispielhaft wird im Folgenden die Installation eines Paketes gezeigt. In diesem Schritt können auch weitere Pakete installiert werden. Nach der Installation können Sie mit `sysinstall` zusätzliche Pakete installieren.

```
                    User Confirmation Requested
The FreeBSD package collection is a collection of hundreds of
ready-to-run applications, from text editors to games to WEB servers
and more. Would you like to browse the collection now?
```

```
                                [ Yes -]   No
```

Nachdem Sie [Yes] ausgewählt und Enter gedrückt haben, gelangen Sie in die Paketauswahl:

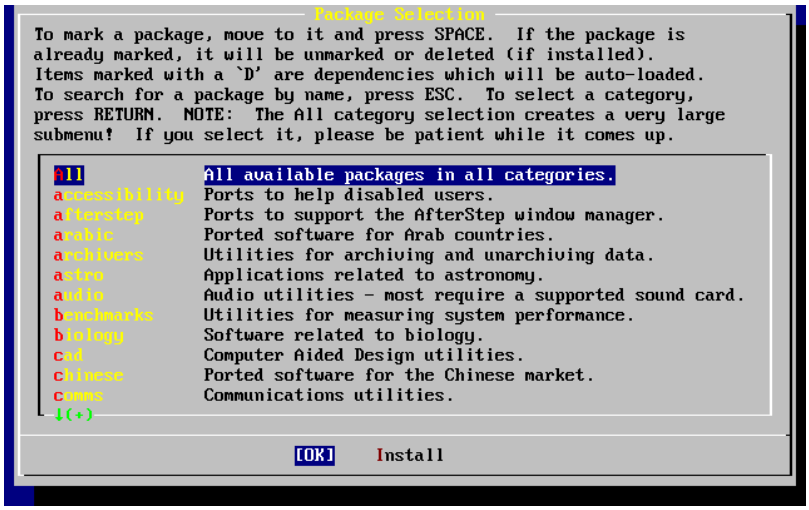


Abbildung 2.47. Die Paketkategorie aussuchen

Es stehen nur die Pakete zur Auswahl, die sich auf dem momentanen Installationsmedium befinden.

Wenn Sie All auswählen, werden alle Pakete angezeigt. Sie können die Anzeige auf die Pakete einer Kategorie beschränken. Wählen Sie mit den Pfeiltasten die Kategorie aus und drücken Sie die Taste Enter.

Ein Menü mit allen Paketen der ausgewählten Kategorie erscheint:

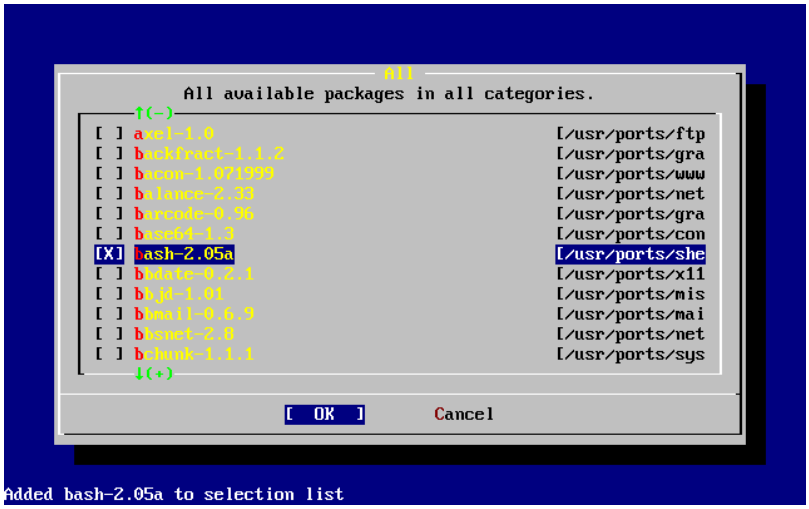


Abbildung 2.48. Pakete auswählen

Im gezeigten Bildschirm ist das Paket bash ausgewählt. Sie können weitere Pakete auswählen, indem Sie die Pakete mit den Pfeiltasten markieren und die Taste Space drücken. In der unteren linken Ecke des Bildschirms wird eine Kurzbeschreibung des ausgewählten Pakets angezeigt.

Die Taste Tab wechselt zwischen dem zuletzt ausgesuchten Paket, [OK] und [Cancel].

Wenn Sie die zu installierenden Pakete ausgewählt haben, drücken Sie einmal Tab, um [OK] zu markieren. Drücken Sie dann Enter, um wieder in die Paketauswahl zu gelangen.

Die rechte und die linke Pfeiltaste wechseln ebenfalls zwischen [OK] und [Cancel]. Mit diesen Tasten können Sie auch [OK] auswählen und dann mit Enter zur Paketauswahl zurückkehren.

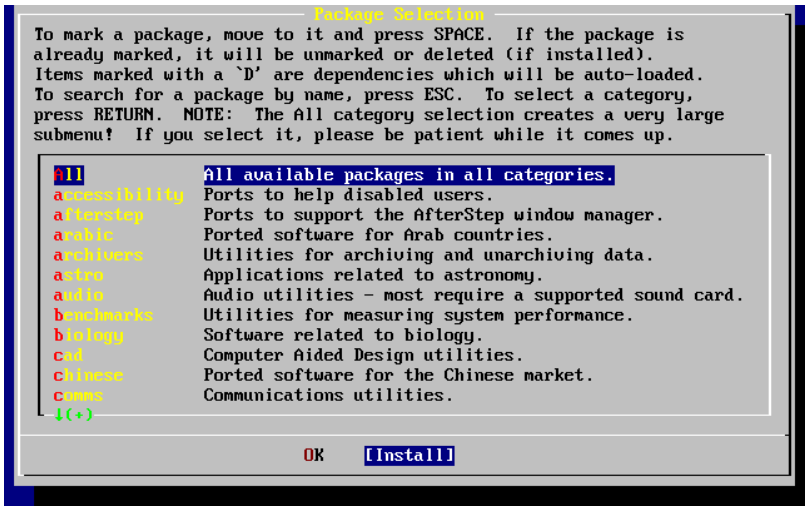


Abbildung 2.49. Pakete installieren

Benutzen Sie die Taste Tab und die Pfeiltasten um [Install] auszuwählen. Drücken Sie anschließend die Taste Enter. Sie müssen jetzt die Installation der Pakete bestätigen:

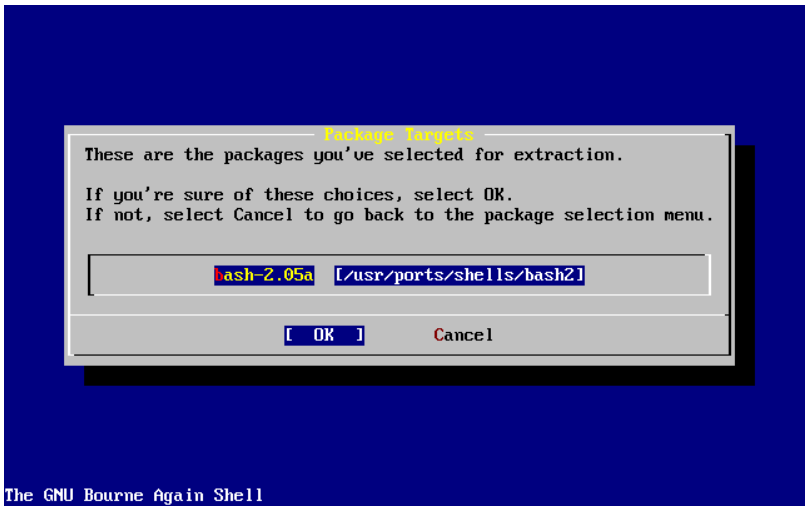


Abbildung 2.50. Paketinstallation bestätigen

Die Paketinstallation wird gestartet, wenn Sie [OK] auswählen und Enter drücken. Den Verlauf der Installation können Sie anhand der angezeigten Meldungen verfolgen; achten Sie dabei auf Fehlermeldungen.

Nach der Paketinstallation können Sie die Nacharbeiten fortsetzen. Wenn Sie keine Pakete ausgewählt haben und die Nacharbeiten fortsetzen möchten, wählen Sie trotzdem [Install] aus.

2.10.12. Benutzer und Gruppen anlegen

Während der Installation sollten Sie mindestens ein Benutzerkonto anlegen, sodass Sie das System ohne das Konto `root` benutzen können. Normalerweise ist die Root-Partition recht klein und läuft schnell voll, wenn Sie Anwendungen unter dem `root`-Konto laufen lassen. Vor der größten Gefahr warnt der nachstehende Hinweis:

```
                    User Confirmation Requested
Would you like to add any initial user accounts to the system? Adding
at least one account for yourself at this stage is suggested since
working as the "root" user is dangerous (it is easy to do things which
adversely affect the entire system).
```

```
                              [ Yes -]   No
```

Der Bildschirm auf Deutsch:

```
                        Bestätigung erforderlich
Wollen Sie Benutzerkonten anlegen?  Wir empfehlen, mindestens
ein Konto für sich selbst anzulegen, da es gefährlich
ist, unter "root" zu arbeiten (es ist leicht, Befehle einzugeben,
die das System nachhaltig beeinträchtigen).

                              [ Yes -]   No
```

Um ein Benutzerkonto anzulegen, wählen Sie [Yes] aus und drücken Enter.

Abbildung 2.51. Benutzerkonto auswählen

Markieren Sie User mit den Pfeiltasten und drücken Sie die Taste Enter.

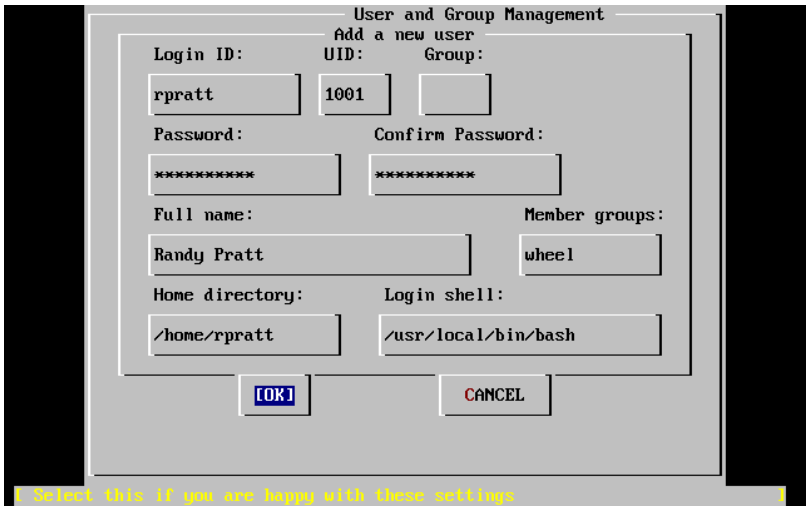


Abbildung 2.52. Benutzerkonto anlegen

Wählen Sie die Felder zum Ausfüllen mit der Taste Tab aus. Zur Hilfe werden die nachstehenden Beschreibungen werden im unteren Teil des Bildschirms angezeigt:

Login ID
: Der Name des Benutzerkontos (verpflichtend).

UID
: Die numerische ID dieses Kontos. Wenn Sie das Feld leer lassen, wird eine ID automatisch zugeteilt.

Group
: Die diesem Konto zugeordnete Login-Gruppe. Wenn Sie das Feld leer lassen, wird automatisch eine Gruppe zugeteilt.

Password
: Das Passwort des Benutzerkontos. Füllen Sie dieses Feld sehr sorgfätig aus.

Full name
: Der vollständige Name des Benutzers (Kommentarfeld).

Member groups
: Die Gruppen, in denen dieses Konto Mitglied ist (das Konto erhält Zugriffsrechte auf Dateien dieser Gruppe).

Home directory
: Das Heimatverzeichnis des Benutzerkontos. Wenn Sie das Feld leer lassen, wird das Verzeichnis automatisch festgelegt.

Login shell
: Die Login-Shell des Kontos. Wenn Sie das Feld leer lassen, wird `/bin/sh` als Login-Shell festgesetzt.

Im Beispiel wurde die Login-Shell von `/bin/sh` zu der vorher installierten `/usr/local/bin/bash` geändert. Tragen Sie keine Shell ein, die nicht existiert, da sich sonst nicht anmelden können. In der BSD-Welt wird häufig die C-Shell benutzt, die Sie mit `/bin/tcsh` angeben können.

Damit ein Wechsel auf den Superuser `root` möglich ist, wurde dem Benutzerkonto die Gruppe `wheel` zugeordnet.

Wenn Sie zufrieden sind, drücken Sie [OK]. Es erscheint wieder das Benutzer-Menü:

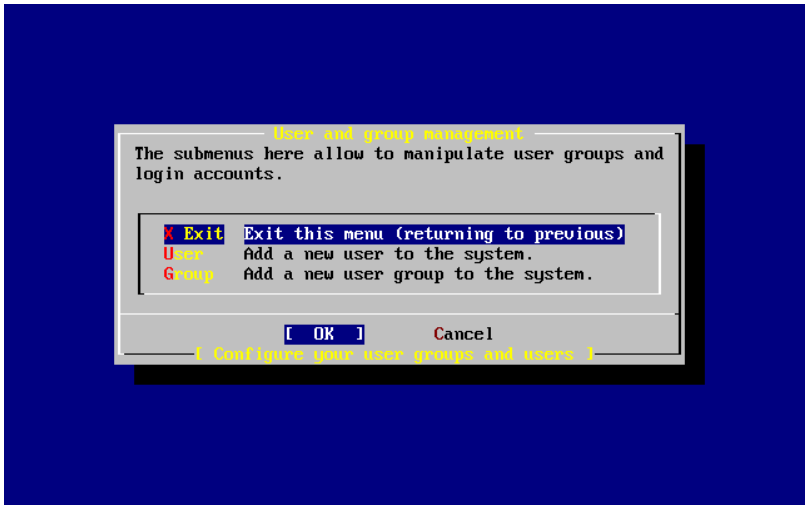


Abbildung 2.53. Benutzermenü verlassen

Weitere Gruppen können, wenn Sie die Anforderungen schon kennen, zu diesem Zeitpunkt angelegt werden. Nach der Installation können Sie Gruppen mit dem Werkzeug `sysinstall` anlegen.

Wenn Sie alle Benutzer angelegt haben, wählen Sie mit den Pfeiltasten Exit aus und drücken Sie die Taste Enter.

2.10.13. Das `root`-Passwort festlegen

```
                        Message
Now you must set the system manager's password.
```

```
 This is the password you'll use to log in as "root".

                          [ OK -]

               [ Press enter or space -]
```

Um das `root`-Passwort festzulegen, drücken Sie die Taste Enter.

Sie müssen das Passwort zweimal eingeben. Stellen Sie sicher, dass Sie das Passwort nicht vergessen. Beachten Sie, dass bei der Eingabe das Passwort weder ausgegeben wird noch Sterne angezeigt werden.

```
New password :
Retype new password :
```

Nach der erfolgreichen Eingabe des Passworts kann die Installation fortgesetzt werden.

2.10.14. Die Installation beenden

Wenn Sie noch weitere Netzwerkkarten konfigurieren oder weitere Einstellungen vornehmen wollen, können Sie das jetzt tun. Sie können die Einstellungen auch nach der Installation mit `sysinstall` vornehmen.

```
                     User Confirmation Requested
 Visit the general configuration menu for a chance to set any last
 options?

                                Yes   [ No -]
```

Um in das Hauptmenü zurückzukehren, wählen Sie mit den Pfeiltasten [No] aus und drücken Sie Enter.

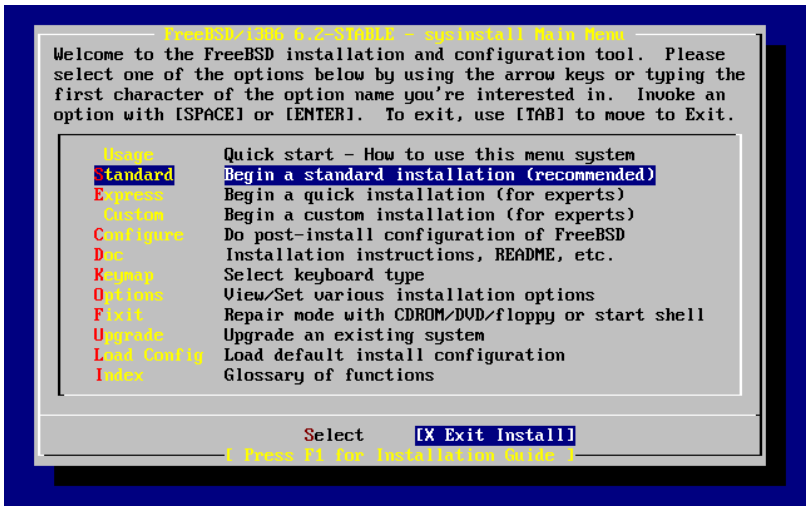


Abbildung 2.54. Die Installation beenden

Wählen Sie mit den Pfeiltasten [X Exit Install] aus und drücken Sie die Taste Enter. Sie müssen das Beenden der Installation bestätigen:

```
                     User Confirmation Requested
 Are you sure you wish to exit? The system will reboot.

                              [ Yes -]   No
```

Wählen Sie [Yes]. Wenn Sie von einer CD-ROM gestartet haben, erhalten Sie die folgende Meldung, die Sie daran erinnert, die CD-ROM aus dem Laufwerk zu entfernen:

```
                       Message
Be sure to remove the media from the drive.
```

```
                    [ OK -]
          [ Press enter or space -]
```

Das CD-Laufwerk ist bis zum Neustart des Systems verriegelt. Entfernen Sie die CD zügig, wenn der Rechner startet.

Achten Sie beim Neustart des Systems auf eventuell auftauchende Fehlermeldungen (lesen Sie Abschnitt 2.10.16, „FreeBSD starten“ für weitere Informationen).

2.10.15. Weitere Netzwerkdienste einrichten

Beigetragen von Tom Rhodes.

Anfänger ohne Vorwissen finden das Einrichten von Netzwerkdiensten oft deprimierend. Netzwerke und das Internet sind für moderne Betriebssysteme von entscheidender Bedeutung. Es ist daher wichtig, die Netzwerkfunktionen von FreeBSD zu kennen. Die von FreeBSD angebotenen Netzwerkdienste können Sie während der Installation kennen lernen.

Netzwerkdienste sind Programme, die Eingaben aus dem Netzwerk entgegennehmen. Es wird große Mühe darauf verwendet, dass diese Programme keinen Schaden verursachen. Leider können auch Programmierern Fehler unterlaufen und es gibt Fälle, in denen Fehler in Netzwerkdiensten von Angreifern ausgenutzt wurden. Es ist daher wichtig, dass Sie nur Dienste aktivieren, die Sie benötigen. Im Zweifallsfall sollten Sie einen Dienst solange nicht aktivieren, bis Sie herausfinden, dass Sie den Dienst benötigen. Einen Dienst können Sie später immer noch mit sysinstall oder in der Datei `/etc/rc.conf` aktivieren.

Wählen Sie den Menüpunkt Networking und es erscheint ein Menü wie das nachstehende:

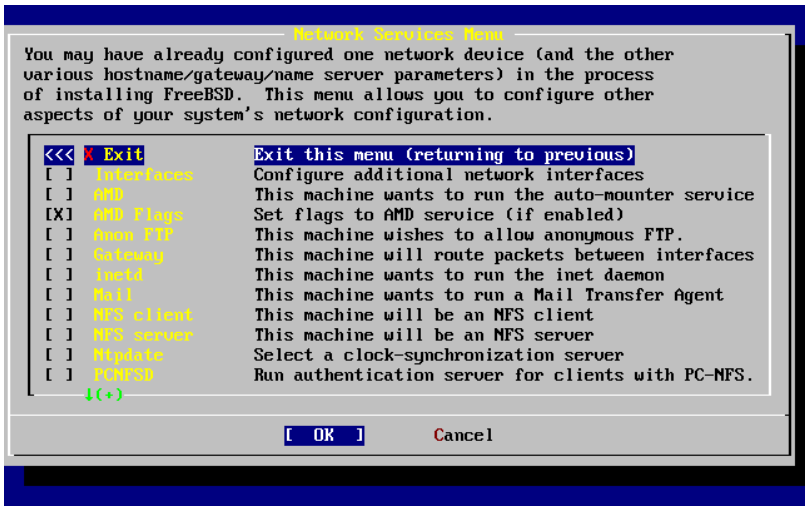


Abbildung 2.55. Netzwerkdienste – obere Hälfte

Die erste Option, Interfaces, wurde schon in Abschnitt 2.10.1, „Netzwerkkonfiguration“ konfiguriert. Sie können daher diesen Punkt überspringen.

Der Punkt AMD aktiviert einen Dienst, der automatisch Dateisysteme einhängt. Normalerweise wird der Dienst zusammen mit dem NFS-Protokoll (siehe unten) verwendet, um automatisch entfernte Dateisysteme einzuhängen. Dieser Menüpunkt erfordert keine weitere Konfiguration.

Der nächste Menüpunkt ist AMD Flags. Wenn Sie den Punkt auswählen, erscheint ein Fenster, in dem Sie AMD-spezifische Optionen eingeben können. Die nachstehenden Optionen sind schon vorgegeben:

```
-a /.amd_mnt -l syslog /host /etc/amd.map /net /etc/amd.map
```

Die Option `-a` legt das Verzeichnis fest (hier `/.amd_mnt`), unter dem Dateisysteme eingehangen werden. Die Option `-l` legt die Protokolldatei fest. Wenn syslogd verwendet wird, werden alle Meldungen an den Daemon syslogd

gesendet. Das Verzeichnis `/host` dient zum Zugriff auf exportierte Verzeichnisse von entfernten Rechnern, das Verzeichnis `/net` dient zum Zugriff auf exportierte Verzeichnisse von entfernten IP-Adressen. Die Datei `/etc/amd.map` enthält die Einstellungen für von AMD verwaltete Dateisysteme.

Die Auswahl Anon FTP erlaubt Anonymous-FTP-Verbindungen. Wählen Sie diese Option, wenn Sie einen Anonymous-FTP-Server einrichten wollen. Seien Sie sich über die Sicherheitsrisiken bewusst, wenn Sie Anonymous-FTP erlauben. Die Sicherheitsrisiken und die Konfiguration von Anonymous-FTP werden in einem gesonderten Fenster erklärt, das aufgeht, wenn Sie diese Option auswählen.

Der Menüpunkt Gateway konfiguriert das System, wie vorher erläutert, als Gateway. Wenn Sie während der Installation den Rechner aus Versehen als Gateway konfiguriert haben, können Sie dies hier wieder rückgängig machen.

Der Menüpunkt Inetd konfiguriert, wie schon oben besprochen, den Daemon inetd(8).

Die Auswahl Mail konfiguriert den Mail Transfer Agent (MTA) des Systems. Wenn Sie diesen Punkt auswählen, erscheint das folgende Menü:

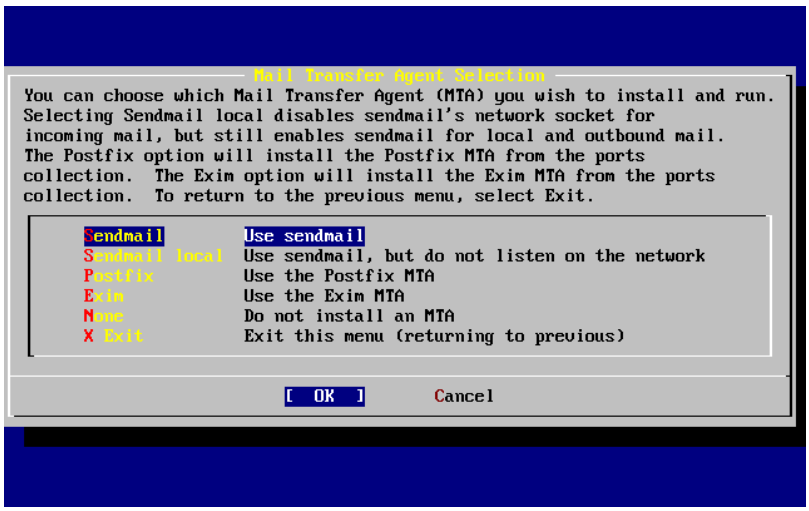


Abbildung 2.56. Den MTA festlegen

In diesem Menü wählen Sie aus, welcher MTA installiert und benutzt wird. Ein MTA ist ein Mail-Server, der E-Mails an lokale Empfänger oder an Empfänger im Internet ausliefert.

Die Auswahl Sendmail installiert das verbreitete sendmail (in FreeBSD die Voreinstellung). Die Auswahl Sendmail local verwendet sendmail als MTA, deaktiviert aber den Empfang von E-Mails aus dem Internet. Postfix und Exim sind ähnlich wie Sendmail. Beide Programme liefern E-Mails aus und einige Anwender verwenden lieber eines der beiden Programme als MTA.

Nachdem Sie einen MTA ausgewählt haben (oder beschlossen haben, keinen MTA zu benutzen), erscheint wieder das Menü Netzwerkdienste. Der nächste Menüpunkt ist NFS client.

Die Auswahl NFS client erlaubt es dem System, mit einem NFS-Server zu kommunizieren. Ein NFS-Server stellt mithilfe des NFS-Protokolls Dateisysteme für andere Systeme auf dem Netzwerk bereit. Wenn der Rechner alleine für sich steht, können Sie diesen Menüpunkt auslassen. Wahrscheinlich müssen Sie noch weitere Einstellungen vornehmen; der Abschnitt 29.3, „Network File System (NFS)“ beschreibt die Einstellungen für NFS-Server und NFS-Clients.

Der Menüpunkt NFS server richtet einen NFS-Server auf dem Rechner ein. Durch die Auswahl dieses Punktes werden die für Remote-Procedure-Call (RPC) benötigten Dienste gestartet. Mit RPC werden Routinen auf entfernten Rechnern aufgerufen.

Der nächste Punkt, Ntpdate, konfiguriert die Zeitsynchronisation. Wenn Sie diesen Punkt auswählen, erscheint das folgende Menü:

Abbildung 2.57. Ntpdate konfigurieren

Wählen Sie aus diesem Menü einen nahe liegenden Server aus. Die Zeitsynchronisation mit einem nahe liegenden Server ist, wegen der geringeren Latenzzeit, genauer als die Synchronisation mit einem weiter entfernten Server.

Der nächste Menüpunkt ist PCNFSD. Wenn Sie diesen Punkt auswählen, wird net/pcnfsd aus der Ports-Sammlung installiert. Dieses nützliche Werkzeug stellt NFS-Authentifizierungsdienste für Systeme bereit, die diese Dienste nicht anbieten (beispielsweise Microsofts MS-DOS®).

Um die nächsten Menüpunkte zu sehen, müssen Sie herunterblättern:

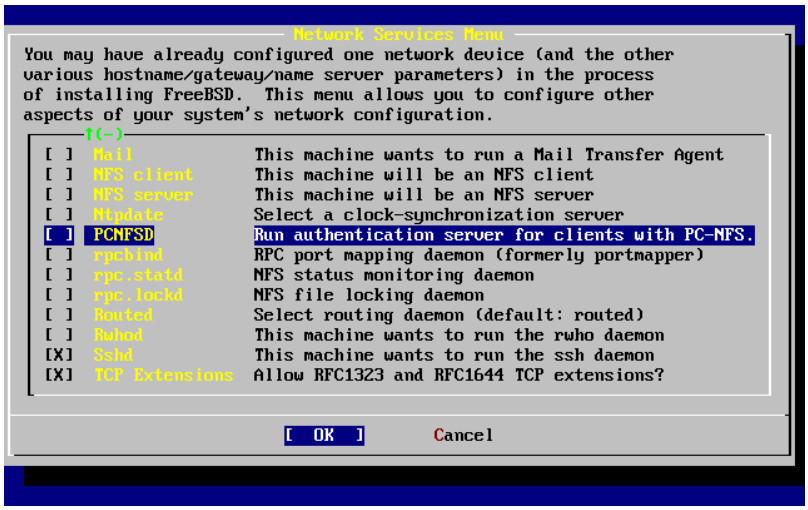


Abbildung 2.58. Netzwerkdienste – untere Hälfte

Die Programme rpcbind(8), rpc.statd(8) und rpc.lockd(8) werden für Remote-Procedure-Calls (RPC) benutzt. Das Programm `rpcbind` verwaltet die Kommunikation zwischen NFS-Servern und NFS-Clients und ist für den Betrieb eines NFS-Servers erforderlich. Der Daemon rpc.statd hält zusammen mit dem Daemon rpc.statd des entfernten Rechners den Status der Verbindung. Der Status einer Verbindung wird normalerweise in der Datei `/var/db/statd.status` festgehalten. Der nächste Menüpunkt ist rpc.lockd, der Dateisperren (*file locks*) bereitstellt. rpc.lockd wird normalerweise zusammen mit dem Daemon rpc.statd benutzt, der festhält welche Rechner Sperren anfordern und wie oft Sperren angefordert werden. Beide Dienste sind wunderbar zur Fehlersuche geeignet, doch werden Sie zum Betrieb von NFS-Servern und NFS-Clients nicht benötigt.

Der nächste Punkt in der Auswahl ist Routed, der Routing-Daemon. Das Programm routed(8) verwaltet die Routing-Tabelle, entdeckt Multicast-Router und stellt die Routing-Tabelle auf Anfrage jedem mit dem Netz verbundenen Rechner zur Verfügung. Der Daemon wird hauptsächlich auf Gateways eines lokalen Netzes eingesetzt. Wenn

Sie den Punkt auswählen müssen Sie den Ort des Programms angeben. Die Vorgabe können Sie mit der Taste Enter übernehmen. Anschließend werden Sie nach den Kommandozeilenoptionen für `routed` gefragt. Vorgegeben ist die Option `-q`.

Der nächste Menüpunkt ist Rwhod. Wenn Sie diesen Punkt auswählen, wird während des Systemstarts der Daemon rwhod(8) gestartet. Das Kommando `rwhod` schickt Broadcast-Meldungen in das Netz oder empfängt diese im Consumer-Mode. Die Funktion der Werkzeuge wird in den Hilfeseiten ruptime(1) und rwho(1) beschrieben.

Der vorletzte Menüpunkt aktiviert den Daemon sshd(8), den OpenSSH Secure-Shell-Server. Wo möglich sollte SSH anstelle von telnet und FTP eingesetzt werden. Der Secure-Shell-Server erstellt verschlüsselte und daher sichere Verbindungen zwischen zwei Rechnern.

TCP Extensions ist der letzte Menüpunkt. Diese Auswahl aktiviert die TCP-Erweiterungen aus RFC 1323 und RFC 1644. Obwohl dies auf vielen Rechnern die Verbindungsgeschwindigkeit erhöht, können durch diese Option auch Verbindungsabbrüche auftreten. Auf Servern sollte diese Option nicht aktiviert werden, auf Einzelmaschinen kann diese Option nützlich sein.

Wenn Sie die Netzwerkdienste eingerichtet haben, blättern Sie zum Menüpunkt Exit hoch, um die Nacharbeiten fortzusetzen oder verlassen Sie sysinstall, indem Sie zweimal X Exit und danach [X Exit Install] wählen.

2.10.16. FreeBSD starten

2.10.16.1. Start von FreeBSD auf FreeBSD/i386

Wenn alles funktioniert hat, laufen viele Meldungen über den Bildschirm und schließlich erscheint ein Anmeldeprompt. Um sich die Meldungen anzusehen. drücken Sie die Taste Scroll-Lock. Sie können dann mit den Tasten PgUp und PgDn blättern. Wenn Sie erneut Scroll-Lock drücken, kehren Sie zum Anmeldeprompt zurück.

Es kann sein, dass der Puffer zu klein ist, um alle Meldungen anzuzeigen. Nachdem Sie sich angemeldet haben, können Sie sich mit dem Kommando `dmesg` alle Meldungen ansehen.

Melden Sie sich bitte mit dem Benutzerkonto an (`rpratt` im Beispiel), das Sie während der Installation eingerichtet haben. Arbeiten Sie mit `root` nur dann wenn es erforderlich ist.

Die nachfolgende Abbildung zeigt typische Startmeldungen (Versionsangaben entfernt):

```
Copyright (c) 1992-2002 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.

Timecounter "i8254"  frequency 1193182 Hz
CPU: AMD-K6(tm) 3D processor (300.68-MHz 586-class CPU)
  Origin = "AuthenticAMD"  Id = 0x580  Stepping = 0
  Features=0x8001bf<FPU,VME,DE,PSE,TSC,MSR,MCE,CX8,MMX>
  AMD Features=0x80000800<SYSCALL,3DNow!>
real memory  = 268435456 (262144K bytes)
config> di sn0
config> di lnc0
config> di le0
config> di ie0
config> di fe0
config> di cs0
config> di bt0
config> di aic0
config> di aha0
config> di adv0
config> q
avail memory = 256311296 (250304K bytes)
Preloaded elf kernel "kernel" at 0xc0491000.
Preloaded userconfig_script "/boot/kernel.conf" at 0xc049109c.
md0: Malloc disk
Using $PIR table, 4 entries at 0xc00fde60
npx0: <math processor> on motherboard
npx0: INT 16 interface
```

```
pcib0: <Host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcib1: <VIA 82C598MVP (Apollo MVP3) PCI-PCI (AGP) bridge> at device 1.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <Matrox MGA G200 AGP graphics accelerator> at 0.0 irq 11
isab0: <VIA 82C586 PCI-ISA bridge> at device 7.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <VIA 82C586 ATA33 controller> port 0xe000-0xe00f at device 7.1 on pci0
ata0: at 0x1f0 irq 14 on atapci0
ata1: at 0x170 irq 15 on atapci0
uhci0: <VIA 83C572 USB controller> port 0xe400-0xe41f irq 10 at device 7.2 on pci0
usb0: <VIA 83C572 USB controller> on uhci0
usb0: USB revision 1.0
uhub0: VIA UHCI root hub, class 9/0, rev 1.00/1.00, addr 1
uhub0: 2 ports with 2 removable, self powered
chip1: <VIA 82C586B ACPI interface> at device 7.3 on pci0
ed0: <NE2000 PCI Ethernet (RealTek 8029)> port 0xe800-0xe81f irq 9 at
device 10.0 on pci0
ed0: address 52:54:05:de:73:1b, type NE2000 (16 bit)
isa0: too many dependant configs (8)
isa0: unexpected small tag 14
fdc0: <NEC 72065B or clone> at port 0x3f0-0x3f5,0x3f7 irq 6 drq 2 on isa0
fdc0: FIFO enabled, 8 bytes threshold
fd0: <1440-KB 3.5” drive> on fdc0 drive 0
atkbdc0: <keyboard controller (i8042)> at port 0x60-0x64 on isa0
atkbd0: <AT Keyboard> flags 0x1 irq 1 on atkbdc0
kbd0 at atkbd0
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: model Generic PS/2 mouse, device ID 0
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
sc0: <System console> at flags 0x1 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
sio0 at port 0x3f8-0x3ff irq 4 flags 0x10 on isa0
sio0: type 16550A
sio1 at port 0x2f8-0x2ff irq 3 on isa0
sio1: type 16550A
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
ppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/15 bytes threshold
ppbus0: IEEE1284 device found /NIBBLE
Probing for PnP devices on ppbus0:
plip0: <PLIP network interface> on ppbus0
lpt0: <Printer> on ppbus0
lpt0: Interrupt-driven port
ppi0: <Parallel I/O> on ppbus0
ad0: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata0-master using UDMA33
ad2: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata1-master using UDMA33
acd0: CDROM <DELTA OTC-H101/ST3 F/W by OIPD> at ata0-slave using PIO4
Mounting root from ufs:/dev/ad0s1a
swapon: adding /dev/ad0s1b as swap device
Automatic boot in progress...
/dev/ad0s1a: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1a: clean, 48752 free (552 frags, 6025 blocks, 0.9% fragmentation)
/dev/ad0s1f: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1f: clean, 128997 free (21 frags, 16122 blocks, 0.0% fragmentation)
/dev/ad0s1g: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1g: clean, 3036299 free (43175 frags, 374073 blocks, 1.3% fragmentation)
/dev/ad0s1e: filesystem CLEAN; SKIPPING CHECKS
/dev/ad0s1e: clean, 128193 free (17 frags, 16022 blocks, 0.0% fragmentation)
Doing initial network setup: hostname.
ed0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
        inet6 fe80::5054::5ff::fede:731b%ed0 prefixlen 64 tentative scopeid 0x1
        ether 52:54:05:de:73:1b
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x8
```

```
        inet6 ::1 prefixlen 128
        inet 127.0.0.1 netmask 0xff000000
Additional routing options: IP gateway=YES TCP keepalive=YES
routing daemons:.
additional daemons: syslogd.
Doing additional network setup:.
Starting final network daemons: creating ssh RSA host key
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
cd:76:89:16:69:0e:d0:6e:f8:66:d0:07:26:3c:7e:2d root@k6-2.example.com
 creating ssh DSA host key
Generating public/private dsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
f9:a1:a9:47:c4:ad:f9:8d:52:b8:b8:ff:8c:ad:2d:e6 root@k6-2.example.com.
setting ELF ldconfig path: /usr/lib /usr/lib/compat /usr/X11R6/lib
/usr/local/lib
a.out ldconfig path: /usr/lib/aout /usr/lib/compat/aout /usr/X11R6/lib/aout
starting standard daemons: inetd cron sshd usbd sendmail.
Initial rc.i386 initialization:.
rc.i386 configuring syscons: blank_time screensaver moused.
Additional ABI support: linux.
Local package initialization:.
Additional TCP options:.

FreeBSD/i386 (k6-2.example.com) (ttyv0)

login: rpratt
Password:
```

Das Erzeugen der RSA- und DSA-Schlüssel kann auf langsamen Maschinen lange dauern. Die Schlüssel werden nur beim ersten Neustart erzeugt, spätere Neustarts sind schneller.

Wenn der X-Server konfiguriert ist und eine Oberfläche ausgewählt wurde, können Sie X mit dem Kommando `startx` starten.

2.10.17. FreeBSD herunterfahren

Es ist wichtig, dass Sie das Betriebssystem richtig herunterfahren. Wechseln Sie zunächst mit dem Befehl `su` zum Superuser; Sie müssen dazu das `root`-Passwort eingeben. Der Wechsel auf den Superuser gelingt nur, wenn der Benutzer ein Mitglied der Gruppe `wheel` ist. Ansonsten melden Sie sich direkt als Benutzer `root` an. Der Befehl `shutdown -h now` hält das System an.

```
The operating system has halted.
Please press any key to reboot.
```

Sie können den Rechner ausschalten, nachdem die Meldung `Please press any key to reboot` erschienen ist. Wenn Sie stattdessen eine Taste drücken, startet das System erneut.

Sie können das System auch mit der Tastenkombination Ctrl+Alt+Del neu starten. Sie sollten diese Tastenkombination allerdings nicht gewohnheitsmäßig benutzen.

2.11. Fehlersuche

Dieser Abschnitt behandelt häufig auftretende Installationsprobleme. Weiterhin enthält er Hinweise, wie FreeBSD parallel mit MS-DOS® oder Windows® betrieben wird.

2.11.1. Wenn etwas schief geht

Aufgrund der Beschränkungen der PC-Architektur ist eine zuverlässige Geräteerkennung nicht möglich. Falls die Geräteerkennung fehlschlägt, können Sie einige Dinge versuchen.

Sehen Sie in den Hardware Notes Ihrer FreeBSD-Version nach, ob Ihre Hardware unterstützt wird.

Wenn Ihre Hardware unterstützt wird und sich der Installationsprozess aufhängt oder sonstige Probleme auftauchen, müssen Sie einen angepassten Kernel erstellen, da Ihre Hardware in diesem Fall nicht vom `GENERIC` -Kernel unterstützt wird. Der Kernel auf den Startdisketten verwendet die Werkseinstellungen für IRQs, IO-Adressen und DMA-Kanäle. Geänderte Einstellungen müssen Sie daher in der Kernelkonfigurationsdatei angeben, damit FreeBSD diese Geräte korrekt erkennt.

Es ist auch möglich, dass die Suche nach einem nicht vorhandenen Gerät dazu führt, dass die Erkennung eines vorhandenen Geräts fehlschlägt. In diesem Fall sollten Sie nicht vorhandene Geräte, deren Einstellungen sich mit vorhandenen Geräten überschneiden, deaktivieren.

Anmerkung

Einige Installationsprobleme können Sie vermeiden oder umgehen, indem Sie die Firmware der Hardware, insbesondere die Firmware der Systemplatine, aktualisieren. Die Firmware der Systemplatine ist das BIOS. Die meisten Hardware-Hersteller bieten aktuelle Firmware und Anleitungen zur Aktualisierung der Firmware auf dem Internet an.

Viele Hersteller raten davon ab, ohne guten Grund das BIOS zu aktualisieren. Die Aktualisierung *kann* fehlschlagen und den BIOS-Chip dauerhaft beschädigen.

2.11.2. MS-DOS®- und Windows®-Dateisysteme benutzen

Mit Double Space™ komprimierte Dateisysteme werden zurzeit von FreeBSD nicht unterstützt. Damit FreeBSD auf die Daten zugreifen kann, müssen Sie das Dateisystem daher dekomprimieren. Rufen Sie dazu den Compression Agent aus dem Menü Start > Programs > System Tools auf.

FreeBSD unterstützt MS-DOS®-Dateisysteme (manchmal auch als FAT-Dateisysteme bezeichnet). Der Befehl mount_msdosfs(8) bindet diese Dateisysteme in den FreeBSD-Verzeichnisbaum ein und erlaubt dadurch den Zugriff auf die darin enthaltenen Daten. mount_msdosfs(8) wird normalerweise nicht direkt, sondern über einen Eintrag in der Datei `/etc/fstab` oder durch den Aufruf des Befehls mount(8) (in Kombination mit den korrekten Parametern).

Ein typischer Eintrag in `/etc/fstab` sieht so aus:

```
/dev/ad0sN  /dos  msdosfs rw  0   0
```

Anmerkung

Das Verzeichnis `/dos` muss bereits vorhanden sein, damit dieser Eintrag funktioniert. Weitere Informationen zu den Einstellungen in der Datei `/etc/fstab` finden sich in der Manualpage fstab(5).

Ein typischer Aufruf von mount(8) zum Einhängen eines MS-DOS®-Dateisystems sieht so aus:

```
# mount -t msdosfs /dev/ad0s1 /mnt
```

Das MS-DOS®-Dateisystem befindet sich hier auf der ersten Partition der primären Platte. Dies kann bei Ihnen anders sein. Die Anordnung der Partitionen entnehmen Sie den Ausgaben von `dmesg` und `mount`.

Anmerkung

FreeBSD numeriert Platten (genauer MS-DOS®-Partitionen) anders als andere Betriebssysteme. Die Nummern von erweiterten Partitionen sind in der Regel höher als die Nummern von primären Partitionen. Das Werkzeug fdisk(8) kann Ihnen dabei helfen, festzustellen, welche Partitionen zu FreeBSD und welche zu einem anderen Betriebssystem gehören.

Analog werden NTFS-Partitionen mit dem Kommando mount_ntfs(8) eingehangen.

2.11.3. Fragen und Antworten zu häufig auftretenden Problemen

F: Mein System hängt sich beim Testen der Hardware auf, oder es verhält sich seltsam während der Installation oder das Diskettenlaufwerk wird nicht getestet.

A: FreeBSD 5.0 und neuer machen ausgiebig Gebrauch von den ACPI-Systemdiensten zur Systemkonfiguration der i386-, amd64- und ia64-Plattformen, falls diese während des Bootvorgangs gefunden werden. Leider enthalten sowohl der ACPI-Treiber als auch manche Motherboard- und BIOS-Implementierungen für ACPI noch einige Fehler. Kommt es auf Ihrem System zu Problemen, können Sie ACPI daher deaktivieren, indem während des Bootvorganges den „Hint“ `hint.acpi.0.disabled` aktivieren:

```
set hint.acpi.0.disabled="1"
```

Da diese Einstellung bei jedem Neustart verloren geht, aktivieren Sie sie dauerhaft, indem Sie die Zeile `hint.acpi.0.disabled="1"` in die Datei `/boot/loader.conf`. Weitere Informationen über den Bootloader finden Sie in Abschnitt 13.1, „Übersicht“ des FreeBSD-Handbuchs.

F: Direkt nach der Installation beginnt das System zwar zu booten, der Kernel wird geladen und meine Hardware getestet. Dann bricht der Bootvorgang aber mit der folgenden (oder einer ähnlichen) Fehlermeldung ab:

```
changing root device to ad1s1a panic: cannot mount root
```

Was läuft hier falsch? Was kann/muss ich tun?

Was soll ich mit diesem `bios_drive:interface(unit,partition)kernel_name` anfangen, das mir die Hilfefunktion ausgibt?

A: Dabei handelt es sich um ein lange bekanntes Problem, das nur dann auftritt, wenn es sich bei der Bootplatte nicht um die erste Platte im System handelt. Das BIOS numeriert die Festplatten anders als FreeBSD, daher ist das System manchmal nicht in der Lage, diese Numerierungen selbst automatisch in Einklang zu bringen.

Sollte Ihre Bootplatte nicht die erste Platte im System sein, können Sie FreeBSD dabei helfen, diese Platte zu finden. Es gibt zwei Situationen, in denen Sie FreeBSD mitteilen müssen, wo sich das root-Dateisystem befindet. Dazu müssen Sie die Nummer der Platte im BIOS, den Plattentyp sowie die Nummer der Platte unter FreeBSD angeben.

Im ersten Fall verfügen Sie über zwei IDE-Platten, die beide als Master an ihrem jeweiligen IDE-Controller konfiguriert sind. FreeBSD soll dabei von der zweiten Platte booten. Ihr BIOS erkennt die beiden Platten als „Platte 1“ und „Platte 2“, während FreeBSD die Platten als `ad0` und `ad2` erkennt.

Für das BIOS befindet sich FreeBSD auf der Platte Nummer 1, der Typ ist `ad`, und FreeBSD erkennt die Platte als Platte Nummer 2. Daher geben Sie Folgendes ein:

```
1:ad(2,a)kernel
```

Beachten Sie, dass dieser Eintrag nicht notwendig ist, wenn die zweite Platte als Slave am primären IDE-Controller konfiguriert ist (sondern sogar falsch wäre).

Die zweite Situation entsteht, wenn Sie von einer SCSI-Platte booten und zusätzlich eine oder mehrere IDE-Platten installiert haben. In diesem Fall ist die Plattennummer unter FreeBSD kleiner als die Plattennummer im BIOS. Verfügen Sie über zwei IDE-Platten und eine SCSI-Platte, hat die SCSI-Platte im BIOS die Nummer 2, den Typ `da`, und wird von FreeBSD als Platte Nummer 0 erkannt. In diesem Fall geben Sie daher Folgendes ein:

```
2:da(0,a)kernel
```

Durch diese Zeile teilen Sie FreeBSD mit, dass Sie von der BIOS-Platte Nummer 2 booten wollen (bei der es sich um die erste SCSI-Platte Ihres Systems handelt). Verfügen Sie nur über eine IDE-Platte, geben Sie hingegen `1:` ein.

Nachdem Sie die korrekten Werte ermittelt haben, können Sie die entsprechende Zeile in exakt der gleichen Form in die Datei `/boot.config` aufnehmen. In der Voreinstellung verwendet FreeBSD den Inhalt dieser Datei als Standardantwort am `boot:`-Prompt.

F: Nach der Installation beginnt das System zu booten, der Bootmanager zeigt im Bootmenü aber immer nur `F?` an und das System startet nicht.

A: Sie haben bei der FreeBSD-Installation eine falsche Plattengeometrie angegeben. Starten Sie den Partitionseditor neu und geben Sie die korrekte Plattengeometrie an. Danach installieren Sie FreeBSD erneut (diesmal mit der korrekten Plattengeometrie).

Ist es Ihnen nicht möglich, die korrekte Plattengeometrie herauszufinden, hilft Ihnen vielleicht der folgende Tipp weiter: Legen Sie eine kleine MS-DOS®-Partition am Beginn Ihrer Bootplatte an und installieren Sie anschließend FreeBSD auf diese Platte. Das FreeBSD-Installationsprogramm wird die MS-DOS®-Partition erkennen und ist dadurch normalerweise in der Lage, die korrekte Plattenkonfiguration automatisch zu erkennen.

Die Vorgangsweise im folgenden Tipp wird zwar nicht länger empfohlen, soll aber trotzdem dokumentiert werden:

> Wenn Sie ein reines FreeBSD-System aufsetzen wollen (als Server oder als Workstation) und daher nie auf Kompatibilitält zu MS-DOS®, Linux oder anderen Betriebssystemen angewiesen sein werden, haben Sie auch die Möglichkeit, die komplette Platte (durch die Wahl von A im Partitionseditor) für FreeBSD zu verwenden. Danach wird FreeBSD die komplette Platte vom ersten bis zum letzten Sektor verwenden und die tatsächliche Plattengeometrie ignorieren. Danach ist es allerdings nicht mehr möglich, ein anderes Betriebssystem auf die gleiche Platte zu installieren (ohne auch FreeBSD neu zu installieren).

F: FreeBSD erkennt meine ed(4)-Netzwerkkarte. Trotzdem erhalte ich weiterhein Timeout-Meldungen für dieses Gerät.

A: Ihre Karte verwendet wahrscheinlich einen anderen IRQ als den, der in der Datei `/boot/device.hints` angegeben wurde. Der ed(4)-Treiber verwendet in der Voreinstellung keine „Soft“-Konfiguration (also Werte, die durch EZSETUP unter MS-DOS® eingeben wurden). Sie können dies allerdings erzwingen, indem Sie die Option `-1` in den „Hints“ für dieses Gerät angeben.

Entweder verändern Sie die Jumper-Konfiguration der Karte (und, falls notwendig, die Kerneleinstellungen). Oder Sie geben den IRQ als `-1` an, indem Sie `hint.ed.0.irq="-1"` eingeben. Dadurch wird der Kernel angewiesen, die „Soft“-Konfiguration zu verwenden.

Prüfen Sie auch, ob Ihre Karte nicht etwa IRQ 9 verwendet, da dieser mit IRQ 2 geteilt wird. Diese Einstellung verursacht häufig Probleme (insbesondere dann, wenn IRQ 2 durch eine VGA-Grafikkarte belegt ist!). Wenn irgend möglich, sollten Sie daher IRQ 2 oder 9 nicht verwenden.

F: Wenn ich sysinstall aus einen X-Terminal starte, ist die gelbe Schritt auf dem grauen Hintergrund nur schwer zu erkennen. Gibt es eine Möglichkeit, den Kontrast für dieses Programm zu erhöhen?

A: Haben Sie X11 bereits installiert und die von sysinstall verwendeten Farben bereiten Ihnen beim Lesen von Text Probleme (wenn Sie ein X-Terminal verwenden), sollten Sie die Zeile `XTerm*color7: #c0c0c0` in die Datei `~/.Xdefaults` aufnehmen. Dadurch wird der Hintergrund in einem dunkleren Grauton dargestellt.

2.12. Anspruchsvollere Installationen

Beigetragen von Valentino Vaschetto.
Aktualisiert von Marc Fonvieille.

Dieser Abschnitt beschreibt die Installation von FreeBSD in besonderen Situationen.

2.12.1. FreeBSD auf einem System ohne Monitor oder Tastatur installieren

Diese Methode wird als „headless install“ (kopflose Installation) bezeichnet, da die Maschine, auf die FreeBSD installiert werden soll, entweder keinen Monitor angeschlossen hat oder über keine VGA-Karte verfügt. Wie kann FreeBSD dennoch installiert werden? Eben mithilfe einer seriellen Konsole. Im Wesentlichen ist eine serielle Konsole eine andere Maschine, die Ein- und Ausgaben für eine andere Maschine bereitstellt. Um über eine serielle Konsole zu installieren, erstellen Sie zunächst (wie in Abschnitt 2.3.7, „Das Startmedium vorbereiten“ beschrieben) einen bootbaren USB-Stick oder laden Sie das passende CD-ISO-Abbild herunter.

Um von diesen Medien in eine serielle Konsole booten zu können, müssen Sie die folgenden Schritte durchführen (bei Verwendung einer Boot-CD kann der erste Schritt entfallen):

1. Den USB-Stick für eine serielle Konsole anpassen

   Wenn Sie ein System mit den frisch erstellten USB-Stick starten, läuft der normale FreeBSD-Installationsprozess an. Diese Installation soll aber über die serielle Konsole gesteuert werden. Daher müssen Sie den USB-Stick mit dem Befehl mount(8) in den Verzeichnisbaum einhängen:

   ```
   # mount /dev/da0a /mnt
   ```

   > **Anmerkung**
   >
   > Passen Sie den Mountpunkt und die Gerätedatei falls nötig an Ihre Gegebenheiten an.

   Nachdem Sie den USB-Stick eingehängt haben, müssen Sie ihn rekonfigurieren, damit er in eine serielle Konsole startet. Dazu nehmen Sie in die Datei `loader.conf` des USB-Sticks eine Zeile auf, die die serielle Konsole als Systemkonsole festlegt:

   ```
   # echo 'console="comconsole"' >> /mnt/boot/loader.conf
   ```

   Damit ist Ihr USB-Stick für die Installation vorbereitet. Sie können ihn daher wieder aus dem Dateisystem aushängen:

   ```
   # umount /mnt
   ```

   Entfernen Sie nun den USB-Stick und machen Sie direkt mit Schritt 3 weiter.

2. Die Installations-CD für eine serielle Konsole anpassen

   Wenn Sie von dem soeben heruntergeladenen CD-ISO-Abbild (siehe Abschnitt 2.13.1, „Eine Installations-CD-ROM erzeugen“) starten, gelangen Sie in den normalen Installationsmodus von FreeBSD. Da wir aber in eine

serielle Konsole booten wollen, muss das CD-Image extrahiert, modifiziert und neu erzeugt werden, bevor Sie es auf eine CD-R brennen.

Entpacken Sie alle Dateien des CD-ISO-Abbilds (beispielsweise `FreeBSD-10.3-RELEASE-i386-disc1.iso`) auf dem System, auf das Sie das Abbild heruntergeladen haben unter Verwendung von tar(1):

```
# mkdir /path/to/headless-iso
# tar -C /path/to/headless-iso -pxvf FreeBSD-10.3-RELEASE-i386-disc1.iso
```

Nun müssen Sie das entpackte ISO-Abbild rekonfigurieren, damit es künftig in eine serielle Konsole startet. Dazu nehmen Sie in die Datei `loader.conf` des entpackten ISO-Abbild eine Zeile auf, die die serielle Konsole als Systemkonsole festlegt:

```
# echo 'console="comconsole"' >> /path/to/headless-iso/boot/loader.conf
```

Damit ist der Dateibaum des entpackten ISO-Abbilds für die Installation vorbereitet und Sie können über den Befehl mkisofs(8) (das Sie über den Port sysutils/cdrtools installieren können) ein neues CD-ISO-Abbild erzeugen:

```
# mkisofs -v -b boot/cdboot -no-emul-boot -r -J -V "Headless_install" \
     -o Headless-FreeBSD-10.3-RELEASE-i386-disc1.iso /path/to/headless-iso
```

Dieses rekonfigurierte ISO-Abbild brennen Sie nun mit dem Brennprogramm Ihrer Wahl auf eine CD-R.

3. Das Nullmodemkabel anschließen

   Sie müssen beide Maschinen mit einem Nullmodemkabel verbinden. Schließen Sie das Nullmodemkabel an die seriellen Schnittstellen beider Maschinen an. *Sie können kein direktes serielles Kabel verwenden*, Nullmodemkabel besitzen gekreuzte Leitungen.

4. Die Installation starten

   Sie können die Installation jetzt starten. Stöpseln Sie den vorbereiteten USB-Stick ein und starten Sie Ihren Computer. Alternativ starten Sie Ihren Computer und legen die vorbereitete Installations-CD ein.

5. Die Verbindung mit der zur installierenden Maschine herstellen

   Mit dem Kommando cu(1) verbinden Sie sich mit der zu installierenden Maschine:

   ```
   # cu -l /dev/cuau0
   ```

   Unter FreeBSD 7.X verwenden Sie hingegen den folgenden Befehl:

   ```
   # cu -l /dev/cuad0
   ```

Fertig! Über die `cu`-Sitzung können Sie nun die zu installierende Maschine steuern. Der Kernel wird automatisch geladen und Sie können anschließend den Terminaltyp festlegen. Wählen Sie die `FreeBSD color console` aus und fahren wie gewohnt mit der Installation fort.

2.13. Eigene Installationsmedien herstellen

Anmerkung

Im Folgenden ist mit „Installations-CD“ eine CD-ROM oder DVD gemeint, die Sie gekauft oder selbst hergestellt haben.

Oft müssen Sie eigene Installationsmedien erzeugen. Dies können physische Medien wie Bänder sein oder Installationsquellen sein, aus denen sysinstall Dateien herunterlädt, beispielsweise ein lokaler FTP-Server oder eine MS-DOS®-Partition.

Beispiele:

- Im lokalen Netzwerk befinden sich viele Maschinen, Sie besitzen allerdings nur eine Installations-CD. Den Inhalt der Installations-CD wollen Sie auf einem lokalem FTP-Server bereitstellen. Zur Installation wird der lokale FTP-Server anstelle eines Internet-Servers benutzt.

- Sie haben eine Installations-CD, allerdings erkennt FreeBSD im Gegensatz zu MS-DOS®/Windows® das CD- oder DVD-Laufwerk nicht. Sie können die Installationsdateien auf eine MS-DOS®-Partition desselben Rechners kopieren und FreeBSD von der MS-DOS®-Partition installieren.

- Der Rechner, auf dem Sie FreeBSD installieren wollen, besitzt kein CD- oder DVD-Laufwerk. Ein anderer Rechner, zu dem eine serielle oder parallele Verbindung besteht, besitzt allerdings ein CD- oder DVD-Laufwerk.

- Sie wollen ein Band erzeugen, mit dem Sie FreeBSD installieren können.

2.13.1. Eine Installations-CD-ROM erzeugen

Mit jeder Release stellt das FreeBSD-Project für jede unterstützte Architektur mindestens zwei CD-Abbilder („ISO-Images“) zur Verfügung. Wenn Sie einen CD-Brenner besitzen, können Sie diese Abbilder brennen und damit FreeBSD installieren. Wenn Sie einen CD-Brenner besitzen und über eine gute Internet-Verbindung verfügen, ist das die preiswerteste Art, FreeBSD zu installieren.

1. Das richtige Abbild herunterladen

   Die ISO-Abbilder für jedes Releases können Sie von `ftp://ftp.FreeBSD.org/pub/FreeBSD/ISO-IMAGES-arch/version` oder einem nahe gelegenen Spiegel herunterladen. Ersetzen Sie *`arch`* und *`version`* durch passende Werte.

   Das Verzeichnis enthält die folgenden Abbilder:

   Tabelle 2.4. FreeBSD 8.*X* ISO-Abbilder

Dateiname	Inhalt
`FreeBSD-version-RELEASE-arch-bootonly.iso`	Enthält alles, was Sie benötigen, um um den FreeBSD-Kernel zu laden und das Installationsprogramm zu starten. Die zu installierenden Dateien müssen allerdings über FTP oder eine andere geeignete Quelle bezogen werden, da sie in diesem Abbild nicht enthalten sind.
`FreeBSD-version-RELEASE-arch-dvd1.iso.gz`	Dieses DVD-Abbild enthält alle zur Installation von FreeBSD nötigen Dateien, eine Auswahl an Paketen Dritter sowie die Dokumentation. Zusätzlich ermöglicht es Ihnen dieses Abbild, einen „livefs“-basierten Rettungsmodus zu starten.
`FreeBSD-version-RELEASE-arch-memstick.img`	Dieses Abbild kann auf einen USB-Stick geschrieben werden. Dieser kann danach als Installationsmedium verwendet werden (wenn Ihr System dies unterstützt). Zusätzlich ermöglicht es Ihnen dieses Abbild, einen „livefs“-basierten Rettungsmodus zu starten. Die FreeBSD-Dokumentation ist ebenfalls enthalten, aber keine Pakete Dritter. Dieses Abbild ist erst ab FreeBSD 8.0 verfügbar.

Dateiname	Inhalt
`FreeBSD-version-RELEASE-arch-disc1.iso`	Dieses CD-Abbild enthält alle für die Installation von FreeBSD nötigen Dateien sowie die Dokumentation. Es sind allerdings keine Pakete Dritter enthalten.
`FreeBSD-version-RELEASE-arch-disc2.iso`	So viele Pakete Dritter, wie auf dem Installationsmedium Platz hatten. Dieses Abbild ist für FreeBSD 8.*X* nicht mehr verfügbar.
`FreeBSD-version-RELEASE-arch-disc3.iso`	Ein weiteres Abbild mit so vielen Paketen Dritter, wie auf dem Installationsmedium Platz hatten. Dieses Abbild ist für FreeBSD 8.*X* nicht mehr verfügbar.
`version-RELEASE-arch-docs.iso`	Die FreeBSD-Dokumentation.
`FreeBSD-version-RELEASE-arch-livefs.iso`	Dieses Abbild enthät einen „livefs“-basierten Rettungsmodus. Eine Installation von FreeBSD ist mit diesem Abbild allerdings nicht möglich.

Anmerkung

Die Abbilder für FreeBSD 7.*X*-Releases vor FreeBSD 7.3 sowie für FreeBSD 8.0 wurden noch unterschiedlich benannt. Bei Ihnen fehlt die Bezeichnung `FreeBSD-` am Anfang des Abbildnamens.

Sie benötigen nur eines der beiden Abbilder `bootonly` oder `disc1`. Laden Sie bitte nicht beide Abbilder herunter, das `disc1`-Abbild enthält alles, was das `bootonly`-Abbild enthält.

Benutzen Sie das `bootonly`-Abbild, wenn Sie eine preiswerte Internet-Anbindung besitzen. Mit diesem Abbild können Sie FreeBSD installieren. Software Dritter können Sie anschließend mithilfe des Ports-Systems (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*) herunterladen.

Benutzen Sie das `dvd1`-Abbild, wenn Sie FreeBSD installieren wollen und das Installationsmedium eine angemessene Auswahl an Software Dritter enthalten soll.

Die zusätzlichen Abbilder sind nützlich, aber nicht notwendig, insbesondere wenn Sie eine schnelle Internet-Verbindung besitzen.

2. Die CDs brennen

   Sie müssen die Abbilder auf eine CD brennen. Das Brennen von CDs unter FreeBSD wird in Abschnitt 18.6, „Erstellen und Verwenden von CDs“ erläutert (sehen Sie sich insbesondere Abschnitt 18.6.3, „Einen ATAPI-Brenner benutzen“ und Abschnitt 18.6.4, „Einen USB- oder SCSI-Brenner benutzen“ an).

   Wenn Sie die CDs unter einem anderen Betriebssystem erstellen, benutzen Sie die entsprechenden Werkzeuge des Betriebssystems. Die Abbilder sind Standard-ISO-Abbilder und können von vielen Brennprogrammen verarbeitet werden.

Anmerkung

Wenn Sie eine angepasste Version von FreeBSD erstellen wollen, sollten Sie den Release Engineering Article lesen.

2.13.2. Einen lokalen FTP-Server einrichten

Die Dateien auf der Installations-CD sind genauso angeordnet wie auf den FreeBSD-FTP-Servern. Daher ist es einfach, einen lokalen FTP-Server für die FreeBSD-Installation über ein Netzwerk einzurichten.

1. Hängen Sie auf dem FTP-Server die CD-ROM in das Verzeichnis `/cdrom` ein:

   ```
   # mount /cdrom
   ```

2. Legen Sie ein Konto für Anonymous-FTP an. Dazu editieren Sie die Datei `/etc/passwd` mit dem Kommando vipw(8) und fügen die nachstehende Zeile hinzu:

   ```
   ftp:*:99:99::0:0:FTP:/cdrom:/nonexistent
   ```

3. Stellen Sie sicher, dass der FTP-Dienst in der Datei `/etc/inetd.conf` aktiviert ist.

Jeder, der Ihren Rechner über das Netzwerk erreicht, kann nun FreeBSD über FTP installieren. In sysinstall wird dazu FTP als Installationsmedium wählt. Der FTP-Server wird durch die Auswahl Other (andere als die vorgegebenen Server) und anschließende Eingabe von **`ftp://Ihr Rechner`** festgelegt.

Anmerkung

Wenn die Version der für die FTP-Installation Ihrer Clients verwendeten Bootmedien (normalerweise Disketten) nicht exakt der von Ihnen auf Ihrem lokalen FTP-Server angebotenen Version entspricht, ist sysinstall nicht in der Lage, die Installation abzuschließen. Sind die Versionsnummern unterschiedlich, können Sie durch das Aufrufen des Punktes Options sysinstall dazu zwingen, die Installation dennoch abzuschließen. Dazu setzen Sie den Namen der Distribution auf any.

Warnung

Diese Vorgehensweise ist in Ihrem lokalen Netzwerk, das durch eine Firewall geschützt ist, völlig in Ordnung. Wenn Sie FTP für Rechner auf dem Internet (und nicht für lokale Rechner) anbieten, zieht Ihr Server die Aufmerksamkeit von Crackern und andere Unannehmlichkeiten auf sich. Achten Sie in diesem Fall darauf, dass Sie gute Sicherheitsverfahren anwenden.

2.13.3. Installationsdisketten erstellen

Wenn Sie, was wir *nicht* empfehlen, von Disketten installieren müssen, weil Disketten das einzig unterstützte Installationsmedium sind oder Sie es sich einfach schwer machen wollen, müssen Sie zunächst einige Disketten vorbereiten.

Sie müssen mindestens den Inhalt des Verzeichnisses `base` auf 1.44 MB Disketten kopieren. Wenn Sie die Disketten unter MS-DOS® erstellen, *müssen* Sie die Disketten mit dem MS-DOS®-Kommando `FORMAT` formatieren. Unter Windows® können Sie Disketten mithilfe des Explorers formatieren (klicken Sie mit der rechten Maustaste auf das `A:`-Laufwerk und wählen Sie Format aus).

Vertrauen Sie vorformatierten Disketten nicht; formatieren Sie die Disketten zur Sicherheit immer selbst. In der Vergangenheit waren vorformatierte Disketten der Verursacher vieler Probleme.

Falls Sie die Disketten auf einer FreeBSD-Maschine erstellen, sollten Sie immer noch formatieren. Allerdings brauchen Sie kein MS-DOS®-Dateisystem auf den Disketten anzulegen. Mit den Kommandos `bsdlabel` und `newfs` können Sie das Dateisystem UFS verwenden, wie im nachstehenden Beispiel für 3.5" 1.44 MB Disketten gezeigt:

```
# fdformat -f 1440 fd0.1440
# bsdlabel -w fd0.1440 floppy3
# newfs -t 2 -u 18 -l 1 -i 65536 /dev/fd0
```

Anschließend können Sie die Disketten wie ein normales Dateisystem einhängen und beschreiben.

Nachdem Sie die Disketten formatiert haben, kopieren Sie die Dateien der Distribution auf die Disketten. Die Dateien der Distribution sind in Stücke geteilt, sodass fünf Dateien auf eine 1.44 MB Diskette passen. Kopieren Sie die gewünschten Distribution auf Disketten, wobei Sie so viele Dateien wie möglich auf eine Diskette kopieren. Jede Distribution wird auf der Diskette in einem eigenen Verzeichnis abgelegt, beispielsweise `a:\base\base.aa` , `a:\base\base.ab` und so weiter.

Wichtig

Die Datei `base.inf` muss unbedingt auf die erste Diskette des `base`-Diskettensatzes kopiert werden, damit das Installationsprogramm feststellen kann, wie viele Disketten geladen werden müssen, um die Distribution wieder zusammenzusetzen.

Im Installationsprozess wählen Sie als Installationsmedium Floppy aus. Folgen Sie dann den gegebenen Anweisungen.

2.13.4. Von einer MS-DOS®-Partition installieren

Um eine Installation von einer MS-DOS®-Partition vorzubereiten, kopieren Sie Dateien der Distributionen in das Verzeichnis `freebsd` direkt unterhalb des Wurzelverzeichnisses (zum Beispiel `c:\freebsd`). In diesem Verzeichnis muss sich dieselbe Verzeichnisstruktur wie auf dem Installationsmedium befinden. Wenn Sie die Dateien von einer Installations-CD kopieren, empfehlen wir den MS-DOS®-Befehl `xcopy`. Das nachstehende Beispiel bereitet eine minimale Installation von FreeBSD vor:

```
C:\> md c:\freebsd
C:\> xcopy e:\bin c:\freebsd\bin\ /s
C:\> xcopy e:\manpages c:\freebsd\manpages\ /s
```

Im Beispiel wurde angenommen, dass auf Laufwerk `C:` ausreichend Platz vorhanden ist und die CD-ROM Laufwerk `E:` ist.

Wenn Sie kein CD-Laufwerk besitzen, können Sie die Distributionen von ftp.FreeBSD.org herunterladen. Jede Distribution liegt in einem eigenen Verzeichnis. Beispielsweise liegt die Base-Distribution im Verzeichnis 10.3/base/.

Kopieren Sie jede Distribution, die Sie von einer MS-DOS®-Partition installieren wollen (und für die Platz ist) in das Verzeichnis `c:\freebsd` . Für eine minimale Installation benötigen Sie nur die Base-Distribution.

2.13.5. Ein Installationsband erstellen

Falls Sie nicht über FTP oder von einer CD-ROM installieren können, ist die Installation von Band wahrscheinlich die einfachste Methode. Das Installationsprogramm erwartet, dass sich die Distributionen im `tar`-Format auf dem Band befinden. Von den Distributions-Dateien erstellen Sie das Installationsband einfach mit dem Kommando `tar`:

```
# cd /freebsd/distdir
# tar cvf /dev/rwt0 dist1 ... dist2
```

Stellen Sie während der Installation sicher, dass Sie über genügend freien Platz in einem temporären Verzeichnis (das Sie festlegen können) verfügen. Das temporäre Verzeichnis muss den *gesamten* Inhalt des Bands aufnehmen können. Da auf Bänder nicht wahlfrei zugegriffen werden kann, benötigt diese Installationsmethode temporär sehr viel Platz.

Anmerkung

Das Band muss sich vor dem Neustart mit der Startdiskette im Laufwerk befinden. Ansonsten wird das Band während der Geräteerkennung vielleicht nicht erkannt.

2.13.6. Eine Netzwerkinstallation vorbereiten

Sie können drei Verbindungsarten für eine Netzwerkinstallation benutzen: Eine Ethernet-Verbindung, eine serielle Verbindung (PPP), oder eine parallele Verbindung (PLIP, Laplink-Kabel).

Die schnellste Netzwerkinstallation ist natürlich mit einer Netzwerkkarte möglich. FreeBSD unterstützt die meisten der üblichen Netzwerkkarten. Eine Liste der unterstützten Netzwerkkarten ist in den Hardware-Notes jedes Releases enthalten. Wenn Sie eine unterstützte PCMCIA-Netzwerkkarte benutzen, stellen Sie sicher, dass die Karte eingesteckt ist, *bevor* der Laptop eingeschaltet wird. Leider unterstützt FreeBSD das Einstecken von PCMCIA-Karten während der Installation noch nicht.

Für eine Netzwerkinstallation müssen Sie Ihre IP-Adresse, die Netzwerkmaske und den Namen Ihres Rechner kennen. Wenn Sie über eine PPP-Verbindung installieren und keine feste IP-Adresse besitzen, braucht Sie der vorgehende Satz nicht zu beunruhigen. Sie können eine IP-Adresse dynamisch von Ihrem ISP beziehen. Fragen Sie Ihren Systemadministrator nach den richtigen Netzwerkeinstellungen. Wenn Sie andere Rechner über Namen anstatt über IP-Adressen erreichen wollen, brauchen Sie zudem einen Nameserver und möglicherweise die Adresse eines Gateways (mit PPP ist das die Adresse des ISPs), über den Sie den Nameserver erreichen. Wenn Sie von einem FTP-Server über einen HTTP-Proxy installieren wollen, benötigen Sie außerdem noch die Adresse des Proxy-Servers. Wenn Sie nicht alle oder zumindest die meisten der benötigten Daten kennen, sollten Sie wirklich *vor* der Installation mit Ihrem Systemadministrator oder ISP reden!

Wenn Sie ein Modem benutzen, ist PPP ziemlich sicher die einzige Wahl. Stellen Sie sicher, dass Sie die Daten Ihres Service Providers bereitliegen haben, da Sie während der Installation die Daten früh benötigen.

Wenn Sie PAP oder CHAP benutzen, um sich mit Ihrem ISP zu verbinden (wenn Sie unter Windows® kein Skript benötigen, um die Verbindung herzustellen), brauchen Sie an der ppp-Eingabeaufforderung nur das Kommando `dial` abzusetzen. Ansonsten müssen Sie sich mit Modem-spezifischen AT-Kommandos bei Ihrem ISP einwählen (PPP stellt nur einen einfachen Terminal-Emulator zur Verfügung). Weiteres über PPP erfahren Sie im Abschnitt User-PPP des Handbuchs und im PPP-Abschnitt der FAQ. Bei Problemen können Sie mit dem Kommando `set log local` Meldungen auf den Bildschirm umleiten.

Wenn eine feste Verbindung zu einer anderen FreeBSD-Maschine besteht, sollten Sie ein paralleles Laplink-Kabel in Betracht ziehen. Über eine parallele Verbindung sind höhrere Geschwindigkeiten als über eine serielle Verbindung (typischerweise bis zu 50 kByte/s) möglich. Daher ist die Installation über eine parallele Verbindung schneller als eine Installation über eine serielle Verbindung.

2.13.6.1. Eine NFS-Installation vorbereiten

Eine NFS-Installation ist unkompliziert. Kopieren Sie einfach die Distributionen auf einen NFS-Server und wählen Sie NFS als Installationsmedium aus.

Wenn der NFS-Server nur Verbindungen über privilegierte Ports (Ports kleiner 1024) annimmt, setzen Sie vor der Installation die Option `NFS Secure` im Menü Options.

Wenn Sie eine schlechte Netzwerkkarte besitzen, die sehr langsam ist, wählen Sie die Option `NFS Slow`.

Damit die NFS-Installation funktioniert, muss der NFS-Server auch Unterverzeichnisse von exportierten Verzeichnissen zum Einhängen freigeben. Wenn beispielsweise die Distribution von FreeBSD 10.3 unter `ziggy:/usr/archi-`

`ve/stuff/FreeBSD` liegt, muss der Rechner `ziggy` erlauben, das Verzeichnis `/usr/archive/stuff/FreeBSD` einzuhängen. Es reicht nicht, dass `ziggy` erlaubt das Verzeichnis `/usr` oder `/usr/archive/stuff` einzuhängen.

Unter FreeBSD werden diese Freigaben in der Datei `/etc/exports` mit der Option `-alldirs` eingestellt. Die nötigen Einstellungen können auf einem anderen NFS-Server unterschiedlich sein. Wenn Sie vom NFS-Server die Fehlermeldung permission denied erhalten, dann haben Sie wahrscheinlich die Freigaben nicht richtig konfiguriert.

Kapitel 3. FreeBSD 9.X (und neuer) installieren

Restructured, reorganized, and parts rewritten by Jim Mock.
Updated for bsdinstall by Gavin Atkinson und Warren Block.
Erweitert um root-on-ZFS von Allan Jude.
Übersetzt von Benedict Reuschling.

3.1. Übersicht

Beginnend mit FreeBSD 9.0-RELEASE enthält FreeBSD ein text-basiertes, einfach zu verwendendes Installationsprogramm namens bsdinstall. Dieses Kapitel beschreibt die Installation von FreeBSD mit Hilfe von bsdinstall. Der Einsatz von sysinstall, das Installationsprogramm von FreeBSD-nbsp;8.x, wird in Kapitel 2, *FreeBSD 8.X (und älter) installieren* behandelt.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie wissen:

- wie man FreeBSD Installationsmedien erstellt.
- wie man bsdinstall startet.
- welche Fragen bsdinstall stellt, was sie bedeuten und wie man diese beantwortet.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- Die Liste von unterstützter Hardware lesen, die mit der zu installierenden Version von FreeBSD ausgeliefert wird, um sicherzustellen, dass die Hardware auch unterstützt wird.

Anmerkung

Generell wurden diese Installationsanweisungen für Rechner der i386™ („PC-kompatibel“) Architektur verfasst. An Stellen, an denen sich die Anweisungen speziell auf eine andere Plattform beziehen, wird darauf hingewiesen. Es mag kleinere Unterschiede geben zwischen dem Installationsprogramm und dem, was hier beschrieben steht. Sie sollten daher dieses Kapitel als eine Art Wegweiser und keine exakte Anleitung betrachten.

3.2. Minimale Hardwarenaforderungen

Die Hardwaranforderungen zur Installation von FreeBSD variieren mit der Version von FreeBSD und der Hardwarearchitektur. Hardwarearchitekturen und von FreeBSD unterstützte Geräte werden in den Hardware Notes aufgelistet. Normalerweise heißt diese Datei `HARDWARE.TXT` und befindet sich im Wurzelverzeichnis des Installationsmediums. Kopien dieser Hardwareliste finden Sie in den Release Information der FreeBSD Webseite (http://www.FreeBSD.org/releases/index.html).

Für die Installation von FreeBSD sind mindestens 64 MB RAM und 1.5 GB freier Festplattenspeicher erforderlich. Das ist jedoch eine minimale Installation, die kaum freien Speicherplatz bietet. Die Anforderungen an den Arbeitsspeicher hängen stark vom Einsatzzweck ab. Während spezialisierte FreeBSD-Systeme mit weniger als 128 MB RAM auskommen, sollten Desktop-Systeme mindestens 4 GB RAM haben.

Die Anforderungen an den Prozessor für jede Architektur lassen sich wie folgt zusammenfassen:

amd64
: Dies ist die häufigste Art von Prozessor für Desktop- und Laptop-Computern. Andere Anbieter nennen diese Architektur auch x86-64.

  Es gibt zwei primäre Anbieter von amd64-Prozessoren: Intel® (produziert Intel64-Prozessoren) und AMD (produziert AMD64).

  Beispiele für amd64-kompatible Prozessoren umfassen: AMD Athlon™64, AMD Opteron™, multi-core Intel® Xeon™ und Intel® Core™ 2 sowie neuere Prozessoren.

i386
: Die 32-Bit x86-Architektur.

  Fast alle i386-kompatiblen Prozessoren mit einer Floating-Point-Einheit werden unterstützt. Alle Intel®-Prozessoren 486 oder neuer werden unterstützt.

  FreeBSD nutzt die *Physical Adress Extensions* (PAE), falls die CPU diese Funktion unterstützt. Wenn PAE im Kernel aktiviert ist, wird Speicher über 4 GB vom Kernel erkannt und kann von System verwendet werden. Diese Funktion bringt auch Einschränkungen für die Gerätetreiber und anderen Komponenten von FreeBSD. Einzelheiten finden Sie in pae(4).

ia64
: Die derzeit unterstützten Prozessoren sind Itanium® und Itanium® 2. Zu den unterstützten Chipsätzen zählen HP zx1, Intel® 460GX und Intel® E8870. Sowohl Einprozessorsysteme (UP) als auch Mehrprozessorsysteme (SMP werden unerstützt.

pc98
: Die NEC PC-9801/9821-Serie mit fast allen i386-kompatiblen Prozessoren, einschließlich 80486, Pentium®, Pentium® Pro und Pentium® II werden unterstützt. Alle i386-kompatiblen Prozessoren von AMD, Cyrix, IBM und IDT werden ebenfalls unterstützt. Ebenso die EPSON PC-386/486/586-Serie, welche mit der NEC-Serie PC-9801 kompatibel ist. NEC FC-9801/9821 und die NEC SV-98-Serie sollte ebenfalls unterstützt werden.

  Der hochauflösende Modus wird nicht unterstützt. NEC PC-98XA/XL/RL/XL^2 und die NEC PC-H98-Serie werden nur im normalen (PC-9801 kompatibel) Modus unterstützt. Die SMP bezogenen Funktionen von FreeBSD werden nicht unterstützt. Auch der *New Extend Standard Architecture* (NESA) Bus, verwendet in den Serien PC-H98, SV-H98 und FC-H98, wird nicht unterstützt.

powerpc
: Alle New Word ROM Apple® Mac®-Systeme mit integriertem USB werden unterstützt. SMP wird auf Maschinen mit mehreren CPUs unterstützt.

  Ein 32-Bit Kernel kann jedoch nur die ersten 2 GB RAM verwenden.

sparc64
: Systeme, die von FreeBSD/sparc64 unterstützt werden, sind auf der FreeBSD/sparc64-Projektseite aufgelistet (http://www.FreeBSD.org/platforms/sparc.html").

  SMP wird auf allen Systemen mit mehr als einem Prozessor unterstützt. Eine dedizierte Platte wird benötigt, da es nicht möglich ist, eine Platte mit einem anderen Betriebssystem zur gleichen Zeit zu teilen.

3.3. Vor der Installation

3.3.1. Sichern Sie wichtige Daten

Sichern Sie alle wichtigen Daten auf dem Zielcomputer, auf dem FreeBSD installiert werden soll. überprüfen Sie diese Sicherungen, bevor Sie fortfahren. Die FreeBSD Installation wird Sie vor Änderungen an den Platten danach fragen, jedoch kann dies nicht mehr rückgängig gemacht werden, sobald der Prozess gestartet wurde.

3.3.2. Den Installationsort von FreeBSD festlegen

Falls FreeBSD das einzige installierte Betriebssystem sein wird und die gesamte Platte dazu verwenden kann, kann der Rest dieses Abschnitts übersprungen werden. Sollten Sie allerdings die Platte mit anderen Betriebssystemen teilen, ist ein Verständnis des Plattenlayouts hilfreich für die Installation.

3.3.2.1. Festplattenlayout für FreeBSD/i386 und FreeBSD/amd64

Festplatten können in mehrere verschiedene Bereiche aufgeteilt werden. Diese Bereiche werden *Partitionen* genannt.

Es gibt zwei Arten, eine Festplatte in mehrere Partitionen einzuteilen. Traditionell enthält ein *Master Boot Record* (MBR) eine Partitionstabelle, welche bis zu vier *primäre Partitionen* aufnehmen kann. Aus historischen Gründen werden diese primären Partitionen in FreeBSD *slices* genannt. Eine Begrenzung von nur vier Partitionen ist für große Platten sehr beschränkt, so dass eine dieser primären Partitionen als *erweiterte Partition* eingesetzt wird. Mehrere *logische Partitionen* können dann innerhalb der erweiterten Partition angelegt werden. Dies mag etwas unhandlich erscheinen und das ist auch der Fall.

Die *GUID-Partitionstabelle* (GPT) ist eine neuere und einfachere Methode zur Partition einer Festplatte. GPT ist weitaus flexibler als die traditionelle MBR-Partitionstabelle. Geläufige GPT-Implementierungen erlauben bis zu 128 Partitionen pro Platte, was die Notwendigkeit von umständlichen Behelfen wie logische Partitionen eliminiert.

Warnung

Manche älteren Betriebssysteme wie Windows® XP sind mit dem GPT-Partitionsschema nicht kompatibel. Wenn sich FreeBSD die Platte mit einem solchen Betriebssystem teilen soll, werden MBR Partitionen benötigt.

FreeBSDs Standard-Bootloader benötigt entweder eine primäre oder eine GPT-Partition. Lesen Sie dazu Kapitel 13, *FreeBSDs Bootvorgang* für weitere Informationen zum FreeBSD Bootvorgang. Wenn alle der primären oder GPT-Partitionen bereits in Verwendung sind, muss eine davon für FreeBSD zur Verfügung gestellt werden.

Eine Minimalinstallation von FreeBSD braucht ungefähr 1 GB Plattenplatz. Dies ist jedoch eine *sehr* minimale Installation, die fast gar keinen freien Speicherplatz übrig lässt. Eine etwas realistischere Minimalangabe sind 3 GB ohne eine graphische Umgebung und 5 GB oder mehr, falls eine graphische Benutzeroberfläche verwendet werden soll. Anwendungen von Drittanbietern benötigt sogar noch mehr Platz.

Eine Vielzahl freier und kommerzieller Werkzeuge zur Veränderung der Partitionsgrößen finden Sie unter http://en.wikipedia.org/wiki/List_of_disk_partitioning_software. GParted Live (http://gparted.sourceforge.net/livecd.php) ist eine freie Live-CD, die den GParted-Partitionseditor enthält. GParted ist auch in einer Vielzahl von anderen Linux Live-CD Distributionen enthalten.

Warnung

Anwendungen zur Festplattenpartition kann Daten zerstören. Erstellen Sie eine Vollsicherung und überprüfen Sie deren Integrität bevor Sie die Partitionen auf der Platte verändern.

Die Veränderung der Grösse von Microsoft® Vista-Partitionen kann schwierig sein. Eine Vista Installations-CD kann hilfreich sein, wenn eine solche Aktion versucht wird.

Beispiel 3.1. Eine existierende Partition verändern

Ein Windows®-Computer besitzt eine einzelne 40 GB Platte, die in zwei 20 GB Partitionen aufgeteilt wurde. Windows® nennt diese `C:` und `D:`. Die `C:` Partition enthält 10 GB und die `D:` Partition 5 GB an Daten.

Durch kopieren der Daten von `D:` nach `C:` macht die zweite Partition frei, so dass FreeBSD sie benutzen kann.

Beispiel 3.2. Verkleinern einer bestehenden Partition

Ein Windows®-Computer besitzt eine einzelne 40 GB Platte und eine grosse Partition, welche die gesamte Platte einnimmt. Windows® zeigt diese 40 GB Partition als einzelne `C:` Partition. 15 GB Plattenplatz wird verwendet. Das Ziel ist, für Windows® eine 20 GB Partition einzurichten und eine weitere 20 GB Partition für FreeBSD bereitzustellen.

Es gibt zwei Wege, dieses Ziel zu erreichen:

1. Sichern Sie alle Windows®-Daten. Installieren Sie dann Windows® neu, indem Sie eine 20 GB Partition während der Installation anlegen.

2. Verwenden Sie ein Werkzeug zur Veränderung einer Partition wie GParted, um die Windows®-Partition zu verkleinern und eine neue Partition im freigewordenen Plattenplatz für FreeBSD anzulegen.

Festplattenpartitionen, die unterschiedliche Betriebssysteme enthalten, ermöglichen es, jeweils eines dieser Systeme zu verwenden. Eine andere Methode, die es erlaubt, mehrere Betriebssysteme gleichzeitig einzusetzen, wird im Kapitel 22, *Virtualisierung* behandelt.

3.3.3. Netzwerkparameter ermitteln

Manche FreeBSD Installationsarten benötigen eine Netzwerkverbindung, um Dateien herunter zu laden. Um zu einem Ethernet-Netzwerk (bzw. Kabel oder DSL-Modem mit einem Ethernet-Anschluss) eine Verbindung herzustellen, wird das Installationsprogramm bestimmte Information zum Netzwerk abfragen.

DHCP wird allgemein verwendet, um automatisch Netzwerkeinstellungen vorzunehmen. Falls DHCP nicht verfügbar ist, müssen diese Netzwerkeinstellungen vom lokalen Netzwerkadministrator oder Provider erfragt werden:

1. IP-Adresse

2. Subnetz-Maske

3. Default-Router IP-Adresse

4. Domänenname des lokalen Netzwerks

5. DNS-Server IP-Adresse(n)

3.3.4. Lesen Sie die FreeBSD-Errata

Obwohl das FreeBSD Projekt sich bemüht, jede veröffentlichte Version von FreeBSD so stabil wie möglich zu machen, können sich doch gelegentlich Fehler in den Veröffentlichungsprozess einschleichen. In sehr seltenen

Fällen betreffen diese Fehler den Installationsvorgang. Wenn diese Probleme entdeckt und behoben sind, werden dazu Hinweise in der FreeBSD Errata (http://www.freebsd.org/releases/10.3R/errata.html auf der FreeBSD Webseite veröffentlicht. Prüfen Sie die Errata vor der Installation, um sicherzustellen, dass es keine Probleme gibt, welche die Installation betreffen.

Informationen und Errata für all diese Veröffentlichungen finden Sie unter den Release Informationen auf der FreeBSD Webseite (http://www.freebsd.org/releases/index.html).

3.3.5. Die Installationsmedien beschaffen

Eine FreeBSD-Installation wird durch das starten des Computers mit einer eingelegten FreeBSD-Installations-CD/DVD oder eines USB-Sticks begonnen. Das Installationsprogramm ist kein Programm das aus einem anderen Betriebssystem heraus gestartet werden kann.

Zusätzlich zum Standardinstallationsmedium, welches Kopien von allen FreeBSD-Installationsdateien enthält, gibt es auch eine *bootonly*-Variante. Ein solches Installationsmedium besitzt keine Kopien der Installationsdateien, jedoch kann es diese während der Installation aus dem Netzwerk nachladen. Die bootonly Installations-CD ist dadurch viel kleiner und reduziert die benötigte Bandbreite während der Installation durch herunterladen der allernötigsten Dateien.

Kopien der FreeBSD-Installationsmedien sind unter www.freebsd.org/where.html#download verfügbar. Laden Sie auch `CHECKSUM.SHA256` aus dem gleichen Verzeichnis herunter, um die Integrität der Installationsmedien anhand einer Prüfsumme zu berechnen. FreeBSD bietet hierfür sha256(1) an, andere Betriebsysteme haben ähnliche Programme. Vergleichen Sie die Prüfsumme mit der in `CHECKSUM.SHA256` . Wenn die beiden Prüfsummen nicht übereinstimmen, ist die Datei beschädigt und sollte erneut heruntergeladen werden.

Tipp

Falls Sie bereits eine Kopie von FreeBSD auf CD, DVD oder USB-Stick besitzen, kann dieser Abschnitt übersprungen werden.

CD und DVD-Images von FreeBSD sind startfähige ISO-Dateien. Nur eine CD oder DVD wird für eine Installation benötigt. Brennen Sie ein ISO-Image auf eine startfähige CD oder DVD mit Hilfe eines CD-Brennprogramms, das für das aktuelle Betriebssystem zur Verfügung steht.

Um einen startfähigen USB-Stick zu erstellen, führen Sie die folgenden Schritte durch:

1. Das Image für den USB-Stick herunterladen

   Das Image für FreeBSD 9.0-RELEASE und höhere Versionen kann aus dem `ISO-IMAGES/`-Verzeichnis unter `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/` *`arch`*`/`*`arch`*`/ISO-IMAGES/version/FreeBSD-` *`version`*`-RELEASE-` *`arch`*`-memstick.img` bezogen werden. Ersetzen Sie jeweils *`arch`* und *`version`* mit der Architektur und der Versionsnummer, die installiert werden soll. Beispielsweise sind die USB-Stick Images für FreeBSD/i386 9.0-RELEASE verfügbar unter `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/i386/ISO-IMAGES/9.0/FreeBSD-9.0-RELEASE-i386-memstick.img` .

   Tipp

   Für FreeBSD 8.*X* und frühere Versionen wird ein anderer Pfad verwendet. Details für das Herunterladen und Installieren von FreeBSD 8.*X* und frühere werden im Kapitel 2, *FreeBSD 8.X (und älter) installieren* behandelt.

Das USB-Stick Image hat die Endung `.img`. Das Verzeichnis `ISO-IMAGES/` enthält eine Vielzahl von verschiedenen Installations-Images und die jeweils benötigte Version von FreeBSD, sowie in manchen Fällen die Zielhardware.

Wichtig

Bevor Sie fortfahren, *machen Sie Sicherungskopien* der Daten auf dem USB-Stick, da die folgende Prozedur alle Daten *löscht*.

2. Das Image auf den USB-Stick schreiben

Prozedur 3.1. Den USB-Stick unter FreeBSD vorbereiten

Warnung

Das Beispiel unten verwendet `/dev/da0` als das Zielgerät, auf welches das Image geschrieben werden soll. Seien Sie vorsichtig, dass das richtige Gerät als Ausgabe benutzt wird, da das Kommando alle vorhandenen Daten zerstört.

- Das Image mit dd(1) schreiben

  Die `.img`-Datei ist *keine* gewöhnliche Datei. Es ist ein *Image* des kompletten späteren Inhalts des USB-Sticks. Es kann *nicht* wie eine gewöhnliche Datei kopiert werden, sondern muss direkt auf das Zielgerät mit dd(1) geschrieben werden:

```
# dd if=FreeBSD-9.0-RELEASE-i386-memstick.img of=/dev/   da0 bs=64k
```

Prozedur 3.2. Das Image unter Windows® schreiben

Warnung

Versichern Sie sich, dass Sie den korrekten Laufwerksbuchstaben als Ausgabe angeben, da die bestehenden Daten überschrieben und zerstört werden.

1. Image Writer für Windows® herunterladen

   Image Writer für Windows® ist eine frei verfügbare Anwendung, welche eine Imagedatei korrekt auf einen USB-Stick schreiben kann. Laden Sie diese von `https://launchpad.net/win32-image-writer/` herunter und entpacken Sie sie in einen Ordner.

2. Das Image mit Image Writer auf den USB-Stick schreiben

   Klicken Sie doppelt auf das Win32DiskImager-Icon, um das Programm zu starten. Prüfen Sie dabei, dass der Laufwerksbuchstabe unter `Device` dem Gerät entspricht, in dem sich der USB-Stick befindet. Klicken Sie auf das Ordnersymbol und wählen Sie das Image aus, welches auf den USB-Stick geschrieben werden soll. Um den Image-Dateinamen zu akzeptieren, klicken Sie auf [Save]. Überprüfen Sie erneut, ob alles stimmt und dass keine Ordner auf dem USB-Stick in anderen Fenstern geöffnet sind. Sobald alles bereit ist, klicken Sie auf [Write], um die Imagedatei auf den USB-Stick zu schreiben.

Anmerkung

Die Installation von Disketten wird nicht mehr unterstützt.

Sie sind jetzt dazu bereit, mit der Installation von FreeBSD zu beginnen.

3.4. Die Installation starten

Wichtig

Es werden bei Installation so lange keine Änderungen an den Festplatten durchgeführt, bis die folgende Meldung erscheint:

```
Your changes will now be written to disk.  If you
have chosen to overwrite existing data, it will
be PERMANENTLY ERASED. Are you sure you want to
commit your changes?
```

Die Installation kann vor dieser Warnung zu jeder Zeit abgebrochen werden, ohne dass die Inhalte der Festplatte davon betroffen sind. Falls Zweifel bestehen, dass etwas falsch konfiguriert wurde, schalten Sie einfach den Computer vor diesem Punkt aus und es wird kein Schaden angerichtet.

3.4.1. Der Systemstart

3.4.1.1. Systemstart von i386™ und amd64

1. Falls Sie einen „startfähigen“ USB-Stick einsetzen, wie in Abschnitt 3.3.5, „Die Installationsmedien beschaffen“ beschrieben ist, dann stecken Sie diesen vor dem Anschalten des Computers hinein.

   Falls die Installation von einer CD startet, müssen Sie den Computer anschalten und die CD so bald wie möglich einlegen.

2. Konfigurieren Sie den Rechner so, dass er entweder von der CD oder dem USB-Stick startet, abhängig vom verwendeten Installationsmedium. Die Konfiguration im BIOS erlaubt es, das Gerät, von dem gestartet werden soll, auszuwählen. Die meisten Systeme erlauben es auch, das Startgerät während des Startvorgangs zu wählen, typischerweise durch drücken von entweder F10, F11, F12 oder Escape.

3. Falls der Computer wie normal startet und das bestehende Betriebssystem lädt, befolgen Sie einen der hier aufgeführten Schritte:

   1. Die Medien wurden während des Startvorgangs nicht früh genug eingelegt. Lassen Sie diese wo sie sind und versuchen Sie, den Rechner davon neu zu starten.

   2. Die Änderungen am BIOS haben nicht richtig funktioniert. Wiederholen Sie diesen Schritt, um die richtige Option auszuwählen.

   3. Das verwendete BIOS unterstützt das starten vom gewählten Medium nicht. Der Plop Boot Manager (http://www.plop.at/en/bootmanager.html) kann in diesem Fall verwendet werden, um ältere Computer von CD oder USB-Medien zu starten.

4. FreeBSD wird anfangen zu starten. Falls das System von CD startet, werden Meldungen ähnlich wie die folgenden angezeigt (Versionsinformationen wurden hier entfernt):

```
Booting from CD-ROM...
645MB medium detected
CD Loader 1.2

Building the boot loader arguments
Looking up /BOOT/LOADER... Found
Relocating the loader and the BTX
Starting the BTX loader

BTX loader 1.00 BTX version is 1.02
Consoles: internal video/keyboard
BIOS CD is cd0
BIOS drive C: is disk0
BIOS drive D: is disk1
BIOS 636kB/261056kB available memory

FreeBSD/i386 bootstrap loader, Revision 1.1

Loading /boot/defaults/loader.conf
/boot/kernel/kernel text=0x64daa0 data=0xa4e80+0xa9e40 syms=[0x4+0x6cac0+0x4+0x88e9d]
\
```

5. Der FreeBSD-Bootloader wird angezeigt:

Abbildung 3.1. Das FreeBSD-Bootloader Menü

Warten Sie entweder zehn Sekunden oder drücken Sie Enter.

3.4.1.2. Systemstart beim Macintosh® PowerPC®

Auf den meisten Maschinen können Sie C auf der Tastatur gedrückt halten, um von der CD zu starten. Andernfalls, halten Sie Command+Option+O+F, oder Windows+Alt+O+F auf nicht-Apple® Tastaturen gedrückt. Geben Sie an der `0 >`-Eingabeaufforderung folgendes ein:

```
boot cd:,\ppc\loader cd:0
```

3.4.1.3. Systemstart für SPARC64®

Die meisten SPARC64®-Systeme sind so eingerichtet, dass diese automatisch von CD starten. Um FreeBSD zu installieren, muss es über das Netzwerk oder von einer CD gestartet werden, was es nötig macht, in die PROM OpenFirmware einzubrechen.

Um dies zu tun, starten Sie das System neu und warten Sie bis die Startmeldungen erscheinen. Abhängig vom Modell sollte dies in etwa folgendermaßen aussehen:

```
Sun Blade 100 (UltraSPARC-IIe), Keyboard Present
Copyright 1998-2001 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.2, 128 MB memory installed, Serial #51090132.
Ethernet address 0:3:ba:b:92:d4, Host ID: 830b92d4.
```

Falls das System jetzt damit fortfährt von Platte zu starten, müssen Sie L1+A oder Stop+A auf der Tastatur eingeben oder ein `BREAK`-Kommando (indem Sie z.B. `~#` in tip(1) oder cu(1) absetzen) über die serielle Konsole senden, um zur PROM Befehlszeile zu gelangen. Es sieht dann so aus:

```
ok    ❶
ok {0} ❷
```

❶ Dies ist die Eingabeaufforderung, welche auf Systemen mit einer CPU verwendet wird.
❷ Dies ist die Eingabeaufforderung auf SMP-Systemen. Die Zahl gibt die Nummer der aktiven CPU an.

Legen Sie nun die CD in das Laufwerk und geben Sie am PROM-Prompt `boot cdrom` ein.

3.4.2. Die Geräteerkennung prüfen

Die letzten hundert Zeilen, die am Bildschirm angezeigt wurden, sind gespeichert worden und können erneut abgerufen werden.

Um diesen Puffer anzusehen, drücken Sie Scroll Lock. Das bewirkt, dass Sie die Bildschirmanzeige hoch und runter bewegen (scrollen) können. Benutzen Sie die Pfeiltasten oder PageUp und PageDown, um die Meldungen zu sehen. Drücken Sie Scroll Lock erneut, um das scrollen zu stoppen.

Tun Sie dies jetzt, um den Text, der aus den Bildschirm gelaufen ist, als der Kernel die Geräteerkennung durchgeführt hat, erneut zu prüfen. Sie werden einen Text ähnlich zu Abbildung 3.2, „Typical Device Probe Results“ sehen, obwohl sich der genaue Text, abhängig von den Geräten im Computer, unterscheiden wird.

```
Copyright (c) 1992-2011 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.
FreeBSD 9.0-RELEASE #0 r225473M: Sun Sep 11 16:07:30 BST 2011
    root@psi:/usr/obj/usr/src/sys/GENERIC amd64
CPU: Intel(R) Core(TM)2 Duo CPU     T9400  @ 2.53GHz (2527.05-MHz K8-class CPU)
  Origin = "GenuineIntel"  Id = 0x10676  Family = 6  Model = 17  Stepping = 6
  ↺
Features=0xbfebfbff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,APIC,SEP,MTRR,PGE,MCA,CMOV,PAT,PSE36,CLFLUSH,DTS,ACPI
  Features2=0x8e3fd<SSE3,DTES64,MON,DS_CPL,VMX,SMX,EST,TM2,SSSE3,CX16,xTPR,PDCM,SSE4.1>
  AMD Features=0x20100800<SYSCALL,NX,LM>
  AMD Features2=0x1<LAHF>
  TSC: P-state invariant, performance statistics
real memory  = 3221225472 (3072 MB)
avail memory = 2926649344 (2791 MB)
Event timer "LAPIC" quality 400
ACPI APIC Table: <TOSHIB A0064   >
FreeBSD/SMP: Multiprocessor System Detected: 2 CPUs
FreeBSD/SMP: 1 package(s) x 2 core(s)
 cpu0 (BSP): APIC ID:  0
 cpu1 (AP): APIC ID:  1
ioapic0: Changing APIC ID to 1
ioapic0 <Version 2.0> irqs 0-23 on motherboard
kbd1 at kbdmux0
acpi0: <TOSHIB A0064> on motherboard
acpi0: Power Button (fixed)
acpi0: reservation of 0, a0000 (3) failed
acpi0: reservation of 100000, b6690000 (3) failed
Timecounter "ACPI-safe" frequency 3579545 Hz quality 850
acpi_timer0: <24-bit timer at 3.579545MHz> port 0xd808-0xd80b on acpi0
cpu0: <ACPI CPU> on acpi0
ACPI Warning: Incorrect checksum in table [ASF!] - 0xFE, should be 0x9A (20110527/
tbutils-282)
cpu1: <ACPI CPU> on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci0: <ACPI PCI bus> on pcib0
vgapci0: <VGA-compatible display> port 0xcff8-0xcfff mem ↺
0xff400000-0xff7fffff,0xe0000000-0xefffffff irq 16 at device 2.0 on pci0
agp0: <Intel GM45 SVGA controller> on vgapci0
agp0: aperture size is 256M, detected 131068k stolen memory
vgapci1: <VGA-compatible display> mem 0xffc00000-0xffcfffff at device 2.1 on pci0
pci0: <simple comms> at device 3.0 (no driver attached)
em0: <Intel(R) PRO/1000 Network Connection 7.2.3> port 0xcf80-0xcf9f mem ↺
0xff9c0000-0xff9dffff,0xff9fe000-0xff9fefff irq 20 at device 25.0 on pci0
em0: Using an MSI interrupt
em0: Ethernet address: 00:1c:7e:6a:ca:b0
uhci0: <Intel 82801I (ICH9) USB controller> port 0xcf60-0xcf7f irq 16 at device 26.0 on ↺
pci0
usbus0: <Intel 82801I (ICH9) USB controller> on uhci0
uhci1: <Intel 82801I (ICH9) USB controller> port 0xcf40-0xcf5f irq 21 at device 26.1 on ↺
pci0
usbus1: <Intel 82801I (ICH9) USB controller> on uhci1
uhci2: <Intel 82801I (ICH9) USB controller> port 0xcf20-0xcf3f irq 19 at device 26.2 on ↺
pci0
usbus2: <Intel 82801I (ICH9) USB controller> on uhci2
ehci0: <Intel 82801I (ICH9) USB 2.0 controller> mem 0xff9ff800-0xff9ffbff irq 19 at ↺
device 26.7 on pci0
usbus3: EHCI version 1.0
usbus3: <Intel 82801I (ICH9) USB 2.0 controller> on ehci0
hdac0: <Intel 82801I High Definition Audio Controller> mem 0xff9f8000-0xff9fbfff irq 22 ↺
at device 27.0 on pci0
pcib1: <ACPI PCI-PCI bridge> irq 17 at device 28.0 on pci0
pci1: <ACPI PCI bus> on pcib1
iwn0: <Intel(R) WiFi Link 5100> mem 0xff8fe000-0xff8fffff irq 16 at device 0.0 on pci1
pcib2: <ACPI PCI-PCI bridge> irq 16 at device 28.1 on pci0
pci2: <ACPI PCI bus> on pcib2
pcib3: <ACPI PCI-PCI bridge> irq 18 at device 28.2 on pci0
pci4: <ACPI PCI bus> on pcib3
pcib4: <ACPI PCI-PCI bridge> at device 30.0 on pci0
pci5: <ACPI PCI bus> on pcib4
cbb0: <RF5C476 PCI-CardBus Bridge> at device 11.0 on pci5
cardbus0: <CardBus bus> on cbb0
pccard0: <16-bit PCCard bus> on cbb0
isab0: <PCI-ISA bridge> at device 31.0 on pci0
```

Abbildung 3.2. Typical Device Probe Results

Prüfen Sie die Ergebnisse der Geräteerkennung genau, um sicher zu stellen, dass FreeBSD alle Geräte gefunden hat. Falls ein Gerät nicht gefunden wurde, wird es auch nicht aufgelistet. Abschnitt 9.2, „Wieso einen eigenen Kernel bauen?“ erklärt, wie Sie Unterstützung für Geräte, die nicht im `GENERIC` -Kernel vorhanden sind, hinzuzufügen.

Nach der Geräteerkennungsprozedur, werden Sie das Menü aus Abbildung 3.3, „Auswahl der Verwendung des Installationsmediums“ sehen. Das Installationsmedium kann auf drei Arten verwendet werden: um FreeBSD zu installieren, als Live-CD wie in Abschnitt 3.11, „Verwendung der Live-CD“ beschrieben, oder um eine FreeBSD-Shell zu öffnen. Benutzen Sie die Pfeiltasten, um eine Option auszuwählen und drücken Sie Enter zum bestätigen.

Abbildung 3.3. Auswahl der Verwendung des Installationsmediums

Wählen Sie hier [Install], gelangen Sie in das Installationsprogramm.

3.5. Das bsdinstall-Werkzeug

bsdinstall ist ein textbasiertes FreeBSD Installationsprogramm, das von Nathan Whitehorn <nwhitehorn@FreeBSD.org> geschrieben und im Jahr 2011 für FreeBSD 9.0 vorgestellt wurde.

Anmerkung

Kris Moore <kmoore@FreeBSD.org>s pc-sysinstall ist in PC-BSD enhalten und kann verwendet werden, um FreeBSD nach den Anweisungen in http://wiki.pcbsd.org/index.php/Use_PC-BSD_Installer_to_Install_FreeBSD zu installieren. Obwohl es manchmal mit bsdinstall verwechselt wird, sind die beiden Programme nicht miteinander verwandt.

Das bsdinstall Menüsystem wird durch die Pfeiltasten gesteuert, Enter, Tab, Space und andere Tasten.

3.5.1. Die Tastaturbelegung auswählen

Abhängig davon, welche Systemkonsole verwendet wird, fragt bsdinstall am Anfang ab, ob eine nicht-Standard Tastaturbelegung festgelegt werden soll.

Abbildung 3.4. Tastaturbelegung festlegen

Wenn [YES] ausgewählt wird, wird der folgende Tastaturauswahlbildschirm angezeigt. Andernfalls wird dieser Auswahlbildschirm nicht gezeigt und eine Standardtastaturbelegung genutzt.

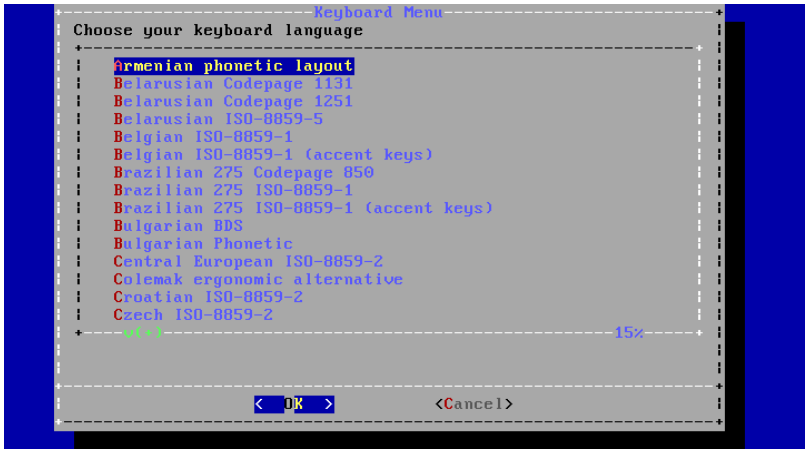


Abbildung 3.5. Tastaturauswahlbildschirm

Wählen Sie die Tastenbelegung, die der am System angeschlossenen Tastatur am nächsten kommt, indem Sie die Pfeiltasten Hoch/Runter verwenden und anschliessend Enter drücken.

Anmerkung

Durch drücken von Esc wird die Standardbelegung eingestellt. United States of America ISO-8859-1 ist eine sichere Option, falls Sie sich unsicher sind, welche Auswahl Sie treffen sollen.

In FreeBSD 10.0-RELEASE und neueren Versionen wurde dieses Menü erweitert. Die vollständige Auswahl der Tastaturbelegungen wird nun angezeigt und die Voreinstellung ist ausgewählt. Wird die Tastenbelegung geändert, erscheint ein Dialog, in dem der Benutzer die Einstellung testen kann, bevor die Installation fortgesetzt wird.

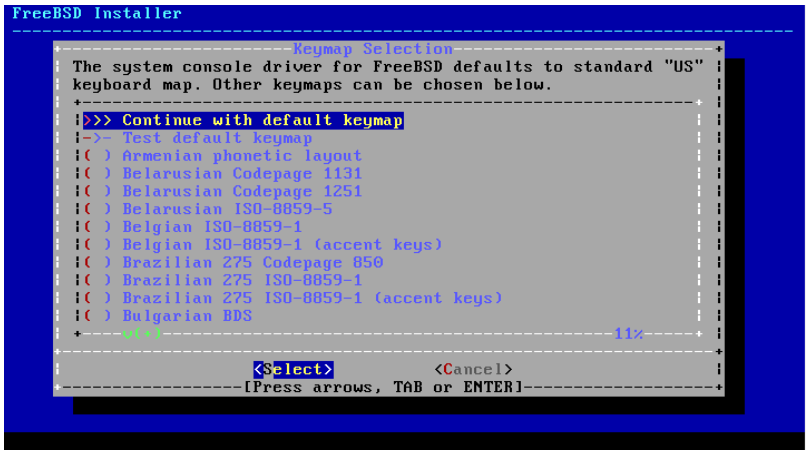


Abbildung 3.6. Erweiterte Tastaturbelegung

3.5.2. Den Rechnernamen festlegen

Als nächstes fragt Sie bsdinstall nach dem Rechnernamen, der in dem neu zu installierenden System verwendet werden soll.

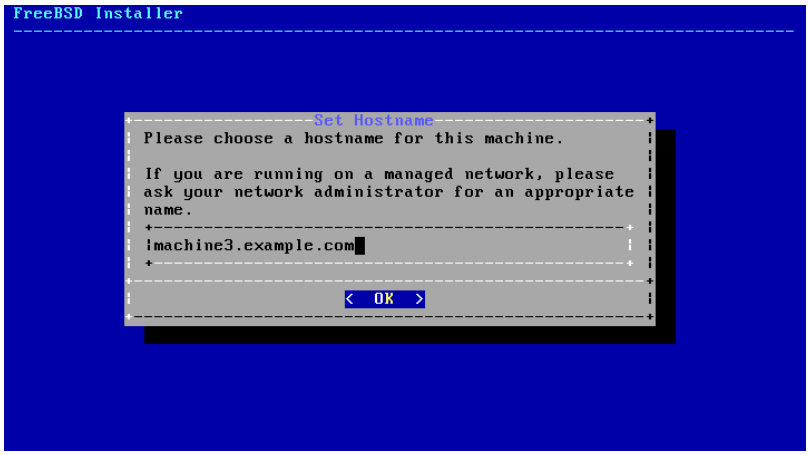


Abbildung 3.7. Festlegen des Rechnernamens

Der eingegebene Rechnername sollte ein voll-qualifizierter Rechnername sein, so wie z.B. `machine3.example.com`

3.5.3. Auswahl der zu installierenden Komponenten

Im nächsten Schritt fragt Sie bsdinstall, die optionalen Komponenten für die Installation auszuwählen.

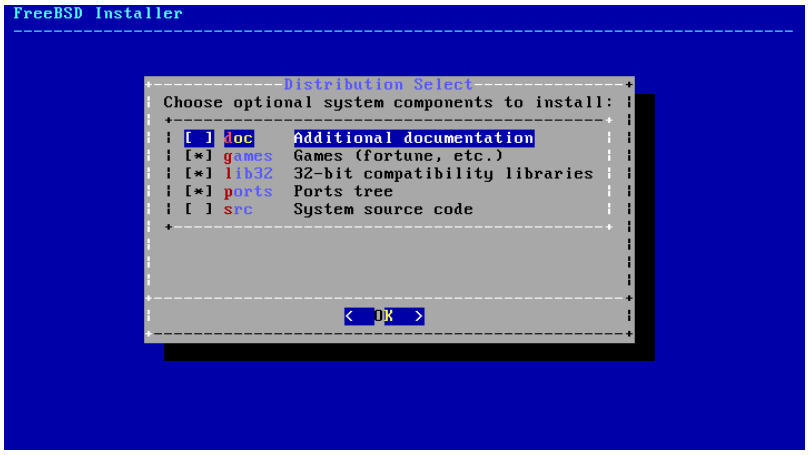


Abbildung 3.8. Komponenten für die Installation auswählen

Die Entscheidung, welche Komponenten auszuwählen sind, hängt grösstenteils davon ab, für was das System künftig eingesetzt werden soll und der zur Verfügung stehende Plattenplatz. Der FreeBSD-Kernel und die Systemprogramme (zusammengenommen auch als „Basissystem“ bezeichnet) werden immer installiert.

Abhängig vom Typ der Installation, werden manche dieser Komponenten nicht erscheinen.

- `doc` - Zusätzliche Dokumentation, meistens eher von historischem Interesse. Dokumentation, wie Sie vom FreeBSD Dokumentationsprojekt bereitgestellt wird, kann zu einem späteren Zeitpunkt noch installiert werden.

- `games` - Mehrere traditionelle BSD-Spiele, darunter fortune, rot13 und andere.

- `lib32` - Kompatibilitäts-Bibliotheken, um 32-bit Anwendungen auf der 64-bit Version von FreeBSD laufen zu lassen.

- `ports` - Die FreeBSD Ports-Sammlung.

  Die Ports-Sammlung stellt eine einfache und praktische Art dar, Software zu installieren. Die Ports-Sammlung enthält nicht den nötigen Quellcode, um die Software zu erstellen. Stattdessen handelt es sich um einen Sammlung von Dateien, die das herunterladen, erstellen und installieren von Drittanbietersoftware automatisiert. Kapitel 5, *Installieren von Anwendungen: Pakete und Ports* behandelt die Verwendung der Ports-Sammlung.

  > **Warnung**
  >
  > Das Installationsprogramm prüft nicht, ob genügend Plattenplatz zur Verfügung steht. Wählen Sie diese Option nur, wenn die Festplatte über ausreichend Speicher verfügt. Die Ports-Sammlung nimmt etwa 500 MB Plattenplatz ein.

- `src` - Quellcode für das System.

  FreeBSD wird mit allen Quellen für den Kernel und die Systemprogramme ausgeliefert. Obwohl dies für die meisten Anwendungen nicht benötigt wird, kann es doch für manche Software, die als Quellcode verbreitet wird (beispielsweise Gerätetreiber oder Kernelmodule), oder um an FreeBSD selbst mitzuentwickeln, notwendig sein.

  Der komplette Quellcodebaum benötigt 1 GB Plattenplatz und um das gesamte Betriebssystem neu zu erstellen, werden zusätzliche 5 GB Platz benötigt.

3.6. Installation aus dem Netzwerk

Die *bootonly*-Installationsmedien enthält keine Kopien der Installationsdateien. Wenn eine *bootonly*-Installationsmethode verwendet wird, müssen die Dateien über eine Netzwerkverbindung übertragen werden, sobald diese benötigt werden.

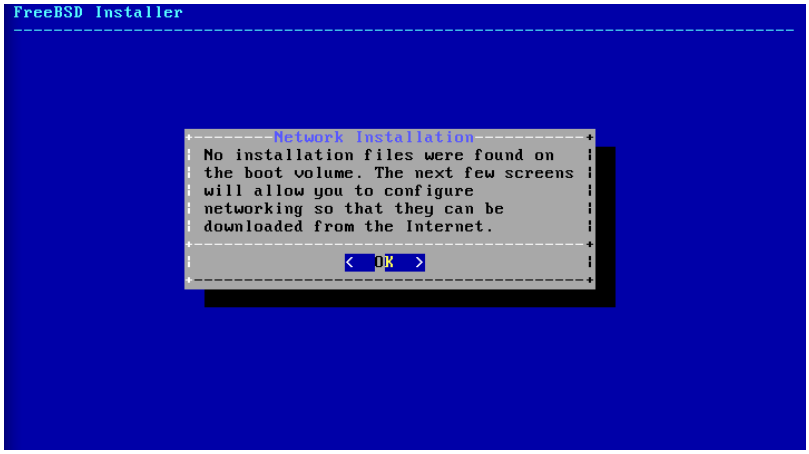


Abbildung 3.9. Installation über das Netzwerk

Nachdem die Netzwerkverbindung wie in Abschnitt 3.9.2, „Die Netzwerkschnittstelle konfigurieren“ konfiguriert wurde, kann ein Spiegelserver ausgewählt werden. Spiegelserver dienen zur Zwischenspeicherung von Kopien der FreeBSD-Dateien. Wählen Sie einen Spiegelserver, welcher in der gleichen Region auf der Welt beheimatet ist, wie der Computer, auf dem FreeBSD installiert werden soll. Dateien können so viel schneller übertragen werden, wenn der Spiegelserver sich näher am Zielcomputer befindet und die Installationszeit wird somit reduziert.

Abbildung 3.10. Einen Spiegelserver wählen

Die Installation wird auf die gleiche Weise fortfahren, als würden die Installationsdateien auf einem lokalen Medium vorliegen.

3.7. Plattenplatz bereitstellen

Es gibt vier Arten, Plattenplatz für FreeBSD zur Verfügung zu stellen. *Geführte* Partitionierung richtet Partitionen automatisch ein, während *manuelle* Partitionierung es fortgeschrittenen Anwendern erlaubt, selbstgewählte Partitionen zu erzeugen. *ZFS* Partitionierung, die ab FreeBSD 10 verfügbar ist, erstellt ein *root-on-ZFS* System mit Unterstützung für *Boot Environments*, welches optional verschlüsselt werden kann. Schließlich gbt es noch die Option

eine Shell zu starten, in der Kommandozeilenprogramme wie gpart(8), fdisk(8) und bsdlabel(8) direkt ausgeführt werden können.

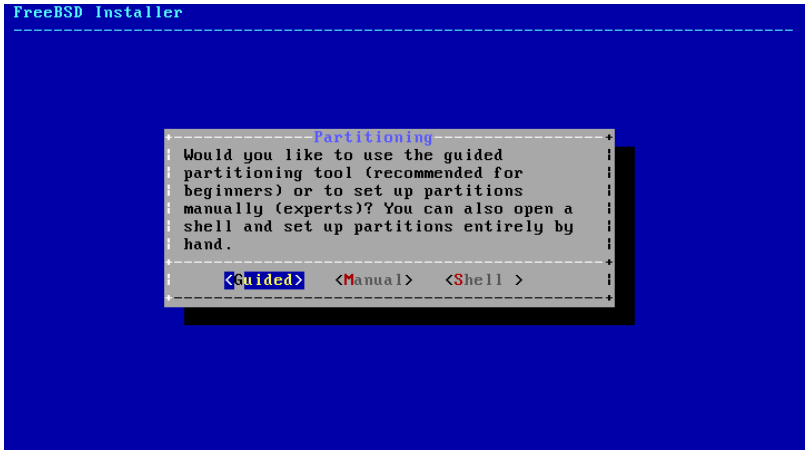


Abbildung 3.11. Geführte, manuelle oder Shell Partitionierung auswählen

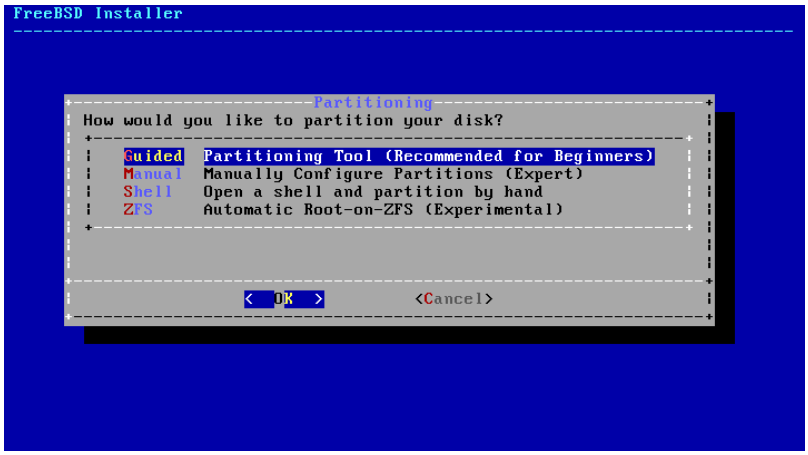


Abbildung 3.12. Geführte, manuelle, Shell oder ZFS Partitionierung auswählen

Dieser Abschnitt beschreibt, was bei der Partitionierung der Platten zu beachten ist und wie die einzelnen Methoden zur Partitionierung angewendet werden.

3.7.1. Ein Partitionslayout entwerfen

Wenn Sie Dateisysteme anlegen, sollten Sie beachten, dass Festplatten auf Daten in den äußeren Spuren schneller zugreifen können als auf Daten in den inneren Spuren. Daher sollten die kleineren und oft benutzten Dateisysteme an den äußeren Rand der Platte gelegt werden. Die größeren Partitionen wie `/usr` sollten in die inneren Bereiche gelegt werden. Es empfiehlt sich, die Partitionen in folgender Reihenfolge anzulegen: `/`, swap, `/var` und `/usr`.

Die Größe der `/var`-Partition ist abhängig vom Zweck der Maschine. Diese Partition enthält hauptsächlich Postfächer, Logdateien und Druckwarteschlangen. Abhängig von der Anzahl an Systembenutzern und der Aufbewahrungszeit für Logdateien, können Postfächer und Logdateien unerwartete Größen annehmen. Die meisten Benutzer benötigen nur selten mehr als ein Gigabyte für `/var`.

Anmerkung

Ein paar Mal wird es vorkommen, dass viel Festplattenspeicher in `/var/tmp` benötigt wird. Wenn neue Software mit pkg_add(1) installiert wird, extrahieren die Paketwerkzeuge eine

vorübergehende Kopie der Pakete unter `/var/tmp`. Die Installation grosser Softwarepakete wie Firefox, Apache OpenOffice oder LibreOffice kann sich wegen zu wenig Speicherplatz in `/var/tmp` als trickreich herausstellen.

Die `/usr` Partition enthält viele der Hauptbestandteile des Systems, einschließlich der FreeBSD Ports-Sammlung und den Quellcode des Systems. Für diese Partition werden mindestens zwei Gigabyte empfohlen.

Behalten Sie bei der Auswahl der Partitionsgrößen den Platzbedarf im Auge. Wenn Sie den Platz auf einer Partition vollständig aufgebraucht haben, eine andere Partition aber kaum benutzen, kann die Handhabung des Systems schwierig werden.

Als Daumenregel sollten Sie doppelt soviel Speicher für die Swap-Partition vorsehen, als Sie Hauptspeicher haben, da die VM-Paging-Algorithmen im Kernel so eingestellt sind, dass sie am besten laufen, wenn die Swap-Partition mindestens doppelt so groß wie der Hauptspeicher ist. Zu wenig Swap kann zu einer Leistungsverminderung im *VM page scanning* Code führen, sowie Probleme verursachen, wenn später mehr Speicher in die Maschine eingebaut wird.

Auf größeren Systemen mit mehreren SCSI-, oder IDE-Laufwerken an unterschiedlichen Controllern, wird empfohlen, Swap-Bereiche auf bis zu vier Laufwerken einzurichten. Diese Swap-Partitionen sollten ungefähr dieselbe Größe haben. Der Kernel kann zwar mit beliebigen Größen umgehen, aber die internen Datenstrukturen skalieren bis zur vierfachen Größe der größten Partition. Ungefähr gleich große Swap-Partitionen erlauben es dem Kernel, den Swap-Bereich optimal über die Laufwerke zu verteilen. Große Swap-Bereiche, auch wenn sie nicht oft gebraucht werden, sind nützlich, da sich ein speicherfressendes Programm unter Umständen auch ohne einen Neustart des Systems beenden lässt.

Indem Sie ein System richtig partitionieren, verhindern Sie, dass eine Fragmentierung in den häufig beschriebenen Partitionen auf die meist nur gelesenen Partitionen übergreift. Wenn Sie die häufig beschriebenen Partitionen an den Rand der Platte legen, dann wird die I/O-Leistung dieser Partitionen steigen. Die I/O-Leistung ist natürlich auch für große Partitionen wichtig, doch erzielen Sie eine größere Leistungssteigerung, wenn Sie `/var` an den Rand der Platte legen.

3.7.2. Geführte Partitionierung

Sollten mehrere Platten angeschlossen sein, wählen Sie diejenige aus, auf der FreeBSD installiert werden soll.

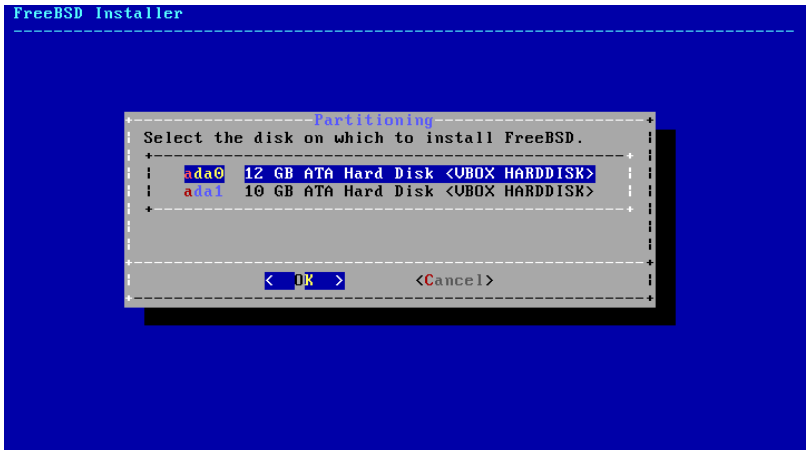


Abbildung 3.13. Aus mehreren Platten eine auswählen

Die gesamte Festplatte oder nur ein Teil davon kann für FreeBSD verwendet werden. Ein allgemeines Partitionslayout, das die gesamte Platte einnimmt wird erstellt, wenn [Entire Disk] ausgewählt wird. Durch die Wahl von [Partition] wird ein Partitionslayout im unbenutzten Speicherplatz der Platte eingerichtet.

Abbildung 3.14. Auswahl der gesamten Platte oder einer Partition

Nachdem das Partitionslayout nun erstellt wurde, sollten Sie es danach noch einmal auf Korrektheit prüfen. Sollten Sie einen Fehler gemacht haben, können Sie durch Auswahl von [Revert] wieder die ursprünglichen Partitionen setzen oder durch [Auto] die automatischen FreeBSD Partitionen wiederherstellen. Partitionen können manuell erstellt, geändert oder gelöscht werden. Sollte die Partitionierung richtig sein, wählen Sie [Finish] aus, um mit der Installation fortzufahren.

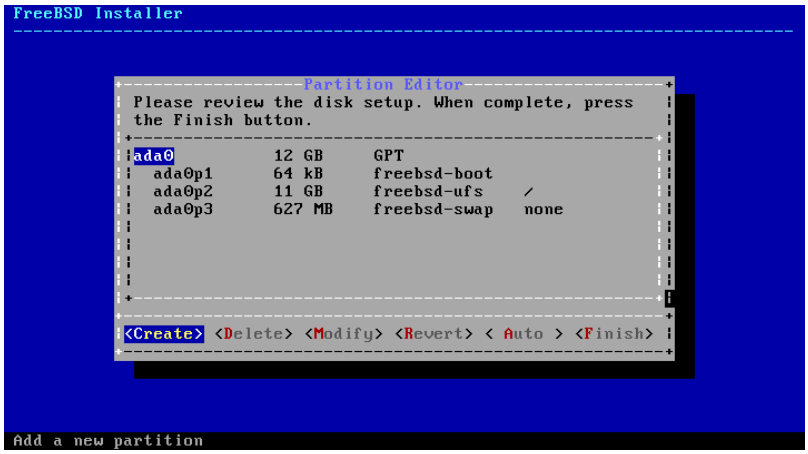


Abbildung 3.15. Überprüfen der erstellen Partitionen

3.7.3. Manuelle Partitionierung

Manuelle Partitionierung führt Sie direkt zum Partitionseditor.

Abbildung 3.16. Partitionen manuell erstellen

Durch hervorheben einer Platte (in diesem Fall `ada0`) und die Auswahl von [Create], wird ein Menü zur Wahl des *Partitionierungsschemas* angezeigt.

Abbildung 3.17. Partitionen manuell anlegen

GPT-Partitionierung ist normalerweise die passendste Auswahl für PC-kompatible Rechner. Ältere PC Betriebssysteme, die nicht mit GPT kompatibel und benötigen stattdessen MBR-Partitionen. Die anderen Partitionsschemata werden für gewöhnlich für ältere Computersysteme benutzt.

Tabelle 3.1. Partitionierungsschemas

Abkürzung	Beschreibung
APM	Apple Partition Map, von PowerPC® Macintosh® verwendet (http://support.apple.com/kb/TA21692).
BSD	BSD-Labels ohne einen MBR, manchmal auch „dangerously dedicated mode“ genannt. Lesen Sie dazu bsdlabel(8).
GPT	GUID Partition Table (http://en.wikipedia.org/wiki/GUID_Partition_Table).
MBR	Master Boot Record (http://en.wikipedia.org/wiki/Master_boot_record).
PC98	MBR-Variante, verwendet von NEC PC-98 Computern (http://en.wikipedia.org/wiki/Pc9801).

Abkürzung	Beschreibung
VTOC8	Volume Table Of Contents, von Sun SPARC64 und UltraSPARC Computern verwendet.

Nachdem das Partitionierungsschema ausgewählt und erstellt wurde, werden durch erneute Auswahl von [Create] neue Partitionen erzeugt.

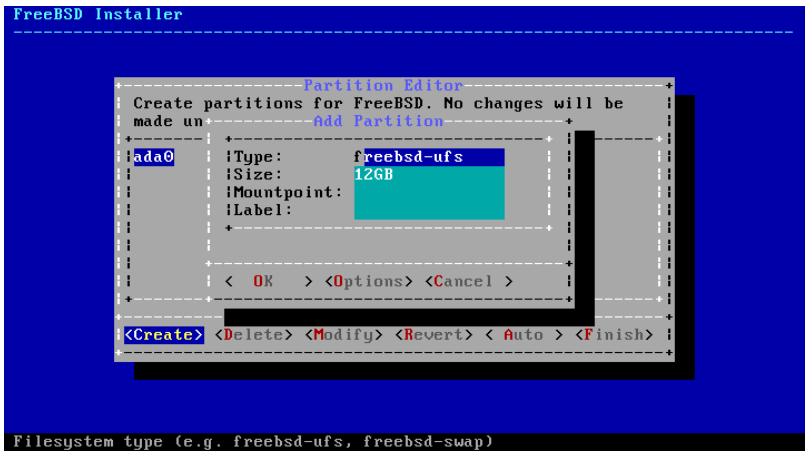


Abbildung 3.18. Partitionen manuell erzeugen

Eine FreeBSD-Standardinstallation mit GPT legt mindestens die folgenden drei Partitionen an:

- `freebsd-boot` - FreeBSD-Bootcode.
- `freebsd-ufs` - Ein FreeBSD UFS-Dateisystem.
- `freebsd-swap` - FreeBSD Auslagerungsbereich (swap space).

Ein weiter Partitionstyp ist `freebsd-zfs` . Dieser Typ ist für Partitionen bestimmt, die ein FreeBSD ZFS-Dateisystem verwenden sollen. gpart(8) beschreibt weitere verfügbare Partitionstypen.

Mehere Dateisystempartitionen können benutzt werden und manche Leute ziehen es vor, ein traditionelles Layout mit getrennten Partitionen für die Dateisysteme `/`, `/var`, `/tmp` und `/usr` zu erstellen. Lesen Sie dazu Beispiel 3.3, „Ein traditionelles, partitioniertes Dateisystem erstellen“, um ein Beispiel zu erhalten.

Grössenangaben können mit gängigen Abkürzungen eingegeben werden: *K* für Kilobytes, *M* für Megabytes oder *G* für Gigabytes.

Tipp

Korrekte Sektorausrichtung ermöglicht grösstmögliche Geschwindigkeit und das Anlegen von Partitionsgrössen als vielfaches von 4K-Bytes hilft, die passende Ausrichtung auf Platten mit entweder 512-Bytes oder 4K-Bytes Sektorgrössen, festzulegen. Generell sollte die Verwendung von Partitionsgrössen, die sogar vielfache von 1M oder 1G sind, den einfachsten Weg darstellen, um sicher zu stellen, dass jede Partition an einem vielfachen von 4K beginnt. Eine Ausnahme gibt es: momentan sollte die *freebsd-boot*-Partition aufgrund von Beschränkungen im Bootcode nicht grösser sein als 512K.

Ein Einhägepunkt wird benötigt, falls diese Partition ein Dateisystem enthält. Falls nur eine einzelne UFS-Partition erstellt wird, sollte der Einhängepunkt `/` lauten.

Ein *label* wird ebenfalls benötigt. Ein Label ist ein Name, durch den diese Partition angesprochen wird. Festplattennamen oder -nummern können sich ändern, falls die Platte einmal an einem anderen Controller oder Port angeschlossen sein sollte, doch das Partitionslabel ändert sich dadurch nicht. Anstatt auf Plattennamen und Partitionsnummern in Dateien wie `/etc/fstab` zu verweisen, sorgen Labels dafür, dass das System Hardwäreänderungen eher toleriert. GPT-Labels erscheinen in `/dev/gpt/` , wenn eine Platte angeschlossen wird. Andere Partitionierungsschemas besitzen unterschiedliche Fähigkeiten, Labels zu verwenden und diese erscheinen in anderen `/dev/`-Verzeichnissen.

Tipp

Vergeben Sie ein einzigartiges Label auf jedem Dateisystem um Konflikte mit identischen Labels zu verhindern. Ein paar Buchstaben des Computernamens, dessen Verwendungszweck oder Ortes kann dem Label hinzugefügt werden. Beispielsweise „labroot“ oder „rootfslab“ für die UFS root-Partition auf einem Laborrechner.

Beispiel 3.3. Ein traditionelles, partitioniertes Dateisystem erstellen

Für ein traditionelles Partitionslayout, in dem sich `/`, `/var`, `/tmp` und `/usr` in getrennten Partitionen befinden sollen, erstellen Sie ein GPT-Partitionsschema und anschliessend die Partitionen selbst. Die gezeigten Partitionsgrössen sind typisch für eine Festplatte von 20 G. Falls mehr Platz verfügbar ist, sind größere Swap oder `/var`-Partitionen nützlich. Den hier gezeigten Beschreibungen sind `bsp` für „Beispiel“ vorangestellt, jedoch sollten Sie andere, einzigartige Beschreibungen verwenden, wie oben beschrieben.

Standardmäßig erwartet FreeBSDs `gptboot`, dass die erste erkannte UFS-Partition die `/`-Partition ist.

Partitionstyp	Grösse	Eingehängt als	Beschreibung
`freebsd-boot`	`512K`		
`freebsd-ufs`	`2G`	`/`	`bsprootfs`
`freebsd-swap`	`4G`	`bspswap`	
`freebsd-ufs`	`2G`	`/var`	`bspvarfs`
`freebsd-ufs`	`1G`	`/tmp`	`bsptmpfs`
`freebsd-ufs`	Akzeptieren Sie die Standardeinstellungen (Rest der Platte)	`/usr`	`bspusrfs`

Nachdem die selbstgewählten Partitionen erzeugt wurden, wählen Sie [Finish], um mit der Installation fortzusetzen.

3.7.4. Root-on-ZFS automatische Partitionierung

Die Unterstützung für die automatische Erstellung von Root-on-ZFS Installationen wurde in FreeBSD 10.0-RELEASE hinzugefügt. Dieser Modus funktioniert nur mit ganzen Laufwerken und wird alle vorhandenen Daten auf der Platte löschen. Das Installationsprogramm wird die Partitionen für ZFS automatisch an 4k Sektoren ausrichten. Dies funktioniert auch zuverlässig bei einer Sektorgröße von 512 Byte und hat den Vorteil, dass nachträglich Laufwerke mit einer Sektorgröße von 4k hinzugefügt werden können, entweder als zusätzlicher Speicherplatz oder als Ersatz für eine ausgefallene Platte. Das Installationsprogramm kann auch optional die Platte mit GELI verschlüsseln, wie

in Abschnitt 18.13.2, „Plattenverschlüsselung mit `geli`“ beschrieben. Wird die Verschlüsselung aktiviert, wird eine 2 GB große unverschlüsselte Partition für `/boot` erstellt. Diese Partition enthält den Kernel und weitere Dateien, die nötig sind um den Kernel zu booten. Zudem wird eine Swap-Partition erstellt. Der verbleibende Platz wird für den ZFS-Pool verwendet.

Das Konfigurationsmenü für ZFS bietet einige Optionen, um die Erstellung des Pools zu beeinflussen.

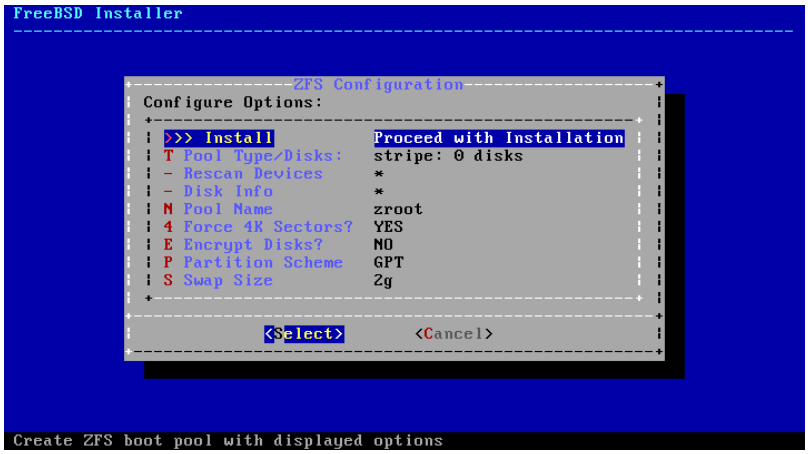


Abbildung 3.19. ZFS Konfigurationsmenü

Die wichtigste Option ist die Auswahl des vdev-Typs und der Festplatten, die das vdev für den Pool bilden werden. Mit Ausnahme des Stripe Modus, unterstützt das automatische ZFS Installationsprogramm derzeit nur die Erstellung eines einzigen Top-Level-vdevs. Wenn Sie komplexere Pools erstellen möchten, folgen Sie den Anweisungen in Abschnitt 3.7.5, „Shell Partitionierung“. Das Installationsprogramm unterstützt verschiedene Pool Typen, einschließlich Stripe (nicht empfohlen, keine Redundanz), Spiegel (beste Leistung, weniger nutzbarer Speicherplatz), und RAID-Z 1, 2 und 3 (übersteht den Ausfall von jeweils einer, zwei und drei Platten). Am unteren Rand des Bildschirms werden Hinweise zur Anzahl der benötigten Platten angezeigt. Im Fall von RAID-Z wird die optimale Anzahl von Platten für die jeweilige Konfiguration angezeigt.

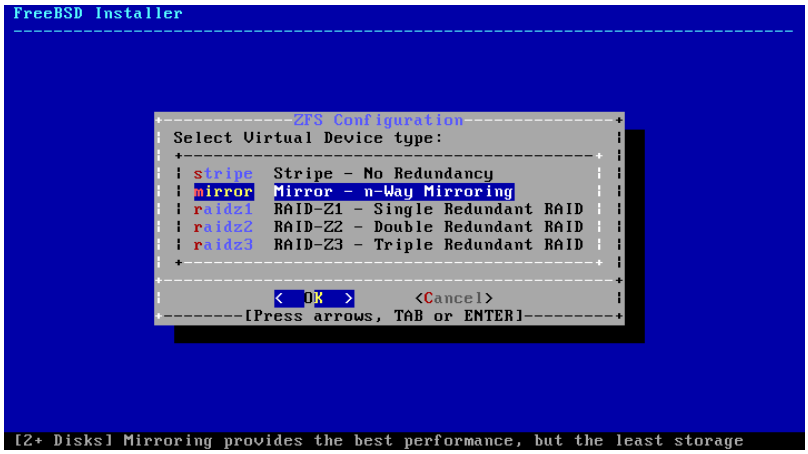


Abbildung 3.20. ZFS Pool Typen

Sobald ein Pool Typ ausgewählt wurde, wird eine Liste der verfügbaren Laufwerke angezeigt und der Benutzer wird aufgefordert, eine oder mehrere Festplatten auszuwählen um den Pool zu bilden. Anschließend wie die Konfiguration geprüft um zu gewährleisten, dass genug Platten ausgewählt wurden. Wählen Sie <Change Selection> um zur Auswahl der Laufwerke zurückzukehren, oder <Cancel> um den Pool Typ zu ändern.

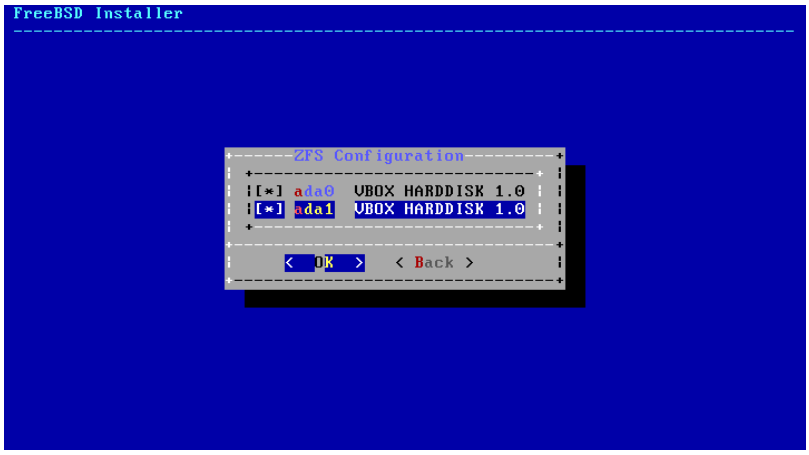


Abbildung 3.21. Auswahl der Laufwerke

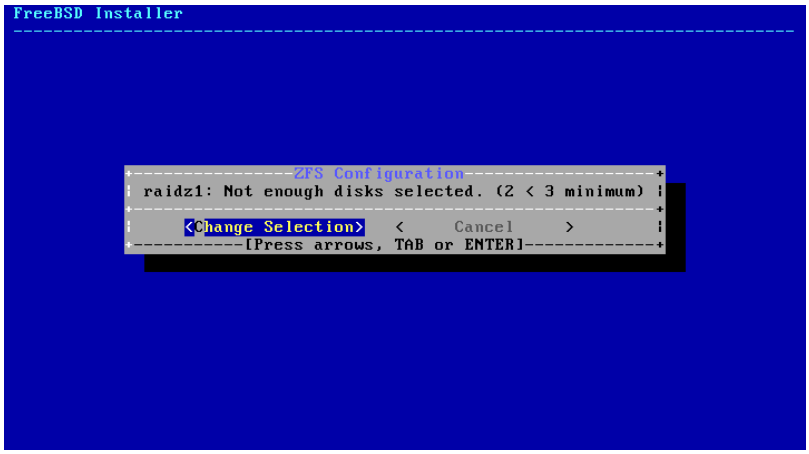


Abbildung 3.22. Ungültige Auswahl

Wenn eine oder mehrere Platten in der Liste fehlen, oder wenn Festplaten angebunden wurden, nachdem das Installationsprogramm gestartet wurde, wählen Sie - Rescan Devices um die Laufwerke nochmals zu suchen und anzuzeigen. Um sicherzustellen, dass die richtigen Platten ausgewählt wurden, können Sie das - Disk-Info Menü verwenden. Dieses Menü zeigt verschiedene Informationen, einschließlich der Partitionstabelle, der Modelnummer und der Seriennummer, falls verfügbar.

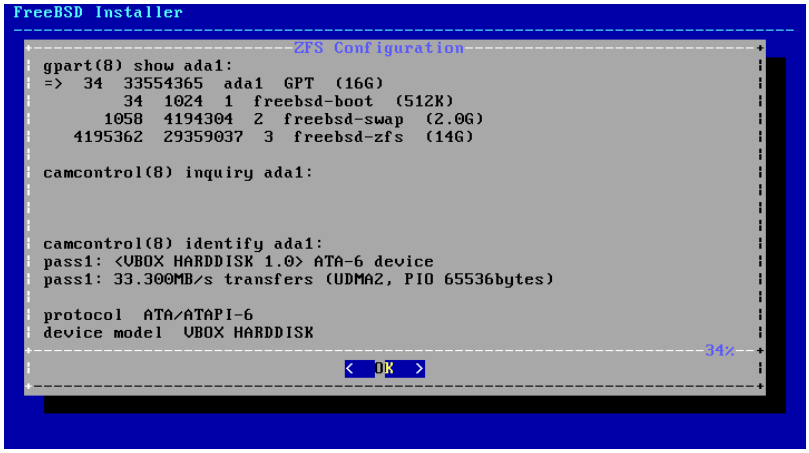


Abbildung 3.23. Eigenschaften einer Festplatte

Das Menü ermöglicht es dem Benutzer auch, einen Namen für den Pool zu vergeben, 4k Sektoren zu deaktivieren, Tabellentypen zu aktivieren oder deaktivieren, und die Größe des Swap-Bereichs zu wählen. Wenn alle Optionen auf die gewünschten Werte eingestellt sind, wählen Sie >>> Install am oberen Rand des Menüs.

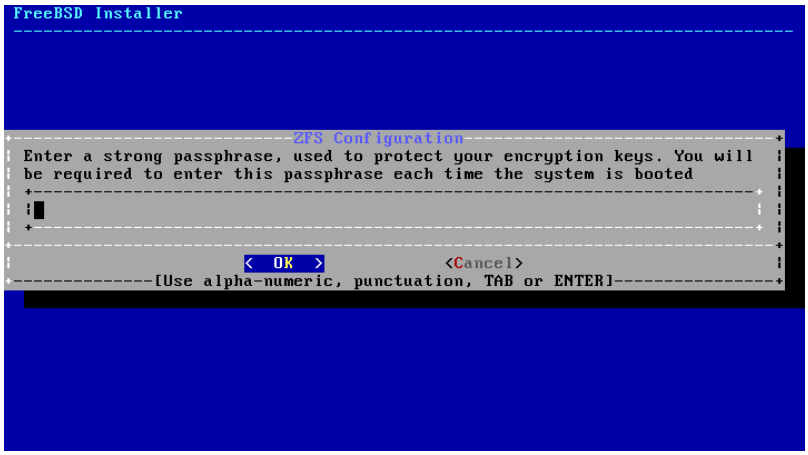


Abbildung 3.24. Passwort für die Verschlüsselung der Platte

Das Installationsprogramm bietet dann eine letzte Chance, abzubrechen, bevor der Inhalt der ausgewählten Laufwerken zerstört wird, um den neuen ZFS-Pool zu erstellen.

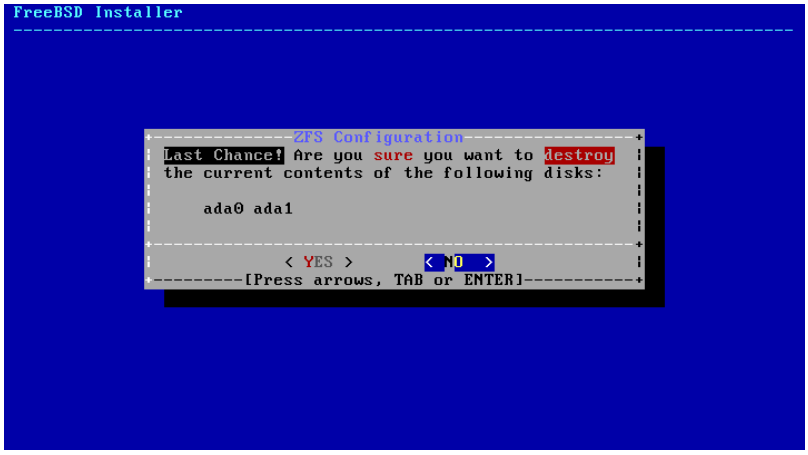


Abbildung 3.25. Letzte Chance

Danach wird die Installation normal weitergeführt.

3.7.5. Shell Partitionierung

bsdinstall bietet bei fortgeschrittenen Installationen womöglich nicht die benötigte Flexibilität. Erfahrene Benutzer können die Option Shell im Menü auswählen, um die Laufwerke manuell zu partitionieren, Dateisysteme zu erstellen, `/tmp/bsdinstall_etc/fstab` zu befüllen und Dateisysteme unter `/mnt` einzuhängen. Geben Sie anschließend `exit` ein, um zu bsdinstall zurückzukehren und die Installation fortzusetzen.

3.8. Die Installation festschreiben

Dies ist die letzte Chance, die Installation abzubrechen, ohne Änderungen an den Festplatten vorzunehmen.

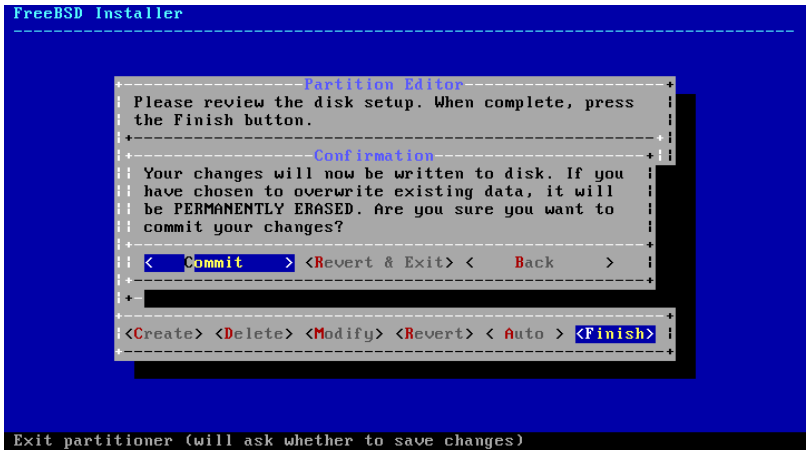


Abbildung 3.26. Letzte Bestätigung

Wählen Sie [Commit] und drücken Sie Enter, um fortzufahren. Fall noch Änderungen zu machen sind, wählen Sie [Back], um zum Partitionseditor zurück zu gelangen. Mittels [Revert & Exit] wird das Installationsprogramm beendet, ohne Änderungen an den Festplatten durchzuführen.

Die Installationsdauer hängt von den gewählten Distributionen, dem Installationsmedium und der Geschwindigkeit des Computers ab. Eine Reihe von Nachrichten werden angezeigt, um den Fortschritt darzustellen.

Zuerst wird das Installationsprogramm die Partitionen auf die Platte schreiben und den Befehl `newfs` ausführen, um die Partitionen zu initialisieren.

Falls Sie eine Netzwerkinstallation vornehmen, wird bsdinstall dann mit dem herunterladen der benötigten Distributionsdateien fortfahren.

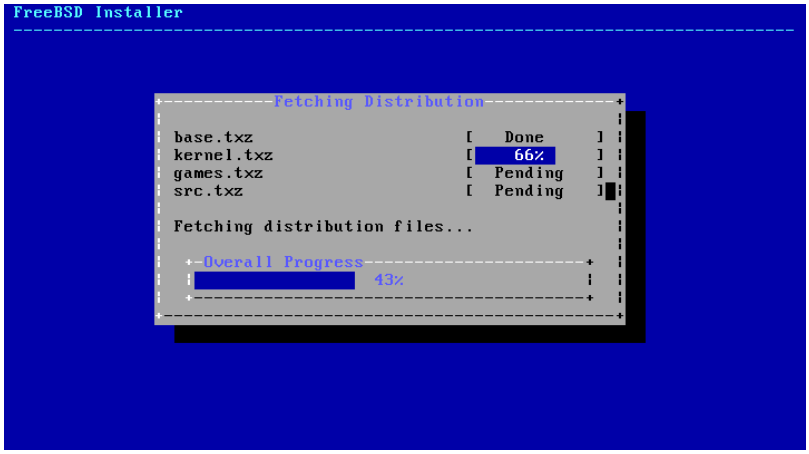


Abbildung 3.27. Herunterladen der Distributionsdateien

Als nächstes wird die Integrität der Distributionsdateien überprüft, um sicherzustellen, dass diese während des Ladevorgangs nicht beschädigt oder unsauber vom Installationsmedium gelesen wurden.

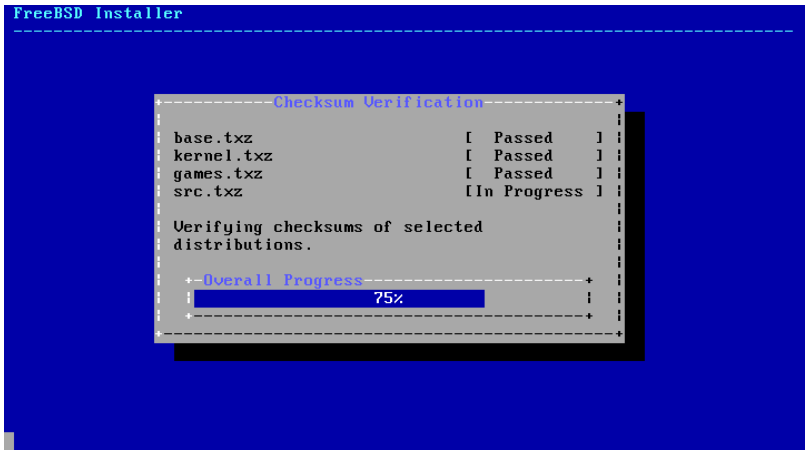


Abbildung 3.28. Überprüfen der Distributionsdateien

Zum Schluss werden die überprüften Distributionsdateien auf die Festplatte entpackt.

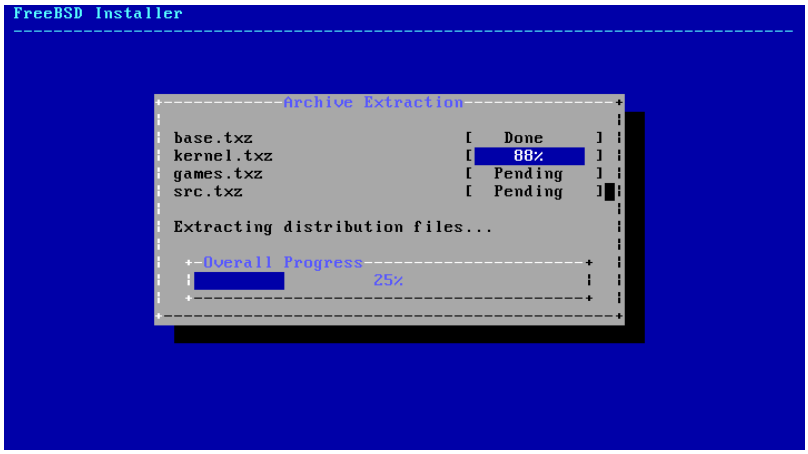


Abbildung 3.29. Entpacken der Distributionsdateien

Sobald alle benötigten Distributionsdateien entpackt wurden, wird bsdinstall direkt mit den Arbeiten nach der Installation, wie in Abschnitt 3.9, „Arbeiten nach der Installation“ beschrieben, fortfahren.

3.9. Arbeiten nach der Installation

Die Konfiguration von verschiedenen Optionen folgt auf eine erfolgreiche FreeBSD-Installation. Eine solche Option kann durch das erneute betreten der Konfigurationsoptionen aus dem letzten Menü vor dem Neustart in das gerade installierte FreeBSD-System angepasst werden.

3.9.1. Setzen des `root`-Passworts

Das `root`-Passwort muss gesetzt werden. Wichtig ist dabei zu wissen, dass die eingegebenen Zeichen nicht auf dem Bildschirm angezeigt werden. Nachdem das Passwort eingegeben wurde, muss es zur Bestätigung erneut eingetippt werden. Damit werden auch Tippfehler verhindert.

Abbildung 3.30. Das `root`-Passwort setzen

Nachdem das Passwort erfolgreich gesetzt wurde, wird die Installation nun fortgesetzt.

3.9.2. Die Netzwerkschnittstelle konfigurieren

> **Anmerkung**
>
> Die Netzwerkkonfiguration wird übersprungen, falls dies bereits als Teil der *bootonly* durchgeführt worden ist.

Eine Liste aller gefundenen Netzwerkschnittstellen, die auf diesem Computer gefunden wurden, wird als nächstes angezeigt. Wählen Sie davon eine aus, um diese zu konfigurieren.

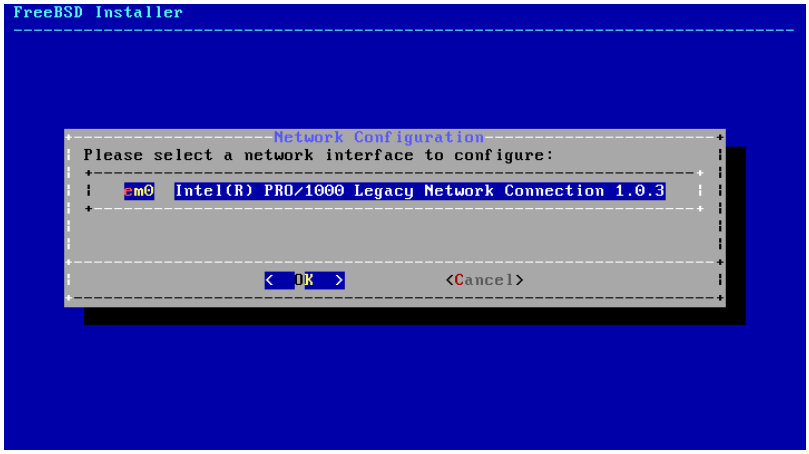


Abbildung 3.31. Eine zu konfigurierende Netzwerkschnittstelle auswählen

3.9.2.1. Eine drahtlose Netzwerkverbindung einrichten

Sollte eine drahtlose Netzwerkverbindung ausgewählt worden sein, müssen WLAN-Identifikation und Sicherheitsparameter nun eingegeben werden, um die Verbindung mit dem Netzwerk herzustellen.

Drahtlose Netzwerke werden durch einen Service Set Identifier oder auch SSID genannt, identifiziert. Der SSID ist ein kurzer, eindeutiger Name, der für jedes Netzwerk vergeben wird.

Die meisten drahtlosen Netzwerke verschlüsseln die übertragenen Daten, um die Information darin vor unautorisiertem Zugriff zu schützen. Die Verwendung von WPA2-Verschlüsselung wird empfohlen. Ältere Verschlüsselungstypen, wie WEP, bieten nur sehr wenig Sicherheit.

Der erste Schritt des Verbindungsaufbaus ist das drahtlose Netzwerk nach drahtlosen Zugriffspunkten (access points) zu scannen.

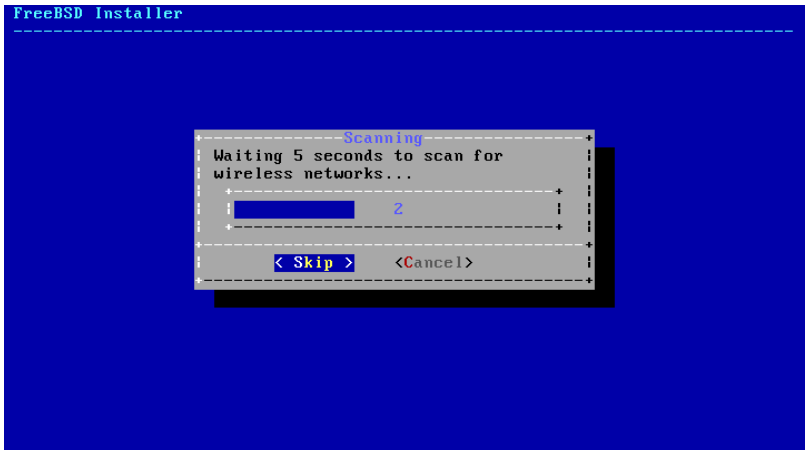


Abbildung 3.32. Nach drahtlosen Access Points scannen

SSIDs, die während des scannens gefunden wurden, werden aufgelistet, gefolgt von einer Beschreibung der Verschlüsselungsarten, die für dieses Netzwerk verfügbar sind. Falls die gewünschte SSID nicht in der Liste auftaucht, wählen Sie [Rescan], um erneut einen Scanvorgang durchzuführen. Falls dann das gewünschte Netzwerk immer noch nicht erscheint, überprüfen Sie die Antenne auf Verbindungsprobleme oder versuchen Sie, näher an den Access point zu gelangen. Scannen Sie erneut nach jeder vorgenommenen Änderung.

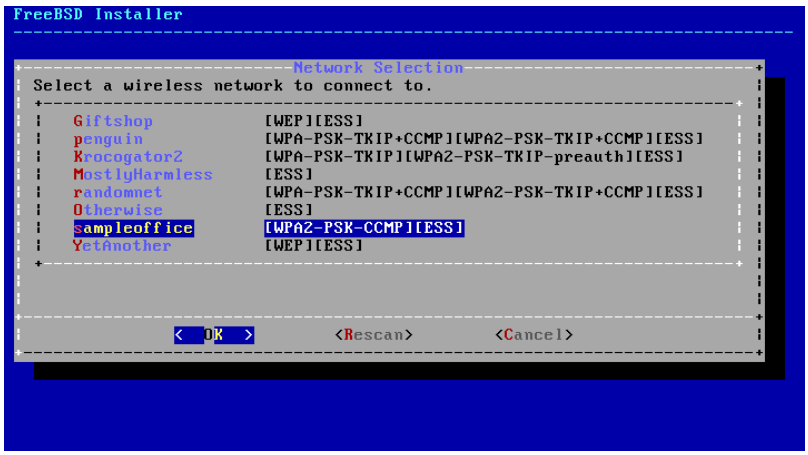


Abbildung 3.33. Ein drahtloses Netzwerk auswählen

Die Verschlüsselungsinformationen, um sich mit dem Netzwerk zu verbinden, werden nach der Auswahl des Netzwerks eingegeben. Mit WPA2 wird nur ein Passwort (auch bekannt als Pre-Shared Key oder PSK) benötigt. Zeichen, die in die Eingabebox getippt werden, erscheinen aus Sicherheitsgründen als Sternchen.

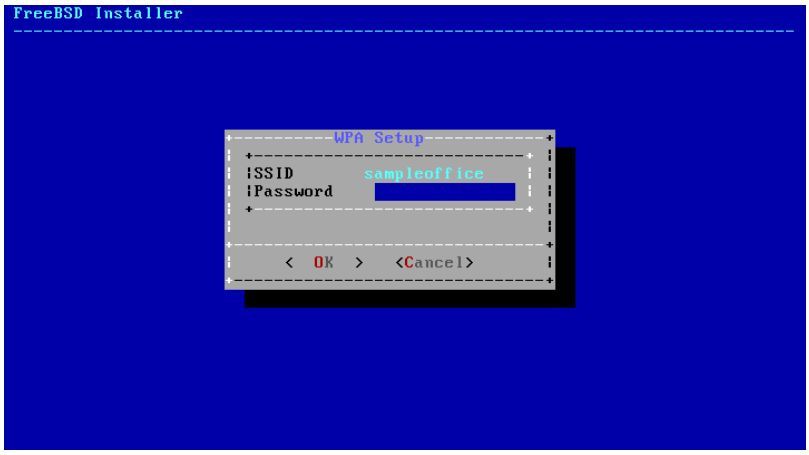


Abbildung 3.34. Verbindungsaufbau mit WPA2

Die Netzwerkkonfiguration wird fortgesetzt, nachdem das drahtlose Netzwerk und die Verbindungsinformationen eingegeben wurden.

3.9.2.2. Konfiguration des IPv4-Netzwerks

Wählen Sie, ob Sie ein IPv4-Netzwerk verwenden möchten. Dies ist der am häufigsten vorkommende Typ einer Netzwerkverbindung.

Abbildung 3.35. Auswahl von IPv4

Es gibt zwei Arten, ein IPv4-Netzwerk zu konfigurieren. *DHCP* wird automatisch die Netzwerkschnittstelle richtig konfigurieren und sollte als bevorzugte Methode verwendet werden. *Statische* Konfiguration erfordert die manuelle Eingabe von Netzwerkinformationen.

Anmerkung

Geben Sie keine zufällig gewählten Netzwerkinformationen ein, da dies nicht funktionieren wird. Holen Sie sich die in Abschnitt 3.3.3, „Netzwerkparameter ermitteln“ gezeigten Informationen vom Netzwerkadministrator oder Serviceprovider.

3.9.2.2.1. Netzwerkkonfiguration von IPv4 mittels DHCP

Falls ein DHCP-Server zur Verfügung steht, wählen Sie [Yes], um die Netzwerkschnittstelle automatisch einrichten zu lassen.

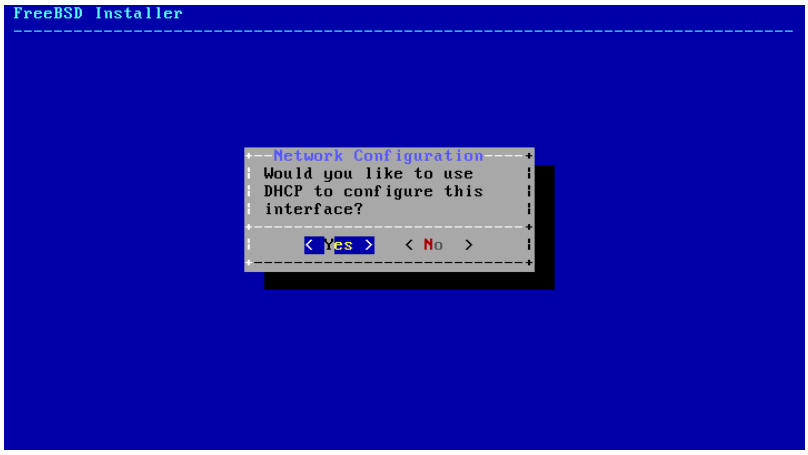


Abbildung 3.36. Auswählen der IPv4-Konfiguration über DHCP

3.9.2.2.2. Statische IPv4-Netzwerkkonfiguration

Statische Konfiguration der Netzwerkschnittstelle erfordert die die Eingabe einiger IPv4-Informationen.

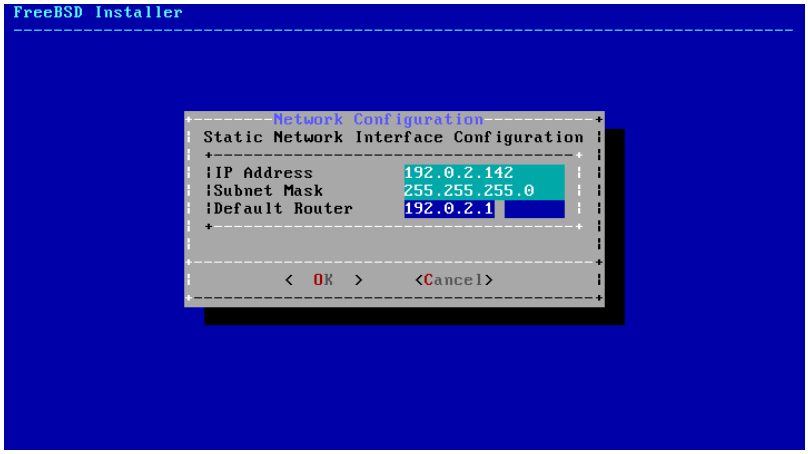


Abbildung 3.37. Statische IPv4-Konfiguration

- `IP-Adresse` - Die manuell festgelegte IPv4-Adresse, welche diesem Computer zugewiesen werden soll. Diese Adresse muss eindeutig sein und darf nicht bereits von einem anderen Gerät im lokalen Netzwerk verwendet werden.

- `Subnetzmaske` - Die Subnetzmaske, die im lokalen Netzwerk Verwendung findet. Typischerweise ist dies `255.255.255.0` .

- `Defaultrouter` - Die IP-Adresse des Defaultrouters in diesem Netzwerk. Normalerweise ist das die Adresse des Routers oder einer anderen Netzwerkkomponente, die das lokale Netzwerk mit dem Internet verbindet. Auch bekannt als das *Default Gateway*.

3.9.2.3. Konfiguration des IPv6-Netzwerks

IPv6 ist eine neuere Methode der Netzwerkkonfiguration. Falls IPv6 verfügbar ist und verwendet werden soll, wählen Sie [Yes] aus.

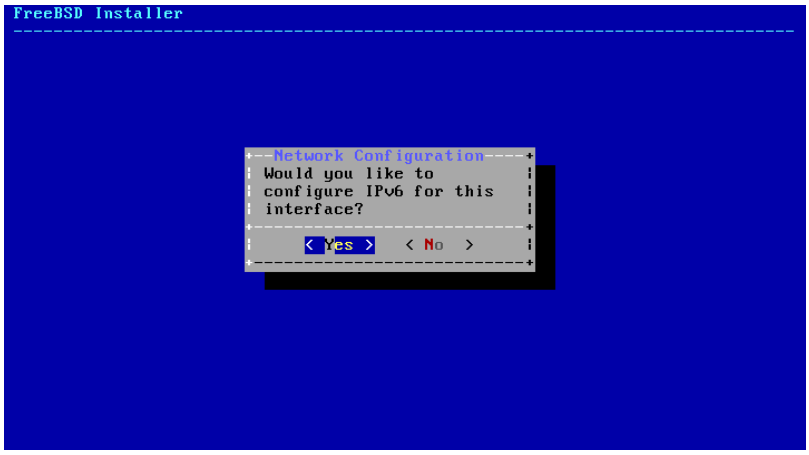


Abbildung 3.38. Auswahl von IPv6

IPv6 besitzt ebenfalls zwei Arten der Konfiguration. *SLAAC*, oder *StateLess Address AutoConfiguration*, wird die Netzwerkschnittstelle automatisch richtig konfigurieren. *Statische* Konfiguration verlangt die manuelle Eingabe von Netzwerkinformationen.

3.9.2.3.1. IPv6 Stateless Address Autoconfiguration

SLAAC erlaubt es einer IPv6-Netzwerkkomponente, die Information zur automatischen Konfiguration von einem lokalen Router abzufragen. Lesen Sie http://tools.ietf.org/html/rfc4862 für weitere Informationen.

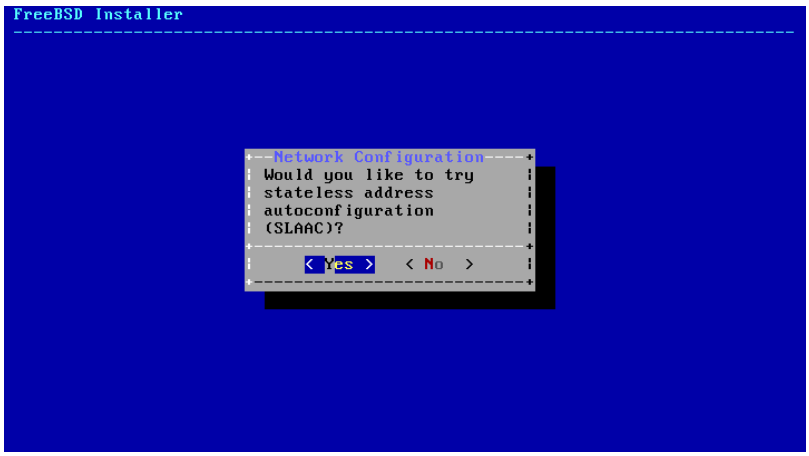


Abbildung 3.39. Auswahl der IPv6 SLAAC-Konfiguration

3.9.2.3.2. Statische IPv6-Netzwerkkonfiguration

Statische Konfiguration der Netzwerkschnittstelle benötigt die Eingabe von IPv6-Konfigurationsinformationen.

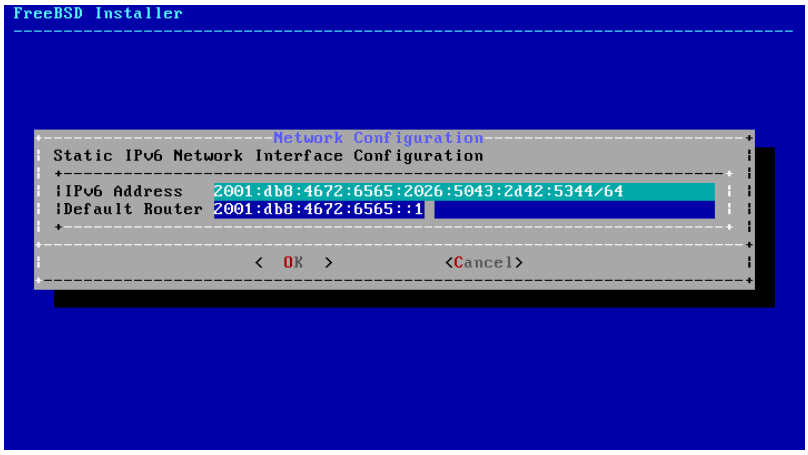


Abbildung 3.40. Statische IPv6-Konfiguration

- `IPv6-Adresse` - The manuell zugewiesene IP-Adresse, welche dem Computer zugeteilt werden soll. Diese Adresse muss eindeutig sein und nicht bereits von einer anderen Netzwerkkomponente im lokalen Netzwerk verwendet werden.

- `Defaultrouter` - Die IPv6-Adresse des Defaultrouters in diesem Netzwerk. Normalerweise ist dies die Adresse des Routers oder einer anderen Netzwerkkomponente, welche das lokale Netz mit dem Internet verbindet. Auch bekannt als *Default Gateway*.

3.9.2.4. DNS-Konfiguration

Der *Domain Name System* (oder auch *DNS*) Resolver wandelt Hostnamen von und zu Netzwerkadressen um. Falls DHCP oder SLAAC verwendet wurde, um die Netzwerkschnittstelle zu konfigurieren, ist die Konfiguration für den Resolver möglicherweise bereits vorhanden. Andernfalls geben Sie den lokalen Netzwerkdomänennamen in das Feld `Search` ein. `DNS #1` und `DNS #2` sind die IP-Adressen der lokalen DNS-Server. Zumindest ein DNS-Server wird benötigt.

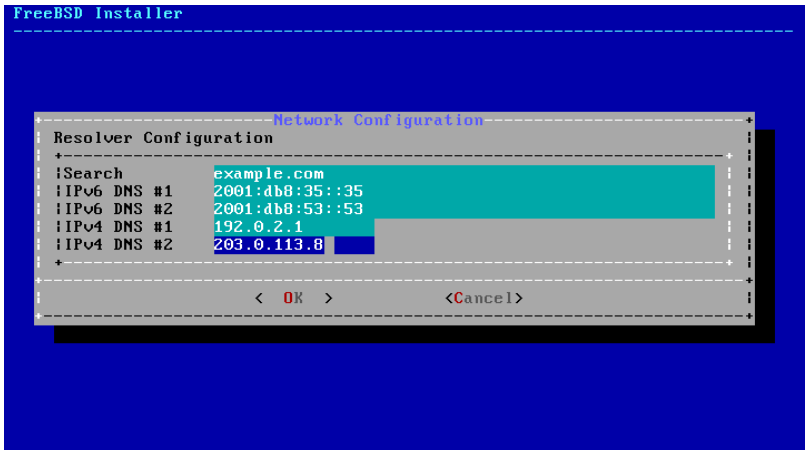


Abbildung 3.41. DNS-Konfiguration

3.9.3. Setzen der Zeitzone

Das Setzen der Zeitzone für die Maschine erlaubt es, diese auf regionale Zeitveränderungen hin anzupassen und um andere zeitzonenbezogene Funktionen richtig durchzuführen.

Das hier gezeigte Beispiel bezieht sich auf einen Rechner in der östlichen Zeitzone der Vereinigten Staaten. Die Auswahl wird von der geographischen Position abhängig sein.

Abbildung 3.42. Lokale oder UTC-Zeit

Wählen Sie [Yes] oder [No], abhängig davon, wie die Rechneruhr konfiguriert ist und drücken Sie dann Enter. Wenn Sie nicht wissen, ob das System UTC oder lokale Zeit verwendet, wählen Sie [No], um die am häufigsten verwendete lokale Zeit zu setzen.

Abbildung 3.43. Das Gebiet auswählen

Das passende Gebiet wird durch die Pfeiltasten und das anschliessende druücken von Enter gewählt.

Abbildung 3.44. Das Land auswählen

Wählen Sie das zutreffende Land mit den Pfeiltasten und durch anschliessendes drücken von Enter aus.

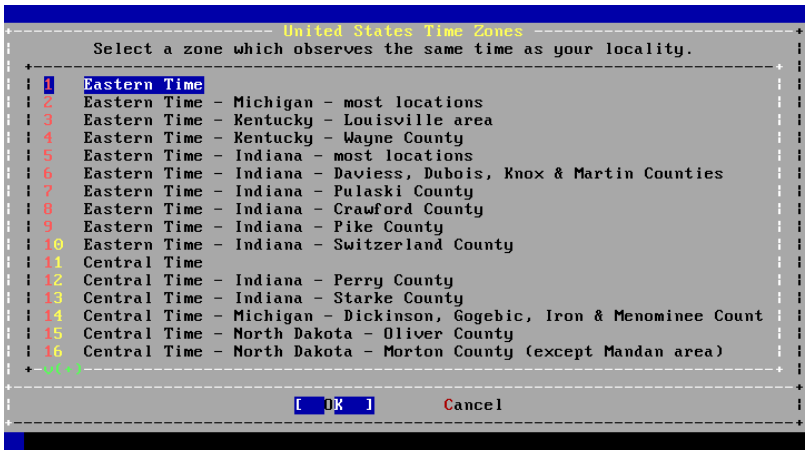


Abbildung 3.45. Wählen einer Zeitzone

Die passende Zeitzone wird durch die Pfeiltasten und anschliessendes drücken von Enter ausgewählt.

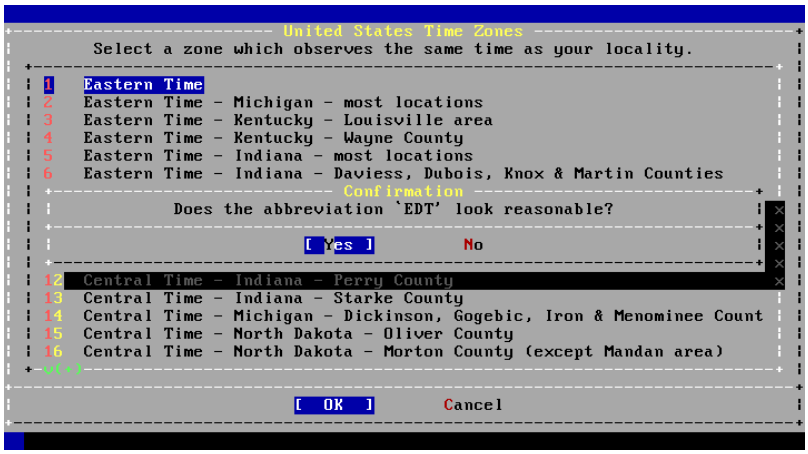


Abbildung 3.46. Bestätigen der Zeitzone

Bestätigen Sie, dass die Abkürzung für die Zeitzone richtig ist. Wenn Ihnen diese richtig erscheint, drücken Sie Enter, um mit dem Rest der Konfiguration nach der Installation fortzufahren.

3.9.4. Zu aktivierende Dienste auswählen

Zusätzliche Systemdienste, die zur Startzeit aktiviert werden sollen, können eingeschaltet werden. All diese Dienste sind optional.

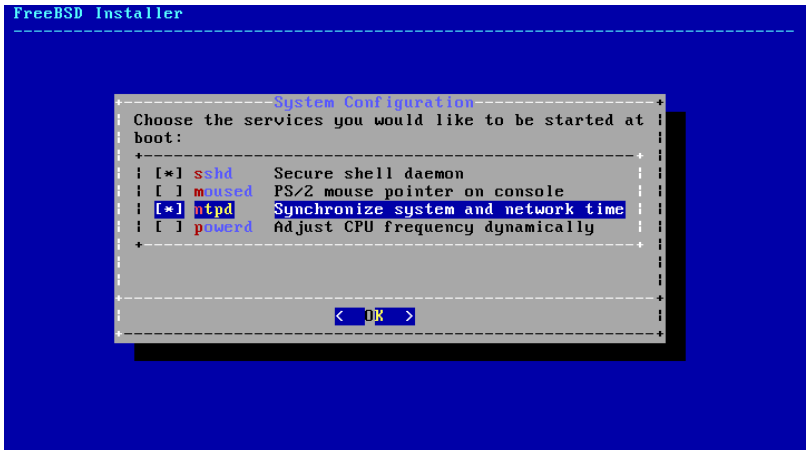


Abbildung 3.47. Auswahl zusätzlicher Dienste

- `sshd` - Secure Shell (SSH)-Dienst für sicheren Fernzugriff.
- `moused` - Sorgt für Mausunterstützung innerhalb der Systemkonsole.
- `ntpd` - Network Time Protocol (NTP)-Dienst zur automatischen Uhrzeitsynchronisation.
- `powerd` - Systemleistungskontrollwerkzeug zur Leistungsregelung und für Stromsparfunktionen.

3.9.5. Absturzaufzeichnung aktivieren

bsdinstall wird Sie fragen, ob die Absturzaufzeichnung auf dem Zielsystem aktiviert werden soll. Die Aktivierung von Absturzaufzeichnungen kann sehr nützlich sein, um Systemfehler aufzuspüren, deswegen wird Anwendern empfohlen, diese so oft wie möglich einzusetzen. Wählen Sie [Yes], um Absturzaufzeichnungen zu aktivieren oder [No], um ohne die Aufzeichnung von Abstürzen fortzufahren.

Abbildung 3.48. Aktivierung der Absturzaufzeichnung

3.9.6. Benutzer hinzufügen

Das hinzufügen von mindestens einem Benutzer während der Installaton erlaubt das Benutzen des Systems ohne als `root`-Benutzer angemeldet zu sein. Wenn man als `root` angemeldet ist, gibt es so gut wie keine Beschränkungen oder Schutz vor dem, was man tun kann. Anmelden als normaler Benutzer ist daher sicherer und bietet mehr Schutz.

Wählen Sie [Yes], um neue Benutzer hinzuzufügen.

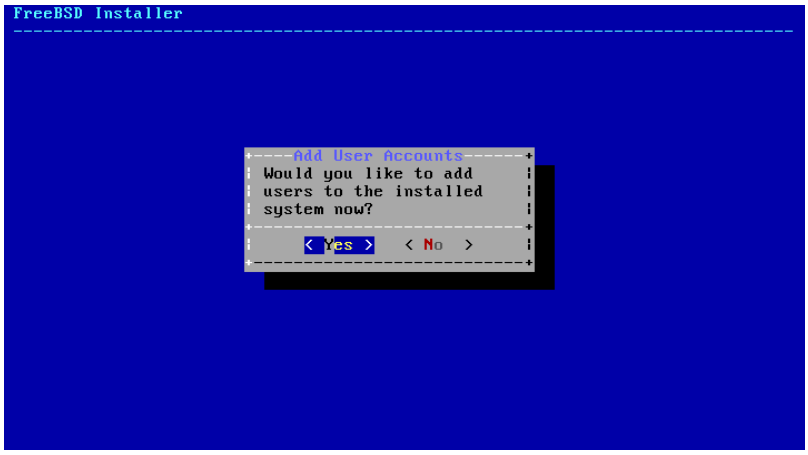


Abbildung 3.49. Benutzerkonten hinzufügen

Geben Sie die nötigen Informationen für den Benutzer ein, der dem System hinzugefügt werden soll.

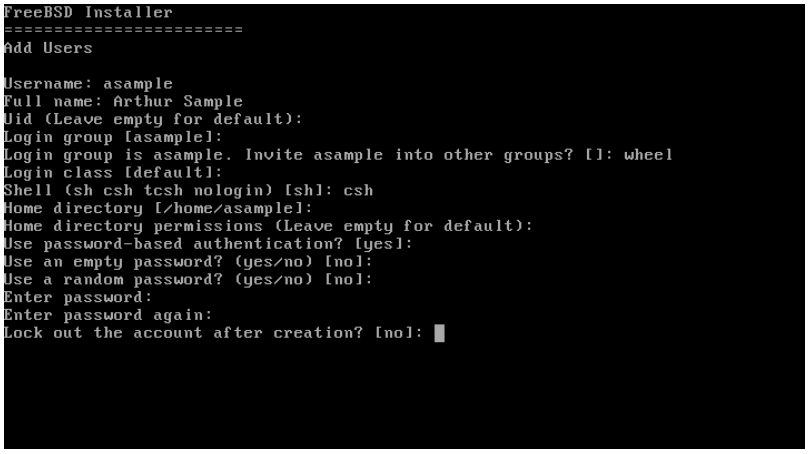


Abbildung 3.50. Benutzerinformationen eingeben

- `Username` - Der Name des Benutzers, den man zur Anmeldung eingeben muss. Typischerweise der erste Buchstabe des Vornamens, gefolgt vom Nachnamen.
- `Full name` - Der volle Name des Benutzers.
- `Uid` - User ID. Normalerweise wird dieses Feld leer gelassen, so dass das System einen Wert vergibt.
- `Login group` - Die Benutzergruppe. Normalerweise bleibt dieses Feld leer, um die Standardgruppe zu akzeptieren.
- `Invite` *`user`* `into other groups?` - Zusätzliche Gruppen zu denen der Benutzer als Mitglied hinzugefügt werden soll.
- `Login class` - In der Regel bleibt dieses Feld leer.
- `Shell` - Die interaktive Shell für diesen Benutzer. In diesem Beispiel wurde csh(1) ausgewält.
- `Home directory` - Das Heimatverzeichnis des Benutzers. Die Vorgabe ist für gewöhnlich richtig.
- `Home directory permissions` - Zugriffsrechte auf das Heimatverzeichnis des Benutzers. Die Vorgabe ist normalerweise die passende.
- `Use password-based authentication?` Normalerweise `yes`.
- `Use an empty password?` - Normalerweise `no`.

- `Use a random password?` - Normalerweise `no`.
- `Enter password` - Das Passwort für diesem Benutzer. Eingegebene Zeichen werden nicht am Bildschirm angezeigt.
- `Enter password again` - Das Passwort muss zur Überprüfung erneut eingegeben werden.
- `Lock out the account after creation?` - Normalerweise `no`.

Nachdem alles eingegeben wurde, wird eine Zusammenfassung angezeigt und das System fragt Sie, dies so korrekt ist. Falls ein Eingabefehler gemacht wurde, geben Sie `no` ein und versuchen es erneut. Falls alles in Ordnung ist, drücken Sie `yes`, um den neuen Benutzer anzulegen.

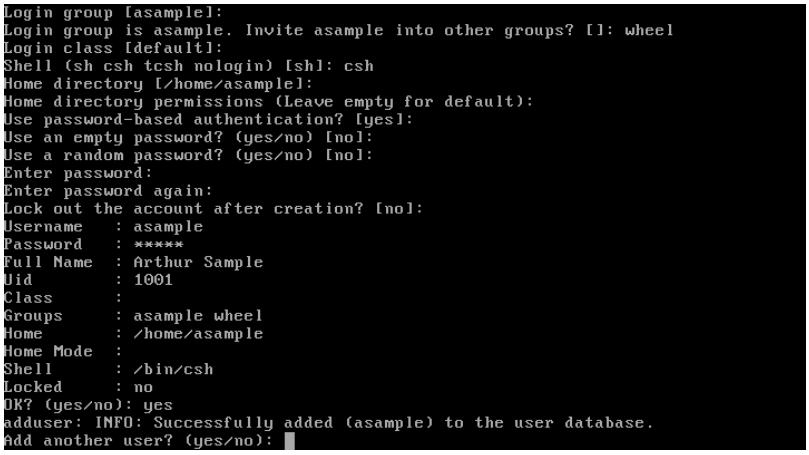


Abbildung 3.51. Verlassen der Benutzer- und Gruppenverwaltung

Falls es mehr Benutzer hinzuzufügen gibt, beantworten Sie die Frage `Add another user?` mit `yes`. Geben Sie `no` ein, wird das hinzufügen von Benutzern beendet und die Installation fortgesetzt.

Für weitere Informationen zum hinzufügen von Benutzern und deren Verwaltung, lesen Sie Abschnitt 4.3, „Benutzer und grundlegende Account-Verwaltung“.

3.9.7. Letzte Konfigurationsschritte

Nachdem alles installiert und konfiguriert wurde, bekommen Sie noch eine letzte Chance, um Einstellungen zu verändern.

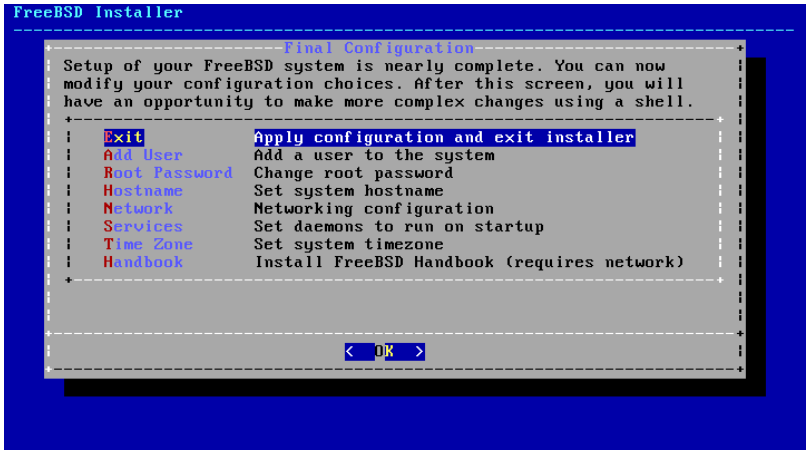


Abbildung 3.52. Letzte Schritte der Konfiguration

Verwenden Sie dieses Menü, um noch letzte Änderungen oder zusätzliche Konfigurationen vor dem Abschliessen der Installation zu tätigen.

- `Add User` - Beschrieben in Abschnitt 3.9.6, „Benutzer hinzufügen“.
- `Root Password` - Beschrieben in Abschnitt 3.9.1, „Setzen des `root`-Passworts“.
- `Hostname` - Beschrieben in Abschnitt 3.5.2, „Den Rechnernamen festlegen“.
- `Network` - Beschrieben in Abschnitt 3.9.2, „Die Netzwerkschnittstelle konfigurieren“.
- `Services` - Beschrieben in Abschnitt 3.9.4, „Zu aktivierende Dienste auswählen“.
- `Time Zone` - Beschrieben in Abschnitt 3.9.3, „Setzen der Zeitzone“.
- `Handbook` - Herunterladen und installieren des FreeBSD Handbuchs.

Nachdem die lezten Konfigurationsschritte beendet sind, wählen Sie Exit, um die Installation zu verlassen.

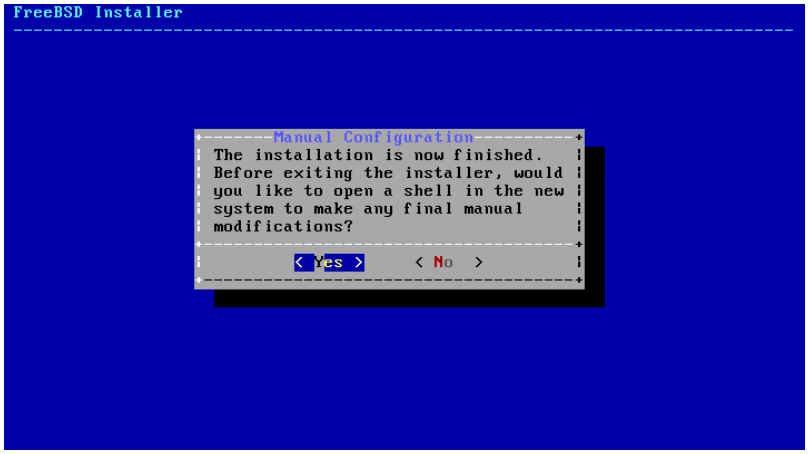


Abbildung 3.53. Manuelle Konfiguration

bsdinstall wird nach zusätzlichen Konfigurationen, die noch zu tätigen sind, fragen, bevor in das neue System gebootet wird. Wählen Sie [Yes], um in eine Shell innerhalb des neuen Systems zu wechseln oder [No], um mit dem letzten Schritt der Installation zu beginnen.

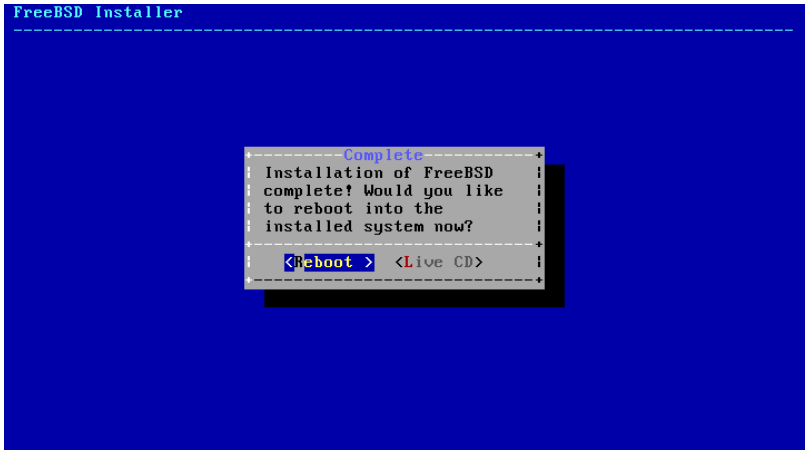


Abbildung 3.54. Die Installation vervollständigen

Wenn weitere Konfigurationen oder besondere Einstellungen benötigt werden, kann durch auswählen von [Live CD] das Installationsmedium im Live-CD Modus gestartet werden.

Wenn die Installation vollständig ist, wählen Sie [Reboot], um den Computer neu zu starten und das neu installierte FreeBSD-System zu booten. Vergessen Sie nicht, die FreeBSD Installations-CD, -DVD oder den USB-Stick zu entfernen, oder der Computer wird erneut davon starten.

3.9.8. FreeBSD starten und herunterfahren

3.9.8.1. FreeBSD/i386 starten

Wenn FreeBSD startet, werden viele Informationsmeldungen ausgegeben. Die meisten davon werden aus dem Bildschirm verschwinden, das ist normal. Nachdem das System den Startvorgang abgeschlossen hat, wird eine Anmeldeaufforderungen angezeigt. Um Nachrichten, die aus dem Bildschirm gelaufen sind, zu sehen, aktivieren Sie durch drücken von Scroll-Lock den *scroll-back buffer*. Die Tasten PgUp, PgDn und die Pfeiltasten dienen zur Navigation durch die Nachrichten. Durch erneutes drücken von Scroll-Lock wird der Bildschirm wieder entsperrt und kehrt zur normalen Anzeige zurück.

Am `login:`-Bildschirm geben Sie den Benutzernamen ein, den Sie während der Installation angelegt haben, in diesem Fall ist das `asample`. Vermeiden Sie die Anmeldung als `root`, ausser wenn es wirklich notwendig ist.

Der oben beschriebene scroll-back buffer ist in der Größe beschränkt, somit werden vielleicht nicht alle Nachrichten sichtbar sein. Nach dem Anmelden können die meisten davon aus der Kommandozeile aus durch eingabe von `dmesg | less` betrachtet werden. Durch drücken von q kehren Sie wieder zur Kommandozeile zurück.

Typische Startmeldungen (Versionsinformationen wurden hier weggelassen):

```
Copyright (c) 1992-2011 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
        The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.

    root@farrell.cse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC amd64
CPU: Intel(R) Core(TM)2 Duo CPU     E8400  @ 3.00GHz (3007.77-MHz K8-class CPU)
  Origin = "GenuineIntel"  Id = 0x10676  Family = 6  Model = 17  Stepping = 6
  ↺
Features=0x783fbff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,APIC,SEP,MTRR,PGE,MCA,CMOV,PAT,PSE36,MMX,FXSR,SSE,SS
  Features2=0x209<SSE3,MON,SSSE3>
  AMD Features=0x20100800<SYSCALL,NX,LM>
  AMD Features2=0x1<LAHF>
real memory  = 536805376 (511 MB)
avail memory = 491819008 (469 MB)
Event timer "LAPIC" quality 400
ACPI APIC Table: <VBOX   VBOXAPIC>
ioapic0: Changing APIC ID to 1
ioapic0 <Version 1.1> irqs 0-23 on motherboard
kbd1 at kbdmux0
acpi0: <VBOX VBOXXSDT> on motherboard
acpi0: Power Button (fixed)
acpi0: Sleep Button (fixed)
Timecounter "ACPI-fast" frequency 3579545 Hz quality 900
acpi_timer0: <32-bit timer at 3.579545MHz> port 0x4008-0x400b on acpi0
cpu0: <ACPI CPU> on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci0: <ACPI PCI bus> on pcib0
isab0: <PCI-ISA bridge> at device 1.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <Intel PIIX4 UDMA33 controller> port ↺
0x1f0-0x1f7,0x3f6,0x170-0x177,0x376,0xd000-0xd00f at device 1.1 on pci0
ata0: <ATA channel 0> on atapci0
ata1: <ATA channel 1> on atapci0
vgapci0: <VGA-compatible display> mem 0xe0000000-0xe0ffffff irq 18 at device 2.0 on pci0
em0: <Intel(R) PRO/1000 Legacy Network Connection 1.0.3> port 0xd010-0xd017 mem ↺
0xf0000000-0xf001ffff irq 19 at device 3.0 on pci0
em0: Ethernet address: 08:00:27:9f:e0:92
pci0: <base peripheral> at device 4.0 (no driver attached)
pcm0: <Intel ICH (82801AA)> port 0xd100-0xd1ff,0xd200-0xd23f irq 21 at device 5.0 on pci0
pcm0: <SigmaTel STAC9700/83/84 AC97 Codec>
ohci0: <OHCI (generic) USB controller> mem 0xf0804000-0xf0804fff irq 22 at device 6.0 ↺
on pci0
usbus0: <OHCI (generic) USB controller> on ohci0
pci0: <bridge> at device 7.0 (no driver attached)
```

```
acpi_acad0: <AC Adapter> on acpi0
atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0
atkbd0: <AT Keyboard> irq 1 on atkbdc0
kbd0 at atkbd0
atkbd0: [GIANT-LOCKED]
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: model IntelliMouse Explorer, device ID 4
attimer0: <AT timer> port 0x40-0x43,0x50-0x53 on acpi0
Timecounter "i8254" frequency 1193182 Hz quality 0
Event timer "i8254" frequency 1193182 Hz quality 100
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
atrtc0: <AT realtime clock> at port 0x70 irq 8 on isa0
Event timer "RTC" frequency 32768 Hz quality 0
ppc0: cannot reserve I/O port range
Timecounters tick every 10.000 msec
pcm0: measured ac97 link rate at 485193 Hz
em0: link state changed to UP
usbus0: 12Mbps Full Speed USB v1.0
ugen0.1: <Apple> at usbus0
uhub0: <Apple OHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0
cd0 at ata1 bus 0 scbus1 target 0 lun 0
cd0: <VBOX CD-ROM 1.0> Removable CD-ROM SCSI-0 device
cd0: 33.300MB/s transfers (UDMA2, ATAPI 12bytes, PIO 65534bytes)
cd0: Attempt to query device size failed: NOT READY, Medium not present
ada0 at ata0 bus 0 scbus0 target 0 lun 0
ada0: <VBOX HARDDISK 1.0> ATA-6 device
ada0: 33.300MB/s transfers (UDMA2, PIO 65536bytes)
ada0: 12546MB (25694208 512 byte sectors: 16H 63S/T 16383C)
ada0: Previously was known as ad0
Timecounter "TSC" frequency 3007772192 Hz quality 800
Root mount waiting for: usbus0
uhub0: 8 ports with 8 removable, self powered
Trying to mount root from ufs:/dev/ada0p2 [rw]...
Setting hostuuid: 1848d7bf-e6a4-4ed4-b782-bd3f1685d551.
Setting hostid: 0xa03479b2.
Entropy harvesting: interrupts ethernet point_to_point kickstart.
Starting file system checks:
/dev/ada0p2: FILE SYSTEM CLEAN; SKIPPING CHECKS
/dev/ada0p2: clean, 2620402 free (714 frags, 327461 blocks, 0.0% fragmentation)
Mounting local file systems:.
vboxguest0 port 0xd020-0xd03f mem 0xf0400000-0xf07fffff,0xf0800000-0xf0803fff irq 20 at ↺
device 4.0 on pci0
vboxguest: loaded successfully
Setting hostname: machine3.example.com.
Starting Network: lo0 em0.
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
        options=3<RXCSUM,TXCSUM>
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3
        inet 127.0.0.1 netmask 0xff000000
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=9b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM>
        ether 08:00:27:9f:e0:92
        nd6 options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL>
        media: Ethernet autoselect (1000baseT <full-duplex>)
        status: active
Starting devd.
Starting Network: usbus0.
DHCPREQUEST on em0 to 255.255.255.255 port 67
DHCPACK from 10.0.2.2
bound to 192.168.1.142 -- renewal in 43200 seconds.
add net ::ffff:0.0.0.0: gateway ::1
```

```
add net ::0.0.0.0: gateway ::1
add net fe80::: gateway ::1
add net ff02::: gateway ::1
ELF ldconfig path: /lib /usr/lib /usr/lib/compat /usr/local/lib
32-bit compatibility ldconfig path: /usr/lib32
Creating and/or trimming log files.
Starting syslogd.
No core dumps found.
Clearing /tmp (X related).
Updating motd:.
Configuring syscons: blanktime.
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
10:a0:f5:af:93:ae:a3:1a:b2:bb:3c:35:d9:5a:b3:f3 root@machine3.example.com
The key's randomart image is:
+--[RSA1 1024]----+
|    o..          |
|   o . .         |
|  .   o          |
|       o         |
|    o   S        |
|   + + o         |
|o . + *          |
|o+ ..+ .         |
|==o..o+E         |
+-----------------+
Generating public/private dsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
7e:1c:ce:dc:8a:3a:18:13:5b:34:b5:cf:d9:d1:47:b2 root@machine3.example.com
The key's randomart image is:
+--[ DSA 1024]----+
|       ..     . .|
|      o  .   . + |
|     . ..   . E .|
|    . .  o o . . |
|     +  S = .    |
|    +  . = o     |
|     +  . * .    |
|    . .  o .     |
|      .o. .      |
+-----------------+
Starting sshd.
Starting cron.
Starting background file system checks in 60 seconds.

Thu Oct  6 19:15:31 MDT 2011

FreeBSD/amd64 (machine3.example.com) (ttyv0)

login:
```

Das Generieren der RSA- und DSA-Schlüssel kann auf langsameren Rechnern einige Zeit benötigen. Dies geschieht nur während der Startphase einer neuen Installation und auch nur, wenn sshd zum automatischen Start gesetzt ist. Die nachfolgenden Startvorgänge werden schneller sein.

FreeBSD installiert standardmässig keine graphische Umgebung, jedoch stehen viele zur Verfügung. Lesen Sie Kapitel 6, *Das X-Window-System* für weitere Informationen.

3.9.9. FreeBSD herunterfahren

Das korrekte herunterfahren eines FreeBSD-Computers hilft, beugt dem Datenverlust vor und schützt sogar die Hardware vor Schäden. Schalten Sie nicht den Strom ab, bevor das System ordnungsgemäß heruntergefahren wurde. Wenn der Benutzer ein Mitglied der `wheel`-Gruppe ist, können Sie zum Superuser durch die Eingabe von `su` und der anschliessenden Eingabe des Passworts von `root` werden. Andernfalls melden Sie sich mit `root` an und verwenden den Befehl `shutdown -p now`. Das System wird jetzt sauber heruntergefahren und den Rechner ausschalten.

Die Ctrl+Alt+Del Kombination kann verwendet werden, um das System neu zu starten, jedoch wird dies nicht während des normalen Betriebs empfohlen.

3.10. Fehlerbehebung

Der folgende Abschnitt behandelt einfache Fehlerbehebungen für die Installation, wie beispielsweise häufig auftretende Fehler, die von Anwendern berichtet wurden.

3.10.1. Was man tun sollte, wenn etwas schiefgeht

Wegen verschiedener Limitierungen der PC-Architektur ist es unmöglich dass die Geräteerkennung 100% verlässlich funktioniert. Jedoch gibt es ein paar Dinge, die man tun kann, wenn es fehlschlägt.

Überprüfen Sie die Hardware Notes (http://www.FreeBSD.org/releases/index.html) nach der Version von FreeBSD, um sicher zu stellen, dass die Hardware auch unterstützt wird.

Wenn die Hardware unterstützt wird und Sie immer noch Abstürze oder andere Probleme erleben, müssen Sie einen eigenen Kernel bauen. Diese Prozedur wird in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben. Das erlaubt es, Unterstützung für Geräte, die im `GENERIC`-Kernel nicht vorhanden sind, hinzuzufügen. Der Kernel auf den Bootmedien ist mit der Annahme konfiguriert, dass die Hardwaregeräte sich in ihren Fabrikeinstellungen in Bezug auf IRQs, I/O-Adressen und DMA-Kanälen befinden. Wenn die Hardware neu konfiguriert wurde, werden Sie möglicherweise die Konfiguration des Kernels bearbeiten und diesen neu erstellen müssen, um FreeBSD mitzuteilen, wo es gewisse Dinge finden kann.

Es ist auch möglich, dass ein fehlerhaft erkanntes Gerät die Erkennung eines vorhandenen, späteren Geräts ebenfalls fehlschlagen lässt. In diesem Fall sollte die Erkennung des fehlerhaften Gerätetreibers deaktiviert werden.

Anmerkung

Manche Installationsprobleme können Aktualisierung der Firmware auf verschiedenen Hardwarekomponenten verhindert oder verringert werden, meistens am Mainboard. Mit Mainboard-Firmware ist für gewöhnlich das BIOS gemeint. Die meisten Mainboard- und Computerhersteller haben eine Webseite mit Aktualisierungen und Informationen zur Durchführung.

Hersteller raten meist von einer Aktualisierung des Mainboard-BIOS ab, ausser es gibt einen guten Grund dafür, wie beispielsweise eine kritische Aktualisierung. Der Aktualisierungsvorgang *kann* schiefgehen, was das BIOS unvollständig macht und den Computer nicht mehr starten lässt.

3.10.2. Fragen und Antworten zur Fehlerbehebung

F: Mein System hängt während die Geräteerkennung beim Starten durchgeführt wird oder verhält sich merkwürdig während der Installation.

A: FreeBSD macht starken Gebrauch vom ACPI-Dienst des Systems auf den i386-, amd64-, and ia64-Platformen, um den System bei der Konfiguration während des Startvorgangs zu helfen. Leider existieren immer noch Fehler im ACPI-Treiber, in den Mainboards und der BIOS-Firmware. ACPI kann durch setzen der Einstellung `hint.acpi.0.disabled` im dritten Teil des Bootloaders deaktiviert werden:

```
set hint.acpi.0.disabled="1"
```

Dies wird nach jedem Neustart des Systems wieder zurückgesetzt, also ist es notwendig, die Zeile `hint.acpi.0.disabled="1"` zu der Datei `/boot/loader.conf` hinzuzufügen. Weitere Informationen über den Bootloader lassen sich in Abschnitt 13.1, „Übersicht“ nachlesen.

3.11. Verwendung der Live-CD

Neben dem Installationsprogramm steht auch die Live-CD von FreeBSD auf der CD zur Verfügung. Die Live-CD ist für Benutzer, die sich fragen, ob FreeBSD das richtige Betriebssystem für sie ist und die vor der Installation noch einige Merkmale und Eigenschaften testen wollen.

Anmerkung

Die folgenden Punkte sollten beachtet werden, während die Live-CD benutzt wird:

- Um Zugriff auf das System zu bekommen, wird eine Authentifizierung benötigt. Der Benutzername ist `root` und das Kennwort bleibt leer.
- Da das System direkt von der CD ausgeführt wird, ist die Geschwindigkeit deutlich langsamer als bei einem System, das auf einer Festplatte installiert ist.
- Die Live-CD enthält nur eine Eingabeaufforderung und keine graphische Oberfläche.

Kapitel 4. Grundlagen des UNIX Betriebssystems

4.1. Übersicht

Dieses Kapitel umfasst die grundlegenden Kommandos und Funktionsweisen des FreeBSD-Betriebssystems. Viel von diesem Material gilt auch für jedes andere UNIX®-artige System. Neue Benutzer von FreeBSD sollten dieses Kapitel aufmerksam lesen.

Dieser Abschnitt behandelt die folgenden Themen:

- virtuelle Konsolen,
- Erstellung und Verwaltung von Benutzern und Gruppen in FreeBSD,
- Zugriffsrechte unter UNIX® sowie Datei-Flags unter FreeBSD,
- Zugriffskontrolllisten für Dateisysteme,
- die Verzeichnisstruktur von FreeBSD,
- Organisation von Dateisystemen unter FreeBSD,
- Ein- und Abhängen von Dateisystemen,
- Prozesse, Dämonen und Signale,
- Shells und die Login-Umgebung,
- Texteditoren,
- Geräte und Gerätedateien,
- wie Sie in den Manualpages nach weiteren Informationen suchen können.

4.2. Virtuelle Konsolen und Terminals

Wenn das FreeBSD-System so konfiguriert wurde, dass es ohne eine grafische Benutzeroberfläche startet, wird das System nach dem Start einen Anmeldeprompt ausgeben, wie in diesem Beispiel zu sehen:

```
FreeBSD/amd64 (pc3.example.org) (ttyv0)

login:
```

Die erste Zeile enthält einige Informationen über das System. `amd64` zeigt an, dass auf dem System in diesem Beispiel eine 64-Bit Version von FreeBSD läuft. Der Hostname ist `pc3.example.org` und `ttyv0` gibt an, dass dies die „Systemkonsole“ ist. Die zweite Zeile zeigt den Anmeldeprompt.

Da FreeBSD ein Mehrbenutzersystem ist, muss es die verschiedenen Benutzer voneinander unterscheiden können. Dies wird dadurch erreicht, dass sich jeder Benutzer zuerst am System anmelden muss, um Zugriff auf die Programme zu bekommen. Jeder Benutzer hat einen eindeutigen „Benutzernamen“ und ein persönliches „Kennwort“.

Um sich auf der Systemkonsole anzumelden, geben Sie den Benutzernamen ein, der während der Systeminstallation, wie in ??? beschrieben, konfiguriert wurde und drücken Sie Enter. Geben Sie dann das zum Benutzernamen zugeordnete Passwort ein und drücken Enter. Das Passwort wird aus Sicherheitsgründen *nicht angezeigt*.

Sobald das richtige Passwort eingegeben wird, wird die Nachricht des Tages (MOTD) gefolgt von einer Eingabeaufforderung ausgegeben. In Abhängigkeit der verwendeten Shell des Benutzers wird der Prompt mit dem Zeichen `#`, `$` oder `%` dargestellt. Der Prompt zeigt an, dass der Benutzer jetzt an der FreeBSD Systemkonsole angemeldet ist und nun alle verfügbaren Befehle probieren kann.

4.2.1. Virtuelle Konsolen

Obwohl die Systemkonsole dazu verwendet werden kann, um mit dem System zu interagieren, wird sich ein Benutzer in der Regel an einer virtuellen Konsole im FreeBSD-System anmelden. Das liegt daran, dass die Systemmeldungen standardmäßig auf der Systemkonsole angezeigt werden und somit die Meldungen des Befehls oder einer Datei, die der Benutzer gerade bearbeitet, überschrieben werden.

In der Voreinstellung ist FreeBSD so konfiguriert, dass viele virtuelle Konsolen zur Eingabe von Befehlen zur Verfügung stehen. Jede virtuelle Konsole verfügt über einen eigenen Anmeldeprompt und eine Shell. Sie können ganz einfach zwischen den virtuellen Konsolen umschalten. Dies ist vergleichbar mit mehreren geöffneten Fenstern in einer graphischen Umgebung.

Die Tastenkombinationen Alt+F1 bis Alt+F8 sind in FreeBSD zum Umschalten zwischen virtuellen Konsolen reserviert. Verwenden Sie Alt+F1 um auf die Systemkonsole (`ttyv0`) zu wechseln, Alt+F2 für die erste virtuelle Konsole (`ttyv1`, Alt+F3 für die zweite virtuelle Konsole (`ttyv2`, und so weiter.

Beim Wechsel von einer Konsole zur nächsten wird die Bildschirmausgabe von FreeBSD verwaltet. Dies erzeugt die Illusion mehrerer Bildschirme und Tastaturen, an denen Kommandos abgesetzt werden können. Die Programme, die in einer virtuellen Konsole gestartet werden, laufen auch dann weiter, wenn der Benutzer auf eine andere virtuelle Konsole wechselt.

Lesen Sie syscons(4), atkbd(4), vidcontrol(1) und kbdcontrol(1) für eine recht technische Beschreibung der FreeBSD-Konsole und der Tastatur-Treiber.

In FreeBSD wird die Anzahl der verfügbaren virtuellen Konsolen in diesem Abschnitt von `/etc/ttys` konfiguriert:

```
# name getty         type  status comments
#
ttyv0   "/usr/libexec/getty Pc"         cons25  on  secure
# Virtual terminals
ttyv1   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv2   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv3   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv4   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv5   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv6   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv7   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv8   "/usr/X11R6/bin/xdm -nodaemon"  xterm   off secure
```

Um eine virtuelle Konsole zu deaktivieren, setzen Sie ein Kommentarzeichen (`#` an den Anfang der Zeile für die entsprechende Konsole. Um bspw. die Anzahl der verfügbaren virtuellen Konsolen von acht auf vier zu reduzieren, setzen Sie ein `#` an den Anfang der letzten vier Zeilen, den virtuellen Konsolen `ttyv5` bis `ttyv8`. Kommentieren Sie nicht die Zeile für die Systemkonsole `ttyv0` aus! Beachten Sie, dass die letzte virtuelle Konsole (`ttyv8`) zum Wechsel auf die graphische Oberfläche gedacht ist, wenn Xorg wie im Kapitel 6, *Das X-Window-System* installiert und konfiguriert ist.

ttys(5) enthält eine ausführliche Beschreibung der Spalten dieser Datei und der verfügbaren Optionen für virtuelle Konsolen.

4.2.2. Single-User-Modus

Das FreeBSD Boot-Menü verfügt über eine Option „Boot Single User“. Wird diese Option gewählt, bootet das System in einen speziellen Modus, der als „Single-User-Modus“ bekannt ist. Dieser Modus wird normalerweise zur Reparatur des Systems verwendet, bspw. wenn das System nicht mehr startet, oder das `root`-Passwort zurückgesetzt werden muss. Im Single-User-Modus haben Sie keinen Zugriff auf das Netzwerk und es stehen Ihnen keine

weiteren virtuellen Konsolen zur Verfügung. Allerdings haben Sie vollen Zugriff auf das System und in der Voreinstellung wird das `root`-Passwort nicht benötigt. Aus diesem Grund wird ein physischer Zugriff auf die Tastatur benötigt, um in diesem Modus zu booten. Zur Absicherung eines FreeBSD-Systems sollte ermittelt werden, welche Personen physischen Zugriff auf die Tastatur bekommen sollen.

Die Einstellungen für den Single-User-Modus befinden sich diesem Abschnitt von `/etc/ttys` :

```
# name  getty                           type  status  comments
#
# If console is marked "insecure", then init will ask for the root password
# when going to single-user mode.
console none                            unknown  off  secure
```

In der Voreinstellung ist der Status auf `secure` eingestellt. Das setzt voraus, dass der physische Zugriff auf die Tastatur entweder unwichtig ist, oder über eine Sicherheitsrichtlinie geregelt wird. Wenn der Status auf `insecure` eingestellt wird, wird davon ausgegangen, dass die Umgebung selbst unsicher ist, da jeder Zugriff auf die Tastatur hat. FreeBSD wird dann nach dem `root`-Passwort fragen, wenn ein Benutzer versucht in den Single-User-Modus zu booten.

Anmerkung

Setzen Sie `insecure` nicht leichtfertig ein! Wenn das `root`-Passwort vergessen wird, wird es schwierig in den Single-User-Modus zu gelangen, wenn man den Bootprozess von FreeBSD nicht genau versteht.

4.2.3. Den Videomodus der Konsole anpassen

Der Standard-Videomodus der FreeBSD-Konsole kann auf jeden Modus eingestellt werden, der von der Grafikkarte und dem Monitor unterstützt wird (beispielsweise 1024x768 oder 1280x1024). Um eine andere Einstellung zu verwenden, muss das `VESA`-Modul geladen werden:

```
# kldload vesa
```

Um festzustellen, welche Video-Modi von der Hardware unterstützt werden, nutzen Sie vidcontrol(1). Um eine Liste aller unterstützten Modi zu sehen, verwenden Sie diesen Befehl:

```
# vidcontrol -i mode
```

Die Ausgabe dieses Befehls listet alle Videomodi, die von der Hardware unterstützt werden. Um einen neuen Video-Modi zu wählen, wird der entsprechende Modus als `root`-Benutzer an vidcontrol(1) übergeben:

```
# vidcontrol MODE_279
```

Um diese Einstellung dauerhaft zu speichern, muss folgende Zeile in `/etc/rc.conf` hinzugefügt werden:

```
allscreens_flags="MODE_279"
```

4.3. Benutzer und grundlegende Account-Verwaltung

FreeBSD ermöglicht es mehreren Benutzern, den Computer zur selben Zeit zu benutzen. Es kann immer nur ein Benutzer vor der Konsole sitzen, aber es können sich beliebig viele Benutzer über das Netzwerk am System anmelden. Jeder Benutzer muss einen Account haben, um das System benutzen zu können.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie

- die verschiedenen Account-Typen von FreeBSD kennen,
- wissen, wie Sie Accounts angelegen, verändern oder löschen,

- wissen, wie Sie Limits für einen Benutzer oder eine Gruppe setzen, um beispielsweise Ressourcen, wie Speicher oder CPU-Zeit einzuschränken,
- wissen, wie Sie Gruppen erstellen und Benutzer zu diesen Gruppen hinzufügen.

4.3.1. Account-Typen

Jeder Zugriff auf das FreeBSD-System geschieht über Accounts und alle Prozesse werden von Benutzern gestartet, also sind Benutzer- und Account-Verwaltung von wesentlicher Bedeutung.

Es gibt drei Haupttypen von Accounts: Systembenutzer, Benutzer-Accounts und der Superuser-Account.

4.3.1.1. Systembenutzer

Systembenutzer starten Dienste wie DNS, Mail-Server und Web-Server. Der Grund dafür ist die Sicherheit; wenn die Programme von dem Superuser gestartet werden, können Sie ohne Einschränkungen handeln.

Beispiele von Systembenutzern sind `daemon`, `operator`, `bind`, `news` und `www`.

`nobody` ist der generische unprivilegierte Systembenutzer. Bedenken Sie aber, dass je mehr Dienste `nobody` benutzen, desto mehr Dateien und Prozesse diesem Benutzer gehören und dieser Benutzer damit umso privilegierter wird.

4.3.1.2. Benutzer-Accounts

Benutzer-Accounts sind realen Personen zugeordnet und sind das primäre Mittel des Zugriffs das System. Jede Person, die Zugriff auf das System bekommt, sollte einen eindeutigen Benutzer-Account besitzen. Dies erlaubt es dem Administrator herauszufinden, wer was macht. Gleichzeitig werden die Benutzer daran gehindert, die Einstellungen anderer Benutzer zu zerstören.

Jeder Benutzer kann die eigene Umgebung anpassen, bspw. seine voreingestellte Shell, Editor, Tastenbelegungen und Spracheinstellungen.

Mit jedem Account eines FreeBSD-Systems sind bestimmte Informationen verknüpft:

Loginnamen
: Der Loginname wird am `login:` Prompt eingegeben. Jeder Benutzer muss einen eindeutigen Benutzernamen haben. Es gibt eine Reihe von Regeln für die Erstellung von gültigen Loginnamen, die in passwd(5) dokumentiert sind. Es wird aus Kompatibilitätsgründen empfohlen, Benutzernamen zu verwenden, die aus Kleinbuchstaben bestehen und bis zu acht Zeichen lang sind.

Passwort
: Jeder Account ist mit einem Passwort verknüpft.

User ID (UID)
: Die User ID (UID) ist eine Zahl, die verwendet wird, um die Benutzer auf dem FreeBSD-System eindeutig zu identifizieren. Programme, die einen Loginnamen akzeptieren, wandeln diesen zuerst in eine UID um. Es wird empfohlen, nur UIDs kleiner 65535 zu verwenden, da höhere Werte Kompatibilitätsprobleme mit einigen Anwendungen verursachen können.

Group ID (GID)
: Die Group ID (GID) ist eine Zahl, die verwendet wird, um die primäre Gruppe eines Benutzers eindeutig zu identifizieren. Gruppen sind ein Mechanismus zur Steuerung des Zugriffs auf Ressourcen über die GID eines Benutzers anstelle der UID. Dies kann die Größe einiger Konfigurationsdateien signifikant reduzieren und ermöglicht es Benutzern, Mitglied mehreren Gruppen zu sein. Es wird empfohlen, GIDs kleiner 65535 zu verwenden, da höhere Werte bei einigen Anwendungen große Probleme verursachen können.

Login-Klasse
: Login-Klassen erweitern das Gruppenkonzept. Sie erhöhen die Flexibilität des Systems in der Handhabung der verschiedenen Accounts. Login-Klassen werden auch im Abschnitt 4.3.3, „Benutzer einschränken“ diskutiert.

Gültigkeit von Passwörtern
: In der Voreinstellung verfallen Passwörter nicht. Allerdings können Passwortwechsel nach einer gewissen Zeit auf Basis einzelner Accounts erzwungen werden.

Verfallszeit eines Accounts
: In der Voreinstellung verfallen unter FreeBSD keine Accounts. Wenn Sie Accounts einrichten, die nur für eine bestimmte Zeit gültig sein sollen, beispielsweise Accounts für Teilnehmer eines Praktikums, können Sie mit pw(8) die Gültigkeitsdauer des Accounts angeben. Nachdem die angegebene Zeitspanne verstrichen ist, kann dieser Account nicht mehr zum Anmelden verwendet werden, obwohl alle Verzeichnisse und Dateien, die diesem Account gehören, noch vorhanden sind.

vollständiger Benutzername
: FreeBSD identifiziert einen Account eindeutig über den Loginnamen, der aber keine Ähnlichkeit mit dem richtigen Namen des Benutzers haben muss. Ähnlich wie bei einem Kommentar, kann diese Information Leerzeichen, Großbuchstaben und mehr als 8 Zeichen enthalten.

Heimatverzeichnis
: Das Heimatverzeichnis gibt den vollständigen Pfad zu dem Verzeichnis an, in dem sich der Benutzer nach erfolgreicher Anmeldung befindet. Es ist üblich, alle Heimatverzeichnisse unter `/home/Loginname` oder `/usr/home/Loginname` anzulegen. Im Heimatverzeichnis oder in dort angelegten Verzeichnissen werden die Dateien eines Benutzers gespeichert.

Login-Shell
: Grundsätzlich ist die Shell, von denen es viele unterschiedliche gibt, eine Schnittstelle zum System. Die bevorzugte Shell eines Benutzers kann seinem Account zugeordnet werden.

4.3.1.3. Der Superuser-Account

Der Superuser-Account, normalerweise `root` genannt, ist vorkonfiguriert und erleichtert die Systemverwaltung, sollte aber nicht für alltägliche Aufgaben wie das Verschicken und Empfangen von Mails, Erforschen des Systems oder Programmierung benutzt werden.

Der Superuser kann, im Gegensatz zu normalen Benutzer-Accounts, ohne Beschränkungen operieren und die falsche Anwendung des Superuser-Accounts kann in spektakulären Katastrophen resultieren. Benutzer-Accounts sind nicht in der Lage, das System versehentlich zu zerstören, deswegen wird empfohlen, normale Benutzer-Accounts zu verwenden, solange nicht zusätzliche Privilegien benötigt werden.

Kommandos, die Sie als Superuser eingeben, sollten Sie immer doppelt und dreifach überprüfen, da ein zusätzliches Leerzeichen oder ein fehlender Buchstabe irreparablen Datenverlust bedeuten kann.

Es gibt mehrere Möglichkeiten Superuser-Rechte zu bekommen. Obwohl man sich direkt als `root` anmelden kann, wird von dieser Methode dringend abgeraten.

Verwenden Sie stattdessen su(1) um zum Superuser zu werden. Wenn Sie noch ein `-` eingeben, wird der Benutzer auch die Umgebung des Root-Benutzers erben. Der Benutzer, der diesen Befehl ausführt muss Mitglied der Gruppe `wheel` sein, oder der Befehl schlägt fehl. Zudem muss der Benutzer das Kennwort für den Benutzer-Account `root` kennen.

In diesem Besipiel wird der Benutzer nur zum Superuser, um `make install` auszuführen, da dieser Befehl Superuser-Rechte erfordert. Nachdem der Befehl ausgeführt wurde, kann der Benutzer `exit` eingeben, um den Superuser-Account zu verlassen und zu den Privilegien des Benutzer-Accounts zurückkehren.

Beispiel 4.1. Ein Programm als Superuser installieren

```
% configure
% make
```

```
% su -
Password:
# make install
# exit
%
```

Das in FreeBSD enthaltene su(1) funktioniert gut für einzelne Systeme oder in kleineren Netzwerken, mit nur einem Administrator. Eine Alternative ist es, das Paket oder den Port security/sudo zu installieren. Diese Software bietet eine Protokollierung von Aktivitäten und ermöglicht es dem Administrator zu bestimmen, welche Benutzer welche Befehle als Superuser ausführen dürfen.

4.3.2. Accounts verändern

FreeBSD stellt eine Vielzahl an Programmen bereit, um Accounts zu verändern. Die gebräuchlichsten Kommandos sind in Tabelle 4.1, „Programme zur Verwaltung von Benutzer-Accounts“ gefolgt von einer detaillierten Beschreibung, zusammengefasst. Weitere Informationen und Anwendungsbeispiele finden Sie in der Manualpage des jeweiligen Programms.

Tabelle 4.1. Programme zur Verwaltung von Benutzer-Accounts

Programm	Zusammenfassung
adduser(8)	Das empfohlene Werkzeug, um neue Accounts zu erstellen.
rmuser(8)	Das empfohlene Werkzeug, um Accounts zu löschen.
chpass(1)	Ein flexibles Werkzeug, um Informationen in der Account-Datenbank zu verändern.
passwd(1)	Ein Werkzeug, um Passwörter von Accounts zu ändern.
pw(8)	Ein mächtiges und flexibles Werkzeug um alle Informationen über Accounts zu ändern.

4.3.2.1. `adduser`

Das empfohlene Programm zum Hinzufügen neuer Benutzer ist adduser(8). Wenn ein neuer Benutzer hinzugefügt wird, aktualisiert das Programm automatisch `/etc/passwd` und `/etc/group`. Es erstellt auch das Heimatverzeichnis für den Benutzer, kopiert die Standardkonfigurationsdateien aus `/usr/share/skel` und kann optional eine „Willkommen``-Nachricht an den neuen Benutzer versenden. Das Programm muss als Superuser ausgeführt werden.

Das Werkzeug adduser(8) arbeitet interaktiv und führt durch die einzelnen Schritte, wenn ein neues Benutzerkonto erstellt wird. Wie in Beispiel 4.2, „Einen Benutzer unter FreeBSD anlegen“ zu sehen ist, müssen Sie entweder die benötigte Information eingeben oder Return drücken, um den Vorgabewert in eckigen Klammern zu akzeptieren. In diesem Beispiel wird der Benutzer in die Gruppe `wheel` aufgenommen, was es ihm erlaubt mit su(1) zum Superuser zu werden. Wenn Sie fertig sind, können Sie entweder einen weiteren Benutzer erstellen oder das Programm beenden.

Beispiel 4.2. Einen Benutzer unter FreeBSD anlegen

```
# adduser
Username: jru
Full name: J. Random User
Uid (Leave empty for default):
Login group [jru]:
Login group is jru. Invite jru into other groups? []: wheel
```

```
Login class [default]:
Shell (sh csh tcsh zsh nologin) [sh]: zsh
Home directory [/home/jru]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]:
Use an empty password? (yes/no) [no]:
Use a random password? (yes/no) [no]:
Enter password:
Enter password again:
Lock out the account after creation? [no]:
Username   : jru
Password   : ****
Full Name  : J. Random User
Uid        : 1001
Class      :
Groups     : jru wheel
Home       : /home/jru
Shell      : /usr/local/bin/zsh
Locked     : no
OK? (yes/no): yes
adduser: INFO: Successfully added (jru) to the user database.
Add another user? (yes/no): no
Goodbye!
#
```

Anmerkung

Wenn Sie das Passwort eingeben, werden weder Passwort noch Sternchen angezeigt. Passen Sie auf, dass Sie das Passwort korrekt eingeben.

4.3.2.2. `rmuser`

Benutzen Sie rmuser(8) als Superuser, um einen Account vollständig aus dem System zu entfernen. Dieses Programm führt die folgenden Schritte durch:

1. Entfernt den crontab(1) Eintrag des Benutzers, wenn dieser existiert.
2. Entfernt alle at(1) jobs, die dem Benutzer gehören.
3. Schließt alle Prozesse des Benutzers.
4. Entfernt den Benutzer aus der lokalen Passwort-Datei des Systems.
5. Entfernt optional das Heimatverzeichnis des Benutzers, falls es dem Benutzer gehört.
6. Entfernt eingegangene E-Mails des Benutzers aus `/var/mail` .
7. Entfernt alle Dateien des Benutzers aus temporären Dateispeicherbereichen wie `/tmp`.
8. Entfernt den Loginnamen von allen Gruppen, zu denen er gehört, aus `/etc/group` . Wenn eine Gruppe leer wird und der Gruppenname mit dem Loginnamen identisch ist, wird die Gruppe entfernt. Das ergänzt sich mit den einzelnen Benutzer-Gruppen, die von adduser(8) für jeden neuen Benutzer erstellt werden.

Der Superuser-Account kann nicht mit rmuser(8) entfernt werden, da dies in den meisten Fällen das System unbrauchbar macht.

Als Vorgabe wird ein interaktiver Modus benutzt.

Beispiel 4.3. Interaktives Löschen von Accounts mit `rmuser`

```
# rmuser jru
Matching password entry:
jru:*:1001:1001::0:0:J. Random User:/home/jru:/usr/local/bin/zsh
Is this the entry you wish to remove? y
Remove user's home directory (/home/jru)? y
Removing user (jru): mailspool home passwd.
#
```

4.3.2.3. `chpass`

Jeder Benutzer kann chpass(1) verwenden, um die Shell und persönliche Informationen des Benutzerkontos zu verändern. Der Superuser kann dieses Werkzeug benutzen, um zusätzliche Kontoinformationen für alle Benutzer zu ändern.

Werden neben dem optionalen Loginnamen keine weiteren Optionen angegeben, zeigt chpass(1) einen Editor mit Account-Informationen an. Wenn der Benutzer den Editor verlässt, wird die Account-Datenbank mit den neuen Informationen aktualisiert.

Anmerkung

Dieses Programm fragt nach dem Verlassen des Editors nach dem Passwort, es sei denn, man ist als Superuser angemeldet.

In Beispiel 4.4, „`chpass` als Superuser verwenden“ hat der Superuser `chpass jru` eingegeben. Es werden die Felder ausgegeben, die für diesen Benutzer geändert werden können. Wenn stattdessen `jru` diesen Befehl aufruft, werden nur die letzten sechs Felder ausgegeben. Dies ist in Beispiel 4.5, „`chpass` als normaler Benutzer verwenden“ zu sehen.

Beispiel 4.4. `chpass` als Superuser verwenden

```
#Changing user database information for jru.
Login: jru
Password: *
Uid [#]: 1001
Gid [# or name]: 1001
Change [month day year]:
Expire [month day year]:
Class:
Home directory: /home/jru
Shell: /usr/local/bin/zsh
Full Name: J. Random User
Office Location:
Office Phone:
Home Phone:
Other information:
```

Beispiel 4.5. `chpass` als normaler Benutzer verwenden

```
#Changing user database information for jru.
Shell: /usr/local/bin/tcsh
Full Name: J. Random User
Office Location:
Office Phone:
Home Phone:
Other information:
```

Anmerkung

Die Kommandos chfn(1) und chsh(1) sind nur Verweise auf chpass(1), genauso wie ypchpass(1), ypchfn(1) und ypchsh(1). Da NIS automatisch unterstützt wird, ist es nicht notwendig das `yp` vor dem Kommando einzugeben. NIS wird später im Kapitel 29, *Netzwerkserver* besprochen.

4.3.2.4. passwd

Jeder Benutzer kann mit passwd(1) einfach sein Passwort ändern. Um eine versehentliche oder unbefugte Änderung zu verhindern, muss bei einem Passwortwechsel zunächst das ursprüngliche Passwort eingegeben werden, bevor das neue Passwort festgelegt werden kann.

Beispiel 4.6. Das eigene Passwort wechseln

```
% passwd
Changing local password for jru.
Old password:
New password:
Retype new password:
passwd: updating the database...
passwd: done
```

Der Superuser kann jedes beliebige Passwort ändern, indem er den Benutzernamen an passwd(1) übergibt. Das Programm fordert den Superuser nicht dazu auf, das aktuelle Passwort des Benutzers einzugeben. Dadurch kann das Passwort geändert werden, falls der Benutzer sein ursprüngliches Passwort vergessen hat.

Beispiel 4.7. Als Superuser das Passwort eines anderen Accounts verändern

```
# passwd jru
Changing local password for jru.
New password:
Retype new password:
passwd: updating the database...
```

```
passwd: done
```

Anmerkung

Wie bei chpass(1) ist yppasswd(1) nur ein Verweis auf passwd(1). NIS wird von jedem dieser Kommandos unterstützt.

4.3.2.5. `pw`

Mit dem Werkzeug pw(8) können Accounts und Gruppen erstellt, entfernt, verändert und angezeigt werden. Dieses Kommando dient als Schnittstelle zu den Benutzer- und Gruppendateien des Systems. pw(8) besitzt eine Reihe mächtiger Kommandozeilenschalter, die es für die Benutzung in Shell-Skripten geeignet machen, doch finden neue Benutzer die Bedienung des Kommandos komplizierter, als die der anderen hier vorgestellten Kommandos.

4.3.3. Benutzer einschränken

FreeBSD bietet dem Systemadministrator mehrere Möglichkeiten die System-Ressourcen, die ein einzelner Benutzer verwenden kann, einzuschränken. Diese Limitierungen sind in zwei Kategorien eingeteilt: Festplattenkontingente und andere Ressourcenbeschränkungen.

Festplatten-Kontingente schränken den Plattenplatz, der einem Benutzer zur Verfügung steht, ein. Sie bieten zudem, ohne aufwändige Berechnung, einen schnellen Überblick über den verbrauchten Plattenplatz. Kontingente werden im Abschnitt 18.12, „Disk Quotas“ diskutiert.

Die anderen Ressourcenbeschränkungen umfassen die Begrenzung von CPU, Speicher und weitere Ressourcen, die ein Benutzer verbrauchen kann.

Login-Klassen werden in `/etc/login.conf` und sind in login.conf(5) im Detail beschrieben.n. Jeder Benutzer wird einer Login-Klasse zugewiesen (standardmäßig `default`) und jede Login-Klasse ist mit einem Satz von Login-Fähigkeiten verbunden. Eine Login-Fähigkeit ist ein *`Name=Wert`* Paar, in dem *`Name`* die Fähigkeit bezeichnet und *`Wert`* ein beliebiger Text ist, der in Abhänigkeit von *`Name`* entsprechend verarbeitet wird. Login-Klassen und -Fähigkeiten zu definieren ist ziemlich einfach und wird auch in login.conf(5) beschrieben.

Anmerkung

FreeBSD liest die Konfiguration aus `/etc/login.conf` normalerweise nicht direkt, sondern nur über die Datenbank `/etc/login.conf.db`, da diese eine schnellere Abfrage erlaubt. Wenn `/etc/login.conf` verändert wurde, muss die `/etc/login.conf.db` mit dem folgenden Kommando aktualisiert werden:

```
# cap_mkdb /etc/login.conf
```

Ressourcenbeschränkungen unterscheiden sich von normalen Login-Fähigkeiten zweifach. Erstens gibt es für jede Beschränkung ein aktuelles und ein maximales Limit. Das aktuelle Limit kann vom Benutzer oder einer Anwendung beliebig bis zum maximalen Limit verändert werden. Letzteres kann der Benutzer nur heruntersetzen. Zweitens gelten die meisten Ressourcenbeschränkungen für jeden vom Benutzer gestarteten Prozess, nicht für den Benutzer selbst. Beachten Sie jedoch, dass diese Unterschiede durch das spezifische Einlesen der Limits und nicht durch das System der Login-Fähigkeiten entstehen (das heißt, Ressourcenbeschränkungen sind *keine* Login-Fähigkeiten).

Hier befinden sich die am häufigsten benutzten Ressourcenbeschränkungen. Der Rest kann zusammen mit den anderen Login-Fähigkeiten in login.conf(5) gefunden werden:

`coredumpsize`

Das Limit der Größe einer core-Datei, die von einem Programm generiert wird, unterliegt aus offensichtlichen Gründen anderen Limits der Festplattenbenutzung, zum Beispiel `filesize` oder Festplattenkontingenten. Es wird aber trotzdem oft als weniger harte Methode zur Kontrolle des Festplattenplatz-Verbrauchs verwendet: Da Benutzer die core-Dateien nicht selbst erstellen und sie oft nicht löschen, kann sie diese Option davor retten, dass kein Festplattenspeicher mehr zur Verfügung steht, sollte ein großes Programm abstürzen.

`cputime`

Die maximale Rechenzeit, die ein Prozess eines Benutzers verbrauchen darf. Überschreitet der Prozess diesen Wert, wird er vom Kernel beendet.

Anmerkung

Die Rechen*zeit* wird limitiert, nicht die prozentuale Prozessorenbenutzung, wie es in einigen Feldern in top(1) und ps(1) dargestellt wird.

`filesize`

Hiermit lässt sich die maximale Größe einer Datei bestimmen, die der Benutzer besitzen darf. Im Gegensatz zu Festplattenkontingenten ist diese Beschränkung nur für jede einzelne Datei gültig und nicht für den Platz, den alle Dateien eines Benutzers verwenden.

`maxproc`

Das ist die maximale Anzahl von Prozessen, die ein Benutzer starten darf, und beinhaltet sowohl Vordergrund- als auch Hintergrundprozesse. Natürlich darf dieser Wert nicht höher sein als das System-Limit, das in `kern.maxproc` angegeben ist. Vergessen Sie auch nicht, dass ein zu kleiner Wert den Benutzer in seiner Produktivität einschränken könnte; es ist oft nützlich, mehrfach eingeloggt zu sein, oder *Pipelines* [1] zu verwenden. Ein paar Aufgaben, wie die Kompilierung eines großen Programms, starten mehrere Prozesse.

`memorylocked`

Dieses Limit gibt an, wie viel virtueller Speicher von einem Prozess maximal im Arbeitsspeicher festgesetzt werden kann (siehe auch mlock(2)). Ein paar systemkritische Programme, wie amd(8), verhindern damit einen Systemzusammenbruch, der auftreten könnte, wenn sie aus dem Speicher genommen werden.

`memoryuse`

Bezeichnet den maximalen Speicher, den ein Prozess benutzen darf und beinhaltet sowohl Arbeitsspeicher-, als auch Swap- Benutzung. Es ist kein allübergreifendes Limit für den Speicherverbrauch, aber ein guter Anfang.

`openfiles`

Mit diesem Limit lässt sich die maximale Anzahl der von einem Prozess des Benutzers geöffneten Dateien festlegen. In FreeBSD werden Dateien auch verwendet, um Sockets und *IPC*-Kanäle [2] darzustellen. Setzen Sie es deshalb nicht zu niedrig. Das System-Limit ist im `kern.maxfiles` sysctl(8) definiert.

`sbsize`

Dieses Limit beschränkt den Netzwerk-Speicher und damit die mbufs, die ein Benutzer verbrauchen darf. Es stammt aus einer Antwort auf einen DoS-Angriff, bei dem viele Netzwerk-Sockets geöffnet wurden, kann aber generell dazu benutzt werden Netzwerk-Verbindungen zu beschränken.

[1] *Pipeline* = *Leitung*. Mit *Pipes* sind Verbindungen zwischen zwei Sockets in meistens zwei verschiedenen Prozessen gemeint.

[2] *IPC* steht für *Interprocess Communication*.

`stacksize`
: Das ist die maximale Größe, auf die der Stack eines Prozesses heranwachsen darf. Das allein ist natürlich nicht genug, um den Speicher zu beschränken, den ein Programm verwenden darf. Es sollte deshalb in Verbindung mit anderen Limits gesetzt werden.

Beim Setzen von Ressourcenbeschränkungen sind noch andere Dinge zu beachten. Nachfolgend ein paar generelle Tipps, Empfehlungen und verschiedene Kommentare.

- Von `/etc/rc` beim Hochfahren des Systems gestartete Prozesse werden der `daemon` Login-Klasse zugewiesen.
- Obwohl das mitgelieferte `/etc/login.conf` eine Quelle von vernünftigen Limits darstellt, können nur Sie, der Administrator, wissen, was für Ihr System angebracht ist. Ein Limit zu hoch anzusetzen könnte Ihr System für Missbrauch öffnen, und ein zu niedriges Limit der Produktivität einen Riegel vorschieben.
- Benutzer des Xorg sollten wahrscheinlich mehr Ressourcen zugeteilt bekommen als andere Benutzer. Xorg beansprucht selbst schon eine Menge Ressourcen, verleitet die Benutzer aber auch, mehrere Programme gleichzeitig laufen zu lassen.
- Bedenken Sie, dass viele Limits für einzelne Prozesse gelten und nicht für den Benutzer selbst. Setzt man zum Beispiel `openfiles` auf 50, kann jeder Prozess des Benutzers bis zu 50 Dateien öffnen. Dadurch ist die maximale Anzahl von Dateien, die von einem Benutzer geöffnet werden können, `openfiles` mal `maxproc`. Das gilt auch für den Speicherverbrauch.

Weitere Informationen über Ressourcenbeschränkungen, Login-Klassen und -Fähigkeiten finden Sie in cap_mkdb(1), getrlimit(2) und login.conf(5).

4.3.4. Gruppen

Eine Gruppe ist einfach eine Zusammenfassung von Accounts. Gruppen werden durch den Gruppennamen und die GID identifiziert. Der Kernel von FreeBSD entscheidet anhand der UID und der Gruppenmitgliedschaft eines Prozesses, ob er dem Prozess etwas erlaubt oder nicht. Wenn jemand von der GID eines Benutzers oder Prozesses spricht, meint er damit meistens die erste Gruppe der Gruppenliste.

Die Zuordnung von Gruppennamen zur GID steht in `/etc/group`, einer Textdatei mit vier durch Doppelpunkte getrennten Feldern. Im ersten Feld steht der Gruppenname, das zweite enthält ein verschlüsseltes Passwort, das dritte gibt die GID an und das vierte besteht aus einer Komma separierten Liste der Mitglieder der Gruppe. Eine ausführliche Beschreibung der Syntax dieser Datei finden Sie in group(5).

Wenn Sie `/etc/group` nicht händisch editieren möchten, können Sie pw(8) zum Editieren benutzen. Das folgende Beispiel zeigt das Hinzufügen einer Gruppe mit dem Namen `teamtwo`:

Beispiel 4.8. Setzen der Mitgliederliste einer Gruppe mit pw(8)

```
# pw groupadd teamtwo
# pw groupshow teamtwo
teamtwo:*:1100:
```

`1100` ist die GID der Gruppe `teamtwo`. Momentan hat `teamtwo` noch keine Mitglieder. Mit dem folgenden Kommando wird der Benutzer `jru` in die Gruppe `teamtwo` aufgenommen.

Beispiel 4.9. Ein Gruppenmitglied mit pw(8) hinzufügen

```
# pw groupmod teamtwo -M jru
# pw groupshow teamtwo
```

```
teamtwo:*:1100:jru
```

Als Argument von `-M` geben Sie eine Komma separierte Liste von Mitgliedern an, die in die Gruppe aufgenommen werden sollen. Aus den vorherigen Abschnitten ist bekannt, dass die Passwort-Datei ebenfalls eine Gruppe für jeden Benutzer enthält. Das System teilt dem Benutzer automatisch eine Gruppe zu, die aber vom `groupshow` Kommando von pw(8) nicht angezeigt wird. Diese Information wird allerdings von id(1) und ähnlichen Werkzeugen angezeigt. Das heißt, dass pw(8) nur `/etc/group` manipuliert, es wird nicht versuchen, zusätzliche Informationen aus `/etc/passwd` zu lesen.

Beispiel 4.10. Hinzufügen eines neuen Gruppenmitglieds mittels pw(8)

```
# pw groupmod teamtwo -m db
# pw groupshow teamtwo
teamtwo:*:1100:jru,db
```

Die Argumente zur Option `-m` ist eine durch Komma getrennte Liste von Benutzern, die der Gruppe hinzugefügt werden sollen. Anders als im vorherigen Beispiel werden diese Benutzer in die Gruppe aufgenommen und ersetzen nicht die bestehenden Benutzer in der Gruppe.

Beispiel 4.11. Mit `id` die Gruppenzugehörigkeit bestimmen

```
% id jru
uid=1001(jru) gid=1001(jru) groups=1001(jru), 1100(teamtwo)
```

In diesem Beispiel ist `jru` Mitglied von `jru` und `teamtwo`.

Weitere Informationen zu diesem Befehl und dem Format von `/etc/group` finden Sie in pw(8) und group(5).

4.4. Zugriffsrechte

In FreeBSD besitzt jede Datei und jedes Verzeichnis einen Satz von Zugriffsrechten. Es stehen mehrere Programme zum Anzeigen und Bearbeiten dieser Rechte zur Verfügung. Ein Verständnis für die Funktionsweise von Zugriffsrechten ist notwendig, um sicherzustellen, dass Benutzer nur auf die von ihnen benötigten Dateien zugreifen können und nicht auf die Dateien des Betriebssystems oder von anderen Benutzern.

In diesem Abschnitt werden die traditionellen Zugriffsrechte von UNIX® beschrieben. Informationen zu feingranularen Zugriffsrechten für Dateisysteme finden Sie im Abschnitt 14.11, „Zugriffskontrolllisten für Dateisysteme“.

In UNIX® werden die grundlegenden Zugriffsrechte in drei Typen unterteilt: Lesen, Scheiben und Ausführen. Diese Zugriffstypen werden verwendet, um den Dateizugriff für den Besitzer der Datei, die Gruppe und alle anderen zu bestimmen. Die Lese-, Schreib- und Ausführungsberechtigungen werden mit den Buchstaben `r`, `w` und `x` dargestellt. Alternativ können die Berechtigungen als binäre Zahlen dargestellt werden, da jede Berechtigung entweder aktiviert oder deaktiviert (`0`) ist. Wenn die Berechtigung als Zahl dargestellt wird, ist die Reihenfolge immer als `rwx` zu lesen, wobei `r` den Wert `4` hat, `w` den Wert `2` und `x` den Wert `1`.

In Tabelle 4.1 sind die einzelnen nummerischen und alphabetischen Möglichkeiten zusammengefasst. Das Zeichen `-` in der Spalte „Auflistung im Verzeichnis“ besagt, dass eine eine Berechtigung deaktiviert ist.

Tabelle 4.2. UNIX® Zugriffsrechte

Wert	Zugriffsrechte	Auflistung im Verzeichnis
0	Kein Lesen, Kein Schreiben, Kein Ausführen	`---`
1	Kein Lesen, Kein Schreiben, Ausführen	`--x`
2	Kein Lesen, Schreiben, Kein Ausführen	`-w-`
3	Kein Lesen, Schreiben, Ausführen	`-wx`
4	Lesen, Kein Schreiben, Kein Ausführen	`r--`
5	Lesen, Kein Schreiben, Ausführen	`r-x`
6	Lesen, Schreiben, Kein Ausführen	`rw-`
7	Lesen, Schreiben, Ausführen	`rwx`

Benutzen Sie das Argument `-l` mit ls(1), um eine ausführliche Verzeichnisauflistung zu sehen, die in einer Spalte die Zugriffsrechte für den Besitzer, die Gruppe und alle anderen enthält. Die Ausgabe von `ls -l` könnte wie folgt aussehen:

```
% ls -l
total 530
-rw-r--r--  1 root  wheel     512 Sep  5 12:31 myfile
-rw-r--r--  1 root  wheel     512 Sep  5 12:31 otherfile
-rw-r--r--  1 root  wheel    7680 Sep  5 12:31 email.txt
```

Das erste Zeichen (ganz links) der ersten Spalte zeigt an, ob es sich um eine normale Datei, ein Verzeichnis, ein zeichenorientiertes Gerät, ein Socket oder irgendeine andere Pseudo-Datei handelt. In diesem Beispiel zeigt `-` eine normale Datei an. Die nächsten drei Zeichen, dargestellt als `rw-`, ergeben die Rechte für den Datei-Besitzer. Die drei Zeichen danach `r--` die Rechte der Gruppe, zu der die Datei gehört. Die letzten drei Zeichen, `r--`, geben die Rechte für den Rest der Welt an. Ein Minus bedeutet, dass das Recht nicht gegeben ist. In diesem Beispiel sind die Zugriffsrechte also: der Eigentümer kann die Datei lesen und schreiben, die Gruppe kann lesen und alle anderen können auch nur lesen. Entsprechend obiger Tabelle wären die Zugriffsrechte für diese Datei `644`, worin jede Ziffer die drei Teile der Zugriffsrechte dieser Datei verkörpert.

Wie kontrolliert das System die Rechte von Hardware-Geräten? FreeBSD behandelt die meisten Hardware-Geräte als Dateien, welche Programme öffnen, lesen und mit Daten beschreiben können. Diese speziellen Gerätedateien sind in `/dev` gespeichert.

Verzeichnisse werden ebenfalls wie Dateien behandelt. Sie haben Lese-, Schreib- und Ausführ-Rechte. Das Ausführungs-Bit hat eine etwas andere Bedeutung für ein Verzeichnis als für eine Datei. Die Ausführbarkeit eines Verzeichnisses bedeutet, dass in das Verzeichnis, zum Beispiel mit cd(1), gewechselt werden kann. Das bedeutet auch, dass in dem Verzeichnis auf Dateien, deren Namen bekannt sind, zugegriffen werden kann, vorausgesetzt die Zugriffsrechte der Dateien lassen dies zu.

Das Leserecht auf einem Verzeichnis erlaubt es, sich den Inhalt des Verzeichnisses anzeigen zu lassen. Um eine Datei mit bekanntem Namen in einem Verzeichnis zu löschen, müssen auf dem Verzeichnis Schreib- *und* Ausführ-Rechte gesetzt sein.

Es gibt noch mehr Rechte, aber die werden vor allem in speziellen Umständen benutzt, wie zum Beispiel bei SetUID-Binaries und Verzeichnissen mit gesetztem Sticky-Bit. Mehr über Zugriffsrechte von Dateien und wie sie gesetzt werden, finden Sie in chmod(1).

4.4.1. Symbolische Zugriffsrechte

Beigesteuert von Tom Rhodes.

Symbolische Zugriffsrechte verwenden Zeichen anstelle von oktalen Werten, um die Berechtigungen für Dateien oder Verzeichnisse festzulegen. Zugriffsrechte verwenden die Syntax *`Wer`*, *`Aktion`* und *`Berechtigung`*. Die folgenden Werte stehen zur Auswahl:

Option	Symbol	Bedeutung
`Wer`	u	Benutzer (*user*)
`Wer`	g	Gruppe (*group*)
`Wer`	o	Andere (*other*)
`Wer`	a	Alle
`Aktion`	+	Berechtigungen hinzufügen
`Aktion`	-	Berechtigungen entziehen
`Aktion`	=	Berechtigungen explizit setzen
`Berechtigung`	r	lesen (*read*)
`Berechtigung`	w	schreiben (*write*)
`Berechtigung`	x	ausführen (*execute*)
`Berechtigung`	t	Sticky-Bit
`Berechtigung`	s	Set-UID oder Set-GID

Diese symbolischen Werte werden zusammen mit chmod(1) verwendet. Beispielsweise würde der folgende Befehl den Zugriff auf *`FILE`* für alle anderen Benutzer verbieten:

```
% chmod go= FILE
```

Wenn Sie mehr als eine Änderung der Rechte einer Datei vornehmen wollen, können Sie eine durch Kommata getrennte Liste der Rechte angeben. Das folgende Beispiel entzieht der Gruppe und der Welt die Schreibberechtigung auf *`FILE`* und fügt für jeden Ausführungsrechte hinzu:

```
% chmod go-w,a+x FILE
```

4.4.2. FreeBSD Datei-Flags

Beigetragen von Tom Rhodes.

Zusätzlich zu den Zugriffsrechten unterstützt FreeBSD auch die Nutzung von „Datei-Flags“. Diese erhöhen die Sicherheit des Systems, indem sie eine verbesserte Kontrolle von Dateien erlauben. Verzeichnisse werden allerdings nicht unterstützt. Mit dem Einsatz von Datei-Flags kann sogar `root` daran gehindert werden, Dateien zu löschen oder zu verändern.

Datei-Flags werden mit chflags(1) verändert. Um beispielsweise auf der Datei `file1` das „unlöschbar“-Flag zu aktivieren, geben Sie folgenden Befehl ein:

```
# chflags sunlink file1
```

Um dieses Flag zu deaktivieren, setzen Sie ein „no“ vor `sunlink`:

```
# chflags nosunlink file1
```

Um die Flags einer Datei anzuzeigen, verwenden Sie ls(1) zusammen mit `-lo`:

```
# ls -lo file1
```

```
-rw-r--r--  1 trhodes  trhodes  sunlnk 0 Mar  1 05:54 file1
```

Einige Datei-Flags können nur vom `root`-Benutzer gesetzt oder gelöscht werden. Andere wiederum können auch vom Eigentümer der Datei gesetzt werden. Weitere Informationen hierzu finden sich in chflags(1) und chflags(2).

4.4.3. Die Berechtigungen setuid, setgid, und sticky

Beigetragen von Tom Rhodes.

Anders als die Berechtigungen, die bereits angesprochen wurden, existieren drei weitere Einstellungen, über die alle Administratoren Bescheid wissen sollten. Dies sind die Berechtigungen `setuid`, `setgid` und `sticky`.

Diese Einstellungen sind wichtig für manche UNIX®-Operationen, da sie Funktionalitäten zur Verfügung stellen, die normalerweise nicht an gewöhnliche Anwender vergeben wird. Um diese zu verstehen, muss der Unterschied zwischen der realen und der effektiven Benutzer-ID erwähnt werden.

Die reale Benutzer-ID ist die UID, welche den Prozess besitzt oder gestartet hat. Die effektive UID ist diejenige, als die der Prozess läuft. Beispielsweise wird passwd(1) mit der realen ID des Benutzers ausgeführt, der sein Passwort ändert. Um jedoch die Passwortdatenbank zu bearbeiten, wird es effektiv als `root`-Benutzer ausgeführt. Das ermöglicht es normalen Benutzern, ihr Passwort zu ändern, ohne einen Permission Denied-Fehler angezeigt zu bekommen.

Die setuid-Berechtigung kann durch das Voranstellen bei einer Berechtigungsgruppe mit der Nummer Vier (4) gesetzt werden, wie im folgenden Beispiel gezeigt wird:

```
# chmod 4755 suidexample.sh
```

Die Berechtigungen auf *`suidexample.sh`* sehen jetzt wie folgt aus:

```
-rwsr-xr-x   1 trhodes  trhodes    63 Aug 29 06:36 suidexample.sh
```

Beachten Sie, dass ein `s` jetzt Teil der Berechtigungen des Dateibesitzers geworden ist, welches das Ausführen-Bit ersetzt. Dies ermöglicht es Werkzeugen mit erhöhten Berechtigungen zu laufen, wie z.B. `passwd`.

Anmerkung

Die `nosuid` mount(8)-Option bewirkt, dass solche Anwendungen stillschweigend scheitern, ohne den Anwender darüber zu informieren. Diese Option ist nicht völlig zuverlässig, da ein `nosuid`-Wrapper in der Lage wäre, dies zu umgehen.

Um dies in Echtzeit zu beobachten, öffnen Sie zwei Terminals. Starten Sie auf einem `passwd` als normaler Benutzer. Während es auf die Passworteingabe wartet, überprüfen Sie die Prozesstabelle und sehen Sie sich die Informationen für passwd(1) an:

Im Terminal A:

```
Changing local password for trhodes
Old Password:
```

Im Terminal B:

```
# ps aux | grep passwd
```

```
trhodes  5232  0.0  0.2  3420  1608   0  R+    2:10AM   0:00.00 grep passwd
root     5211  0.0  0.2  3620  1724   2  I+    2:09AM   0:00.01 passwd
```

Obwohl passwd(1) als normaler Benutzer ausgeführt wird, benutzt es die effektive UID von `root`.

Die `setgid`-Berechtigung führt die gleiche Aktion wie die `setuid`-Berechtigung durch, allerdings verändert sie die Gruppenberechtigungen. Wenn eine Anwendung oder ein Werkzeug mit dieser Berechtigung ausgeführt wird, erhält es die Berechtigungen basierend auf der Gruppe, welche die Datei besitzt und nicht die des Benutzers, der den Prozess gestartet hat.

Um die `setgid`-Berechtigung auf einer Datei zu setzen, geben Sie chmod(1) eine führende Zwei (2) mit:

```
# chmod 2755 sgidexample.sh
```

Beachten Sie in der folgenden Auflistung, dass das `s` sich jetzt in dem Feld befindet, das für die Berechtigungen der Gruppe bestimmt ist:

```
-rwxr-sr-x   1 trhodes  trhodes    44 Aug 31 01:49 sgidexample.sh
```

Anmerkung

Obwohl es sich bei dem in diesen Beispielen gezeigten Shellskript um eine ausführbare Datei handelt, wird es nicht mit einer anderen EUID oder effektiven Benutzer-ID ausgeführt. Das ist so, weil Shellskripte keinen Zugriff auf setuid(2)-Systemaufrufe erhalten.

Die `setuid` und `setgid` Berechtigungs-Bits können die Systemsicherheit verringern, da sie erhöhte Rechte ermöglichen. Das dritte Berechtigungs-Bit, das `sticky bit` kann die Sicherheit eines Systems erhöhen.

Wenn das `sticky bit` auf einem Verzeichnis angewendet wird, erlaubt es das Löschen von Dateien nur durch den Besitzer der Datei. Dies ist nützlich, um die Löschung von Dateien in öffentlichen Verzeichnissen wie `/tmp`, durch Benutzer denen diese Dateien nicht gehören, zu verhindern. Um diese Berechtigung anzuwenden, stellen Sie der Berechtigung eine Eins (1) voran:

```
# chmod 1777 /tmp
```

Das `sticky bit` kann anhand des `t` ganz am Ende der Berechtigungen abgelesen werden.

```
# ls -al / | grep tmp
```

```
drwxrwxrwt  10 root  wheel         512 Aug 31 01:49 tmp
```

4.5. Verzeichnis-Strukturen

Die FreeBSD-Verzeichnishierarchie ist die Grundlage, um ein umfassendes Verständnis des Systems zu erlangen. Das wichtigste Verzeichnis ist das Root-Verzeichnis „/“. Dieses Verzeichnis ist das erste, das während des Bootens eingehangen wird. Es enthält das notwendige Basissystem, um das Betriebssystem in den Mehrbenutzerbetrieb zu bringen. Das Root-Verzeichnis enthält auch die Mountpunkte für Dateisysteme, die beim Wechsel in den Multiuser-Modus eingehängt werden.

Ein Mountpunkt ist ein Verzeichnis, in das zusätzliche Dateisysteme (in der Regel unterhalb des Wurzelverzeichnisses) eingehängt werden können. Dieser Vorgang wird in Abschnitt 4.6, „Festplatten, Slices und Partitionen“ ausführlich beschrieben. Standard-Mountpunkte sind `/usr`, `/var`, `/tmp`, `/mnt` sowie `/cdrom`. Auf diese Verzeichnisse verweisen üblicherweise Einträge in `/etc/fstab` . Diese Datei ist eine Tabelle mit verschiedenen Dateisystemen und Mountpunkten, vom System gelesen werden. Die meisten der Dateisysteme in `/etc/fstab` werden beim Booten automatisch durch das Skript rc(8) gemountet, wenn die zugehörigen Einträge nicht mit `noauto` versehen sind. Weitere Informationen zu diesem Thema finden Sie im Abschnitt 4.7.1, „Die `fstab` Datei“.

Eine vollständige Beschreibung der Dateisystem-Hierarchie finden Sie in hier(7). Die folgende Aufstellung gibt einen kurzen Überblick über die am häufigsten verwendeten Verzeichnisse:

Verzeichnis	Beschreibung
`/`	Wurzelverzeichnis des Dateisystems.
`/bin/`	Grundlegende Werkzeuge für den Single-User-Modus sowie den Mehrbenutzerbetrieb.
`/boot/`	Programme und Konfigurationsdateien, die während des Bootens benutzt werden.

Verzeichnis	Beschreibung
`/boot/defaults/`	Vorgaben für die Boot-Konfiguration. Weitere Details finden Sie in loader.conf(5).
`/dev/`	Gerätedateien. Weitere Details finden Sie in intro(4).
`/etc/`	Konfigurationsdateien und Skripten des Systems.
`/etc/defaults/`	Vorgaben für die System Konfigurationsdateien. Weitere Details finden Sie in rc(8).
`/etc/mail/`	Konfigurationsdateien von MTAs wie sendmail(8).
`/etc/namedb/`	Konfigurationsdateien von named(8).
`/etc/periodic/`	Täglich, wöchentlich oder monatlich laufende Skripte, die von cron(8) gestartet werden. Weitere Details finden Sie in periodic(8).
`/etc/ppp/`	Konfigurationsdateien von ppp(8).
`/mnt/`	Ein leeres Verzeichnis, das von Systemadministratoren häufig als temporärer Mountpunkt genutzt wird.
`/proc/`	Prozess Dateisystem. Weitere Details finden Sie in procfs(5) und mount_procfs(8).
`/rescue/`	Statisch gelinkte Programme zur Wiederherstellung des Systems, wie in rescue(8) beschrieben.
`/root/`	Home Verzeichnis von `root`.
`/sbin/`	Systemprogramme und administrative Werkzeuge, die grundlegend für den Single-User-Modus und den Mehrbenutzerbetrieb sind.
`/tmp/`	Temporäre Dateien, die für gewöhnlich bei einem Neustart des Systems verloren gehen. Häufig wird ein speicherbasiertes Dateisystem unter `/tmp` eingehängt. Dieser Vorgang kann automatisiert werden, wenn tmpmfs-bezogene Variablen von rc.conf(5) verwendet werden, oder ein entsprechender Eintrag in `/etc/fstab` existiert. Weitere Informationen finden Sie in mdmfs(8).
`/usr/`	Der Großteil der Benutzerprogramme und Anwendungen.
`/usr/bin/`	Gebräuchliche Werkzeuge, Programmierhilfen und Anwendungen.
`/usr/include/`	Standard C include-Dateien.
`/usr/lib/`	Bibliotheken.
`/usr/libdata/`	Daten verschiedener Werkzeuge.
`/usr/libexec/`	System-Dämonen und System-Werkzeuge, die von anderen Programmen ausgeführt werden.
`/usr/local/`	Lokale Programme und Bibliotheken. Die Ports-Sammlung von FreeBSD benutzt dieses Verzeichnis als Zielverzeichnis für Anwendungen. Innerhalb von `/usr/local` sollte das von hier(7) beschriebene Layout für `/usr` benutzt werden. Das `man` Verzeichnis wird direkt unter `/usr/local` anstelle unter `/usr/local/share` angelegt. Die Dokumentation der Ports findet sich in `share/doc/`*`port`*`.`

Verzeichnis	Beschreibung
`/usr/obj/`	Von der Architektur abhängiger Verzeichnisbaum, der durch das Bauen von `/usr/src` entsteht.
`/usr/ports/`	Die FreeBSD-Ports-Sammlung (optional).
`/usr/sbin/`	System-Dämonen und System-Werkzeuge, die von Benutzern ausgeführt werden.
`/usr/share/`	Von der Architektur unabhängige Dateien.
`/usr/src/`	Quelldateien von BSD und/oder lokalen Ergänzungen.
`/var/`	Wird für mehrere Zwecke genutzt und enthält Logdateien, temporäre Daten und Spooldateien. Manchmal wird ein speicherbasiertes Dateisystem unter `/var` eingehängt. Dieser Vorgang kann automatisiert werden, wenn die varmfs-bezogenen Variablen von rc.conf(5) verwendet werden, oder ein entsprechender Eintrag in `/etc/fstab` existiert. Weitere Informationen finden Sie in mdmfs(8).
`/var/log/`	Verschiedene Logdateien des Systems.
`/var/mail/`	Postfächer der Benutzer.
`/var/spool/`	Verschiedene Spool-Verzeichnisse der Drucker- und Mailsysteme.
`/var/tmp/`	Temporäre Dateien, die in der Regel auch bei einem Neustart des Systems erhalten bleiben, es sei denn, bei `/var` handelt es sich um ein speicherbasiertes Dateisystem.
`/var/yp/`	NIS maps.

4.6. Festplatten, Slices und Partitionen

FreeBSD identifiziert Dateien anhand eines Dateinamens. In Dateinamen wird zwischen Groß- und Kleinschreibung unterschieden: `readme.txt` und `README.TXT` bezeichnen daher zwei verschiedene Dateien. FreeBSD benutzt keine Dateiendungen, um den Typ der Datei zu bestimmen, egal ob es sich um ein Programm, ein Dokument oder um andere Daten handelt.

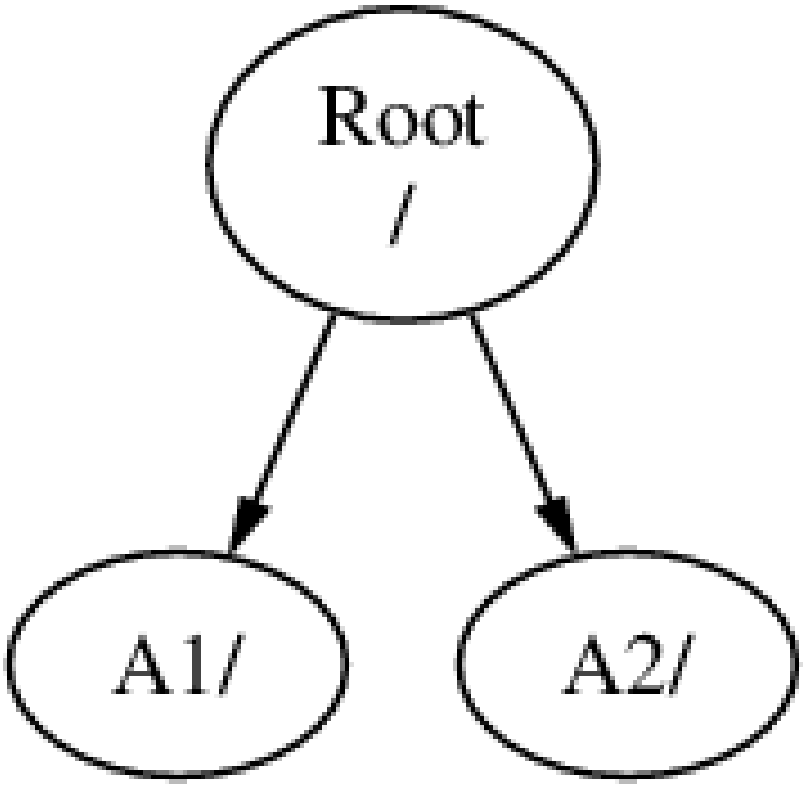
Dateien werden in Verzeichnissen gespeichert. In einem Verzeichnis können sich keine oder hunderte Dateien befinden. Ein Verzeichnis kann auch andere Verzeichnisse enthalten und so eine Hierarchie von Verzeichnissen aufbauen, die die Ablage von Daten erleichtert.

In Dateinamen werden Verzeichnisse durch einen Schrägstrich (`/`, *Slash*) getrennt. Wenn z.B. das Verzeichnis `foo` ein Verzeichnis `bar` enthält, in dem sich die Datei `readme.txt` befindet, lautet der vollständige Name der Datei (oder der *Pfad* zur Datei) `foo/bar/readme.txt` . Beachten Sie, dass sich dies von Windows® unterscheidet, wo der `\` (*Backslash* für die Trennung von Datei- und Verzeichnisnamen verwendet wird. FreeBSD benutzt keine Laufwerkbuchstaben oder Laufwerknamen im Pfad. Beispielsweise würde man unter FreeBSD nicht `c:\foo\bar\readme.txt` eingeben.

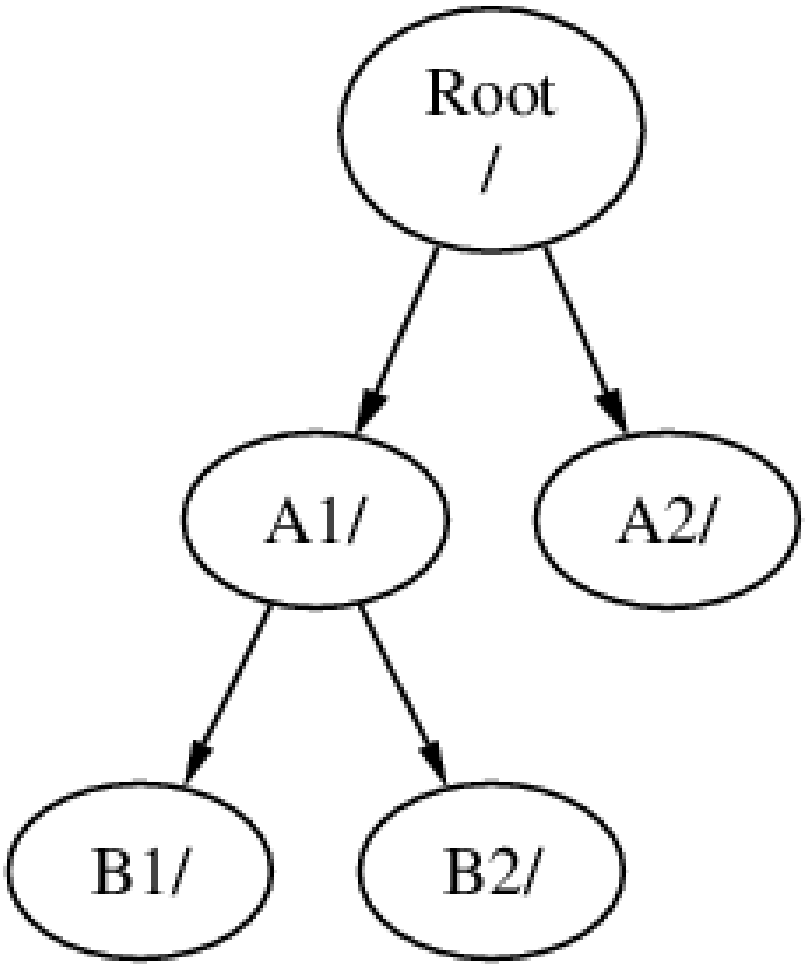
Verzeichnisse und Dateien werden in einem Dateisystem gespeichert. Jedes Dateisystem besitzt genau ein *Wurzelverzeichnis*, das so genannte *Root-Directory*. Dieses Wurzelverzeichnis kann weitere Verzeichnisse enthalten. Ein Dateisystem wird als Wurzeldateisystem festgelegt, und jedes weitere Dateisystem wird unter dem Wurzeldateisystem *eingehangen*. Daher scheint jedes Verzeichnis, unabhängig von der Anzahl der Platten, auf der selben Platte zu liegen.

Betrachten wir die drei Dateisysteme `A`, `B` und `C`. Jedes Dateisystem besitzt ein eigenes Wurzelverzeichnis, das zwei andere Verzeichnisse enthält: `A1`, `A2`, `B1`, `B2`, `C1` und `C2`.

Das Wurzeldateisystem soll `A` sein. ls(1) zeigt darin die beiden Verzeichnisse `A1` und `A2` an. Der Verzeichnisbaum sieht wie folgt aus:

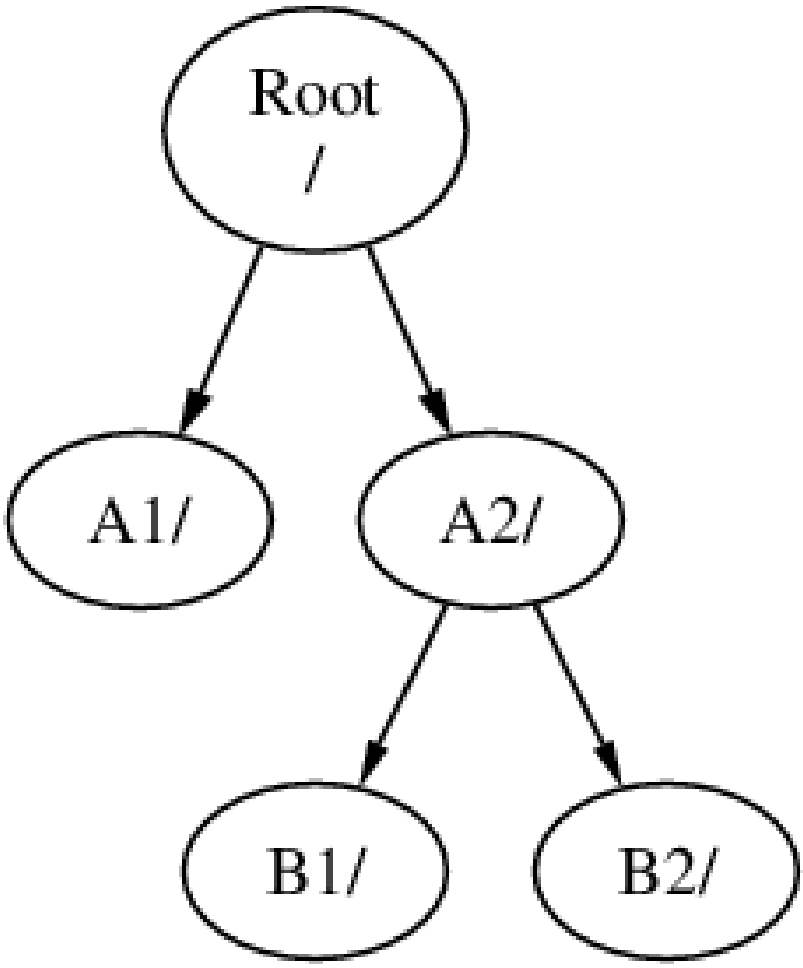


Ein Dateisystem wird in einem Verzeichnis eines anderen Dateisystems eingehangen. Wir hängen nun das Dateisystem `B` in das Verzeichnis `A1` ein. Das Wurzelverzeichnis von `B` ersetzt nun das Verzeichnis `A1` und die Verzeichnisse des Dateisystems `B` werden sichtbar:



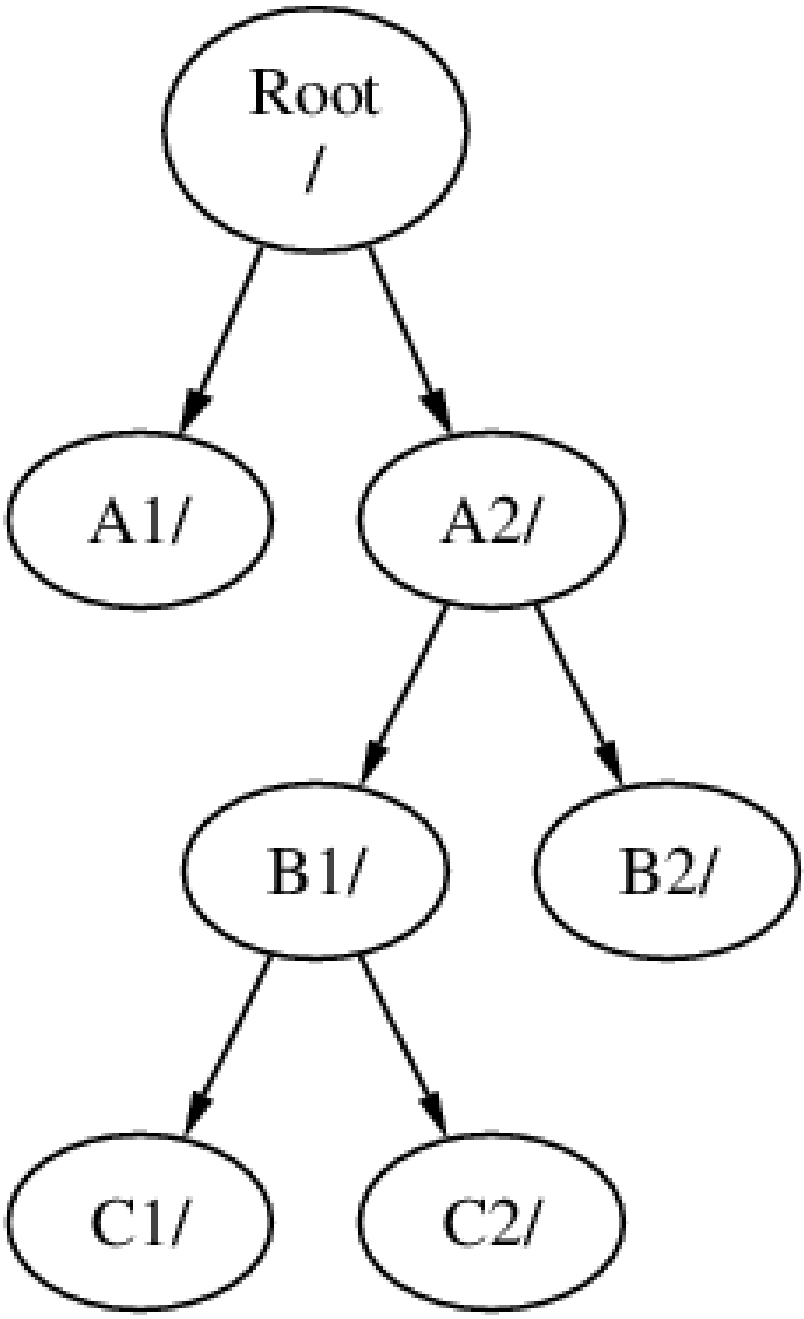
Jede Datei in den Verzeichnissen `B1` oder `B2` kann über den Pfad `/A1/B1` oder `/A1/B2` erreicht werden. Dateien aus dem Verzeichnis `/A1` sind jetzt verborgen. Wenn das Dateisystem `B` wieder *abgehangen* wird (*umount*), erscheinen die verborgenen Dateien wieder.

Wenn das Dateisystem `B` unter dem Verzeichnis `A2` eingehangen würde, sähe der Verzeichnisbaum so aus:

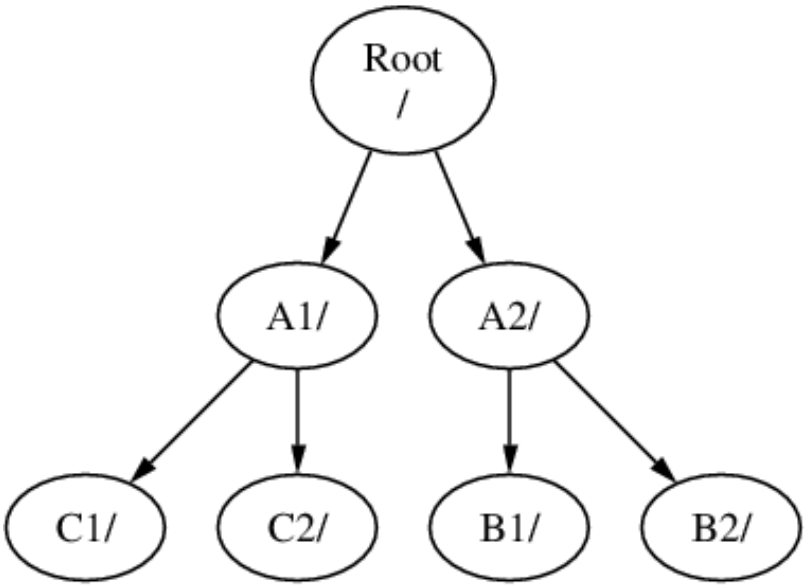


Die Dateien des Dateisystems `B` wären unter den Pfaden `/A2/B1` und `/A2/B2` erreichbar.

Dateisysteme können übereinander eingehangen werden. Der folgende Baum entsteht, wenn im letzten Beispiel das Dateisystem `C` in das Verzeichnis `B1` des Dateisystems `B` eingehangen wird:



`C` könnte auch im Verzeichnis `A1` eingehangen werden:



Sie können sogar mit nur einem großen Dateisystem auskommen. Dies hat mehrere Nachteile und einen Vorteil.

- Die Dateisysteme können mit unterschiedlichen Optionen (*mount options*) eingehangen werden. Beispielsweise kann das Wurzeldateisystem schreibgeschützt eingehangen werden, sodass es für Benutzer nicht möglich ist, versehentlich kritische Dateien zu editieren oder zu löschen. Von Benutzern beschreibbare Dateisysteme wie `/home` können mit der Option *nosuid* eingehangen werden, wenn sie von anderen Dateisystemen getrennt sind. Die *SUID*- und *GUID*-Bits verlieren auf solchen Dateisystemen ihre Wirkung und die Sicherheit des Systems kann dadurch erhöht werden.

- Die Lage von Dateien im Dateisystem wird, abhängig vom Gebrauch des Dateisystems, automatisch von FreeBSD optimiert. Ein Dateisystem mit vielen kleinen Dateien, die häufig geschrieben werden, wird anders behandelt als ein Dateisystem mit wenigen großen Dateien. Mit nur einem Dateisystem ist diese Optimierung unmöglich.

- In der Regel übersteht ein FreeBSD-Dateisystem auch einen Stromausfall. Allerdings kann ein Stromausfall zu einem kritischen Zeitpunkt das Dateisystem beschädigen. Wenn die Daten über mehrere Dateisysteme verteilt sind, lässt sich das System mit hoher Wahrscheinlichkeit noch starten. Dies erleichtert das Zurückspielen von Datensicherungen.

- Dateisysteme haben eine festgelegte Größe. Es kann passieren, dass Sie eine Partition vergrößern müssen. Dies ist nicht leicht: Sie müssen die Daten sichern, das Dateisystem vergrößert anlegen und die gesicherten Daten zurückspielen.

Wichtig

FreeBSD kennt den Befehl growfs(8), mit dem man Dateisysteme im laufenden Betrieb vergrößern kann.

Dateisysteme befinden sich in Partitionen (damit sind nicht die normalen MS-DOS®-Partitionen gemeint). Jede Partition wird mit einem Buchstaben von `a` bis `h` bezeichnet und kann nur ein Dateisystem enthalten. Dateisysteme können daher über ihren Mount-Point, den Punkt an dem sie eingehangen sind, oder den Buchstaben der Partition, in der sie liegen, identifiziert werden.

FreeBSD benutzt einen Teil der Platte für den *Swap-Bereich*, um *virtuellen Speicher* zur Verfügung zu stellen. Dadurch kann der Rechner Anwendungen mehr Speicher zur Verfügung stellen als tatsächlich eingebaut ist. Wenn der Speicher knapp wird, kann FreeBSD nicht benutzte Daten in den Swap-Bereich auslagern. Die ausgelagerten Daten können später wieder in den Speicher geholt werden (dafür werden dann andere Daten ausgelagert).

Für einige Partitionen gelten besondere Konventionen:

Partition	Konvention
a	Enthält normalerweise das Wurzeldateisystem.
b	Enthält normalerweise den Swap-Bereich.
c	Ist normalerweise genauso groß wie die Slice in der die Partition liegt. Werkzeuge, die auf der kompletten Slice arbeiten, wie ein Bad-Block-Scanner, können so die c-Partition benutzen. Für gewöhnlich wird in dieser Partition kein Dateisystem angelegt.
d	Früher hatte die d-Partition eine besondere Bedeutung. Heute ist dies nicht mehr der Fall und die Partition d kann wie jede andere Partition auch verwendet werden.

In FreeBSD werden Festplatten in Slices, welche in Windows® als Partitionen bekannt sind, aufgeteilt und von 1 bis 4 durchnummeriert. Diese werden dann in Partitionen unterteilt, welche wiederum Dateisysteme enthalten und mit Buchstaben benannt werden.

Die Slice-Nummern werden mit vorgestelltem s hinter den Gerätenamen gestellt: „da0*s1*“ ist die erste Slice auf dem ersten SCSI-Laufwerk. Auf einer Festplatte gibt es höchstens vier Slices. In einer Slice des passenden Typs kann es weitere logische Slices geben. Diese erweiterten Slices werden ab fünf durchnummeriert: „ad0*s5*“ ist die erste erweiterte Slice auf einer IDE-Platte. Diese Geräte werden von Dateisystemen benutzt, die sich in einer kompletten Slice befinden müssen.

Slices, „dangerously dedicated“-Festplatten und andere Platten enthalten Partitionen, die mit Buchstaben von a bis h bezeichnet werden. Der Buchstabe wird an den Gerätenamen gehangen: „da0*a*“ ist die a-Partition des ersten da-Laufwerks. Dieses Laufwerk ist „dangerously dedicated“. „ad1s3*e*“ ist die fünfte Partition in der dritten Slice der zweiten IDE-Platte.

Schließlich wird noch jede Festplatte des Systems eindeutig bezeichnet. Der Name einer Festplatte beginnt mit einem Code, der den Typ der Platte bezeichnet. Es folgt eine Nummer, die angibt, um welche Festplatte es sich handelt. Anders als bei Slices werden Festplatten von Null beginnend durchnummeriert. Gängige Festplatten-Namen sind in Tabelle 4.3, „Laufwerk-Codes“ aufgeführt.

Wenn Sie eine Partition angeben, beinhaltet das den Plattennamen, s, die Slice-Nummer und den Buchstaben der Partition. Einige Beispiele finden Sie in Beispiel 4.12, „Namen von Platten, Slices und Partitionen“.

Der Aufbau einer Festplatte wird in Beispiel 4.13, „Aufteilung einer Festplatte“ dargestellt.

Bei der Installation von FreeBSD legen Sie Slices auf der Festplatte an, erstellen Partitionen für FreeBSD innerhalb der Slice, erstellen ein Dateisystem oder Auslagerungsbereiche und entscheiden, welche Dateisysteme wo eingehangen werden.

Tabelle 4.3. Laufwerk-Codes

Code	Bedeutung
ad	ATAPI (IDE) Festplatte
da	SCSI-Festplatte
acd	ATAPI (IDE) CD-ROM
cd	SCSI-CD-ROM
fd	Disketten-Laufwerk

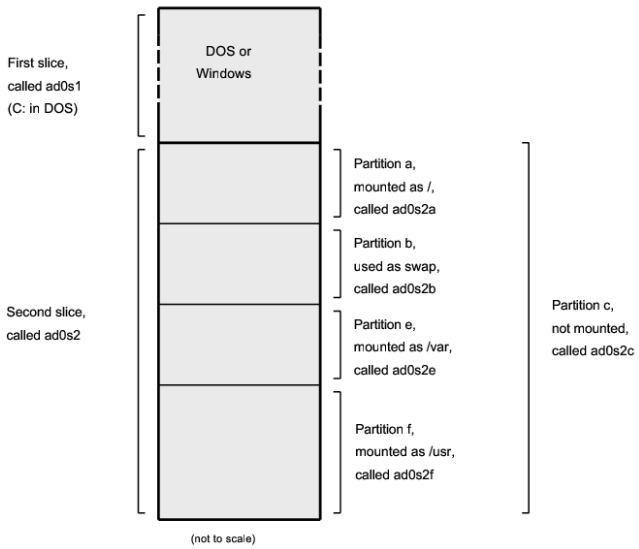
Beispiel 4.12. Namen von Platten, Slices und Partitionen

Name	Bedeutung
`ad0s1a`	Die erste Partition (`a`) in der ersten Slice (`s1`) der ersten IDE-Festplatte (`ad0`).
`da1s2e`	Die fünfte Partition (`e`) der zweiten Slice (`s2`) auf der zweiten SCSI-Festplatte (`da1`).

Beispiel 4.13. Aufteilung einer Festplatte

Das folgende Diagramm zeigt die Sicht von FreeBSD auf die erste IDE-Festplatte des Systems. Die Platte soll 4 GB groß sein und zwei Slices (MS-DOS®-Partitionen) mit je 2 GB besitzen. Die erste Slice enthält ein MS-DOS®-Laufwerk (`C:`), die zweite Slice wird von FreeBSD benutzt. Die FreeBSD-Installation in diesem Beispiel verwendet drei Datenpartitionen und einen Auslagerungsbereich.

Jede der drei Partitionen enthält ein Dateisystem. Das Wurzeldateisystem ist die `a`-Partition. In der `e`-Partition befindet sich der `/var`-Verzeichnisbaum und in der `f`-Partition befindet sich der Verzeichnisbaum unterhalb von `/usr`.



4.7. Anhängen und Abhängen von Dateisystemen

Ein Dateisystem wird am besten als ein Baum mit der Wurzel `/` veranschaulicht. `/dev`, `/usr`, und die anderen Verzeichnisse im Rootverzeichnis sind Zweige, die wiederum eigene Zweige wie `/usr/local` haben können.

Es gibt verschiedene Gründe, bestimmte dieser Verzeichnisse auf eigenen Dateisystemen anzulegen. `/var` enthält `log/`, `spool/` sowie verschiedene andere temporäre Dateien und kann sich daher schnell füllen. Es empfiehlt sich, `/var` von `/` zu trennen, da es schlecht ist, wenn das Root-Dateisystem voll läuft.

Ein weiterer Grund bestimmte Verzeichnisbäume auf andere Dateisysteme zu legen, ist gegeben, wenn sich die Verzeichnisbäume auf gesonderten physikalischen oder virtuellen Platten, wie Network File System oder CD-ROM-Laufwerken, befinden.

4.7.1. Die `fstab` Datei

Während des Boot-Prozesses (Kapitel 13, *FreeBSDs Bootvorgang*) werden in `/etc/fstab` aufgeführte Verzeichnisse, sofern sie nicht mit der Option `noauto` versehen sind, automatisch angehangen. Diese Datei enthält Einträge in folgendem Format:

```
device /mount-point  fstype options  dumpfreq  passno
```

`device`
: Ein existierender Gerätename wie in Abschnitt 18.2, „Gerätenamen“ beschrieben.

`mount-point`
: Ein existierendes Verzeichnis, auf dem das Dateisystem gemountet wird.

`fstype`
: Der Typ des Dateisystems, der an mount(8) weitergegeben wird. FreeBSDs Standarddateisystem ist `ufs`.

`options`
: Entweder `rw` für beschreibbare Dateisysteme oder `ro` für schreibgeschützte Dateisysteme, gefolgt von weiteren benötigten Optionen. Eine häufig verwendete Option ist `noauto` für Dateisysteme, die während der normalen Bootsequenz nicht angehangen werden sollen. Weitere Optionen finden sich in mount(8).

`dumpfreq`
: Wird von dump(8) benutzt, um bestimmen zu können, welche Dateisysteme gesichert werden müssen. Fehlt der Wert, wird `0` angenommen.

`passno`
: Bestimmt die Reihenfolge, in der die Dateisysteme überprüft werden sollen. Für Dateisysteme, die übersprungen werden sollen, ist `passno` auf `0` zu setzen. Für das Root-Dateisystem, das vor allen anderen überprüft werden muss, sollte der Wert von `passno` `1` betragen. Allen anderen Dateisystemen sollten Werte größer `1` zugewiesen werden. Wenn mehrere Dateisysteme den gleichen Wert besitzen, wird fsck(8) versuchen, diese parallel zu überprüfen.

Lesen Sie fstab(5) für weitere Informationen über das Format von `/etc/fstab` und dessen Optionen.

4.7.2. Verwendung von mount(8)

Dateisysteme werden mit mount(8) eingehängt. In der grundlegenden Form wird es wie folgt benutzt:

```
# mount device
mountpoint
```

Dieser Befehl bietet viele Optionen, die in mount(8) beschrieben werden. Die am häufigsten verwendeten Optionen sind:

`-a`
: Hängt alle Dateisysteme aus `/etc/fstab` an. Davon ausgenommen sind Dateisysteme, die mit „noauto“ markiert sind, die mit der Option `-t` ausgeschlossen wurden und Dateisysteme, die schon angehangen sind.

`-d`
: Führt alles bis auf den `mount`-Systemaufruf aus. Nützlich ist diese Option in Verbindung mit `-v`. Damit wird angezeigt, was mount(8) tatsächlich versuchen würde, um das Dateisystem anzuhängen.

`-f`
: Erzwingt das Anhängen eines unsauberen Dateisystems (riskant) oder die Rücknahme des Schreibzugriffs, wenn der Status des Dateisystems von beschreibbar auf schreibgeschützt geändert wird.

`-r`
: Hängt das Dateisystem schreibgeschützt ein. Dies kann auch durch Angabe von `-o ro` erreicht werden.

`-t` *`fstype`*
: Hängt das Dateisystem mit dem angegebenen Typ an, oder hängt nur Dateisysteme mit dem angegebenen Typ an, wenn `-a` angegeben wurde. „ufs“ ist das Standarddateisystem.

`-u`
: Aktualisiert die Mountoptionen des Dateisystems.

`-v`
: Geschwätzig sein.

`-w`
: Hängt das Dateisystem beschreibbar an.

Die folgenden Optionen können durch eine Kommata separierte Liste an `-o` übergeben werden:

nosuid
: SetUID und SetGID Bits werden auf dem Dateisystem nicht beachtet. Dies ist eine nützliche Sicherheitsfunktion.

4.7.3. Verwendung von umount(8)

umount(8) hängt ein Dateisysstem ab. Dieser Befehl akzeptiert als Parameter entweder einen Mountpoint, einen Gerätenamen, `-a` oder `-A`.

Jede Form akzeptiert `-f`, um das Abhängen zu erzwingen, und `-v`, um etwas geschwätziger zu sein. Seien Sie bitte vorsichtig mit `-f`, da der Computer abstürzen kann oder es können Daten auf dem Dateisystem beschädigt werden.

Um alle Dateisysteme abzuhängen, oder nur diejenigen, die mit `-t` gelistet werden, wird `-a` oder `-A` benutzt. Beachten Sie, dass `-a` das Root-Dateisystem nicht aushängt.

4.8. Prozesse und Dämonen

FreeBSD ist ein Multitasking-Betriebssystem. Jedes Programm, das zu irgendeiner Zeit läuft wird als *Prozess* bezeichnet. Jedes laufende Kommando startet mindestens einen neuen Prozess. Dazu gibt es eine Reihe von Systemprozessen, die von FreeBSD ausgeführt werden.

Jeder Prozess wird durch eine eindeutige Nummer identifiziert, die *Prozess-ID* (*PID*) genannt wird. Prozesse haben ebenso wie Dateien einen Besitzer und eine Gruppe, die festlegen, welche Dateien und Geräte der Prozess benutzen kann. Die meisten Prozesse haben auch einen Elternprozess, der sie gestartet hat. Beispielsweise ist die Shell ein Prozess. Jedes in Shell gestartete Kommando ist dann ein neuer Prozess, der die Shell als Elternprozess besitzt. Die Ausnahme hiervon ist ein spezieller Prozess namens init(8), der beim booten immer als erstes gestartet wird und der immer die PID `1` hat.

Manche Programme erwarten keine Eingaben vom Benutzer und lösen sich bei erster Gelegenheit von ihrem Terminal. Ein Webserver zum Beispiel antwortet auf Web-Anfragen und nicht auf Benutzereingaben. Mail-Server sind ein weiteres Beispiel für diesen Typ von Anwendungen. Diese Programme sind als *Dämonen* bekannt. Der Begriff Dämon stammt aus der griechischen Mythologie und bezeichnet ein Wesen, das weder gut noch böse ist und welches unsichtbar nützliche Aufgaben verrichtet. Deshalb ist das BSD Maskottchen dieser fröhlich aussehende Dämon mit Turnschuhen und Dreizack.

Programme, die als Dämon laufen, werden entsprechend einer Konvention mit einem „d“ am Ende benannt. BIND steht beispielsweise für Berkeley Internet Name Domain, das tatsächlich laufende Programm heißt aber `named`. Der Apache Webserver wird `httpd` genannt und der Druckerspool-Dämon heißt lpd(8). Dies ist allerdings nur eine Konvention. Der Dämon der Anwendung Sendmail heißt beispielsweise `sendmail` und nicht `maild`.

4.8.1. Prozesse beobachten

Um die Prozesse auf dem System zu sehen, benutzen Sie ps(1) und top(1). Eine statische Liste der laufenden Prozesse, deren PIDs, Speicherverbrauch und die Kommandozeile, mit der sie gestartet wurden, erhalten Sie mit ps(1). Um alle laufenden Prozesse in einer Anzeige zu sehen, die alle paar Sekunden aktualisiert wird, so dass Sie interaktiv sehen können was der Computer macht, benutzen Sie top(1).

In der Voreinstellung zeigt ps(1) nur die laufenden Prozesse, die dem Benutzer gehören. Zum Beispiel:

```
% ps
  PID TT  STAT    TIME COMMAND
 8203  0  Ss   0:00.59 /bin/csh
 8895  0  R+   0:00.00 ps
```

Die Ausgabe von ps(1) ist in einer Anzahl von Spalten organisiert. Die `PID` Spalte zeigt die Prozess-ID. PIDs werden von 1 beginnend bis 99999 zugewiesen und fangen wieder von vorne an. Ist eine PID bereits vergeben, wird diese allerdings nicht erneut vergeben. Die Spalte `TT` zeigt den Terminal, auf dem das Programm läuft. `STAT` zeigt den Status des Programms und `TIME` gibt die Zeit an, die das Programm auf der CPU gelaufen ist. Dies ist nicht unbedingt die Zeit, die seit dem Start des Programms vergangen ist, da die meisten Programme hauptsächlich auf bestimmte Dinge warten, bevor sie wirklich CPU-Zeit verbrauchen. Unter der Spalte `COMMAND` findet sich schließlich die Kommandozeile, mit der das Programm gestartet wurde.

ps(1) besitzt viele Optionen, um die angezeigten Informationen zu beeinflussen. Eine nützliche Kombination ist `auxww`. `a` zeigt Information über alle laufenden Prozesse aller Benutzer. Der Name des Besitzers des Prozesses, sowie Informationen über den Speicherverbrauch werden mit `u` angezeigt. `x` zeigt auch Dämonen-Prozesse an, und `ww` veranlasst ps(1) die komplette Kommandozeile für jeden Befehl anzuzeigen, anstatt sie abzuschneiden, wenn sie zu lang für die Bildschirmausgabe wird.

Die Ausgabe von top(1) sieht ähnlich aus:

```
% top
last pid:  9609;  load averages:  0.56,  0.45,  0.36              up 0+00:20:03  10:21:46
107 processes: 2 running, 104 sleeping, 1 zombie
CPU:  6.2% user,  0.1% nice,  8.2% system,  0.4% interrupt, 85.1% idle
Mem: 541M Active, 450M Inact, 1333M Wired, 4064K Cache, 1498M Free
ARC: 992M Total, 377M MFU, 589M MRU, 250K Anon, 5280K Header, 21M Other
Swap: 2048M Total, 2048M Free

  PID USERNAME    THR PRI NICE   SIZE    RES STATE   C   TIME   WCPU COMMAND
  557 root          1 -21  r31   136M 42296K select  0   2:20  9.96% Xorg
 8198 dru           2  52    0   449M 82736K select  3   0:08  5.96% kdeinit4
 8311 dru          27  30    0  1150M   187M uwait   1   1:37  0.98% firefox
  431 root          1  20    0 14268K  1728K select  0   0:06  0.98% moused
 9551 dru           1  21    0 16600K  2660K CPU3    3   0:01  0.98% top
 2357 dru           4  37    0   718M   141M select  0   0:21  0.00% kdeinit4
 8705 dru           4  35    0   480M    98M select  2   0:20  0.00% kdeinit4
 8076 dru           6  20    0   552M   113M uwait   0   0:12  0.00% soffice.bin
 2623 root          1  30   10 12088K  1636K select  3   0:09  0.00% powerd
 2338 dru           1  20    0   440M 84532K select  1   0:06  0.00% kwin
 1427 dru           5  22    0   605M 86412K select  1   0:05  0.00% kdeinit4
```

Die Ausgabe ist in zwei Abschnitte geteilt. In den ersten fünf Kopfzeilen finden sich die zuletzt zugeteilte PID, die Systemauslastung (engl. *load average*), die Systemlaufzeit (die Zeit seit dem letzten Reboot) und die momentane Zeit. Die weiteren Zahlen im Kopf beschreiben wie viele Prozesse momentan laufen, wie viel Speicher und Swap verbraucht wurde und wie viel Zeit das System in den verschiedenen CPU-Modi verbringt. Wenn das ZFS-Kernelmodul geladen ist, dann zeigt die Zeile `ARC`, wie viele Daten aus dem Cache gelesen wurden.

Darunter befinden sich einige Spalten mit ähnlichen Informationen wie in der Ausgabe von ps(1), beispielsweise die PID, den Besitzer, die verbrauchte CPU-Zeit und das Kommando, das den Prozess gestartet hat. top(1) zeigt in zwei Spalten den Speicherverbrauch des Prozesses an. Die erste Spalte gibt den gesamten Speicherverbrauch des Prozesses an, in der zweiten Spalte wird der aktuelle Verbrauch angegeben.

Die Anzeige wird von top(1) automatisch alle zwei Sekunden aktualisiert. Ein anderer Intervall kann mit `-s` spezifiziert werden.

4.8.2. Stoppen von Prozessen

Eine Möglichkeit mit einem laufenden Prozess zu kommunizieren, ist über das Versenden von *Signalen* mittels kill(1). Es gibt eine Reihe von verschiedenen Signalen. Manche haben eine feste Bedeutung, während andere in der Dokumentation der Anwendung beschrieben sind. Ein Benutzer kann ein Signal nur an einen Prozess senden, welcher ihm gehört. Wird versucht ein Signal an einen Prozess eines anderen Benutzers zu senden, resultiert dies in einem Zugriffsfehler mangels fehlender Berechtigungen. Die Ausnahme ist der `root`-Benutzer, welcher jedem Prozess Signale senden kann.

FreeBSD kann auch ein Signal an einen Prozess senden. Wenn eine Anwendung schlecht geschrieben ist und auf Speicher zugreift, auf den sie nicht zugreifen soll, so sendet FreeBSD dem Prozess das *Segmentation Violation* Signal (`SIGSEGV`). Wenn eine Anwendung programmiert wurde, den alarm(3) Systemaufruf zu benutzen, um nach einiger Zeit benachrichtigt zu werden, bekommt sie das „Alarm“-Signal (`SIGALRM`) gesendet.

Zwei Signale können benutzt werden, um einen Prozess zu stoppen: `SIGTERM` und `SIGKILL`. `SIGTERM` fordert den Prozess höflich zum Beenden auf. Der Prozess kann das Signal abfangen und hat dann Gelegenheit Logdateien zu schließen und die Aktion, die er durchführte, abzuschließen. In manchen Situationen kann der Prozess `SIGTERM` ignorieren, wenn er eine Aktion durchführt, die nicht unterbrochen werden darf.

`SIGKILL` kann von keinem Prozess ignoriert werden. Wird einem Prozess `SIGKILL` geschickt, dann wird FreeBSD diesen sofort beenden[3].

Andere häufig verwendete Signale sind `SIGHUP`, `SIGUSR1` und `SIGUSR2`. Da diese Signale für allgemeine Zwecke vorgesehen sind, werden verschiedene Anwendungen unterschiedlich auf diese Signale reagieren.

Ändern Sie beispielsweise die Konfiguration eines Webservers, so muss dieser angewiesen werden, seine Konfiguration neu zu lesen. Ein Neustart von `httpd` würde dazu führen, dass der Server für kurze Zeit nicht erreichbar ist. Senden Sie dem Dämon stattdessen das `SIGHUP`-Signal. Es sei erwähnt, dass verschiedene Dämonen sich anders verhalten. Lesen Sie bitte die Dokumentation des entsprechenden Dämonen um zu überprüfen, ob der Dämon bei einem `SIGHUP` die gewünschten Ergebnisse erzielt.

Prozedur 4.1. Verschicken von Signalen

Das folgende Beispiel zeigt, wie Sie inetd(8) ein Signal schicken. Die Konfigurationsdatei von inetd(8) ist `/etc/inetd.conf`. Diese Konfigurationsdatei liest inetd(8) ein, wenn er `SIGHUP` empfängt.

1. Suchen Sie mit pgrep(1) die PID des Prozesses, dem Sie ein Signal schicken wollen. In diesem Beispiel ist die PID von inetd(8) 198:

```
% pgrep -l inetd
198 inetd -wW
```

2. Benutzen Sie kill(1), um ein Signal zu senden. Da inetd(8) dem Benutzer `root` gehört, müssen Sie zuerst mit su(1) `root` werden:

```
% su
Password:
```

[3] Es gibt Fälle, in denen ein Prozess nicht unterbrochen werden kann. Wenn ein Prozess zum Beispiel eine Datei von einem anderen Rechner auf dem Netzwerk liest und dieser Rechner nicht erreichbar ist, dann ist der Prozess nicht zu unterbrechen. Wenn der Prozess den Lesezugriff nach einem Timeout von typischerweise zwei Minuten aufgibt, dann wird er beendet.

```
# /bin/kill -s HUP 198
```

kill(1) wird, wie andere UNIX® Kommandos auch, keine Ausgabe erzeugen, wenn das Kommando erfolgreich war. Wird versucht, einem Prozess der nicht dem Benutzer gehört, ein Signal zu senden, dann wird die Meldung kill: *PID*: Operation not permitted ausgegeben. Ein Tippfehler bei der Eingabe der PID führt dazu, dass das Signal an einen falschen Prozess gesendet wird, was zu negativen Ergebnissen führen kann, oder das Signal wird an eine PID gesendet die derzeit nicht in Gebrauch ist, was zu dem Fehler kill: *PID*: No such process führt.

Warum sollte man `/bin/kill` benutzen?

Viele Shells stellen `kill` als internes Kommando zur Verfügung, das heißt die Shell sendet das Signal direkt, anstatt `/bin/kill` zu starten. Beachten Sie, dass die unterschiedlichen Shells eine andere Syntax benutzen, um die Namen der Signale anzugeben. Anstatt jede Syntax zu lernen, kann es einfacher sein, `/bin/kill` direkt aufzurufen.

Beim Versenden von anderen Signalen, ersetzen Sie `TERM` oder `KILL` in der Kommandozeile mit dem Namen des Signals.

Wichtig

Das zufällige Beenden eines Prozesses kann gravierende Auswirkungen haben. Insbesondere init(8), mit der PID 1, ist ein Spezialfall. `/bin/kill -s KILL 1` ist ein schneller, jedoch nicht empfohlener Weg, das System herunterzufahren. Überprüfen Sie die Argumente von kill(1) *immer* zweimal *bevor* Sie Return drücken.

4.9. Shells

Eine *Shell* stellt eine Kommandozeilen-Schnittstelle zur Interaktion mit dem Betriebssystem zur Verfügung. Sie empfängt Befehle von einem Eingabekanal und führt diese aus. Viele Shells bieten eingebaute Funktionen, die die tägliche Arbeit erleichtern, beispielsweise eine Dateiverwaltung, die Vervollständigung von Dateinamen (Globbing), Kommandozeilen-Editor, sowie Makros und Umgebungsvariablen. FreeBSD enthält einige Shells, darunter die Bourne Shell (sh(1)) und die verbesserte C-Shell (tcsh(1)). Weitere Shells, wie `zsh` oder `bash`, befinden sich in der Ports-Sammlung.

Die verwendete Shell ist letztlich eine Frage des Geschmacks. Ein C-Programmierer, findet vielleicht eine C-artige Shell wie tcsh(1) angenehmer. Ein Linux®-Benutzer bevorzugt vielleicht `bash`. Jede Shell hat ihre speziellen Eigenschaften, die mit der bevorzugten Arbeitsumgebung des Benutzers harmonieren kann oder nicht. Deshalb stehen mehrere Shells zur Auswahl.

Ein verbreitetes Merkmal in Shells ist die Dateinamen-Vervollständigung. Nachdem der Benutzer einige Buchstaben eines Kommandos oder eines Dateinamen eingeben hat, vervollständigt die Shell den Rest durch drücken der Tab-Taste. Angenommen, Sie haben zwei Dateien `foobar` und `football`. Um `foobar` zu löschen, kann der Benutzer `rm foo` eingeben und Tab drücken um den Dateinamen zu vervollständigen.

Die Shell wird lediglich `rm foo` anzeigen. Sie konnte den Dateinamen nicht vervollständigen, da sowohl `foobar` als auch `football` mit `foo` anfangen. Einige Shells geben einen Signalton aus, oder zeigen alle Möglichkeiten an, wenn mehr als ein Name mit dem gegebenen Muster übereinstimmt. Der Benutzer muss dann weitere Zeichen eingeben, damit die Shell den gewünschten Dateinamen bestimmen kann. Durch Eingabe von `t` und erneutes Drücken von Tab ist die Shell in der Lage, den gewünschten Dateinamen zu vervollständigen.

Ein weiteres Merkmal der Shell ist der Gebrauch von Umgebungsvariablen. Dies sind veränderbare Schlüsselpaare im Umgebungsraum der Shell, die jedes von der Shell aufgerufene Programm lesen kann. Daher enthält der Umgebungsraum viele Konfigurationsdaten für Programme. Tabelle 4.4, „Gebräuchliche Umgebungsvariablen“ zeigt verbreitete Umgebungsvariablen und deren Bedeutung. Beachten Sie, dass die Namen der Umgebungsvariablen immer in Großbuchstaben geschrieben sind:

Tabelle 4.4. Gebräuchliche Umgebungsvariablen

Variable	Beschreibung
`USER`	Name des angemeldeten Benutzers.
`PATH`	Liste mit Verzeichnissen (getrennt durch Doppelpunkt) zum Suchen nach Programmen.
`DISPLAY`	Der Name des Xorg-Bildschirms, auf dem Ausgaben erfolgen sollen.
`SHELL`	Die aktuelle Shell.
`TERM`	Name des Terminaltyps des Benutzers. Benutzt, um die Fähigkeiten des Terminals zu bestimmen.
`TERMCAP`	Datenbankeintrag der Terminal Escape Codes, benötigt um verschieden Terminalfunktionen auszuführen.
`OSTYPE`	Typ des Betriebssystems.
`MACHTYPE`	Die CPU-Architektur des Systems.
`EDITOR`	Vom Benutzer bevorzugter Text-Editor.
`PAGER`	Vom Benutzer bevorzugter Text-Betrachter.
`MANPATH`	Liste mit Verzeichnissen (getrennt durch Doppelpunkt) zum Suchen nach Manualpages.

Das Setzen von Umgebungsvariablen unterscheidet sich von Shell zu Shell. In tcsh(1) und csh(1) wird dazu `setenv` benutzt. sh(1) und `bash` benutzen `export` um Umgebungsvariablen zu setzen. Dieses Beispiel für die tcsh(1)-Shell setzt die Variable `EDITOR` auf `/usr/local/bin/emacs` :

```
% setenv EDITOR /usr/local/bin/emacs
```

Der entsprechende Befehl für `bash` wäre:

```
% export EDITOR="/usr/local/bin/emacs"
```

Um eine Umgebungsvariable zu expandieren, geben Sie in der Kommandozeile das Zeichen `$` vor dessen Namen ein. Zum Beispiel gibt `echo $TERM` den aktuellen Wert von`$TERM` aus.

Shells behandeln Spezialzeichen, so genannte Metazeichen, als besondere Darstellungen für Daten. Das häufigste Zeichen ist *, das eine beliebige Anzahl Zeichen in einem Dateinamen repräsentiert. Metazeichen können zur Vervollständigung von Dateinamen (Globbing) benutzt werden. Beispielsweise liefert `echo *` nahezu das gleiche wie `ls`, da die Shell alle Dateinamen die mit * übereinstimmen, an `echo` weitergibt.

Um zu verhindern, dass die Shell ein Sonderzeichen interpretiert, schützt man es, indem man einen Backslash (\) voranstellt. Zum Beispiel zeigt `echo $TERM` die Einstellung des Terminals an, wohingegen `echo \$TERM` einfach die Zeichenfolge `$TERM` ausgibt.

4.9.1. Ändern der Shell

Der einfachste Weg die Standard Shell zu ändern, ist `chsh` zu benutzen. `chsh` startet den Editor, welcher durch die Umgebungsvariable `EDITOR` gesetzt ist. Standardmäßig ist dies vi(1). Tragen Sie in die Zeile die mit `Shell:` beginnt, den absoluten Pfad der neuen Shell ein.

Alternativ setzt `chsh -s` die Shell, ohne dabei einen Editor aufzurufen. Um die Shell zum Beispiel auf `bash` zu ändern, geben Sie folgenden Befehl ein:

```
% chsh -s /usr/local/bin/bash
```

Anmerkung

Die neue Shell *muss* in `/etc/shells` aufgeführt sein. Wurde die Shell aus der FreeBSD Ports-Sammlung installiert, so wie in Kapitel 5, *Installieren von Anwendungen: Pakete und Ports* beschrieben, sollte sie automatisch zu dieser Datei hinzugefügt worden sein. Wenn der Eintrag fehlt, nutzen Sie folgenden Befehl, und ersetzen Sie den Pfad mit dem Pfad zur gewünschten Shell:

```
# echo /usr/local/bin/bash  >> /etc/shells
```

Danach kann chsh(1) erneut aufgerufen werden.

4.9.2. Fortgeschrittene Shell Techniken

Geschrieben von Tom Rhodes.

Die UNIX®-Shell ist nicht nur ein Kommandozeileninterpreter, sie ist ein leistungsfähiges Werkzeug, das Benutzern die Ausführung von Befehlen ermöglicht. Es kann die Ein- und Ausgabe umleiten und Befehle miteinander verketten, um die finale Ausgabe zu verbessern. Diese Funktionalität, gepaart mit den eingebauten Befehlen, bietet dem Benutzer eine Umgebung, welche die Effizienz erheblich steigern kann.

Als *Redirection* bezeichnet man die Umleitung der Ein- oder Ausgabe in einen anderen Befehl oder Datei. Um beispielsweise die Ausgabe des Befehls ls(1) in eine Datei zu schreiben, muss die Ausgabe umgeleitet werden:

```
% ls > Verzeichnis_Ausgabe.txt
```

Die Datei `Verzeichnis_Ausgabe.txt` enthält nun den Verzeichnisinhalt. Einige Befehle, wie beispielsweise sort(1), können verwendet werden um von der Eingabe zu lesen. Wenn Sie die Ausgabe sortieren möchten, müssen Sie die Eingabe umleiten:

```
% sort < Verzeichnis_Ausgabe.txt
```

Die Eingabe wird sortiert und auf dem Bildschirm ausgegeben. Um diese Ausgabe wiederum in eine Datei umzuleiten, kann die Ausgabe von sort(1) umgeleitet werden:

```
% sort < Verzeichnis_Ausgabe.txt > Sortierte_Ausgabe.txt
```

In den bisherigen Beispielen wurden für die Umleitung Dateideskriptoren verwendet. Jedes UNIX®-System verfügt über drei Dateideskriptoren: Standardeingabe (stdin), Standardausgabe (stdout) und Stardardfehlerausgabe (stderr). Jeder Deskriptor hat einen bestimmten Zweck. Die Eingabe könnte von einer Tastatur, einer Maus oder einem anderen Eingabegerät stammen. Die Ausgabe könnte der Bildschirm oder ein Drucker sein. Die Standardfehlerausgabe wird zur Diagnose und für Fehlermeldungen verwendet. Alle drei Deskriptoren arbeiten I/O basiert und werden häufig als Streams bezeichnet.

Die Verwendung von Deskriptoren erlaubt es der Shell, die Ein- und Ausgabe von verschiedenen Kommandos umzuleiten und zu teilen. Eine weitere Möglichkeit zur Umleitung bietet der Pipe-Operator.

Der UNIX® Pipe-Operator „|“ wird verwendet, um die Ausgabe eines Kommandos an ein anderes Programm zu übergeben. Grundsätzlich bedeutet dies, dass die Standardausgabe eines Programms als Standardeingabe für ein weiteres Programm verwendet wird. Ein Beispiel:

```
% cat Verzeichnis_Auflistung.txt | sort | less
```

In diesem Beispiel wird der Inhalt von `Verzeichnis_Auflistung.txt` sortiert und die Ausgabe an less(1) übergeben. Dies erlaubt es dem Benutzer, die Ausgabe Schritt für Schritt und im eigenen Tempo zu betrachten.

4.10. Text-Editoren

Die meiste Konfiguration unter FreeBSD wird durch das Editieren von Textdateien erledigt. Deshalb ist es eine gute Idee, mit einem Texteditor vertraut zu werden. FreeBSD hat ein paar davon im Basissystem und sehr viel mehr in der Ports-Sammlung.

Ein einfach zu erlernender Editor ist ee(1), was für *easy editor* steht. Um diesen Editor zu starten, gibt man in der Kommandozeile `ee` *`filename`* ein, wobei *`filename`* den Namen der zu editierenden Datei darstellt. Einmal im Editor, finden sich alle Editor-Funktionen oben im Display aufgelistet. Das Einschaltungszeichen (^) steht für die Ctrl (oder Strg) Taste, mit `^e` ist also die Tastenkombination Ctrl+e gemeint. Um ee(1) zu verlassen, drücken Sie Esc und wählen dann im Hauptmenü `leave editor` aus. Der Editor fragt nach, ob Sie speichern möchten, wenn die Datei verändert wurde.

FreeBSD verfügt über leistungsfähigere Editoren wie vi(1) als Teil des Basissystems. Andere Editoren wie editors/emacs und editors/vim sind Teil der Ports-Sammlung. Diese Editoren bieten höhere Funktionalität, jedoch auf Kosten einer etwas schwierigeren Erlernbarkeit. Das Erlernen eines leistungsfähigeren Editors, wie vim oder Emacs, kann auf lange Sicht Zeit einsparen.

Viele Anwendungen, die Dateien verändern oder Texteingabe erwarten, werden automatisch einen Texteditor öffnen. Um den Standardeditor zu ändern, wird die Umgebungsvariable `EDITOR` gesetzt, wie im Abschnitt Abschnitt 4.9, „Shells“ beschrieben.

4.11. Geräte und Gerätedateien

Der Begriff Gerät wird meist in Verbindung mit Hardware wie Laufwerken, Druckern, Grafikkarten oder Tastaturen gebraucht. Der Großteil der Meldungen, die beim Booten von FreeBSD angezeigt werden, beziehen sich auf gefundene Geräte. Eine Kopie dieser Bootmeldungen wird in `/var/run/dmesg.boot` gespeichert.

Jedes Gerät verfügt über einen Gerätenamen und Gerätenummer. Zum Beispiel steht `acd0` für das erste IDE CD-ROM Laufwerk, während `kbd0` die Tastatur repräsentiert.

Auf die meisten Geräte wird unter FreeBSD über spezielle Gerätedateien im `/dev` Verzeichnis zugegriffen.

4.12. Manualpages

4.12.1. Manualpages

Die umfassendste Dokumentation rund um FreeBSD gibt es in Form von Manualpages. Annähernd jedes Programm im System bringt eine kurze Referenzdokumentation mit, die die grundsätzliche Funktion und verschiedene Parameter erklärt. Diese Manuals können mit `man` eingesehen werden:

```
% man Kommando
```

`Kommando` ist der Name des Kommandos, über das man etwas erfahren will. Um beispielsweise mehr über das Kommando ls(1) zu erfahren, geben Sie ein:

```
% man ls
```

Die Manualpages sind in nummerierte Sektionen unterteilt, die jeweils ein Thema darstellen. In FreeBSD sind die folgenden Sektionen verfügbar:

1. Benutzerkommandos.
2. Systemaufrufe und Fehlernummern.
3. Funktionen der C Bibliothek.
4. Gerätetreiber.
5. Dateiformate.
6. Spiele und andere Unterhaltung.
7. Verschiedene Informationen.
8. Systemverwaltung und -Kommandos.
9. Kernel Schnittstellen.

In einigen Fällen kann dasselbe Thema in mehreren Sektionen auftauchen. Es gibt zum Beispiel ein `chmod` Benutzerkommando und einen `chmod()` Systemaufruf. Um man(1) mitzuteilen, aus welcher Sektion die Information angezeigt werden soll, kann die Sektionsnummer mit angeben werden:

```
% man 1 chmod
```

Dies wird Ihnen die Manualpage für das Benutzerkommando chmod(1) zeigen. Verweise auf eine Sektion der Manualpages werden traditionell in Klammern gesetzt. So bezieht sich chmod(1) auf das Benutzerkommando und chmod(2) auf den Systemaufruf.

Wenn das Kommandos nicht bekannt ist, kann `man -k` benutzt werden, um nach Schlüsselbegriffen in den Kommandobeschreibungen zu suchen:

```
% man -k mail
```

Dieser Befehl zeigt eine Liste von Kommandos, deren Beschreibung das Schlüsselwort „mail“ enthält. Die gleiche Funktionalität erhalten Sie auch, wenn Sie apropos(1) benutzen.

Um die Beschreibungen der Kommandos in `/usr/bin` zu lesen, geben Sie ein:

```
% cd /usr/bin
% man -f * | more
```

Dasselbe erreichen Sie durch Eingabe von:

```
% cd /usr/bin
% whatis * | more
```

4.12.2. GNU Info Dateien

FreeBSD enthält viele Anwendungen und Utilities der Free Software Foundation (FSF). Zusätzlich zu den Manualpages können diese Programme Hypertext-Dokumente enthalten, die `info`-Seiten genannt werden. Diese Dokumente können mit info(1) ansehen kann. Wenn editors/emacs installiert ist, kann auch der info-Modus von emacs benutzt werden.

Um info(1) zu benutzen, geben Sie ein:

```
% info
```

Eine kurze Einführung gibt es mit `h`; eine Befehlsreferenz erhalten Sie durch Eingabe von: `?`.

Kapitel 5. Installieren von Anwendungen: Pakete und Ports

Übersetzt von Uwe Pierau.
Überarbeitet von Björn Heidotting.

5.1. Übersicht

FreeBSD enthält eine umfassende Sammlung von Systemwerkzeugen, die Teil des Basissystems sind. Darüber hinaus stellt FreeBSD zwei sich ergänzende Methoden zur Installation von Drittanbieter-Software zur Verfügung: Die Ports-Sammlung zur Installation aus dem Quellcode sowie Pakete zur Installation von vorkompilierten binären Softwarepaketen. Beide Methoden können benutzt werden, um Anwendungen von lokalen Medien oder über das Netzwerk zu installieren.

Dieses Kapitel behandelt die folgenden Themen:

- Den Unterschied zwischen binären Softwarepaketen und Ports.
- Wie man Drittanbieter-Software findet, die nach FreeBSD portiert wurde.
- Wie Binärpakete mit pkg verwaltet werden.
- Den Bau von Drittanbieter-Software aus dem Quellcode mithilfe der Ports-Sammlung.
- Wie man die Dateien findet, die zusammen mit der Anwendung installiert wurden.
- Was zu tun ist, wenn die Installation einer Software fehlschlägt.

5.2. Installation von Software

Die typischen Schritte zur Installation von Drittanbieter-Software auf einem UNIX® System sind:

1. Download der Software, die als Quelltext oder im Binärformat vorliegen kann.
2. Auspacken der Software. Dies ist typischerweise ein mit compress(1), gzip(1) oder bzip2(1) komprimiertes Tar-Archiv.
3. Durchsuchen der Dokumentation, die sich in `INSTALL`, `README` oder mehreren Dateien im Verzeichnis `doc/` befindet, nach Anweisungen, wie die Software zu installieren ist.
4. Kompilieren der Software, wenn sie als Quelltext vorliegt. Dazu muss vielleicht das `Makefile` angepasst, oder `configure` ausgeführt werden.
5. Testen und installieren der Software.

Wenn die Software nicht speziell für FreeBSD geschrieben wurde, muss vielleicht sogar der Quelltext angepasst werden, damit die Software funktioniert. Zurzeit werden über 24,000 Anwendungen Dritter zur Verfügung gestellt.

Ein FreeBSD-Paket enthält vorkompilierte Kopien aller Befehle für eine Anwendung, sowie zusätzliche Konfigurationsdateien und Dokumentation. Pakete können mit den pkg-Befehlen, wie `pkg install` , manipuliert werden.

Ein FreeBSD-Port ist eine Sammlung von Dateien, die das Kompilieren der Quelltexte einer Anwendung automatisieren. Die Dateien, die ein Port umfasst enthalten alle notwendigen Informationen um die Anwendung herunterzuladen, zu extrahieren, anzupassen und zu installieren.

Das Portsystem kann auch dazu benutzt werden, Pakete zu generieren, die mit den Paketverwaltungswerkzeugen von FreeBSD manipuliert werden können.

Pakete und Ports beachten Abhängigkeiten zwischen Anwendungen. Wenn ein Paket oder die Ports-Sammlung benutzt wird, um eine Anwendung zu installieren, dann werden fehlende Bibliotheken zuerst installiert, sofern sie nicht schon vorher installiert waren.

Obwohl beide Technologien gleichartig sind, so haben Pakete und Ports jeweils ihre eigenen Stärken. Welche Technologie eingesetzt wird, hängt letzten Endes von den Anforderungen ab, die an eine bestimmte Anwendung gestellt werden.

- Das komprimierte Paket einer Anwendung ist normalerweise kleiner als das komprimierte Archiv der Quelltexte.
- Pakete müssen nicht mehr kompiliert werden. Dies ist ein Vorteil, wenn große Pakete wie Mozilla, KDE oder GNOME auf langsamen Maschinen installiert werden.
- Wenn Sie Pakete verwenden, brauchen Sie nicht zu verstehen, wie Software unter FreeBSD kompiliert wird.
- Da die Pakete auf möglichst vielen System laufen sollen, werden Optionen beim Übersetzen zurückhaltend gesetzt. Wird eine Anwendung über die Ports übersetzt, können die Optionen nach eigenen Bedürfnissen angepasst werden.
- Die Eigenschaften einiger Anwendungen werden über Optionen zum Zeitpunkt des Übersetzens festgelegt. Apache kann zum Beispiel über eine große Auswahl an eingebauten Optionen konfiguriert werden.

  Für einige Fälle existieren verschiedene Pakete einer Anwendung, die beim Übersetzen unterschiedlich konfiguriert wurden. Für Ghostscript gibt es ein `ghostscript`-Paket und ein `ghostscript-nox11` -Paket, die sich durch die Xorg Unterstützung unterscheiden. Das Erstellen von verschiedenen Paketen wird aber schnell unhandlich, wenn eine Anwendung mehr als ein oder zwei Optionen zum Zeitpunkt des Übersetzens besitzt.
- Die Lizenzbestimmungen mancher Software verbietet ein Verbreiten in binärer Form. Diese Software muss als Quelltext, der durch den Benutzer kompiliert werden muss, ausgeliefert werden.
- Einige Leute trauen binären Distributionen nicht, oder sie ziehen es vor den Quelltext zu lesen, um diesen nach möglichen Problemen zu durchsuchen.
- Der Quellcode wird benötigt, um individuelle Anpassungen anzuwenden.

Wenn Sie über aktualisierte Ports informiert sein wollen, lesen Sie die Mailinglisten FreeBSD ports und FreeBSD ports bugs.

Warnung

Bevor Sie eine Anwendung installieren, informieren Sie sich auf der Seite http://vuxml.FreeBSD.org/ über mögliche Sicherheitsprobleme mit der Anwendung, oder führen Sie `pkg audit -F` aus, um alle installierten Pakete auf bekannte Sicherheitslücken zu überprüfen.

Der Rest dieses Kapitels beschreibt, wie man Software Dritter mit Paketen und Ports unter FreeBSD installiert und verwaltet.

5.3. Suchen einer Anwendung

Die Anzahl der nach FreeBSD portierten Anwendungen steigt ständig. Es gibt einige Wege, um nach Anwendungen zu suchen:

- Die FreeBSD-Webseite stellt unter http://www.FreeBSD.org/ports/ eine aktuelle und durchsuchbare Liste aller Anwendungen zur Verfügung. Die Ports können nach dem Namen den Anwendung, oder über die Software-Kategorie durchsucht werden.

- Dan Langille verwaltet FreshPorts.org, das eine umfassende Suchfunktion bietet und Änderungen an den Anwendungen in der Ports-Sammlung verfolgt. Registrierte Benutzer können eine Merkliste erstellen, um automatisch eine E-Mail zu erhalten, sobald ein Port von dieser Liste aktualisiert wurde.

- Wenn Sie bei der Suche nach einer bestimmten Anwendung nicht weiter kommen, versuchen Sie eine Webseite wie SourceForge.net oder GitHub.com. Schauen Sie dann auf der FreeBSD-Webseite nach, ob die Anwendung portiert wurde.

- Das Paket Repository nach einer Anwendung durchsuchen:

```
# pkg search subversion
git-subversion-1.9.2
java-subversion-1.8.8_2
p5-subversion-1.8.8_2
py27-hgsubversion-1.6
py27-subversion-1.8.8_2
ruby-subversion-1.8.8_2
subversion-1.8.8_2
subversion-book-4515
subversion-static-1.8.8_2
subversion16-1.6.23_4
subversion17-1.7.16_2
```

Die Paketnamen enthalten jeweils die Versionsnummer. Wenn ein Port von python abhängt, wird auch die Versionsnummer von python ausgegeben, mit der die Anwendung gebaut wurde. Für einige Ports stehen sogar mehrere Versionen zur Verfügung. Im Fall von subversion gibt es drei verschiedene Versionen, mit unterschiedlichen Optionen. In diesem Fall wird die Version von subversion statisch gelinkt. Wenn Sie ein Paket installieren, ist es am besten den Ursprung des Ports anzugeben, also den Pfad in der Ports-Sammlung. Wiederholen Sie `pkg search` mit `-o` um den Ursprung der Pakete anzuzeigen:

```
# pkg search -o subversion
devel/git-subversion
java/java-subversion
devel/p5-subversion
devel/py-hgsubversion
devel/py-subversion
devel/ruby-subversion
devel/subversion16
devel/subversion17
devel/subversion
devel/subversion-book
devel/subversion-static
```

Zudem unterstützt `pkg search` die Suche mit regulären Ausdrücken, nach exakten Treffern, nach der Beschreibung oder nach anderen Feldern in der Repository-Datenbank. Nach der Installation von ports-mgmt/pkg oder ports-mgmt/pkg-devel, finden Sie in pkg-search(8) weitere Details.

- Wenn die Ports-Sammlung bereits installiert ist, gibt es mehrere Methoden, um die lokale Version dieser Port-Sammlung abzufragen. Verwenden Sie `whereis Datei` um herauszufinden, in welcher Kategorie ein Port ist, wobei *Datei* der Name des Programms ist, das installiert werden soll:

```
# whereis lsof
lsof: /usr/ports/sysutils/lsof
```

Alternativ kann der echo(1)-Befehl verwendet werden:

```
# echo /usr/ports/*/*lsof*
/usr/ports/sysutils/lsof
```

Beachten Sie aber, dass dieser Befehl auch alle Dateien im Verzeichnis `/usr/ports/distfiles` findet, auf die der angegebene Suchbegriff passt.

- Ein weiterer Weg nach Software zu suchen besteht darin, die eingebaute Suchfunktion der Ports-Sammlung zu benutzen. Wechseln Sie dazu in das Verzeichnis `/usr/ports` , und rufen Sie `make search name=` *Anwendungsname* auf, wobei *Anwendungsname* der Name der Software ist. Um zum Beispiel nach `lsof` zu suchen:

```
# cd /usr/ports
# make search name=lsof
Port:   lsof-4.88.d,8
Path:   /usr/ports/sysutils/lsof
Info:   Lists information about open files (similar to fstat(1))
Maint:  ler@lerctr.org
Index:  sysutils
B-deps:
R-deps:
```

Tipp

Der integrierte Suchmechanismus verwendet eine Datei mit Index-Informationen. Erscheint eine Meldung, dass der `INDEX` benötigt wird, führen Sie `make fetchindex` aus, um die aktuelle Index-Datei herunterzuladen. Mit einem vorhandenen `INDEX` ist `make search` in der Lage, die gewünschte Suche durchzuführen.

Die „Path:“-Zeile zeigt an, wo der Port zu finden ist.

Um weniger Informationen zu erhalten, benutzen Sie die Funktion `quicksearch`:

```
# cd /usr/ports
# make quicksearch name=lsof
Port: lsof-4.88.d,8
Path: /usr/ports/sysutils/lsof
Info: Lists information about open files (similar to fstat(1))
```

Erweiterte Suchen führen Sie mit `make search key=` *`Text`* oder `make quicksearch key=` *`Text`* aus. Damit werden Portnamen, Kommentare, Beschreibungen und Abhängigkeiten nach *`Text`* durchsucht. Dies kann sehr nützlich sein, wenn der Name des Programms nicht bekannt ist.

Bei der Verwendung von `search` und `quicksearch` wird Groß- und Kleinschreibung bei der Suche ignoriert. Die Suche nach „LSOF“ wird dieselben Ergebnisse wie die Suche nach „lsof“ liefern.

5.4. Benutzen von pkg zur Verwaltung von Binärpaketen

pkg ist der Nachfolger für die traditionellen Paketverwaltungswerkzeuge von FreeBSD. Es bietet viele Funktionen, die den Umgang mit Binärpaketen schneller und einfacher machen.

pkg ist kein Ersatz für die Portverwaltungswerkzeuge wie ports-mgmt/portmaster oder ports-mgmt/portupgrade. Während diese Werkzeuge Drittanbieter-Software sowohl aus Binärpaketen als auch aus der Ports-Sammlung installieren können, so installiert pkg ausschließlich Binärpakete.

5.4.1. Erste Schritte mit pkg

FreeBSD enthält ein Bootstrap-Programm, welches pkg zusammen mit den Manualpages installiert.

Um das Bootstrap Programm zu starten, geben Sie folgendes ein:

```
# /usr/sbin/pkg
```

Für ältere Versionen von FreeBSD muss pkg zunächst aus der Ports-Sammlung oder als Paket installiert werden.

Um den Port zu installieren, geben Sie folgendes ein:

```
# cd /usr/ports/ports-mgmt/pkg
# make
# make install clean
```

Bei der Aktualisierung eines bestehenden Systems, welches ursprünglich die alten Paketverwaltungswerkzeuge verwendet hat, muss die Datenbank in das neue Format konvertiert werden, damit die neuen Werkzeuge wissen, welche Pakete bereits installiert sind. Sobald pkg installiert ist, muss die Paketdatenbank mit dem folgenden Befehl vom traditionellen Format in das neue Format konvertiert werden:

```
# pkg2ng
```

Anmerkung

Auf neu installieren Systemen, auf denen noch keine Software von Drittanbietern installiert wurde, kann dieser Schritt entfallen.

Wichtig

Die Konvertierung ist unwiderruflich. Sobald die Paketdatenbank in das Format von pkg umgewandelt wurde, sollten die traditionellen pkg_* Werkzeuge nicht mehr benutzt werden.

Anmerkung

Bei der Konvertierung der Paketdatenbank können Fehler ausgegeben werden, wenn die Inhalte auf die neue Version umgewandelt werden. Im Allgemeinen können diese Fehler ignoriert werden. Wenn pkg2ng fertig ist, wird eine Liste von Drittanbieter-Software ausgegeben, die nicht erfolgreich konvertiert werden konnte. Diese Anwendungen müssen manuell neu installiert werden.

Um sicherzustellen, dass die FreeBSD Ports-Sammlung neue Pakete mit pkg und nicht mit den traditionellen Formaten registriert, muss in FreeBSD 10.*X* und früheren Versionen folgende Zeile in `/etc/make.conf` hinzugefügt werden:

```
WITH_PKGNG=     yes
```

Das pkg Paketverwaltungssystem benutzt die Pakete der FreeBSD-Spiegel. Wenn Sie ein eigenes Paket-Repository erstellen möchten, lesen Sie Abschnitt 5.6, „Pakete mit Poudriere bauen“

Weitere Konfigurationsoptionen für pkg sind in pkg.conf(5) beschrieben.

Informationen zur Bedienung von pkg ist in pkg(8) verfügbar. Alternativ kann `pkg` ohne zusätzliche Argumente aufgerufen werden.

Jedes Argument von pkg ist in seiner spezifischen Manualpage dokumentiert. Um beispielsweise die Manualpage von `pkg install` zu lesen, geben Sie einen der folgenden Befehle ein:

```
# pkg help install
```

```
# man pkg-install
```

Der Rest dieses Abschnitts beschreibt die typischen Verwaltungsaufgaben für Binärpakete, die mit pkg erledigt werden können. Jedes gezeigte Kommando verfügt über Optionen, um das Verhalten anzupassen. Details und weitere Beispiele finden Sie in den Manualpages der einzelnen Kommandos.

5.4.2. Informationen über installierte Pakete anzeigen

Informationen über bereits installierte Pakete können mit `pkg info` angezeigt werden. Dabei wird, wenn keine weiteren Optionen angegeben werden, die Version und die Beschreibung aller Pakete oder eines einzelnen Pakets ausgegeben.

Um zu ermitteln welche Version von pkg installiert ist, geben Sie folgendes ein:

```
# pkg info pkg
pkg-1.1.4_1
```

5.4.3. Installation und Deinstallation von Paketen

Ein Binärpaket installieren Sie mit dem folgenden Befehl, wobei *`paketname`* der Name des zu installierenden Pakets ist:

```
# pkg install paketname
```

Dieser Befehl verwendet Daten aus dem Repository um zu bestimmen, welche Version der Software und welche Abhängigkeiten installiert werden müssen. Um beispielsweise curl zu installieren:

```
# pkg install curl
Updating repository catalogue
/usr/local/tmp/All/curl-7.31.0_1.txz          100% of 1181 kB 1380 kBps 00m01s

/usr/local/tmp/All/ca_root_nss-3.15.1_1.txz   100% of  288 kB 1700 kBps 00m00s

Updating repository catalogue
The following 2 packages will be installed:

        Installing ca_root_nss: 3.15.1_1
        Installing curl: 7.31.0_1

The installation will require 3 MB more space

0 MB to be downloaded

Proceed with installing packages [y/N]: y
Checking integrity... done
[1/2] Installing ca_root_nss-3.15.1_1... done
[2/2] Installing curl-7.31.0_1... done
Cleaning up cache files...Done
```

Das neue Paket und jedes weitere Paket, das als Abhängigkeit installiert wurde, ist in der Liste der installierten Pakete zu sehen:

```
# pkg info
ca_root_nss-3.15.1_1     The root certificate bundle from the Mozilla Project
curl-7.31.0_1     Non-interactive tool to get files from FTP, GOPHER, HTTP(S) servers
pkg-1.1.4_6       New generation package manager
```

Wird ein Paket nicht mehr benötigt, kann es mit `pkg delete` entfernt werden. Zum Beispiel:

```
# pkg delete curl
```

```
The following packages will be deleted:

      curl-7.31.0_1

The deletion will free 3 MB

Proceed with deleting packages [y/N]: y
[1/1] Deleting curl-7.31.0_1... done
```

5.4.4. Installierte Pakete aktualisieren

Installierte Pakete können mit diesem Kommando auf die neuesten Versionen aktualisiert werden:

```
# pkg upgrade
```

Dieses Kommando vergleicht und aktualisiert die installierten Versionen der Pakete mit denen im Repository.

5.4.5. Installierte Pakete auditieren

Gelegentlich werden Sicherheitslücken in einer Drittanbieter-Software entdeckt. pkg besitzt einen eingebauten Auditing-Mechanismus. Um die auf dem System installierte Software auf Sicherheitslücken zu prüfen, geben Sie folgenden Befehl ein:

```
# pkg audit -F
```

5.4.6. Automatisches Entfernen von nicht mehr benötigten Abhängigkeiten

Das Entfernen eines Pakets kann möglicherweise Abhängigkeiten hinterlassen, die nicht mehr benötigt werden. Unnötige Pakete, die als Abhängigkeit von anderen Paketen installiert wurden, können automatisch erfasst und entfernt werden:

```
# pkg autoremove
Packages to be removed:
    ca_root_nss-3.15.1_1

The autoremoval will free 723 kB

Proceed with autoremoval of packages [y/N]: y
Deinstalling ca_root_nss-3.15.1_1... done
```

5.4.7. Wiederherstellung der Paketdatenbank

Im Gegensatz zum alten Paketverwaltungssystem beinhaltet pkg einen eigenen Mechanismus zur Sicherung der Paketdatenbank. Diese Funktionalität ist standardmäßig aktiviert.

Tipp

Um das Skript daran zu hindern, eine Sicherung der Paketdatenbank zu erstellen, muss in periodic.conf(5) `daily_backup_pkgdb_enable="NO"` gesetzt werden.

Um den Inhalt einer früheren Paketdatenbank wiederherzustellen, geben Sie folgendes Kommando ein und ersetzen Sie */path/to/pkg.sql* durch den Speicherort der gesicherten Datenbank:

```
# pkg backup -r /path/to/pkg.sql
```

Anmerkung

Wenn Sie eine Sicherung wiederherstellen, die von einem `periodic` Skript erstellt wurde, müssen Sie diese zuerst dekomprimieren.

Um eine manuelle Sicherung der pkg Paketdatenbank zu erstellen, führen Sie den folgenden Befehl aus, und ersetzen Sie */path/to/pkg.sql* durch einen geeigneten Dateinamen:

```
# pkg backup -d /path/to/pkg.sql
```

5.4.8. Alte Pakete entfernen

Standardmäßig speichert pkg Pakete in einem Cache-Verzeichnis, welches in pkg.conf(5) in der Variablen `PKG_CACHEDIR` definiert wird. Nur Kopien der neusten installierten Pakete werden beibehalten. Ältere Versionen von pkg haben alle Pakete aufbewahrt. Um diese veralteten Pakete zu entfernen, geben Sie folgendes ein:

```
# pkg clean
```

Um alle Pakte aus dem Cache-Verzeichnis zu löschen, geben Sie ein:

```
# pkg clean -a
```

5.4.9. Manipulation der Paket-Metadaten

Bei Software aus der FreeBSD Ports-Sammlung kann es vorkommen, dass die Hauptversionsnummer geändert wird. Dafür hat pkg ein eingebautes Kommando, um die Quelle eines Pakets zu aktualisieren. Dies ist nützlich, wenn zum Beispiel lang/php5 zu lang/php53 umbenannt wurde, damit lang/php5 jetzt die Version `5.4` integrieren kann.

Um die Quelle des Pakets für das obige Beispiel zu ändern, geben Sie folgendes ein:

```
# pkg set -o lang/php5:lang/php53
```

Ein weiteres Beispiel: Um lang/ruby18 auf lang/ruby19 zu aktualisieren, geben Sie folgendes ein:

```
# pkg set -o lang/ruby18:lang/ruby19
```

In diesem letzten Beispiel wird die Quelle der Bibliotheken von `libglut` von graphics/libglut auf graphics/freeglut geändert:

```
# pkg set -o graphics/libglut:graphics/freeglut
```

Anmerkung

Bei einem Wechsel der Paketquelle ist es notwendig, die Pakete neu zu installieren, welche von dem Paket abhängig sind, das seine Paketquelle geändert hat. Um eine Neuinstallation von abhängigen Paketen zu erzwingen, führen Sie folgenden Befehl aus:

```
# pkg install -Rf graphics/freeglut
```

5.5. Benutzen der Ports-Sammlung

Die Ports-Sammlung ist eine Reihe von `Makefiles`, Patches und Beschreibungen, die unter `/usr/ports` gespeichert sind. Diese Dateien werden für den Bau und die Installation von Anwendungen unter FreeBSD verwendet. Bevor

eine Anwendung aus den Ports erstellt werden kann, muss zuerst die Ports-Sammlung installiert werden. Wenn Sie dies nicht bereits bei der Installation von FreeBSD getan haben, benutzen Sie eine der beiden Methoden um sie zu installieren:

Prozedur 5.1. Installation mit Portsnap

FreeBSDs Basissystem enthält mit Portsnap ein schnelles und benutzerfreundliches Werkzeug zur Installation der Ports-Sammlung und die bevorzugte Wahl für die meisten Benutzer. Dieses Programm stellt eine Verbindung zu einem FreeBSD-Server her, überprüft den gesicherten Schlüssel und lädt eine aktuelle Kopie der Ports-Sammlung herunter. Der Schlüssel wird benötigt, um die Integrität der heruntergeladenen Dateien zu untersuchen.

1. Laden Sie einen komprimierten Snapshot der Ports-Sammlung in `/var/db/portsnap` :

   ```
   # portsnap fetch
   ```

2. Wenn Sie Portsnap das erste Mal verwenden, müssen Sie den Snapshot nach `/usr/ports` extrahieren:

   ```
   # portsnap extract
   ```

3. Nach dem ersten Einsatz von Portsnap, kann `/usr/ports` wie folgt aktualisiert werden:

   ```
   # portsnap fetch
   # portsnap update
   ```

   Bei der Verwendung von `fetch` können die `extract` oder `update` Operationen nacheinander ausgeführt werden, etwa so:

   ```
   # portsnap fetch update
   ```

Prozedur 5.2. Installation mit Subversion

Wird mehr Kontrolle über die Ports-Sammlung benötigt, oder wenn die lokalen Änderungen beibehalten werden sollen, kann Subversion benutzt werden, um die Ports-Sammlung zu laden. Lesen Sie den Subversion Primer für eine detaillierte Beschreibung von Subversion.

1. Subversion muss installiert sein, bevor die Ports-Sammlung geladen werden kann. Ist eine lokale Kopie der Ports-Sammlung bereits vorhanden, installieren Sie Subversion wie folgt:

   ```
   # cd /usr/ports/devel/subversion
   # make install clean
   ```

   Wenn keine lokale Kopie der Ports-Sammlung vorhanden ist, oder pkg zur Verwaltung von Paketen benutzt wird, kann Subversion als Paket installiert werden:

   ```
   # pkg install subversion
   ```

2. Laden Sie eine Kopie der Ports-Sammlung:

   ```
   # svn checkout https://svn.FreeBSD.org/ports/head /usr/ports
   ```

3. Nach dem erstmaligen *checkout* mit Subversion kann `/usr/ports` wie folgt aktualisiert werden:

   ```
   # svn update /usr/ports
   ```

Die Ports-Sammlung installiert eine Reihe von Verzeichnissen, die jeweils eine Softwarekategorie repräsentieren. Jede Kategorie hat für jede einzelne Anwendung ein weiteres Unterverzeichnis. Jedes Unterverzeichnis enthält Dateien, die FreeBSD sagen, wie ein Programm kompiliert und installiert werden muss. Diese Dateien werden auch Port-„Gerüst“ genannt. Jedes Port-„Gerüst“ beinhaltet die folgenden Dateien und Verzeichnisse:

- `Makefile`: enthält Anweisungen, die spezifizieren, wie die Anwendung kompiliert wird und wohin die Komponenten installiert werden sollten.

- `distinfo`: enthält die Namen und die Prüfsummen der Dateien, die heruntergeladen werden müssen, um den Port zu bauen.
- `files`: dieses Verzeichnis enthält Patches, welche das Übersetzen und Installieren der Anwendung unter FreeBSD ermöglichen. Zudem können noch weitere Dateien, die für die Übersetzung des Ports verwendet werden, enthalten sein.
- `pkg-descr` : enthält eine ausführlichere Beschreibung der Anwendung.
- `pkg-plist` : eine Liste aller Dateien, die durch diesen Port installiert werden. Außerdem sind hier Informationen enthalten, die zum Entfernen des Ports benötigt werden.

Einige Ports beinhalten noch `pkg-message` oder weitere Dateien, die vom Port-System benutzt werden, um spezielle Situationen zu handhaben. Wenn Sie mehr über diese Dateien oder das Port-System erfahren wollen, lesen Sie das FreeBSD Porter's Handbook.

Ein Port enthält nicht den eigentlichen Quellcode, der auch als „Distfile“ bekannt ist. Der heruntergeladene Quellcode wird automatisch nach `/usr/ports/distfiles` extrahiert.

5.5.1. Ports installieren

Dieser Abschnitt beschreibt die grundlegende Benutzung der Ports-Sammlung, um Software zu installieren oder zu deinstallieren. Eine ausführliche Beschreibung der einzelnen `make`-Targets finden Sie in ports(7).

Warnung

Stellen Sie sicher, dass die Ports-Sammlung aktuell ist, bevor Sie einen Port kompilieren. Informieren Sie sich vorher zusätzlich unter http://vuxml.FreeBSD.org/ über mögliche Sicherheitsprobleme des zu installierenden Ports. Alternativ können Sie `pkg audit -F` ausführen, bevor Sie einen neuen Port installieren. Die täglich laufende Sicherheitsprüfung des Systems aktualisiert ebenfalls die Datenbank und prüft installierte Anwendungen auf vorhandene Sicherheitsprobleme. Weitere Informationen finden Sie in pkg-audit(8) und periodic(8).

Die Benutzung der Ports-Sammlung setzt eine funktionierende Internetverbindung und Superuser-Rechte voraus.

Um einen Port zu installieren, wechseln Sie in das Verzeichnis des Ports, den Sie installieren möchten. Geben Sie dann `make install` am Prompt ein:

```
# cd /usr/ports/sysutils/lsof
# make install
>> lsof_4.88D.freebsd.tar.gz doesn't seem to exist in /usr/ports/distfiles/.
>> Attempting to fetch from ftp://lsof.itap.purdue.edu/pub/tools/unix/lsof/.
===>  Extracting for lsof-4.88
...
[Ausgabe des Auspackens weggelassen]
...
>> Checksum OK for lsof_4.88D.freebsd.tar.gz.
===>  Patching for lsof-4.88.d,8
===>  Applying FreeBSD patches for lsof-4.88.d,8
===>  Configuring for lsof-4.88.d,8
...
[configure-Ausgabe weggelassen]
...
===>  Building for lsof-4.88.d,8
...
[Ausgabe der Übersetzung weggelassen]
```

```
...
===>  Installing for lsof-4.88.d,8
...
[Ausgabe der Installation weggelassen]
...
===>   Generating temporary packing list
===>   Compressing manual pages for lsof-4.57
===>   Registering installation for lsof-4.57
===>  SECURITY NOTE:
      This port has installed the following binaries which execute with
      increased privileges.
/usr/local/bin/lsof
#
```

Da `lsof` eine Anwendung ist, die mit erhöhten Rechten läuft, wird nach der Installation eine Sicherheitswarnung angezeigt. Sobald die Installation abgeschlossen ist, erscheint wieder der Prompt.

Um die Suche nach Kommandos zu beschleunigen, speichern einige Shells eine Liste der verfügbaren Kommandos in den durch die Umgebungsvariable `PATH` gegebenen Verzeichnissen. Benutzer der `tcsh` müssen eventuell `rehash` eintippen, um die neu installierten Kommandos benutzen zu können, ohne den vollständigen Pfad anzugeben. Benutzer der Shell `sh` müssen stattdessen `hash -r` eintippen. Weitere Informationen finden Sie in der Dokumentation der jeweiligen Shell.

Bei der Installation wird ein Arbeitsverzeichnis erstellt, das alle temporären Dateien enthält, die während des Bauvorgangs benötigt werden. Wenn dieses Verzeichnis nach der Installation entfernt wird, spart dies Plattenplatz und minimiert mögliche Probleme bei der Aktualisierung des Ports auf eine neuere Version:

```
# make clean
===>  Cleaning for lsof-4.88.d,8
#
```

Anmerkung

Sie können zwei Schritte sparen, wenn Sie bei der Kompilierung des Ports gleich `make install clean` eingeben.

5.5.1.1. Port Installation anpassen

Einige Ports bieten Optionen, mit denen zusätzliche Funktionen oder Sicherheitsoptionen eingestellt werden können. Beispiele dafür sind www/firefox, security/gpgme und mail/sylpheed-claws. Wenn ein Port von anderen Ports abhängig ist und diese über zusätzliche Abhängigkeiten und Optionen verfügen, wird mehrmals ein Menü ausgegeben, wo der Benutzer verschiedene Optionen wählen kann. Um dies zu vermeiden, wechseln Sie in das Verzeichnis des Ports und geben Sie `make config-recursive` ein, um die Konfiguration an einem Stück zu erledigen. Führen Sie danach `make install [clean]` aus, um den Port zu kompilieren und zu installieren.

Tipp

Bei der Verwendung von `config-recursive` wird eine Liste von Ports, die konfiguriert werden, vom Target `all-depends-list` erstellt. Es wird empfohlen, `make config-recursive` so lange auszuführen, bis alle Optionen der abhängigen Ports definiert sind und keine Optionen und Menüs mehr erscheinen. Damit soll sichergestellt werden, dass alle Optionen konfiguriert wurden.

Es gibt diverse Möglichkeiten, dieses Menü nach dem Bau eines Ports erneut aufzurufen, um Optionen zu entfernen, hinzuzufügen oder anzupassen. Sie können beispielsweise mit `cd` in das Verzeichnis des Ports wechseln und

dort `make config` eingeben. Eine andere Möglichkeit ist `make showconfig`. Eine weitere Alternative bietet `make rmconfig`, das alle ursprünglich gewählten Optionen zurücksetzt und es Ihnen dadurch ermöglicht, die Konfiguration erneut zu beginnen. Die eben erwähnten Optionen werden ausführlich in ports(7) beschrieben.

Die Ports-Sammlung benutzt zum Herunterladen von Dateien fetch(3), das diverse Umgebungsvariablen unterstützt. Die Variablen `FTP_PASSIVE_MODE`, `FTP_PROXY` und `FTP_PASSWORD` müssen unter Umständen gesetzt werden, wenn das FreeBSD-System hinter einer Firewall oder einem FTP/HTTP-Proxy arbeitet. Eine vollständige Liste der unterstützten Variablen finden Sie in fetch(1).

Benutzer ohne eine ständige Internet-Verbindung können `make fetch` im Verzeichnis `/usr/ports` ausführen, um die benötigten Dateien herunterzuladen. Es ist auch möglich, `make fetch` nur in einem Teil des Baums, wie `/usr/ports/net`, aufzurufen. Die Dateien von allen abhängigen Ports werden mit diesem Kommando allerdings nicht heruntergeladen. Wenn Sie diese Dateien ebenfalls herunterladen wollen, benutzen Sie stattdessen `make fetch-recursive`.

In einigen seltenen Fällen ist es erforderlich, die benötigten Dateien von einem anderen Ort als den im Port definierten `MASTER_SITES` herunterzuladen. Sie können `MASTER_SITES` mit dem folgenden Kommando überschreiben:

```
# cd /usr/ports/ directory
# make MASTER_SITE_OVERRIDE= \
ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/distfiles/     fetch
```

Die Variablen `WRKDIRPREFIX` und `PREFIX` überschreiben das voreingestellte Bau- und Zielverzeichnis. Zum Beispiel:

```
# make WRKDIRPREFIX=/usr/home/example/ports install
```

Dieses Kommando baut den Port unter `/usr/home/example/ports` und installiert ihn unter `/usr/local`.

Die Variable `PREFIX` legt das Installations-Verzeichnis fest:

```
# make PREFIX=/usr/home/example/local install
```

In diesem Beispiel wird der Port unter `/usr/ports` gebaut und nach `/usr/home/example/local` installiert.

Sie können beide Variablen auch zusammen benutzen:

```
# make WRKDIRPREFIX=../ports PREFIX=../local install
```

Alternativ können diese Variablen auch als Umgebungsvariablen gesetzt werden. In der Manualpage Ihrer Shell finden Sie Anweisungen, wie Umgebungsvariablen gesetzt werden.

5.5.2. Entfernen installierter Ports

Installierte Ports können mit `pkg delete` wieder deinstalliert werden. Beispiele für dieses Kommando finden Sie in pkg-delete(8).

Alternativ kann `make deinstall` im Verzeichnis des Ports aufgerufen werden:

```
# cd /usr/ports/sysutils/lsof
make deinstall
===>  Deinstalling for sysutils/lsof
===>   Deinstalling
Deinstallation has been requested for the following 1 packages:

        lsof-4.88.d,8

Thee deinstallation will free 229 kB
[1/1] Deleting lsof-4.88.d,8... done
```

Es wird empfohlen die Nachrichten zu lesen, die ausgegeben werden, wenn ein Port deinstalliert wird. Wenn der Port noch Anwendungen hat, die von ihm abhängig sind, werdenn diese am Bildschirm angezeigt, aber die Deinstal-

lation wird forgesetzt. In solchen Fällen ist es besser, die Anwendung neu zu installieren, um fehlende Abhängigkeiten zu vermeiden.

5.5.3. Ports aktualisieren

Im Laufe der Zeit stehen neuere Versionen der Software in der Ports-Sammlung zur Verfügung. In diesem Abschnitt wird beschrieben, wie Sie bestimmen, welche Software aktualisiert werden kann und wie das Upgrade durchzuführen ist.

Um festzustellen, ob neuere Versionen der installierten Ports verfügbar sind, stellen Sie sicher, dass die neueste Version der Ports-Sammlung installiert ist. Dies wird in Prozedur 5.1, „Installation mit Portsnap“ und Prozedur 5.2, „Installation mit Subversion“ beschrieben. Führen Sie unter FreeBSD 10 und neueren Versionen, bzw. auf Systemen die bereits mit pkg arbeiten, den folgenden Befehl aus, um eine Liste der installierten Ports zu erhalten für die eine aktuelle Version existiert:

```
# pkg version -l "<"
```

Mit FreeBSD 9.*X* und älteren Versionen kann stattdessen dieser Befehl verwendet werden:

```
# pkg_version -l "<"
```

Wichtig

Lesen Sie zuerst `/usr/ports/UPDATING` , bevor Sie einen Port aktualisieren. In dieser Datei werden Probleme und zusätzlich durchzuführende Schritte bei der Aktualisierung einzelner Ports beschrieben. Dazu gehören solche Dinge wie geänderte Dateiformate, verschobene Konfigurationsdateien, aber auch Inkompatibilitäten zu einer Vorgängerversion. Notieren Sie sich alle Anweisungen der Ports, die aktualisiert werden müssen. Folgen Sie den Anweisungen, wenn Sie das Upgrade durchführen.

5.5.3.1. Ports mit Portmaster aktualisieren

ports-mgmt/portmaster ist ein sehr kleines Werkzeug zum Aktualisieren von Ports. Es wurde entwickelt, um mit den in FreeBSD integrierten Werkzeugen zu arbeiten, ohne dabei von anderen Ports oder Datenbanken abhängig zu sein. Sie können das Programm aus der Ports-Sammlung installieren:

```
# cd /usr/ports/ports-mgmt/portmaster
# make install clean
```

Portmaster teilt Ports in vier Kategorien ein:

- Root Port: hat keine Abhängigkeiten und andere Ports sind nicht von diesem Port abhängig.
- Trunk Port: hat keine Abhängigkeiten, aber andere Ports sind von diesem Port abhängig.
- Branch Port: hat Abhängigkeiten und andere Ports sind von diesem Port abhängig.
- Leaf Port: hat Abhängigkeiten, aber andere Ports sind nicht von diesem Port abhängig.

Um eine Liste der installierten Ports anzuzeigen und nach neueren Versionen zu suchen, verwenden Sie:

```
# portmaster -L
===>>> Root ports (No dependencies, not depended on)
===>>> ispell-3.2.06_18
===>>> screen-4.0.3
       ===>>> New version available: screen-4.0.3_1
===>>> tcpflow-0.21_1
===>>> 7 root ports
```

```
...
===>>> Branch ports (Have dependencies, are depended on)
===>>> apache22-2.2.3
        ===>>> New version available: apache22-2.2.8
...
===>>> Leaf ports (Have dependencies, not depended on)
===>>> automake-1.9.6_2
===>>> bash-3.1.17
        ===>>> New version available: bash-3.2.33
...
===>>> 32 leaf ports

===>>> 137 total installed ports
        ===>>> 83 have new versions available
```

Um alle installierten Ports zu aktualisieren, verwenden Sie folgenden Befehl:

```
# portmaster -a
```

Anmerkung

In der Voreinstellung erzeugt Portmaster eine Sicherheitskopie, bevor ein installierter Port gelöscht wird. Ist die Installation der neuen Version erfolgreich, wird dieses Backup wieder gelöscht. Wollen Sie das Backup lieber manuell löschen, verwenden Sie die Option `-b` beim Aufruf von Portmaster. Durch die Verwendung von `-i` wird Portmaster im interaktiven Modus gestartet und fragt bei jedem zu aktualisierenden Port nach, wie weiter vorgegangen werden soll. Viele weitere Optionen stehen zur Verfügung. Lesen Sie die Manualpage von portmaster(8) für weitere Einzelheiten in Bezug auf ihre Nutzung.

Treten während der Aktualisierung Fehler auf, verwenden Sie die Option `-f`, um alle Ports zu aktualisieren beziehungsweise neu zu bauen:

```
# portmaster -af
```

Portmaster ist auch in der Lage, neue Ports zu installieren, wobei zuvor alle abhängigen Ports aktualisiert werden. Um diese Funktion zu nutzen, geben Sie den Pfad des Ports in der Ports-Sammlung an:

```
# portmaster shells/bash
```

5.5.3.2. Ports mit Portupgrade aktualisieren

Ein weiteres Werkzeug zur Aktualisierung von Ports ist Portupgrade, welches als Paket oder Port ports-mgmt/portupgrade zur Verfügung steht. Dieses Programm installiert eine Reihe von Anwendungen, die für die Verwaltung von Ports verwendet werden können. Das Programm ist jedoch von Ruby abhängig. Um den Port zu installieren, geben Sie ein:

```
# cd /usr/ports/ports-mgmt/portupgrade
# make install clean
```

Durchsuchen Sie vor jedem Update die Liste der installierten Ports mit `pkgdb -F` und beheben Sie alle gefundenen Probleme.

Benutzen Sie `portupgrade -a`, um automatisch alle veralteten Ports auf dem System zu aktualisieren. Verwenden Sie zusätzlich den Schalter `-i`, wenn Sie individuell entscheiden wollen, ob ein Port aktualisiert werden soll:

```
# portupgrade -ai
```

Um nur eine spezifische Anwendung zu aktualisieren, verwenden Sie `portupgrade` *`Paketname`*. Es ist wichtig den Schalter `-R` zu benutzen, um zuvor alle Ports zu aktualisieren, die von dem gegebenen Anwendung abhängen.

```
# portupgrade -R firefox
```

Um Pakete anstelle von Ports zu installieren, verwenden Sie den Schalter `-P`. Mit dieser Option durchsucht Portupgrade die in der Umgebungsvariablen `PKG_PATH` aufgeführten Verzeichnisse nach Paketen. Sind lokal keine Pakete vorhanden, versucht Portupgrade die Pakete über das Netz herunterzuladen. Gibt es die Pakete weder lokal noch auf entfernten Rechnern, werden die Ports verwendet. Um die Nutzung von Ports gänzlich zu verhindern, benutzen Sie die Option `-PP`. Portupgrade würde dann abbrechen, falls keine Pakete zur Verfügung stehen.

```
# portupgrade -PP gnome3
```

Wenn Sie nur die Quelldateien des Ports, oder die Pakete mit `-P` herunterladen möchten, ohne die Anwendung zu bauen oder zu installieren, geben Sie den Schalter `-F` an. Weitere Informationen zu den verfügbaren Schaltern finden Sie in der Manualpage von portupgrade(1).

5.5.4. Platzbedarf von Ports

Die Nutzung der Ports-Sammlung wird im Laufe der Zeit viel Plattenplatz verschlingen. Nach dem Bau und der Installation eines Ports, wird `make clean` die temporären Arbeitsverzeichnisse `work` aufräumen. Portmaster wird dieses Verzeichnis automatisch entfernen, wenn die Option `-K` verwendet wird. Wenn Portupgrade installiert ist, wird dieser Befehl alle Arbeitsverzeichnisse der lokalen Ports-Sammlung entfernen:

```
# portsclean -C
```

Zusätzlich werden sich im Laufe der Zeit zahlreiche veraltete Distfiles in `/usr/ports/distfiles` ansammeln. Wenn Portupgrade installiert ist, können mit dem folgenden Befehl alle Distfiles gelöscht werden, die vom keinem Port mehr benötigt werden:

```
# portsclean -D
```

Mit Portupgrade können Sie alle Distfiles löschen, die von keinem derzeit installierten Port benötigt werden:

```
# portsclean -DD
```

Wenn Portmaster installiert ist, benutzen Sie diesen Befehl:

```
# portmaster --clean-distfiles
```

In der Voreinstellung arbeitet dieses Programm interaktiv und fragt den Benutzer um Bestätigung, bevor ein Distfile gelöscht wird.

Zusätzlich zu diesen Kommandos gibt es noch port-mgmt/pkg_cutleaves. Dieses Werkzeug automatisiert die Deinstallation von installierten Ports, die nicht weiter benötigt werden.

5.6. Pakete mit Poudriere bauen

Poudriere ist ein unter der BSD-Lizenz stehendes Werkzeug zum Erstellen und Testen von FreeBSD-Paketen. Dieses Programm nutzt FreeBSD Jails, um die Pakete in einer isolierten Umgebung zu bauen. Diese Jails können verwendet werden, um Pakete für andere Versionen von FreeBSD zu bauen, oder um auf einem amd64-System Pakete für i386 zu bauen. Sobald die Pakete gebaut sind, haben sie das gleiche Format wie auf den offiziellen Spiegeln. Die Pakete können dann mit pkg(8) oder anderen Paketverwaltungswerkzeugen benutzt werden.

Poudriere wird über das Paket oder den Port ports-mgmt/poudriere installiert. Die Installation beinhaltet eine Beispielkonfiguration in `/usr/local/etc/poudriere.conf.sample` . Kopieren Sie diese Datei nach `/usr/local/etc/poudriere.conf` . Bearbeiten Sie dann die kopierte Datei, um die Konfiguration anzupassen.

Obwohl ZFS für poudriere nicht zwingend erforderlich ist, so hat die Nutzung doch einige Vorteile. Wird ZFS eingesetzt, muss in `/usr/local/etc/poudriere.conf` die Variable `ZPOOL` definiert, und die Variable `FREEBSD_HOST` auf einen nahe gelegenen Spiegel gesetzt werden. Die Definition von `CCACHE_DIR` erlaubt die Verwendung von devel/ccache, um die Bauzeit für häufig kompilierten Code verkürzen. Es kann vorteilhaft sein, die poudriere-Datasets in

einem separaten Verzeichnis auf `/poudriere` einzuhängen. Die Werte der anderen Konfigurationsvariablen sind in der Regel angemessen und brauchen nicht geändert werden.

Die Anzahl der Kerne im Prozessor wird verwendet um zu bestimmen, wie viele Bauprozesse parallel ausgeführt werden sollen. Stellen Sie ausreichend virtuellen Speicher bereit, entweder in Form von RAM oder als Swap-Speicher. Ist der virtuelle Speicher aufgebraucht, bricht der Bauprozess ab und die Jails stürzen ab, was zu seltsamen Fehlermeldungen führt.

5.6.1. Jails und Ports-Sammlung initialisieren

Nach der Konfiguration muss poudriere initialisiert werden, damit es eine Jail mit der benötigten Ports-Sammlung startet. Geben Sie mit `-j` den Namen der Jail und mit `-v` die gewünschte FreeBSD-Version an. Auf FreeBSD/amd64-Systemen kann die Architektur mit dem Schalter `-a` und `i386` oder `amd64` gesetzt werden. Der voreingestellte Wert für die Architektur können Sie sich mit `uname` anzeigen lassen.

```
# poudriere jail -c -j  10amd64  -v 10.0-RELEASE
====>> Creating 10amd64 fs... done
====>> Fetching base.txz for FreeBSD 10.0-RELEASE amd64
/poudriere/jails/10amd64/fromftp/base.txz      100% of   59 MB 1470 kBps 00m42s
====>> Extracting base.txz... done
====>> Fetching src.txz for FreeBSD 10.0-RELEASE amd64
/poudriere/jails/10amd64/fromftp/src.txz       100% of  107 MB 1476 kBps 01m14s
====>> Extracting src.txz... done
====>> Fetching games.txz for FreeBSD 10.0-RELEASE amd64
/poudriere/jails/10amd64/fromftp/games.txz     100% of  865 kB  734 kBps 00m01s
====>> Extracting games.txz... done
====>> Fetching lib32.txz for FreeBSD 10.0-RELEASE amd64
/poudriere/jails/10amd64/fromftp/lib32.txz     100% of   14 MB 1316 kBps 00m12s
====>> Extracting lib32.txz... done
====>> Cleaning up... done
====>> Jail 10amd64 10.0-RELEASE amd64 is ready to be used
```

```
# poudriere ports -c -p  local
====>> Creating local fs... done
====>> Extracting portstree "local"...
Looking up portsnap.FreeBSD.org mirrors... 7 mirrors found.
Fetching public key from ec2-eu-west-1.portsnap.freebsd.org... done.
Fetching snapshot tag from ec2-eu-west-1.portsnap.freebsd.org... done.
Fetching snapshot metadata... done.
Fetching snapshot generated at Tue Feb 11 01:07:15 CET 2014:
94a3431f0ce567f6452ffde4fd3d7d3c6e1da143efec76100% of   69 MB 1246 kBps 00m57s
Extracting snapshot... done.
Verifying snapshot integrity... done.
Fetching snapshot tag from ec2-eu-west-1.portsnap.freebsd.org... done.
Fetching snapshot metadata... done.
Updating from Tue Feb 11 01:07:15 CET 2014 to Tue Feb 11 16:05:20 CET 2014.
Fetching 4 metadata patches... done.
Applying metadata patches... done.
Fetching 0 metadata files... done.
Fetching 48 patches.
(48/48) 100.00%  done.
done.
Applying patches...
done.
Fetching 1 new ports or files... done.
/poudriere/ports/tester/CHANGES
/poudriere/ports/tester/COPYRIGHT

[...-]

Building new INDEX files... done.
```

poudriere kann auf einem einzelnen Rechner Ports mit mehreren Konfigurationen bauen, in mehreren Jails und aus unterschiedlichen Ports-Sammlungen. Spezifische Konfigurationen für diese Kombinationen werden *Sets*

genannt. Lesen Sie den Abschnitt CUSTOMIZATION in poudriere(8) für weitere Einzelheiten nach der Installation von port-mgmt/poudriere oder ports-mgmt/poudriere-devel.

Die hier gezeigte Konfiguration verwendet eine einzelne Jail-, Port- und Set-spezifische `make.conf` in `/usr/local/etc/poudriere.d`. Der verwendete Dateiname in diesem Beispiel wird aus einer Kombination von Jailnamen, Portnamen und Setnamen zusammen gesetzt: *`10amd64-local-workstation`*`-make.conf`. Die `make.conf` des Systems und diese neue Datei werden verwendet, um die `make.conf` für die Jail zu erzeugen.

Die zu bauenden Pakete werden in *`10amd64-local-workstation`*`-pkglist` eingetragen:

```
editors/emacs
devel/git
ports-mgmt/pkg
...
```

Die Optionen und Abhängigkeiten für die Ports werden wie folgt konfiguriert:

```
# poudriere options -j 10amd64 -p local -z workstation -f 10amd64-local-workstation-pkglist
```

Schließlich werden die Pakete gebaut und ein Paket-Repository erstellt:

```
# poudriere bulk -j 10amd64 -p local -z workstation -f 10amd64-local-workstation-pkglist
```

Ctrl+t zeigt den aktuellen Status des Baus an. Poudriere speichert zudem Dateien in `/poudriere/logs/bulk/jailname`. Diese Dateien kann ein Webserver nutzen, um Informationen über den Bau anzuzeigen.

Die Pakete stehen jetzt im poudriere Repository für die Installation zur Verfügung.

Weitere Informationen zu poudriere finden Sie in poudriere(8) und unter https://github.com/freebsd/poudriere/wiki.

5.6.2. Konfiguration des pkg-Clients für das Poudriere Repository

Obwohl es möglich ist ein eigenes Repository zusammen mit dem offiziellen Repository zu nutzen, ist es manchmal sinnvoll das offizielle Repository zu deaktivieren. Dazu wird eine Konfigurationsdatei erstellt, welche die offizielle Konfigurationsdatei überschreibt. Erzeugen Sie dazu `/usr/local/etc/pkg/repos/FreeBSD.conf` mit dem folgenden Inhalt:

```
FreeBSD: {
        enabled: no
}
```

Am einfachsten ist es, das poudriere Repository über HTTP zur Verfügung zu stellen. Setzen Sie einen Webserver auf, der die Dateien des Paketverzeichnisses ausliefert, zum Beispiel `/usr/local/poudriere/data/packages/`*`10amd64`*`.10amd64` bezeichnet dabei den Namen des Baus.

Wenn die URL des Paket Repositories `http://pkg.example.com/10amd64` ist, dann sollte die Konfiguration des Repositories in `/usr/local/etc/pkg/repos/custom.conf` wie folgt aussehen:

```
custom: {
 url: "http://pkg.example.com/10amd64 ",
 enabled: yes,
}
```

5.7. Nach der Installation

Unabhängig davon, ob die Software aus einem binären Paket oder aus einem Port installiert wird, benötigen die meisten Anwendungen von Drittanbietern ein gewisses Maß an Konfiguration, nachdem sie installiert wurden. Die folgenden Kommandos und Speicherorte helfen Ihnen dabei festzustellen, was mit der Anwendung zusammen installiert wurde.

- Die meisten Anwendungen installieren mindestens eine Konfigurationsdatei nach `/usr/local/etc` . Falls die Anwendung viele Konfigurationsdateien enthält, wird ein Unterverzeichnis erstellt um die Dateien zu speichern. Oft werden die Konfigurationsdateien mit einem Suffix wie beispielsweise `.sample` installiert. Die Konfigurationsdateien sollten überprüft und ggf. bearbeitet werden, um die Anforderungen des Systems zu erfüllen. Um eine Konfigurationsdatei zu bearbeiten, kopieren Sie diese zunächst ohne die Erweiterung `.sample`.
- Wenn die Anwendung Dokumentation zur Verfügung stellt, wird diese nach `/usr/local/share/doc` installiert. Viele Anwendungen installieren auch Manualpages. Diese Dokumentation sollten Sie lesen, bevor Sie fortfahren.
- Einige Anwendungen laufen als Dienst und müssen vor dem ersten Start in `/etc/rc.conf` eingetragen werden. Diese Anwendungen installieren meist ein Skript in `/usr/local/etc/rc.d` . Weitere Informationen finden Sie im Abschnitt 12.2, „Start von Diensten“.
- Benutzer der csh(1) sollten `rehash` ausführen, um die neu installierten Programme nutzen zu können.
- Benutzen Sie `pkg info`, um die Dateien, Manualpages und Binaries zu ermitteln, die mit der Anwendung installiert wurden.

5.8. Kaputte Ports

Wenn sich ein Port nicht bauen oder installieren lässt, versuchen Sie folgendes:

1. Stellen Sie fest, ob die Datenbank mit den Problemberichten bereits einen Lösungsvorschlag enthält. Ist dies der Fall, kann die vorgeschlagene Lösung getestet werden.
2. Bitten Sie den Betreuer des Ports um Hilfe. Geben Sie dazu `make maintainer` ein oder lesen Sie das `Makefile` im Verzeichnis des Ports, um an die E-Mail-Adresse zu kommen. Vergessen Sie nicht die Zeile mit `$FreeBSD:` aus dem `Makefile` und die Ausgabe bis zur Fehlermeldung mitzuschicken.

   ### Anmerkung

   Einige Ports werden nicht von einer Einzelperson, sondern von einer Mailingliste betreut. Viele (aber nicht alle) dieser Adressen haben die Form `<freebsd-NameDerListe@FreeBSD.org>`. Denken Sie daran, wenn Sie Ihre Fragen formulieren.

   Dies gilt insbesondere für Ports, die als Betreuer den Eintrag `<ports@FreeBSD.org>` aufweisen. Derartige Ports haben überhaupt keinen Betreuer. Korrekturen und Unterstützung kommen daher nur von Personen, die diese Mailingliste abonniert haben. Gerade in diesem Bereich werden jederzeit zusätzliche freiwillige Helfer benötigt!

   Erhalten Sie auf Ihre Anfrage keine Antwort, benutzen Sie Bugzilla, um einen Problembericht zu erstellen. Bevor Sie einen solchen Bericht erstellen, lesen Sie den Artikel Writing FreeBSD Problem Reports.
3. Reparieren Sie ihn! Das FreeBSD Porter's Handbook enthält eine detaillierte Beschreibung des Portsystems. Damit sind Sie in der Lage, einen zeitweilig kaputten Port zu reparieren oder einen eigenen Port zu erstellen.
4. Installieren Sie das Paket anstelle des Ports. Anweisungen hierzu finden Sie in Abschnitt 5.4, „Benutzen von pkg zur Verwaltung von Binärpaketen“.

Kapitel 6. Das X-Window-System

Übersetzt von Martin Heinen.

6.1. Übersicht

Bei einer Installation von FreeBSD mit bsdinstall wird nicht automatisch eine grafische Benutzeroberfläche installiert. Dieses Kapitel beschreibt die Installation und Konfiguration von Xorg, das eine grafische Umgebung über das quelloffene X-Window-System zur Verfügung stellt. Weiterhin wird beschrieben, wie Sie eine Desktop-Umgebung oder einen Window Manager finden und installieren können.

Anmerkung

Benutzer die eine Installationsmethode bevorzugen, welche automatisch Xorg konfiguriert und zudem die Auswahl eines Window Managers während der Installation anbietet, sollten sich die pcbsd.org Webseite ansehen.

Weitere Informationen über Video-Hardware, die von Xorg unterstützt wird, finden Sie auf der x.org Webseite.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie

- Die Komponenten des X-Window-Systems und ihr Zusammenspiel kennen.
- Wissen, wie Xorg installiert und konfiguriert wird.
- Wissen, wie verschiedene Window-Manager und Desktop-Umgebungen installiert und konfiguriert werden.
- Wissen, wie TrueType®-Schriftarten mit Xorg benutzt werden.
- Wissen, wie Sie die grafische Anmeldung (XDM) einrichten.

Bevor Sie dieses Kapitel lesen, sollten Sie

- Wissen, wie Sie Software Dritter, wie in Kapitel 5, *Installieren von Anwendungen: Pakete und Ports* beschrieben, installieren.

6.2. Terminologie

Obwohl es nicht nötig ist, alle Details der verschiedenen Komponenten des X Window Systems und deren Zusammenspiel zu kennen, kann es trotzdem nützlich sein die Grundlagen dieser Komponenten zu verstehen:

X-Server
: X wurde von Anfang an netzwerktransparent entworfen und verwendet ein „Client-Server-Modell“. In diesem Modell läuft der „X-Server“ auf dem Rechner, an dem die Tastatur, der Bildschirm und die Maus angeschlossen ist. Der Server ist für Dinge wie die Verwaltung des Bildschirms und die Verarbeitung von Tastatur- und Maus-Eingaben sowie anderer Ein- und Ausgabegeräte, wie beispielsweise ein Tablet oder ein Videoprojektor, verantwortlich. Dieses Modell verwirrt viele Leute, die erwarten, dass der „X-Server“ der leistungsstarke Rechner im Maschinenraum und der „X-Client“ ihr Arbeitsplatzrechner ist.

X-Client
: Jede X-Anwendung, wie beispielsweise XTerm oder Firefox ist ein „X-Client“. Der Client sendet dem Server Nachrichten wie „Zeichne an diesen Koordinaten ein Fenster“ und der Server sendet dem Client Nachrichten der Art „Der Benutzer hat gerade den Ok-Knopf gedrückt“.

In kleinen Umgebungen laufen der X-Server und die X-Clients auf demselben Rechner. Es ist auch möglich, den X-Server auf einem weniger leistungsfähigen Rechner laufen zu lassen und die X-Anwendungen auf einem leistungsfähigeren Rechner zu betreiben. In diesem Fall kommunizieren der X-Server und die X-Clients über das Netzwerk.

Window-Manager
: X schreibt nicht vor, wie Fenster auf dem Bildschirm auszusehen haben, wie sie mit der Maus zu verschieben sind, welche Tastenkombinationen benutzt werden sollen um zwischen den Fenstern zu wechseln, wie die Fensterrahmen aussehen, oder ob diese Schaltflächen zum schließen haben. Stattdessen gibt X die Verantwortung für all diese Sachen an eine separate *Window-Manager* Anwendung ab. Es stehen `zahlreiche Window-Manager` zur Verfügung. Jeder Window-Manager bietet ein anderes Erscheinungsbild: einige unterstützen virtuelle Bildschirme, andere erlauben Tastenkombinationen zur Verwaltung des Bildschirms. Einige besitzen eine „Start“ Schaltfläche und in manchen lässt sich das Aussehen und Verhalten der Anwendung über *Themes* beliebig einstellen. Window-Manager stehen in der Kategorie `x11-wm` der Ports-Sammlung zur Verfügung.

  Jeder Window-Manager wird unterschiedlich konfiguriert. Einige erwarten eine manuell erstellte Konfigurationsdatei, während andere ein grafisches Werkzeug für die meisten Konfigurationsarbeiten anbieten.

Desktop-Umgebungen
: KDE und GNOME werden als Desktop-Umgebungen bezeichnet, da sie eine ganze Reihe von Anwendungen für typische Desktop-Aufgaben enthalten. Dazu zählen beispielsweise Office-Pakete, Webbrowser und Spiele.

Fokus
: Der Window-Manager ist für die Methode verantwortlich, mit der ein Fenster den Fokus bekommt. Jedes System, das Fenster verwendet muss entscheiden, wie ein Fenster aktiviert wird, damit es Eingaben empfangen kann. Das aktive Fenster sollte zudem sichtbar gekennzeichnet werden.

  Eine Methode wird „click-to-focus“ genannt. Ein Fenster wird aktiv, wenn es mit der Maus angeklickt wird. Eine weitere Methode ist „focus-follows-mouse“. Hier hat liegt der Fokus auf dem Fenster, auf dem sich der Mauszeiger befindet. Wird der Mauszeiger in ein anderes Fenster bewegt, so erhält dieses Fenster den Fokus. Eine dritte Methode ist „sloppy-focus“. Hier wechselt der Fokus nur dann, wenn sich der Mauszeiger in ein neues Fenster bewegt und nicht, wenn er das aktive Fenster verlässt. Ist der Mauszeiger auf der Desktop Oberfläche, so bleibt der Fokus auf dem zuletzt verwendeten Fenster. Bei der Methode „click-to-focus“ wird das aktive Fenster durch einen Mausklick festgelegt. Dabei kann das Fenster vor alle anderen Fenster gesetzt werden. Alle Eingaben werden dann, unabhängig von der Position des Mauszeigers, dem aktiven Fenster zugeordnet.

  Die verschiedenen Window-Manager unterstützen noch andere Methoden. Alle unterstützen jedoch „click-to-focus“ und die meisten von ihnen auch die anderen Methoden. Lesen Sie die Dokumentation des Window-Managers um festzustellen, welche Methoden zur Verfügung stehen.

Widgets
: *Widget* bezeichnet Objekte, die in irgendeiner Weise geklickt oder manipuliert werden können. Dazu gehören *buttons* (Schaltflächen), *check buttons* (Schaltfläche für Mehrfachauswahlen), *radio buttions* (Schaltfläche für Einfachauswahlen), Icons und Auswahllisten. Eine Widget-Sammlung ist eine Reihe von Widgets, die verwendet werden um grafische Anwendungen zu erstellen. Es gibt mehrere populäre Widget-Sammlungen, einschließlich Qt, das von KDE benutzt wird, und GTK+, das von GNOME benutzt wird. Als Folge dessen, haben Anwendungen einen bestimmten *look and feel*, je nachdem welche Widget-Sammlung benutzt wurde, um die Anwendung zu erstellen.

6.3. Xorg installieren

In FreeBSD kann Xorg als Paket oder Port installiert werden.

Die nachstehenden Kommandos bauen und installieren Xorg aus der Ports-Sammlung:

```
# cd /usr/ports/x11/xorg
```

```
# make install clean
```

Die Installation des Pakets ist zwar schneller, dafür können weniger Optionen angepasst werden:

```
# pkg install xorg
```

Bei beiden Vorgehensweisen wird ein vollständiges Xorg-System installiert. Dies ist die beste Option für die meisten Anwender.

Eine kleinere Version des Xorg-Systems für erfahrene Anwender ist mit x11/xorg-minimal verfügbar. Die meisten Dokumente, Bibliotheken und Anwendungen werden hierbei nicht installiert. Einige Anwendungen erfordern jedoch diese zusätzlichen Komponenten, um ordnungsgemäß zu funktionieren.

6.4. Xorg konfigurieren

Warren Block
Björn Heidotting

6.4.1. Schnellstartanleitung

Xorg unterstützt die meisten gängigen Grafikkarten, Tastaturen und Zeigegeräte. Diese Geräte werden automatisch erkannt und müssen nicht manuell konfiguriert werden.

1. Wenn Xorg bereits zuvor auf diesem Computer verwendet wurde, verschieben oder entfernen Sie alle vorhandenen Konfigurationsdateien:

   ```
   # mv /etc/X11/xorg.conf ~/xorg.conf.etc
   # mv /usr/local/etc/X11/xorg.conf ~/xorg.conf.localetc
   ```

2. Fügen Sie die Benutzer, die Xorg verwenden, zur Gruppe `video` oder `wheel` hinzu, um die 3D-Beschleunigung zu aktivieren. Um den Benutzer *`jru`* in eine der verfügbaren Gruppen hinzuzufügen:

   ```
   # pw groupmod video -m jru || pw groupmod wheel -m jru
   ```

3. Der Window-Manager TWM ist standardmäßig enthalten und wird auch gestartet, wenn Xorg startet:

   ```
   % startx
   ```

4. Auf einigen älteren Versionen von FreeBSD muss die Systemkonsole auf vt(4) eingestellt sein, damit der Wechsel auf die Konsole ordnungsgemäß funktioniert. Informationen dazu finden Sie im Abschnitt 6.4.3, „Kernel Mode Setting (KMS)“.

6.4.2. Benutzergruppen für Grafikbeschleunigung

Um die 3D-Beschleunigung für Grafikkarten zu ermöglichen, ist der Zugriff auf `/dev/dri` notwendig. In der Regel ist es am einfachsten, die Benutzer zur Gruppe `video` oder `wheel` hinzuzufügen. In diesem Beispiel wird pw(8) verwendet, um den Benutzer *`slurms`* zu der Gruppe `video` hinzuzufügen, bzw. zur Gruppe `wheel`, falls die Gruppe `video` nicht existiert:

```
# pw groupmod video -m slurms || pw groupmod wheel -m slurms
```

6.4.3. Kernel Mode Setting (KMS)

Wenn der Computer die Anzeige von der Konsole auf eine höhere Bildschirmauflösung für X umstellt, muss der Videoausgabe-Modus eingestellt werden. Neuere Versionen von Xorg verwenden dazu ein System innerhalb des Kernels, um diesen Modus effizienter zu ändern. Ältere Versionen von FreeBSD verwenden dafür sc(4), welches jedoch nicht mit dem KMS-System umgehen kann. Das führt dazu, dass nach dem Schließen von X die Konsole leer bleibt, obwohl sie weiterhin funktioniert. Die neuere vt(4) Konsole vermeidet dieses Problem.

Fügen Sie diese Zeile in `/boot/loader.conf` ein um vt(4) zu aktivieren:

```
kern.vty=vt
```

6.4.4. Konfigurationsdateien

6.4.4.1. Verzeichnis

Xorg sucht in verschiedenen Verzeichnissen nach Konfigurationsdateien. Unter FreeBSD ist `/usr/local/etc/X11/` das bevorzugte Verzeichnis für diese Dateien. Die Verwendung dieses Verzeichnisses hilft dabei, Anwendungsdateien vom Betriebssystem getrennt zu halten.

Das Speichern von Konfigurationsdateien unter `/etc/X11/` funktioniert immer noch, allerdings vermischt diese Methode Anwendungsdateien mit Dateien des Basissystems und wird daher nicht empfohlen.

6.4.4.2. Einzelne oder mehrere Dateien

Anstatt die traditionelle `xorg.conf` zu verwenden, ist es einfacher, mehrere Dateien, die jeweils eine bestimmte Einstellung konfigurieren, zu verwenden. Diese Dateien werden im Unterverzeichnis `xorg.conf.d/` des Hauptverzeichnisses gespeichert. Der vollständige Pfad ist normalerweise `/usr/local/etc/X11/xorg.conf.d/` .

Beispiele für diese Dateien werden später in diesem Abschnitt vorgestellt.

Die traditionelle, einzelne `xorg.conf` funktioniert weiterhin, ist jedoch nicht so übersichtlich und flexibel wie die Verwendung von mehreren Dateien im Unterverzeichnis `xorg.conf.d/`.

6.4.5. Grafikkarten

Intel®
: 3D-Beschleunigung wird von den meisten Intel®-Grafikkarten unterstützt, einschließlich Ivy Bridge (HD Graphics 2500, 4000 und P4000), Iron Lake (HD Graphics) und Sandy Bridge (HD Graphics 2000).

  Treibername: `intel`

  Weitere Informationen finden Sie unter https://en.wikipedia.org/wiki/List_of_Intel_graphics_processing_units.

AMD® Radeon
: 2D- und 3D-Beschleunigung wird von den meisten Radeon-Karten bis zur HD6000-Serie unterstützt.

  Treibername: `radeon`

  Weitere Informationen finden Sie unter https://en.wikipedia.org/wiki/List_of_AMD_graphics_processing_units.

NVIDIA
: Verschiedene NVIDIA Treiber sind in der Kategorie `x11` der Ports-Sammlung enthalten. Installieren Sie den Treiber, der für die Grafikkarte benötigt wird.

  Weitere Informationen finden Sie unter https://en.wikipedia.org/wiki/List_of_Nvidia_graphics/processing_units.

Hybride Kombinationen
: Einige Notebooks besitzen zusätzlich zum Chipsatz oder Prozessor einen Grafikprozessor. *Optimus* kombiniert Intel® und NVIDIA Hardware. *Umschaltbare Grafik* bzw. *Hybride Grafik* ist eine Kombination aus Intel®, oder AMD® Prozessor mit AMD® Radeon GPU.

  Die Implementierungen dieser Hybrid-Grafik-Systeme variieren und Xorg in FreeBSD ist nicht in der Lage, alle Versionen der Hardware zu betreiben.

Einige Computer bieten jedoch eine BIOS-Option, um eine der beiden Grafikkarten zu deaktivieren oder den *diskreten* Modus einzuschalten. Zum Beispiel ist es manchmal möglich, die NVIDIA GPU in einem Optimus-System zu deaktivieren. Intel® Video kann dann mit einem Intel® Treiber verwendet werden.

Die BIOS-Einstellungen sind abhängig vom Modell des Computers. In manchen Situationen können beide GPUs aktiviert bleiben. Um solch ein System lauffähig zu machen genügt es bereits, nur die Haupt-GPU im Abschnitt `Device` der Konfigurationsdatei zu setzen.

Andere Grafikkarten
: Treiber für weniger gebräuchliche Grafikkarten finden Sie in der Kategorie `x11-drivers` der Ports-Sammlung.

Karten, die nicht durch einen speziellen Treiber unterstützt werden, sind vielleicht noch mit dem Treiber x11-drivers/xf86-video-vesa nutzbar. Dieser Treiber wird von x11/xorg installiert. Der Treiber kann auch manuell als x11-drivers/xf86-video-vesa installiert werden. Xorg versucht immer diesen Treiber zu verwenden, wenn für die Grafikkarte kein passender Treiber gefunden wird.

x11-drivers/xf86-video-scfb ist ein ähnlicher Treiber, der mit vielen UEFI und ARM® Computern funktioniert.

Video-Treiber über eine Datei einstellen
: Den Intel® Treiber in einer Konfigurationsdatei einstellen:

Beispiel 6.1. Den Intel® Treiber über eine Datei auswählen

`/usr/local/etc/X11/xorg.conf.d/driver-intel.conf`

```
Section "Device"
 Identifier  "Card0"
 Driver      "intel"
 # BusID     "PCI:1:0:0"
EndSection
```

Wenn mehr als eine Grafikkarte vorhanden ist, kann der Eintrag `BusID` verwendet werden, um die gewünschte Karte auszuwählen. Eine Liste der `BusID`s der Grafikkarten kann mit `pciconf -lv | grep -B3 display` ausgegeben werden.

Den Radeon Treiber in einer Konfigurationsdatei einstellen:

Beispiel 6.2. Den Radeon Treiber über eine Datei auswählen

`/usr/local/etc/X11/xorg.conf.d/driver-radeon.conf`

```
Section "Device"
 Identifier  "Card0"
 Driver      "radeon"
EndSection
```

Den VESA Treiber in einer Konfigurationsdatei einstellen:

Beispiel 6.3. Den VESA Treiber über eine Datei auswählen

`/usr/local/etc/X11/xorg.conf.d/driver-vesa.conf`

```
Section "Device"
 Identifier  "Card0"
 Driver      "vesa"
EndSection
```

6.4.6. Monitore

Fast alle Monitore unterstützen den Extended Display Identification Data Standard (EDID). Xorg verwendet EDID um mit dem Monitor zu kommunizieren und die unterstützten Auflösungen und Bildwiederholfrequenzen zu erkennen. Xorg wählt dann die für den Monitor am besten geeignete Kombination von Einstellungen.

Weitere vom Monitor unterstützte Auflösungen, können in der Konfigurationsdatei, oder nach dem Start des X-Servers mit xrandr(1) gesetzt werden.

xrandr(1) benutzen
: Führen Sie xrandr(1) ohne Parameter aus, um eine Liste von Video-Ausgängen und erkannten Monitor-Modi zu sehen:

```
% xrandr
Screen 0: minimum 320 x 200, current 3000 x 1920, maximum 8192 x 8192
DVI-0 connected primary 1920x1200+1080+0 (normal left inverted right x axis y axis) ↺
495mm x 310mm
   1920x1200     59.95*+
   1600x1200     60.00
   1280x1024     85.02    75.02    60.02
   1280x960      60.00
   1152x864      75.00
   1024x768      85.00    75.08    70.07    60.00
   832x624       74.55
   800x600       75.00    60.32
   640x480       75.00    60.00
   720x400       70.08
DisplayPort-0 disconnected (normal left inverted right x axis y axis)
HDMI-0 disconnected (normal left inverted right x axis y axis)
```

Die Auflistung zeigt, dass der `DVI-0` Ausgang benutzt wird, um eine Bildschirmauflösung von 1920x1200 bei einer Bildwiederholrate von 60 Hz anzuzeigen. An den Anschlüssen `DisplayPort-0` und `HDMI-0` sind keine Monitore angeschlossen.

Die anderen Anzeigemodi können mit xrandr(1) ausgewählt werden. Um beispielsweise auf 1280x1024 bei 60 Hz umzuschalten:

```
% xrandr --mode 1280x1024 --rate 60
```

Häufig wird für einen Videoprojektor der externe Videoausgang eines Notebooks verwendet.

Die Typen und Anzahl der Videoanschlüsse variiert zwischen den Geräten und auch die Ausgabe variiert von Treiber zu Treiber. Was für den einen Treiber `HDMI-1` ist, nennt ein anderer Treiber vielleicht `HDMI1`. Führen Sie daher zunächst xrandr(1) aus, um alle verfügbaren Anschlüsse aufzulisten.

```
% xrandr
Screen 0: minimum 320 x 200, current 1366 x 768, maximum 8192 x 8192
LVDS1 connected 1366x768+0+0 (normal left inverted right x axis y axis) 344mm x 193mm
   1366x768      60.04*+
   1024x768      60.00
   800x600       60.32    56.25
   640x480       59.94
VGA1 connected (normal left inverted right x axis y axis)
   1280x1024     60.02 +  75.02
   1280x960      60.00
   1152x864      75.00
```

```
   1024x768      75.08    70.07    60.00
   832x624       74.55
   800x600       72.19    75.00    60.32    56.25
   640x480       75.00    72.81    66.67    60.00
   720x400       70.08
HDMI1 disconnected (normal left inverted right x axis y axis)
DP1 disconnected (normal left inverted right x axis y axis)
```

Vier Ausgänge wurden gefunden: das integrierte Panel `LVDS1`, sowie die externen Anschlüsse `VGA1`, `HDMI1` und `DP1`.

Der Videoprojektor wurde am Ausgang `VGA1` angeschlossen. xrandr(1) wird nun verwendet, um diese Ausgabe auf die native Auflösung des Projektors einzustellen und den zusätzlichen Platz auf der rechten Seite des Desktops hinzuzufügen:

```
% xrandr --output VGA1 --auto --right-of LVDS1
```

`--auto` wählt die Auflösung und Aktualisierungsrate die von EDID ermittelt wurden. Wenn die Auflösung nicht richtig ermittelt wurde, kann ein fester Wert mit `--mode` anstelle von `--auto` angegeben werden. Beispielsweise können die meisten Projektoren mit einer Auflösung von 1024x768 betrieben werden, die mit `--mode 1024x768` gesetzt wird.

xrandr(1) wird häufig aus `.xinitrc` ausgeführt, um den entsprechenden Modus zu setzen wenn X startet.

Bildschirmauflösung über eine Datei einstellen
: Eine Bildschirmauflösung von 1024x768 in einer Konfigurationsdatei einstellen:

Beispiel 6.4. Die Bildschirmauflösung in eine Datei schreiben

`/usr/local/etc/X11/xorg.conf.d/screen-resolution.conf`

```
Section "Screen"
 Identifier "Screen0"
 Device     "Card0"
 SubSection "Display"
 Modes      "1024x768"
 EndSubSection
EndSection
```

Die wenigen Monitore, die EDID nicht beherrschen, können durch setzen von `HorizSync` und `VertRefresh` auf den Bereich der vom Monitor unterstützten Frequenzen konfiguriert werden.

Beispiel 6.5. Manuelles Einstellen der Monitorfrequenzen

`/usr/local/etc/X11/xorg.conf.d/monitor0-freq.conf`

```
Section "Monitor"
 Identifier   "Monitor0"
 HorizSync    30-83  # kHz
 VertRefresh  50-76  # Hz
EndSection
```

6.4.7. Eingabegeräte

6.4.7.1. Tastaturen

Tastaturlayout
: Die standardisierte Position von Tasten auf einer Tastatur wird als *Layout* bezeichnet. Layouts und andere einstellbare Parameter werden in xkeyboard-config(7) beschrieben.

  In der Voreinstellung ist ein US-amerikanisches Layout aktiv. Um ein alternatives Layout zu wählen, setzen Sie die Optionen `XkbLayout` und `XkbVariant` in der Klasse `InputClass`. Dies wird für alle Eingabegeräte der entsprechenden Klasse angewendet werden.

Beispiel 6.6. Konfiguration eines Tastaturlayouts

`/usr/local/etc/X11/xorg.conf.d/keyboard-fr-oss.conf`

```
Section "InputClass"
 Identifier "KeyboardDefaults"
 Driver  "keyboard"
 MatchIsKeyboard "on"
 Option  "XkbLayout" "fr"
 Option  "XkbVariant" "oss"
EndSection
```

Beispiel 6.7. Konfiguration mehrerer Tastaturlayouts

Hier werden die Tastaturlayouts für Vereinigte Staaten, Spanien und Ukraine gesetzt. Mit Alt+Shift können Sie zwischen den einzelnen Layouts wechseln. Für eine verbesserte Steuerung des Layouts kann x11/xxkb oder x11/sbxkb benutzt werden.

`/usr/local/etc/X11/xorg.conf.d/kbd-layout-multi.conf`

```
Section "InputClass"
 Identifier "All Keyboards"
 MatchIsKeyboard "yes"
 Option  "XkbLayout" "us,es,ua"
EndSection
```

Xorg über die Tastatur beenden
: X kann über eine Tastenkombination geschlossen werden. Standardmäßig ist die Tastenkombination jedoch nicht gesetzt, da sie mit Tastaturbefehlen für einige Anwendungen in Konflikt steht. Die Aktivierung dieser Option erfordert Änderungen in der Sektion `InputDevice` für die Tastatur:

Beispiel 6.8. X über die Tastatur beenden

`/usr/local/etc/X11/xorg.conf.d/keyboard-zap.conf`

```
Section "InputClass"
 Identifier "KeyboardDefaults"
 Driver  "keyboard"
 MatchIsKeyboard "on"
 Option  "XkbOptions" "terminate:ctrl_alt_bksp"
```

```
EndSection
```

6.4.7.2. Mäuse und Zeigegeräte

Viele Parameter für die Maus können über Konfigurationseinstellungen eingestellt werden. mousedrv(4) enthält eine vollständige Liste.

Mauszeiger
: Die Anzahl der Maustasten wird in `xorg.conf` im Abschnitt `InputDevice` für die Maus festgelegt. Um die Anzahl der Tasten auf 7 zu setzen:

Beispiel 6.9. Die Anzahl der Maustasten festlegen

`/usr/local/X11/xorg.conf.d/mouse0-buttons.conf`

```
Section "InputDevice"
 Identifier  "Mouse0"
 Option      "Buttons" "7"
EndSection
```

6.4.8. Manuelle Konfiguration

In einigen Fällen funktioniert die Autokonfiguration nicht mit bestimmter Hardware, oder es wird eine andere Konfiguration benötigt. Für diese Fälle kann eine benutzerdefinierte Konfigurationsdatei erstellt werden.

Eine Konfigurationsdatei kann, basierend auf der von Xorg erfassten Hardware erzeugt werden. Diese Konfigurationsdatei ist ein guter Ausgangspunkt für angepasste Konfigurationen.

Erzeugung einer `xorg.conf`:

```
# Xorg -configure
```

Die Konfigurationsdatei wird in `/root/xorg.conf.new` gespeichert. Machen Sie alle gewünschten Änderungen an dieser Datei. Danach testen Sie die Datei mit:

```
# Xorg -config /root/xorg.conf.new
```

Nachdem die neue Konfiguration angepasst und getestet wurde, kann die Konfiguration in kleinere Dateien unter `/usr/local/etc/X11/xorg.conf.d/` aufgeteilt werden.

6.5. Schriftarten in Xorg benutzen

6.5.1. Type 1 Schriftarten

Die Schriftarten, die mit Xorg ausgeliefert werden, eignen sich ganz und gar nicht für Desktop-Publishing-Anwendungen. Große Schriftarten zeigen bei Präsentationen deutliche Treppenstufen und kleine Schriftarten sind fast unleserlich. Es gibt allerdings mehrere hochwertige Type 1 Schriftarten (PostScript®), die mit Xorg benutzt werden können. Beispielsweise enthalten die URW-Schriftarten (x11-fonts/urwfonts) hochwertige Versionen gängiger Type 1 Schriftarten (unter anderem Times Roman®, Helvetica®, Palatino®). Die Sammlung Freefonts (x11-fonts/freefonts) enthält viele weitere Schriftarten, doch sind diese für den Einsatz in Grafikprogrammen wie Gimp gedacht und nicht für den alltäglichen Gebrauch. Weiterhin kann Xorg mit einem Minimum an Aufwand konfiguriert werden, damit TrueType®-Schriftarten benutzt werden können. Mehr dazu erfahren Sie in der Manualpage X(7) und im Abschnitt 6.5.2, „TrueType®-Schriftarten“.

Die Type 1 Schriftarten lassen sich aus der Ports-Sammlung wie folgt installieren:

```
# cd /usr/ports/x11-fonts/urwfonts
# make install clean
```

Analog lassen sich Freefont und andere Sammlungen installieren. Damit der X-Server diese Schriftarten erkennt, fügen Sie eine entsprechende Zeile in die Konfigurationsdatei des X-Servers (`/etc/X11/xorg.conf`) hinzu:

```
FontPath "/usr/local/share/fonts/urwfonts/"
```

Alternativ kann in der X-Sitzung das folgende Kommando abgesetzt werden:

```
% xset fp+ /usr/local/share/fonts/urwfonts
% xset fp rehash
```

Jetzt kennt der X-Server die neuen Schriftarten, jedoch nur bis zu Ende der Sitzung. Soll die Änderung dauerhaft sein, müssen die Befehle in `~/.xinitrc` eingetragen werden, wenn X mittels `startx` gestartet wird, beziehungsweise in `~/.xsession`, wenn ein grafischer Login-Manager, wie XDM verwendet wird. Eine dritte Möglichkeit besteht darin, `/usr/local/etc/fonts/local.conf` zu verwenden, was im Abschnitt 6.5.3, „Anti-aliasing“ demonstriert wird.

6.5.2. TrueType®-Schriftarten

Xorg besitzt eine eingebaute Unterstützung zur Darstellung von TrueType®-Schriftarten. Hierzu existieren zwei verschiedene Module, die diese Funktionalität aktivieren können. In diesem Beispiel wird das Freetype-Modul benutzt, da es besser mit anderen Werkzeugen, die TrueType®-Schriftarten darstellen, übereinstimmt. Um das Freetype-Modul zu aktivieren, muss die folgende Zeile zum Abschnitt `"Module"` in `/etc/X11/xorg.conf` hinzugefügt werden.

```
Load "freetype"
```

Erstellen Sie ein Verzeichnis für die TrueType®-Schriftarten (beispielsweise `/usr/local/share/fonts/TrueType`) und kopieren Sie alle Schriftarten dorthin. Beachten Sie, dass die Schriftarten für Xorg im UNIX®/MS-DOS®/Windows®-Format vorliegen müssen und nicht direkt von einem Apple® Mac® übernommen werden können. Sobald die Dateien in das Verzeichnis kopiert wurden, verwenden Sie mkfontdir um `fonts.dir` zu erstellen, damit X weiß, dass diese neuen Dateien installiert wurden. `mkfontdir` kann als Paket installiert werden:

```
# pkg install mkfontdir
```

Erstellen Sie dann einen Index der Schriftarten für X:

```
# cd /usr/local/share/fonts/TrueType
# mkfontdir
```

Geben Sie dem System das TrueType®-Verzeichnis, wie im Abschnitt 6.5.1, „Type 1 Schriftarten“ beschrieben, bekannt:

```
# xset fp+ /usr/local/share/fonts/TrueType
# xset fp rehash
```

Oder fügen Sie eine `FontPath`-Zeile in `xorg.conf` ein.

Jetzt sollten Gimp, Apache OpenOffice und alle anderen X-Anwendungen die TrueType®-Schritarten erkennen. Extrem kleine Schriftarten (Webseiten, die mit hoher Auflösung betrachtet werden) und sehr große Schriftarten (in StarOffice™) werden jetzt viel besser aussehen.

6.5.3. Anti-aliasing

Alle Schriftarten in Xorg, die in den Verzeichnissen `/usr/local/share/fonts/` und `~/.fonts/` gefunden werden, werden automatisch für Anti-aliasing an Anwendungen zur Verfügung gestellt, die Xft beherrschen. Die meisten aktuellen Anwendungen beherrschen Xft, dazu gehören auch KDE, GNOME und Firefox.

In der Datei `/usr/local/etc/fonts/local.conf` werden die Schriftarten, die mit dem Anti-aliasing-Verfahren benutzt werden sollen und die Eigenschaften des Verfahrens festgelegt. In diesem Abschnitt wird nur die grundlegende Konfiguration von Xft beschrieben. Weitere Details entnehmen Sie bitte der Hilfeseite fonts-conf(5).

Die Datei `local.conf` ist ein XML-Dokument. Achten Sie beim Editieren der Datei daher auf die richtige Groß- und Kleinschreibung und darauf, dass alle Tags geschlossen sind. Die Datei beginnt mit der üblichen XML-Deklaration gefolgt von einer DOCTYPE-Definition und dem `<fontconfig>`-Tag:

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

Wie vorher erwähnt, stehen schon alle Schriftarten in `/usr/local/share/fonts/` und `~/.fonts/` für Anwendungen, die Xft unterstützen, zur Verfügung. Wenn Sie ein Verzeichnis außerhalb dieser beiden Bäume benutzen wollen, fügen Sie eine Zeile wie die nachstehende zu `/usr/local/etc/fonts/local.conf` hinzu:

```
<dir>/path/to/my/fonts</dir>
```

Wenn Sie neue Schriftarten hinzugefügt haben, müssen Sie den Schriftarten-Cache neu aufbauen:

```
# fc-cache -f
```

Das Anti-aliasing-Verfahren zeichnet Ränder leicht unscharf, dadurch werden kleine Schriften besser lesbar und der Treppenstufen-Effekt bei wird großen Schriften vermieden. Auf normale Schriftgrößen sollte das Verfahren aber nicht angewendet werden, da dies die Augen zu sehr anstrengt. Um kleinere Schriftgrößen als 14 Punkt von dem Verfahren auszuschließen, fügen Sie in `local.conf` die nachstehenden Zeilen ein:

```
       <match target="font">
    <test name="size" compare="less">
        <double>14</double>
    </test>
    <edit name="antialias" mode="assign">
        <bool>false</bool>
    </edit>
  </match>
  <match target="font">
    <test name="pixelsize" compare="less" qual="any">
        <double>14</double>
    </test>
    <edit mode="assign" name="antialias">
        <bool>false</bool>
    </edit>
  </match>
```

Das Anti-aliasing-Verfahren kann die Abstände einiger Fixschriften falsch darstellen, dies fällt besonders unter KDE auf. Sie können das Problem umgehen, indem Sie die Abstände dieser Schriften auf den Wert `100` festsetzen. Fügen Sie die nachstehenden Zeilen hinzu:

```
       <match target="pattern" name="family">
           <test qual="any" name="family">
               <string>fixed</string>
           </test>
           <edit name="family" mode="assign">
               <string>mono</string>
           </edit>
       </match>
       <match target="pattern" name="family">
           <test qual="any" name="family">
               <string>console</string>
           </test>
           <edit name="family" mode="assign">
               <string>mono</string>
           </edit>
```

```
        </match>
```

Damit werden die Namen der gebräuchlichen Fixschriften auf `"mono"` abgebildet. Für diese Schriften setzen Sie dann den Abstand fest:

```
        <match target="pattern" name="family">
            <test qual="any" name="family">
                <string>mono</string>
            </test>
            <edit name="spacing" mode="assign">
                <int>100</int>
            </edit>
        </match>
```

Bestimmte Schriftarten, wie Helvetica, können Probleme mit dem Anti-Aliasing-Verfahren verursachen. In der Regel erscheinen diese Schriftarten dann vertikal halbiert. Im schlimmsten Fall stürzen Anwendungen als Folge davon ab. Sie vermeiden dies, indem Sie betroffene Schriftarten in `local.conf` von dem Verfahren ausnehmen:

```
        <match target="pattern" name="family">
            <test qual="any" name="family">
                <string>Helvetica</string>
            </test>
            <edit name="family" mode="assign">
                <string>sans-serif</string>
            </edit>
        </match>
```

Wenn Sie `local.conf` editiert haben, stellen Sie bitte sicher, dass die Datei mit dem Tag `</fontconfig>` endet. Ist das nicht der Fall, werden die Änderungen nicht berücksichtigt.

Benutzer können eigene Einstellungen in der Datei `~/.fonts.conf` vornehmen. Achten Sie auch hier auf die richtige XML-Syntax.

Mit einem LCD können Sie *sub-pixel sampling* anstelle von Anti-aliasing einsetzen. Dieses Verfahren behandelt die horizontal getrennten Rot-, Grün- und Blau-Komponenten eines Pixels gesondert und verbessert damit (teilweise sehr wirksam) die horizontale Auflösung. Die nachstehende Zeile in `local.conf` aktiviert diese Funktion:

```
        <match target="font">
            <test qual="all" name="rgba">
                <const>unknown</const>
            </test>
            <edit name="rgba" mode="assign">
                <const>rgb</const>
            </edit>
        </match>
```

Anmerkung

Abhängig von der Art Ihres Bildschirms müssen Sie anstelle von `rgb` eines der folgenden verwenden: `bgr`, `vrgb` oder `vbgr`. Experimentieren Sie und vergleichen, was besser aussieht.

6.6. Der X-Display-Manager

Beigetragen von Seth Kingsley.

Xorg enthält den X-Display-Manager XDM, um Sitzungen zu verwalten. XDM stellt eine graphische Anmeldemaske zur Verfügung, in der Sie den Server, auf dem eine Sitzung laufen soll, auswählen können und in der Sie die Autorisierungs-Informationen, wie Benutzername und Passwort, eingeben können.

Dieser Abschnitt zeigt, wie der X-Displaymanager konfiguriert wird. Einige grafische Oberflächen enthalten ihre eigenen graphischen Login-Manager. Eine Anleitung zur Konfiguration des GNOME Display-Managers finden Sie im Abschnitt 6.7.1, „GNOME“. Eine Anleitung zur Konfiguration des KDE Display Managers finden Sie im Abschnitt 6.7.2, „KDE“.

6.6.1. XDM einrichten

XDM kann über das Paket oder den Port x11/xdm installiert werden. Nach der Installation lässt sich XDM durch einen Eintrag in `/etc/ttys` bei jedem Start des Rechners aktivieren:

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

Ändern Sie den Wert `off` zu `on` und speichern Sie die Datei. `ttyv8` zeigt an, dass XDM auf dem neunten virtuellen Terminal ausgeführt wird.

Die Konfigurationsdateien von XDM befinden sich in `/usr/local/lib/X11/xdm` . Dieses Verzeichnis enthält einige Dateien, mit denen das Verhalten und Aussehen von XDM beeinflusst werden kann, sowie ein paar Skripte und Programme zur Einrichtung des Desktops. Eine Zusammenfassung der Aufgaben dieser Dateien beschreibt die Tabelle 6.1, „Die Konfigurationsdateien von XDM“. Die genaue Syntax und Verwendung wird in xdm(1) beschrieben.

Tabelle 6.1. Die Konfigurationsdateien von XDM

Datei	Beschreibung
`Xaccess`	Verbindungen zu XDM werden über das „X Display Manager Connection Protocol“ (XDMCP) hergestellt. `Xaccess` enthält die Client-Berechtigungen zur Steuerung der XDMCP-Verbindungen entfernter Maschinen. In der Voreinstellung erlaubt diese Datei keine Verbindungen von entfernten Maschinen.
`Xresources`	Diese Datei steuert das Erscheinungsbild der Bildschirmauswahl und Anmeldemasken von XDM. In der Voreinstellung erscheint ein rechteckiges Anmeldefenster, dass den Hostnamen und einen Anmeldeprompt mit „Login:“ und „Password“ anzeigt. Das Format dieser Datei entspricht den Dateien im Verzeichnis `app-defaults` , die in der Dokumentation von Xorg beschrieben sind.
`Xservers`	Diese Datei enthält eine Liste entfernter Rechner, die in der Bildschirmauswahl angeboten werden.
`Xsession`	Dieses Skript wird von XDM aufgerufen, nachdem sich ein Benutzer erfolgreich angemeldet hat. Üblicherweise besitzt jeder Benutzer eine angepasste Version dieses Skripts in `~/.xsession`, das dann anstelle von `Xsession` ausgeführt wird.
`Xsetup_*`	Diese Skripten werden automatisch ausgeführt, bevor die Bildschirmauswahl oder die Anmeldemasken angezeigt werden. Für jeden lokalen Bildschirm gibt es ein Skript namens `Xsetup_*`, wobei * die lokale Bildschirmnummer ist. Normalerweise werden damit ein oder zwei Programme, wie `xconsole`, im Hindergrund gestartet.
`xdm-config`	Konfiguration für alle auf der Maschine verwalteten Bildschirme.
`xdm-errors`	Enthält Fehler, die vom Server generiert werden. Wenn ein von XDM verwalteter Bildschirm hängen bleibt, suchen Sie in dieser Datei nach Fehlermeldungen. Für

Datei	Beschreibung
	jede Sitzung werden die Meldungen auch in die Datei `~/.xsession-errors` des Benutzers geschrieben.
`xdm-pid`	Die Prozess-ID des gerade laufenden XDM-Prozesses.

6.6.2. Fernzugriff einrichten

In der Voreinstellung können sich nur Benutzer auf dem selben System über XDM anmelden. Um es Benutzern anderer Systeme zu ermöglichen, sich mit dem Bildschirm-Server zu verbinden, muss der Zugriffsregelsatz bearbeitet und der Listener aktiviert werden.

Um XDM so zu konfigurieren, dass jede Verbindung angenommen wird, kommentieren Sie die Zeile `DisplayManager.requestPort` in `/usr/local/lib/X11/xdm/xdm-config` aus, indem Sie der Zeile ein `!` voranstellen.

```
! SECURITY: do not listen for XDMCP or Chooser requests
! Comment out this line if you want to manage X terminals with xdm
DisplayManager.requestPort:     0
```

Speichern Sie die Änderungen und starten Sie XDM neu. Um den Fernzugriff zu beschränken, sehen Sie sich die Beispiele in `/usr/local/lib/X11/xdm/Xaccess` an. Zusätzliche Informationen finden Sie in xdm(1)

6.7. Grafische Oberflächen

Beigetragen von Valentino Vaschetto.

Dieser Abschnitt beschreibt die Installation der drei beliebtesten grafischen Oberflächen unter FreeBSD. Eine Oberfläche kann alles von einem einfachen Window-Manager bis hin zu kompletten Anwendungen sein. Mehr als einhundert grafische Oberflächen stehen in der Kategorie `x11-wm` der Ports-Sammlung zur Verfügung.

6.7.1. GNOME

GNOME ist eine benutzerfreundliche Oberfläche. Es besitzt eine Leiste, mit der Anwendungen gestartet werden und die Statusinformationen anzeigen kann. Programme und Daten können auf der Oberfläche abgelegt werden und Standardwerkzeuge stehen zur Verfügung. Es gibt Konventionen, die es Anwendungen leicht machen, zusammenzuarbeiten und ein konsistentes Erscheinungsbild garantieren. Weitere Informationen zu GNOME unter FreeBSD finden Sie unter http://www.FreeBSD.org/gnome. Die Webseite enthält zusätzliche Informationen über die Installation, Konfiguration und Verwaltung von GNOME unter FreeBSD.

Diese grafische Oberfläche kann als Paket installiert werden:

```
# pkg install gnome2
```

Um GNOME stattdessen aus der Ports-Sammlung zu übersetzen, nutzen Sie das folgende Kommando. GNOME ist eine große Anwendung, die sogar auf einem schnellen Computer einige Zeit zum Übersetzten benötigt.

```
# cd /usr/ports/x11/gnome2
# make install clean
```

GNOME benötigt ein eingehängtes `/proc` Dateisystem. Fügen Sie daher die folgende Zeile in `/etc/fstab` ein, damit procfs(5) beim Systemstart automatisch eingehängt wird:

```
proc            /proc       procfs  rw  0   0
```

GNOME benötigt D-Bus und HAL für einen Nachrichtenbus und Hardware Abstraktion. Diese Anwendungen werden automatisch als Abhängigkeiten von GNOME installiert. Aktivieren Sie die Dienste in `/etc/rc.conf` , sodass sie automatisch gestartet werden wenn das System bootet:

```
dbus_enable="YES"
hald_enable="YES"
```

Nach der Installation weisen Sie Xorg an, GNOME zu starten. Der einfachste Weg, dies zu tun, ist über den GNOME Display Manager GDM, der als Teil des GNOME-Desktops installiert wird. Um GDM zu aktivieren, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
gdm_enable="YES"
```

In der Regel ist es ratsam, alle GNOME-Dienste zu starten. Um dies zu erreichen, fügen Sie die folgende Zeile in `/etc/rc.conf` ein:

```
gnome_enable="YES"
```

GDM wird nun automatisch gestartet, wenn das System hochfährt.

GNOME kann alternativ auch von der Kommandozeile gestartet werden, wenn eine entsprechend konfigurierte `~/.xinitrc` vorliegt. Existiert diese Datei bereits, ersetzen Sie den Aufruf des Window-Managers durch /usr/local/bin/gnome-session. Wenn `.xinitrc` nicht existiert, erstellen Sie die Datei mit folgendem Befehl:

```
% echo "exec /usr/local/bin/gnome-session" > ~/.xinitrc
```

Eine dritte Methode ist, XDM als Display-Manager zu verwenden. In diesem Fall erstellen Sie eine ausführbare `~/.xsession`:

```
% echo "#!/bin/sh" > ~/.xsession
% echo "exec /usr/local/bin/gnome-session" >> ~/.xsession
% chmod +x ~/.xsession
```

6.7.2. KDE

KDE ist eine weitere, leicht zu benutzende Desktop-Umgebung. Dieser Desktop bietet eine Sammlung von Anwendungen mit einheitlichem Erscheinungsbild (*look and feel*), einheitlichen Menüs, Werkzeugleisten, Tastenkombinationen, Farbschemata, Internationalisierung und einer zentralen, dialoggesteuerten Desktop-Konfiguration. Weitere Informationen zu KDE finden Sie unter http://www.kde.org/. Spezifische Informationen für FreeBSD finden Sie unter http://freebsd.kde.org.

Um das Paket KDE4 zu installieren, geben Sie ein:

```
# pkg install x11/kde4
```

Um KDE stattdessen aus dem Quellcode zu übersetzen, verwenden Sie das folgende Kommando. Bei der Installation wird ein Menü zur Auswahl der Komponenten angezeigt. KDE ist eine große Anwendung, die sogar auf einem schnellen Computer einige Zeit zum Übersetzen benötigt.

```
# cd /usr/ports/x11/kde4
# make install clean
```

KDE benötigt ein eingehängtes `/proc`. Fügen Sie diese Zeile in `/etc/fstab` ein, um das Dateisystem automatisch beim Systemstart einzuhängen:

```
proc           /proc       procfs  rw  0   0
```

KDE benötigt D-Bus und HAL für einen Nachrichtenbus und Hardware Abstraktion. Diese Anwendungen werden automatisch als Abhängigkeiten von KDE installiert. Aktivieren Sie die Dienste in `/etc/rc.conf` , sodass sie automatisch gestartet werden wenn das System bootet:

```
dbus_enable="YES"
```

```
hald_enable="YES"
```

Die Installation von KDE beinhaltet den KDE Display-Manager KDM. Um diesen Display-Manager zu aktivieren, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
kdm4_enable="YES"
```

Eine zweite Möglichkeit KDE zu starten, ist `startx` in der Kommandozeile einzugeben. Damit dies funktioniert, wird folgende Zeile in `~/.xinitrc` benötigt:

```
exec /usr/local/bin/startkde
```

Eine dritte Möglichkeit ist KDE über XDM zu starten. Um dies zu tun, erstellen Sie eine ausführbare `~/.xsession` wie folgt:

```
% echo "#!/bin/sh" > ~/.xsession
% echo "exec /usr/local/bin/startkde" >> ~/.xsession
% chmod +x ~/.xsession
```

Sobald KDE gestartet wird, finden Sie im integrierten Hilfesystem weitere Informationen zur Benutzung der verschiedenen Menüs und Anwendungen.

6.7.3. Xfce

Xfce ist eine Desktop-Umgebung, basierend auf den von GNOME verwendeten GTK+-Bibliotheken. Es hat einen geringeren Speicherbedarf und stellt dabei einen schlichten, effizienten und einfach zu benutzenden Desktop zur Verfügung. Xfce ist vollständig konfigurierbar, verfügt über eine Programmleiste mit Menüs, *Applets* und einen Programmstarter. Zudem sind ein Datei-Manager und ein Sound-Manager enthalten und das Programm ist über *Themes* anpassbar. Da es schnell, leicht und effizient ist, eignet sich Xfce ideal für ältere oder langsamere Rechner mit wenig Speicher. Weitere Informationen zu Xfce finden Sie unter http://www.xfce.org.

Um das Paket Xfce zu installieren, geben Sie folgendes ein:

```
# pkg install xfce
```

Um stattdessen den Port zu übersetzen:

```
# cd /usr/ports/x11-wm/xfce4
# make install clean
```

Im Gegensatz zu GNOME oder KDE, besitzt Xfce keinen eigenen Login-Manager. Damit Xfce von der Kommandozeile mit `startx` gestartet werden kann, muss zunächst ein Eintrag in `~/.xinitrc` hinzugefügt werden:

```
% echo "exec /usr/local/bin/startxfce4 --with-ck-launch" > ~/.xinitrc
```

Alternativ dazu kann XDM verwendet werden. Um diese Methode zu konfigurieren, erstellen Sie eine ausführbare `~/.xsession`:

```
% echo "#!/bin/sh" > ~/.xsession
% echo "exec /usr/local/bin/startxfce4 --with-ck-launch" >> ~/.xsession
% chmod +x ~/.xsession
```

6.8. Compiz Fusion installieren

Der Einsatz von hübschen 3D-Effekten ist eine Möglichkeit, die Benutzerfreundlichkeit eines Desktop-Rechners zu erhöhen.

Die Installation des Compiz Fusion Pakets ist einfach, aber bei der Konfiguration sind ein paar Schritte notwendig, die nicht in der Dokumentation des Ports beschrieben werden.

6.8.1. Konfiguration des FreeBSD nVidia-Treibers

Desktop-Effekte erzeugen eine hohe Last auf der Grafikkarte. Für nVidia-basierte Grafikkarten sind die proprietären Treiber für eine gute Leistung erforderlich. Benutzer anderer Grafikkarten können diesen Abschnitt überspringen und mit der Konfiguration von Xorg fortfahren.

Lesen Sie die FAQ zu diesem Thema, um herauszufinden, wie der richtige nVidia-Treiber ermittelt werden kann.

Nachdem der richtige Treiber für die Karte ermittelt wurde, kann er wie jedes andere Paket installiert werden.

Um beispielsweise den aktuellsten Treiber zu installieren:

```
# pkg install x11/nvidia-driver
```

Der Treiber erstellt ein Kernelmodul, welches beim Systemstart geladen werden muss. Fügen folgende Zeile in `/boot/loader.conf` ein:

```
nvidia_load="YES"
```

Anmerkung

Um das Kernelmodul direkt in den laufenden Kernel zu laden, kann der Befehl `kldload nvidia` eingeben werden. Allerdings wurde festgestellt, dass einige Versionen von Xorg nicht richtig funktionieren, wenn der Treiber nicht beim Systemstart geladen wurde. Nach der Änderung in `/boot/loader.conf` wird daher ein Neustart des Systems empfohlen.

Wenn das Kernelmodul geladen ist, muss in der Regel nur noch eine einzige Zeile in `xorg.conf` geändert werden, um den proprietären Treiber zu aktivieren:

Suchen Sie folgende Zeile in `/etc/X11/xorg.conf` :

```
Driver      "nv"
```

und ändern Sie die Zeile zu:

```
Driver      "nvidia"
```

Wenn Sie nun die grafische Oberfläche starten, sollten Sie vom nVidia Startbildschirm begrüßt werden. Alles sollte wie gewohnt funktionieren.

6.8.2. Konfiguration von Desktop-Effekten in xorg.conf

Um Compiz Fusion zu aktivieren, muss `/etc/X11/xorg.conf` angepasst werden:

Fügen Sie diesen Abschnitt hinzu, um Composite-Effekte zu aktivieren:

```
Section "Extensions"
    Option         "Composite" "Enable"
EndSection
```

Suchen Sie den Abschnitt „Screen“, der ähnlich wie hier gezeigt aussehen sollte:

```
Section "Screen"
    Identifier     "Screen0"
    Device         "Card0"
    Monitor        "Monitor0"
    ...
```

und fügen Sie die beiden folgenden Zeilen hinzu (z.B. nach „Monitor“):

```
DefaultDepth    24
```

```
Option         "AddARGBGLXVisuals" "True"
```

Suchen Sie den Abschnitt „Subsection“, der sich auf die gewünschte Bildschirmauflösung bezieht. Wenn Sie z.B. 1280x1024 verwenden möchten, suchen Sie den folgenden Abschnitt. Sollte die gewünschte Auflösung nicht in allen Unterabschnitten vorhanden sein, können Sie den entsprechenden Eintrag manuell hinzufügen:

```
SubSection     "Display"
    Viewport    0 0
    Modes      "1280x1024"
EndSubSection
```

Für Composite-Effekte wird eine Farbtiefe von 24 Bit benötigt. Ändern Sie dazu den obigen Abschnitt wie folgt:

```
SubSection     "Display"
    Viewport    0 0
    Depth       24
    Modes      "1280x1024"
EndSubSection
```

Zuletzt muss noch sichergestellt werden, dass die Module „glx“ und „extmod“ im Abschnitt „Module“ geladen werden:

```
Section "Module"
    Load           "extmod"
    Load           "glx"
    ...
```

Die vorangegangenen Einstellungen können automatisch mit x11/nvidia-xconfig erledigt werden, indem Sie folgende Kommandos als root ausführen:

```
# nvidia-xconfig --add-argb-glx-visuals
# nvidia-xconfig --composite
# nvidia-xconfig --depth=24
```

6.8.3. Installation und Konfiguration von Compiz Fusion

Die Installation von Compiz Fusion ist so einfach wie die Installation jedes anderen Pakets:

```
# pkg install x11-wm/compiz-fusion
```

Wenn die Installation abgeschlossen ist, starten Sie (als normaler Benutzer) den grafischen Desktop mit folgendem Befehl:

```
% compiz --replace --sm-disable --ignore-desktop-hints ccp &
% emerald --replace &
```

Der Bildschirm wird für einige Sekunden flackern, da der Window Manager (z.B. Metacity, wenn Sie GNOME benutzen) von Compiz Fusion ersetzt wird. Emerald kümmert sich um die Fensterdekoration (z.B. die Schatzflächenn schließen, minimieren und maximieren, Titelleisten, usw.).

Sie können dieses einfache Skript anpassen und es dann beim Start automatisch ausführen lassen (z.B. durch Hinzufügen von „Sessions“ beim GNOME-Desktop):

```
#! /bin/sh
compiz --replace --sm-disable --ignore-desktop-hints ccp &
emerald --replace &
```

Speichern Sie die Datei in Ihrem Heimatverzeichnis, beispielsweise als `start-compiz` und machen Sie die Datei ausführbar:

```
% chmod +x ~/start-compiz
```

Benutzen Sie dann die grafische Oberfläche, um das Skript zu Autostart-Programme hinzuzufügen (beim GNOME-Desktop unter Systemwerkzeuge, Einstellungen, Sessions).

Um die gewünschten Effekte und Einstellungen zu konfigurieren, starten Sie (wieder als normaler Benutzer) den Compiz Config Einstellungs—Manager:

```
% ccsm
```

Anmerkung

In GNOME finden Sie diese Einstellungen wieder im Menü unter Systemwerkzeuge, Einstellungen.

Wenn Sie „gconf support“ während der Installation ausgewählt haben, können Sie diese Einstellungen auch im `gconf-editor` unter `apps/compiz` finden.

6.9. Fehlersuche

Wenn die Maus nicht funktioniert, müssen Sie diese zuerst konfigurieren. In neueren Versionen von Xorg werden die `InputDevice`-Abschnitte in `xorg.conf` ignoriert, um stattdessen die automatisch erkannten Geräte zu verwenden. Um das alte Verhalten wiederherzustellen, fügen Sie folgende Zeile zum Abschnitt `ServerLayout` oder `ServerFlags` dieser Datei hinzu:

```
Option "AutoAddDevices" "false"
```

Anmerkung

Wie zuvor erwähnt, wird standardmäßig der hald-Dienst automatisch die Tastatur erkennen. Es kann jedoch passieren, dass das Tastaturlayout oder das Modell nicht korrekt erkannt wird. Grafische Oberflächen wie GNOME, KDE oder Xfce stellen Werkzeuge für die Konfiguration der Tastatur bereit. Es ist allerdings auch möglich, die Tastatureigenschaften direkt zu setzen, entweder mit Hilfe von setxkbmap(1) oder mit einer Konfigurationsregel von hald.

Wenn Sie zum Beispiel eine PC 102-Tasten Tastatur mit französischem Layout verwenden möchten, müssen sie eine Tastaturkonfigurationsdatei `x11-input.fdi` für hald im Verzeichnis `/usr/local/etc/hal/fdi/policy` anlegen. Diese Datei sollte die folgenden Zeilen enthalten:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<deviceinfo version="0.2">
  <device>
    <match key="info.capabilities" contains="input.keyboard">
      <merge key="input.x11_options.XkbModel" type="string">pc102</
merge>
      <merge key="input.x11_options.XkbLayout" type="string">fr</merge>
    </match>
  </device>
</deviceinfo>
```

Wenn diese Datei bereits existiert, kopieren Sie nur die Zeilen in die Datei, welche die Tastaturkonfiguration betreffen.

Sie müssen Ihren Computer neu starten, um hald zu zwingen, diese Datei einzulesen.

Es ist auch möglich, die gleiche Konfiguration von einem X-Terminal oder einem Skript über den folgenden Befehl heraus zu tätigen:

```
% setxkbmap -model pc102 -layout fr
```

`/usr/local/share/X11/xkb/rules/base.lst` enthält die zur Verfügung stehenden Tastatur- und Layoutoptionen.

Die Konfigurationsdatei `xorg.conf.new` kann nun an bestimmte Bedürfnisse angepasst werden. Öffnen Sie die Datei in einem Editor, wie emacs(1) oder ee(1). Falls der Monitor ein älteres oder ungewöhnliches Modell ist und keine automatische Erkennung unterstützt, können die Synchronisationsfrequenzen im Abschnitt `"Monitor"` der `xorg.conf.new` eingetragen werden.

```
Section "Monitor"
      Identifier   "Monitor0"
      VendorName   "Monitor Vendor"
      ModelName    "Monitor Model"
      HorizSync    30-107
      VertRefresh  48-120
EndSection
```

Die meisten Monitore unterstützen die automatische Erkennung der Synchronisationsfrequenzen, so dass eine manuelle Eingabe der Werte nicht erforderlich ist. Für die wenigen Monitore, die keine automatische Erkennung unterstützen, sollten nur die vom Hersteller zur Verfügung gestellten Werte eingegeben werden, um einen möglichen Schaden zu vermeiden.

X unterstützt die Energiesparfunktionen (DPMS, Energy Star) für Monitore. Mit xset(1) können die Zeitlimits für die DPMS-Modi standby, suspend, off vorgeben, oder zwingend aktiviert werden. Die DPMS-Funktionen können mit der folgenden Zeile im Abschnitt `"Monitor"` aktiviert werden:

```
Option         "DPMS"
```

Die gewünschte Auflösung und Farbtiefe stellen sie im Abschnitt `"Screen"` ein:

```
Section "Screen"
      Identifier "Screen0"
      Device     "Card0"
      Monitor    "Monitor0"
      DefaultDepth 24
      SubSection "Display"
       Viewport  0 0
       Depth     24
       Modes     "1024x768"
      EndSubSection
EndSection
```

Mit `DefaultDepth` wird die standardmäßige Farbtiefe angegeben. Mit der Option `-depth` von Xorg(1) lässt sich die vorgegebene Farbtiefe überschreiben. `Modes` gibt die Auflösung für die angegebene Farbtiefe an. Die Farbtiefe im Beispiel beträgt 24 Bits pro Pixel, die zugehörige Auflösung ist 1024x768 Pixel. Beachten Sie, dass in der Voreinstellung nur Standard-VESA-Modi der Grafikkarte angegeben werden können.

Sichern Sie die Konfigurationsdatei. Testen Sie anschließend die Konfiguration, wie oben beschrieben.

Anmerkung

Bei der Fehlersuche stehen Ihnen die Protokolldateien von Xorg zur Verfügung. Die Protokolle enthalten Informationen über alle Geräte, die mit dem Xorg-Server verbunden ist. Die Namen der Xorg-Protkolldateien haben das Format `/var/log/Xorg.0.log` .Der exakte Name der Datei variiert dabei von `Xorg.0.log` bis `Xorg.8.log`, und so weiter.

Wenn alles funktioniert, installieren Sie die Datei an einen Ort, an dem Xorg(1) sie finden kann. Typischerweise ist dies `/etc/X11/xorg.conf` oder `/usr/local/etc/X11/xorg.conf` .

```
# cp xorg.conf.new /etc/X11/xorg.conf
```

Damit ist die Konfiguration von Xorg abgeschlossesn. Xorg kann nun mit dem Programm startx(1) gestartet werden. Alternativ kann der Xorg-Server auch mithilfe von xdm(1) gestartet werden.

6.9.1. Konfiguration des Intel® `i810` Graphics Chipsets

Der Intel® i810-Chipset benötigt den Treiber `agpgart`, die AGP-Schnittstelle für Xorg. Die Manualpage für den Treiber agp(4) enthält weitere Informationen.

Ab jetzt kann die Hardware wie jede andere Grafikkarte auch konfiguriert werden. Beachten Sie, dass der Treiber agp(4) nicht nachträglich in einen laufenden Kernel geladen werden kann. Er muss entweder fest im Kernel eingebunden sein, oder beim Systemstart über `/boot/loader.conf` geladen werden.

6.9.2. Einen Widescreen-Monitor einsetzen

Dieser Abschnitt geht über die normalen Konfigurationsarbeiten hinaus und setzt ein wenig Vorwissen voraus. Selbst wenn die Standardwerkzeuge zur X-Konfiguration bei diesen Geräten nicht zum Erfolg führen, gibt es in den Protokolldateien genug Informationen, mit denen Sie letztlich doch einen funktionierenden X-Server konfigurieren können. Alles, was Sie dazu benötigen, ist ein Texteditor.

Aktuelle Widescreen-Formate (wie WSXGA, WSXGA+, WUXGA, WXGA, WXGA+, und andere mehr) unterstützen Seitenverhältnisse wie 16:10 oder 10:9, die unter X Probleme verursachen können. Bei einem Seitenverhältnis von 16:10 sind beispielsweise folgende Auflösungen möglich:

- 2560x1600
- 1920x1200
- 1680x1050
- 1440x900
- 1280x800

Irgendwann wird die Konfiguration vereinfacht werden, dass nur noch die Auflösung als `Mode` in `Section "Screen"` eingtragen wird, so wie hier:

```
Section "Screen"
Identifier  "Screen 0"
Device     "Card 0"
Monitor     "Monitor0"
Default Depth 24
SubSection  "Display"
 ViewPort  0 0
 Depth   24
 Modes   "1680x1050"
EndSubSection
EndSection
```

Xorg ist intelligent genug, um die Informationen zu den Auflösungen über I2C/DDC zu beziehen, und weiß daher, welche Auflösungen und Frequenzen der Widescreen-Monitor unterstützt.

Wenn diese `ModeLines` in den Treiberdateien nicht vorhanden sind, kann es sein, dass Sie Xorg beim Finden der korrekten Werte unterstützen müssen. Dazu extrahieren Sie die benötigten Informationen aus `/var/log/Xorg.0.log` und erzeugen daraus eine funktionierende `ModeLine`. Suchen Sie nach Zeilen ähnlich den folgenden:

```
(II) MGA(0): Supported additional Video Mode:
(II) MGA(0): clock: 146.2 MHz   Image Size:  433 x 271 mm
(II) MGA(0): h_active: 1680  h_sync: 1784  h_sync_end 1960 h_blank_end 2240 h_border: 0
(II) MGA(0): v_active: 1050  v_sync: 1053  v_sync_end 1059 v_blanking: 1089 v_border: 0
(II) MGA(0): Ranges: V min: 48  V max: 85 Hz, H min: 30  H max: 94 kHz, PixClock max ↺
170 MHz
```

Diese Informationen werden auch als EDID-Informationen bezeichnet. Um daraus eine funktionierende `ModeLine` zu erzeugen, müssen lediglich die Zahlen in die korrekte Reihenfolge gebracht werden:

```
ModeLine <name> <clock> <4 horiz. timings> <4 vert. timings>
```

Die korrekte `ModeLine` in `Section "Monitor"` würde für dieses Beispiel folgendermaßen aussehen:

```
Section "Monitor"
Identifier      "Monitor1"
VendorName      "Bigname"
ModelName       "BestModel"
ModeLine        "1680x1050" 146.2 1680 1784 1960 2240 1050 1053 1059 1089
Option          "DPMS"
EndSection
```

Nachdem diese Äderungen durchgeführt sind, sollte X auch auf Ihrem neuen Widescreen-Monitor starten.

6.9.3. Fehersuche in Compiz Fusion

F: Ich habe Compiz Fusion installiert und anschließend die hier erwähnten Kommandos eingegeben. Nun fehlen den Fenstern die Titelleisten und Schaltflächen. Was kann ich tun?

A: Wahrscheinlich fehlt eine Einstellung in `/etc/X11/xorg.conf` . Überprüfen Sie diese Datei gründlich, und überprüfen Sie insbesondere die Richtlinien `DefaultDepth` und `AddARGBGLXVisuals`.

F: Wenn ich Compiz Fusion starte, bringt dass den X-Server zum Absturz. Was kann ich tun?

A: Wenn Sie `/var/log/Xorg.0.log` durchsuchen, finden Sie wahrscheinlich Fehlermeldungen, die während des Starts von X ausgegeben werden. Die häufigste Meldung ist:

```
(EE) NVIDIA(0):     Failed to initialize the GLX module; please check in your X
(EE) NVIDIA(0):     log file that the GLX module has been loaded in your X
(EE) NVIDIA(0):     server, and that the module is the NVIDIA GLX module.  If
(EE) NVIDIA(0):     you continue to encounter problems, Please try
(EE) NVIDIA(0):     reinstalling the NVIDIA driver.
```

Dies ist für gewöhnlich der Fall, wenn Sie Xorg aktualisieren. Sie müssen das Paket x11/nvidia-driver neu installieren, damit GLX neu gebaut wird.

Teil II. Oft benutzte Funktionen

Nach den Grundlagen beschäftigt sich das FreeBSD-Handbuch mit oft benutzten Funktionen von FreeBSD. Die Kapitel behandeln die nachstehenden Themen:

- Zeigen Ihnen beliebte und nützliche Werkzeuge wie Browser, Büroanwendungen und Programme zum Anzeigen von Dokumenten.
- Zeigen Ihnen Multimedia-Werkzeuge für FreeBSD.
- Erklären den Bau eines angepassten FreeBSD-Kernels, der die Systemfunktionen erweitert.
- Beschreiben ausführlich das Drucksystem, sowohl für direkt angeschlossene Drucker als auch für Netzwerkdrucker.
- Erläutern, wie Sie Linux-Anwendungen auf einem FreeBSD-System laufen lassen.

Damit Sie einige Kapitel verstehen, sollten Sie vorher andere Kapitel gelesen haben. Die Übersicht zu jedem Kapitel zählt die Voraussetzungen für das erolgreiche Durcharbeiten des Kapitels auf.

Inhaltsverzeichnis

Kapitel 7. Desktop-Anwendungen

7.1. Übersicht

Obwohl FreeBSD wegen seiner Leistung und Stabilität vor allem auf Serversystemen sehr beliebt ist, so ist es auch für den täglichen Einsatz als Desktop geeignet. Mit über 24,000 Anwendungen, die als Pakete oder Ports vorliegen, ist es leicht einen individuellen Desktop zu bauen, auf dem eine Vielzahl von Desktop-Anwendungen laufen. Dieses Kapitel zeigt, wie Sie die zahlreichen Desktop-Anwendungen, wie Web-Browser, Office-Pakete, Dokumentbetrachter und Finanzsoftware, installieren können.

Anmerkung

Benutzer die es vorziehen eine vorkonfigurierte Desktop-Version von FreeBSD zu installieren, anstatt das System von Grund auf zu konfigurieren, sollten sich die pcbsd.org Website ansehen.

Bevor Sie dieses Kapitel lesen, sollten Sie wissen:

- wie zusätzliche Anwendungen als Paket oder aus der Ports-Sammlung installiert werden. Dies wird in Kapitel 5, *Installieren von Anwendungen: Pakete und Ports* beschrieben.
- wie X und ein Window-Manager installiert wird. Dies wird in Kapitel 6, *Das X-Window-System* beschrieben.

Informationen zur Konfiguration von Multimedia-Anwendungen finden Sie in Kapitel 8, *Multimedia*.

7.2. Browser

FreeBSD besitzt keinen vorinstallierten Browser, stattdessen enthält das www-Verzeichnis der Ports-Sammlung viele Browser, die als Paket oder aus der Ports-Sammlung installiert werden können.

Die Desktop-Umgebungen KDE und GNOME verfügen über eigene HTML-Browser. Weitere Informationen zur Einrichtung dieser Umgebungen finden Sie in Abschnitt 6.7, „Grafische Oberflächen“.

Besonders schlanke Browser sind www/dillo2, www/links und www/w3m.

Dieser Abschnitt demonstriert, wie die folgenden gängigen Webbrowser installiert werden, sowie den Ressourcenbedarf, den Installationsaufwand beim Übersetzen des Ports, oder ob die Anwendung wichtige Abhängigkeiten benötigt.

Anwendung	Ressourcenbedarf	Installationsaufwand aus den Ports	Anmerkungen
Firefox	mittel	hoch	FreeBSD, Linux® und lokalisierte Versionen sind verfügbar
Opera	niedrig	niedrig	Es gibt eine FreeBSD- und eine Linux-Version. Die Linux-Version hängt von der Linux-Kompatibilität (*Linux Binary Compatibility*) und linux-openmotif ab.

Anwendung	Ressourcenbedarf	Installationsaufwand aus den Ports	Anmerkungen
Konqueror	mittel	hoch	Benötigt KDE-Biliotheken
Chromium	mittel	hoch	Benötigt Gtk+

7.2.1. Firefox

Firefox ist ein Open-Source Browser, der vollständig auf FreeBSD portiert wurde. Er bietet eine dem HTML-Standard konforme Anzeige, Browserfenster als Tabs, Blockierung von Pop-up-Fenstern, Erweiterungen, verbesserte Sicherheit und mehr. Firefox basiert auf der Mozilla Codebasis.

Installieren Sie das Paket der aktuellen Release-Version von Firefox:

```
# pkg install firefox
```

Um stattdessen die Extended Support Release (ESR) Version zu installieren, benutzen Sie:

```
# pkg install firefox-esr
```

Lokalisierte Versionen finden Sie in www/firefox-i18n und www/firefox-esr-i18n.

Alternativ kann auch die Ports-Sammlung verwendet werden, um die gewünschte Version von Firefox aus dem Quellcode zu installieren. Dieses Beispiel baut www/firefox, wobei sich `firefox` durch die ESR oder die lokalisierte Version ersetzen lässt.

```
# cd /usr/ports/www/firefox
# make install clean
```

7.2.1.1. Firefox und das Java™-Plugin

Die Installation von Firefox enthält keine Unterstützung für Java™. java/icedtea-web stellt ein freies Webbrowser-Plugin für die Ausführung von Java-Applets zur Verfügung. Dieses Plugin kann als Paket, oder wie in diesem Beispiel, über die Ports-Sammlung installiert werden:

```
# cd /usr/ports/java/icedtea-web
# make install clean
```

Verwenden Sie die Standardoptionen beim Übersetzen des Ports.

Starten Sie nun Firefox. Geben Sie in der Adresszeile `about:plugins` ein und bestätigen Sie die Eingabe mit Enter. Dadurch wird eine Seite geöffnet, die alle installierten Plugins auflistet. In dieser Liste sollte sich auch das Java™-Plugin befinden.

Wird das Plugin nicht gefunden, muss für jeden Benutzer der folgende Befehl ausgeführt werden:

```
% ln -s /usr/local/lib/IcedTeaPlugin.so \
  $HOME/.mozilla/plugins/
```

7.2.1.2. Firefox und das Adobe® Flash®-Plugin

Ein natives Adobe® Flash®-Plugin ist für FreeBSD nicht verfügbar. Es existiert jedoch ein Software-Layer (*Wrapper*), der es erlaubt die Linux®-Version des Plugins unter FreeBSD einzusetzen. Dieser Wrapper bietet auch Unterstützung für weitere Webbrowser-Plugins wie zum Beispiel RealPlayer®.

Führen Sie die folgenden Schritte aus, um das Plugin zu installieren und zu aktivieren:

1. Installieren Sie den Port www/nspluginwrapper. Aus lizenzrechtlichen Gründen ist ein Paket nicht verfügbar. Dieser Port benötigt emulators/linux_base-c6.

2. Installieren Sie den Port www/linux-c6-flashplugin11. Aus lizenzrechtlichen Gründen ist ein Paket nicht verfügbar.

3. Bevor das Plugin verwendet werden kann, muss der Benutzer zunächst das folgende Kommando aufrufen:

```
% nspluginwrapper -v -a -i
```

Nach einem Update von Flash® muss der Benutzer folgendes eingeben:

```
% nspluginwrapper -v -a -u
```

Starten Sie den Browser und geben Sie in der Adresszeile `about:plugins` ein. Diese Eingabe muss mit Enter bestätigt werden. Dadurch wird eine Seite geladen, die alle installierten Plugins auflistet.

7.2.1.3. Firefox und das Swfdec Flash®-Plugin

Swfdec ist die Bibliothek zum Dekodieren und Rendern von Flash® Animationen. Swfdec-Mozilla ist ein Plugin für Firefox-Browser, welches die Swfdec-Bibliothek zum Abspielen von SWF-Dateien benutzt.

Um das Paket zu installieren:

```
# pkg install swfdec-plugin
```

Wenn das Paket nicht verfügbar ist, kann es auch über die Ports-Sammlung gebaut und installiert werden:

```
# cd /usr/ports/www/swfdec-plugin
# make install clean
```

Starten Sie den Browser neu, damit das Plugin aktiviert wird.

7.2.2. Opera

Opera ist ein vollwertiger und standardkonformer Browser, der zudem relativ klein und schnell ist. Es enthält einen eingebauten E-Mail- und Newsreader, einen IRC-Client, einen RSS/Atom-Feeds-Reader, sowie weitere Programme. Opera ist als native Version für FreeBSD, oder als Linux®-Version verfügbar.

Der folgende Befehl installiert die FreeBSD-Version von Opera. Ersetzen Sie `opera` durch `linux-opera` , wenn Sie stattdessen die Linux®-Version installieren wollen.

```
# pkg install opera
```

Alternativ können Sie beide Versionen über die Ports-Sammlung installieren. Dieses Beispiel übersetzt die native Version:

```
# cd /usr/ports/www/opera
# make install clean
```

Wenn Sie die Linux®-Version des Browsers verwenden wollen, ersetzen Sie `opera` durch `linux-opera` .

Um Adobe® Flash®-Plugin Unterstützung zu installieren, übersetzten Sie zunächst den Port www/linux-c6-flash-plugin11. Ein Paket ist aus lizenzrechtlichen Gründen nicht verfügbar. Installieren Sie anschließend www/opera-linuxplugins. Dieses Beispiel übersetzt beides aus der Ports-Sammlung:

```
# cd /usr/ports/www/linux-c6-flashplugin11
# make install clean
# cd /usr/ports/www/opera-linuxplugins
# make install clean
```

Um die Existenz des Plugins zu überprüfen, starten Sie den Browser und geben Sie in der Adresszeile `about:plugins` ein. Diese Eingabe muss mit Enter bestätigt werden. Dadurch wird eine Seite geladen, die alle installierten Plugins auflistet.

Um das Java™-Plugin zu installieren, folgen Sie den Anweisungen in Abschnitt 7.2.1.1, „Firefox und das Java™-Plugin“.

7.2.3. Konqueror

Konqueror ist mehr als nur ein Webbrowser, da es ebenfalls Dateimanager und Multimedia-Betrachter ist. Es ist als Paket oder Port x11/kde4-baseapps verfügbar.

Konqueror unterstützt sowohl WebKit als auch sein eigenes KHTML. WebKit wird von vielen modernen Browsern verwendet, einschließlich Chromium. Um WebKit mit Konqueror unter FreeBSD zu verwenden, installieren Sie das Paket oder den Port www/kwebkitpart. Dieses Beispiel übersetzt den Port:

```
# cd /usr/ports/www/kwebkitpart
# make install clean
```

Um WebKit in Konqueror zu aktivieren, klicken Sie auf „Settings“, „Configure Konqueror“. Klicken Sie auf der Seite „General“ das Dropdown-Menü neben „Default web browser engine“ und ändern Sie „KHTML“ auf „WebKit“.

Konqueror kann auch Flash®-Seiten darstellen. Wie Sie die Flash®-Unterstützung aktivieren, können Sie unter `http://freebsd.kde.org/howtos/konqueror-flash.php` nachlesen.

7.2.4. Chromium

Chromium ist ein quelloffenes Browserprojekt mit dem Ziel ein sicheres, schnelleres und stabileres Surferlebnis im Web zu ermöglichen. Chromium ermöglicht surfen mit Tabs, Blockieren von Pop-Ups, Erweiterungen und vieles mehr. Chromium ist das Open Source Projekt, welches auf dem Google Chrome Webbrowser basiert.

Chromium kann als Paket durch die Eingabe des folgenden Befehls installiert werden:

```
# pkg install chromium
```

Als Alternative kann Chromium aus dem Quellcode durch die Ports Collection übersetzt werden:

```
# cd /usr/ports/www/chromium
# make install clean
```

Anmerkung

Die ausführbare Datei für Chromium ist `/usr/local/bin/chrome` und nicht `/usr/local/bin/chromium`.

7.2.4.1. Chromium und das Java™-Plug-In

Die Installation von Chromium enthält keine Unterstützung für Java™. Um Unterstützung für das Java™-Plugin zu installieren, folgen Sie den Anweisungen in Abschnitt 7.2.1.1, „Firefox und das Java™-Plugin“.

Sobald Unterstützung für Java™ installiert ist, starten Sie Chromium und geben Sie `about:plugins` in die Adresszeile ein. IcedTea-Web sollte dort als eines der installierten Plugins aufgelistet sein.

Falls Chromium das IcedTea-Web Plugin nicht anzeigt, geben Sie das folgende Kommando ein und starten Sie den Webbrowser anschließend neu:

```
# mkdir -p /usr/local/share/chromium/plugins
# ln -s /usr/local/lib/IcedTeaPlugin.so \
   /usr/local/share/chromium/plugins/
```

7.2.4.2. Chromium und das Adobe® Flash®-Plugin

Die Konfiguration von Chromium und Adobe® Flash® ist ähnlich zur Anleitung in Abschnitt 7.2.1.2, „Firefox und das Adobe® Flash®-Plugin“. Es sollte keine weitere Konfiguration notwendig sein, da Chromium in der Lage ist, einige Plugins von anderen Browsern zu benutzen.

7.3. Büroanwendungen

Neue Benutzer suchen oft ein komplettes Office-Paket oder eine leicht zu bedienende Textverarbeitung. Einige graphische Oberflächen wie KDE enthalten zwar ein Office-Paket, diese werden unter FreeBSD jedoch nicht standardmäßig installiert. Unabhängig von der installierten graphischen Oberfläche können diverse Office-Pakete jederzeit installiert werden.

Dieser Abschnitt demonstriert, wie die folgenden gängigen Büroanwendungen installiert werden, sowie den Ressourcenbedarf, den Installationsaufwand beim Übersetzen des Ports, oder ob die Anwendung wichtige Abhängigkeiten benötigt.

Anwendung	Ressourcenbedarf	Installationsaufwand aus den Ports	wichtige Abhängigkeiten
Calligra	niedrig	hoch	KDE
AbiWord	niedrig	niedrig	Gtk+ oder GNOME
The Gimp	niedrig	hoch	Gtk+
Apache OpenOffice	hoch	enorm	JDK™ und Mozilla
LibreOffice	etwas hoch	enorm	Gtk+, KDE/ GNOME oder JDK™

7.3.1. Calligra

Die KDE-Gemeinschaft stellt ein Office-Paket bereit, das auch separat von KDE eingesetzt werden kann. Calligra umfasst Standardkomponenten, die auch in anderen Office-Paketen enthalten sind. Words ist die Textverarbeitung, Sheets die Tabellenkalkulation, mit Stage werden Präsentationen erstellt und Karbon ist ein Zeichenprogramm.

In FreeBSD kann editors/calligra als Paket oder Port installiert werden. Um das Paket zu installieren, geben Sie folgendes ein:

```
# pkg install calligra
```

Wenn das Paket nicht verfügbar ist, benutzen Sie stattdessen die Ports-Sammlung:

```
# cd /usr/ports/editors/calligra
# make install clean
```

7.3.2. AbiWord

AbiWord ist eine freie Textverarbeitung, die dem Erscheinungsbild von Microsoft® Word ähnlich ist. Das Programm ist schnell, besitzt viele Funktionen und ist benutzerfreundlich.

AbiWord kann viele Dateiformate importieren oder exportieren, unter anderem auch propietäre wie Microsoft® `.rtf`.

Das AbiWord-Paket installieren Sie wie folgt:

```
# pkg install abiword
```

Sollte das Paket nicht zur Verfügung stehen, kann es über die Ports-Sammlung installiert werden:

```
# cd /usr/ports/editors/abiword
# make install clean
```

7.3.3. The GIMP

The GIMP ist ein ausgereiftes Bildverarbeitungsprogramm mit dem Bilder erstellt oder retuschiert werden können. Es kann sowohl als einfaches Zeichenprogramm oder zum retuschieren von Fotografien benutzt werden. Das Programm besitzt eine eingebaute Skriptsprache und es existieren sehr viele Plugins. The GIMP kann zahlreiche Formate lesen und speichern und stellt Schnittstellen zu Scannern und Tablets zur Verfügung.

Um das Paket zu installieren, geben Sie ein:

```
# pkg install gimp
```

Benutzen Sie alternativ die Ports-Sammlung:

```
# cd /usr/ports/graphics/gimp
# make install clean
```

Die Kategorie *graphics* (freebsd.org/ports/graphics.html) der Ports-Sammlung enthält für The Gimp verschiedene Plugins, Hilfedateien und Handbücher.

7.3.4. Apache OpenOffice

Apache OpenOffice ist eine Open Source Büroanwendung, die unter Leitung der Apache Software Foundation weiterentwickelt wird. Es enthält die typischen Anwendungen eines Office-Pakets: Textverarbeitung, Tabellenkalkulation, Präsentation und ein Zeichenprogramm. Die Bedienung gleicht anderen Office-Paketen und das Programm kann zahlreiche Dateiformate importieren und exportieren. Es gibt lokalisierte Versionen mit angepassten Menüs, Rechtschreibkontrollen und Wörterbüchern.

Die Textverarbeitung von Apache OpenOffice speichert Dateien im XML-Format. Dadurch wird die Verwendbarkeit der Dateien auf anderen Systemen erhöht und die Handhabung der Daten vereinfacht. Die Tabellenkalkulation besitzt eine Makrosprache und eine Schnittstelle zu Datenbanken. Apache OpenOffice läuft stabil auf Windows®, Solaris™, Linux®, FreeBSD und Mac OS® X. Weitere Informationen über Apache OpenOffice finden Sie auf openoffice.org. Spezifische Informationen für FreeBSD finden Sie auf porting.openoffice.org/freebsd/.

Apache OpenOffice installieren Sie wie folgt:

```
# pkg install apache-openoffice
```

Nachdem das Paket installiert ist, geben Sie folgenden ein, um Apache OpenOffice zu starten:

```
% openoffice-X.Y.Z
```

wobei `X.Y.Z` die Versionsnummer von Apache OpenOffice darstellt. Nach dem ersten Start werden einige Fragen gestellt. Außerdem wird im Heimatverzeichnis des Benutzers ein Verzeichnis `.openoffice.org` angelegt.

Falls das gewünschte Apache OpenOffice-Paket nicht verfügbar ist, kann immer noch der Port übersetzt werden. Es erfordert jedoch eine Menge Plattenplatz und ziemlich viel Zeit um die Quellen zu übersetzten.

```
# cd /usr/ports/editors/openoffice-4
# make install clean
```

Anmerkung

Um eine lokalisierte Version zu bauen, ersetzen Sie den letzten Befehl durch:

```
# make LOCALIZED_LANG= Ihre_Sprache  install clean
```

Ersetzen Sie *Ihre_Sprache* durch den korrekten ISO-Code. Eine Liste der unterstützten Codes steht in `files/Makefile.localized` , die sich im Portsverzeichnis befindet.

7.3.5. LibreOffice

LibreOffice ist ein frei verfügbares Office-Paket, welches von documentfoundation.org entwickelt wird. Es mit anderen großen Office-Paketen kompatibel und für eine Vielzahl von Plattformen erhältlich. Es ist ein Fork von Apache OpenOffice unter neuem Namen, das alle Anwendungen in einem kompletten Office-Paket enthält: Textverarbeitung, Tabellenkalkulation, Präsentationsmanager, Zeichenprogramm, Datenbankmanagementprogramm und ein Werkzeug zum Erstellen und Bearbeiten von mathematischen Formeln. Das Programm steht in verschiedenen Sprachen zur Verfügung, und die Internationalisierung wurde auf die Oberfläche, Rechtschreibkorrektur und die Wörterbücher ausgeweitet.

Das Textverarbeitungsprogramm von LibreOffice benutzt ein natives XML-Dateiformat für erhöhte Portabilität und Flexibilität. Die Tabellenkalkulation enthält eine Makrosprache und kann mit externen Datenbanken Verbindungen herstellen. LibreOffice ist stabil und läuft nativ auf Windows®, Linux®, FreeBSD und Mac OS® X. Weitere Informationen zu LibreOffice finden Sie unter libreoffice.org.

Um die englische Version von LibreOffice als Paket zu installieren, geben Sie folgenden Befehl ein:

```
# pkg install libreoffice
```

Die Kategorie *editors* (freebsd.org/ports/editors.html) der Ports-Sammlung enthält viele Lokalisierungen für LibreOffice. Wenn Sie ein lokalisiertes Paket installieren, ersetzen Sie `libreoffice` durch den Namen des lokalisierten Pakets.

Wenn das Paket installiert ist, geben Sie folgendes Kommando ein, um LibreOffice zu starten:

```
% libreoffice
```

Während des ersten Starts werden einige Fragen gestellt. Außerdem wird im Heimatverzeichinis des Benutzers ein Verzeichnis `.libreoffice` angelegt.

Falls das gewünschte LibreOffice-Paket nicht verfügbar ist, kann immer noch der Port übersetzt werden. Es erfordert jedoch eine Menge Plattenplatz und ziemlich viel Zeit um die Quellen zu übersetzten. Dieses Beispiel übersetzt die englische Version:

```
# cd /usr/ports/editors/libreoffice
# make install clean
```

Anmerkung

Um eine lokalisierte Version zu bauen, wechseln Sie mit `cd` in das Portverzeichnis der gewünschten Sprache. Unterstützte Sprachen finden Sie in der Kategorie *editors* (freebsd.org/ports/editors.html) der Ports-Sammlung.

7.4. Anzeigen von Dokumenten

Einige neuere Dokumentformate, die sich aktuell großer Beliebtheit erfreuen, können Sie sich mit den im Basissystem enthaltenen Programmen möglicherweise nicht ansehen. Dieser Abschnitt zeigt, wie Sie die folgenden Dokumentbetrachter installieren können:

Die nachstehenden Anwendungen werden behandelt:

Anwendung	Ressourcenbedarf	Installationsaufwand aus den Ports	wichtige Abhängigkeiten
Xpdf	niedrig	niedrig	FreeType

Anwendung	Ressourcenbedarf	Installationsaufwand aus den Ports	wichtige Abhängigkeiten
gv	niedrig	niedrig	Xaw3d
GQview	niedrig	niedrig	Gtk+ oder GNOME
ePDFView	niedrig	niedrig	Gtk+
Okular	niedrig	hoch	KDE

7.4.1. Xpdf

Für Benutzer, die einen schnellen PDF-Betrachter bevorzugen, bietet Xpdf eine schlanke und effiziente Alternative, die wenig Ressourcen benötigt. Da das Programm die Standard X-Zeichensätze benutzt, ist es nicht auf andere Toolkits angewiesen.

Um das Xpdf-Paket zu installieren, geben Sie folgendes ein:

```
# pkg install xpdf
```

Wenn das Paket nicht verfügbar ist, benutzen Sie die Ports-Sammlung:

```
# cd /usr/ports/graphics/xpdf
# make install clean
```

Starten Sie nach der Installation `xpdf` und aktivieren Sie das Menü mit der rechten Maustaste.

7.4.2. gv

gv kann PostScript®- und PDF-Dokumente anzeigen. Es stammt von ghostview ab, hat aber wegen der Xaw3d-Bibliothek eine schönere Benutzeroberfläche. gv besitzt viele konfigurierbare Funktionen, wie z. B. Ausrichtung, Papiergröße, Skalierung und Kantenglättung (*Anti-Aliasing*). Fast jede Operation kann sowohl mit der Tastatur als auch mit der Maus durchgeführt werden.

Installieren Sie das gv-Paket wie folgt:

```
# pkg install gv
```

Benutzen Sie die Ports-Sammlung, wenn das Paket nicht zur Verfügung steht:

```
# cd /usr/ports/print/gv
# make install clean
```

7.4.3. GQview

Mit GQview lassen sich Bilder verwalten. Unter anderem können Sie sich Bilder (auch auf dem ganzen Bildschirm) anschauen, ein externes Werkzeug aufrufen und eine Vorschau (*thumbnail*) erzeugen. Weiterhin können Sie automatisch ablaufende Präsentationen erstellen und grundlegende Dateioperationen durchführen, Bildersammlungen verwalten und doppelte Bilder aufspüren. GQview ist internationalisiert, das heißt es berücksichtigt die Spracheinstellungen des Systems.

Um das GQview-Paket zu installieren, geben Sie folgendes ein:

```
# pkg install gqview
```

Wenn das Paket nicht verfügbar ist, benutzen Sie die Ports-Sammlung:

```
# cd /usr/ports/graphics/gqview
# make install clean
```

7.4.4. ePDFView

ePDFView ist ein leichtgewichtiger PDF-Betrachter, der nur die Gtk+- und Poppler-Bibliotheken benötigt. Es befindet sich derzeit noch in Entwicklung, kann aber bereits die meisten PDF-Dateien (auch verschlüsselte) öffnen, speichern und über CUPS drucken.

Um das Paket ePDFView zu installieren, geben Sie folgendes ein:

```
# pkg install epdfview
```

Benutzen Sie die Ports-Sammlung, falls das Paket nicht verfügbar ist:

```
# cd /usr/ports/graphics/epdfview
# make install clean
```

7.4.5. Okular

Okular ist ein universeller Dokumentbetrachter der auf KPDF für KDE basiert. Es kann die meisten Formate öffnen, einschließlich PDF, PostScript®, DjVu, CHM, XPS und ePub.

Um das Paket Okular zu installieren, geben Sie folgendes ein:

```
# pkg install okular
```

Benutzen Sie die Ports-Sammlung, falls das Paket nicht verfügbar ist:

```
# cd /usr/ports/graphics/okular
# make install clean
```

7.5. Finanzsoftware

Zur Verwaltung der persönlichen Finanzen können einige leistungsfähige und einfach zu bedienende Anwendungen installiert werden. Einige von ihnen unterstützen verbreitete Formate, darunter Dateiformate, die von Quicken und Excel verwendet werden.

Dieser Abschnitt behandelt die folgenden Anwendungen:

Anwendung	Ressourcenbedarf	Installationsaufwand aus den Ports	wichtige Abhängigkeiten
GnuCash	niedrig	hoch	GNOME
Gnumeric	niedrig	hoch	GNOME
KMyMoney	niedrig	hoch	KDE

7.5.1. GnuCash

GnuCash ist Teil des GNOME-Projekts, mit dem Ziel, leicht zu bedienende und leistungsfähige Anwendungen bereitzustellen. Mit GnuCash können Einnahmen und Ausgaben, Bankkonten und Wertpapiere verwaltet werden. Das Programm ist leicht zu bedienen und genügt dennoch hohen Ansprüchen.

GnuCash stellt ein Register, ähnlich dem in einem Scheckheft und ein hierarchisches System von Konten zur Verfügung. Eine Transaktion kann in einzelne Teile aufgespaltet werden. GnuCash kann Quicken-Dateien (QIF) importieren und einbinden. Weiterhin unterstützt das Programm die meisten internationalen Formate für Zeitangaben und Währungen. Die Bedienung des Programms kann durch zahlreiche Tastenkombinationen und dem automatischen Vervollständigen von Eingaben beschleunigt werden.

Das GnuCash-Paket installieren Sie wie folgt:

```
# pkg install gnucash
```

Wenn das Paket nicht zur Verfügung steht, benutzen Sie die Ports-Sammlung:

```
# cd /usr/ports/finance/gnucash
# make install clean
```

7.5.2. Gnumeric

Gnumeric ist eine Tabellenkalkulation, die von der GNOME-Gemeinschaft entwickelt wird. Das Programm kann Eingaben anhand des Zellenformats oder einer Folge von Eingaben vervollständigen. Dateien verbreiteter Formate, wie die von Excel, Lotus 1-2-3 oder Quattro Pro lassen sich importieren. Es besitzt viele eingebaute Funktionen und Zellenformate, darunter die üblichen wie Zahl, Währung, Datum, Zeit, und viele weitere.

Installieren Sie das Gnumeric-Paket mit folgendem Kommando:

```
# pkg install gnumeric
```

Wenn das Paket nicht zur Verfügung steht, benutzen Sie die Ports-Sammlung:

```
# cd /usr/ports/math/gnumeric
# make install clean
```

7.5.3. KMyMoney

KMyMoney ist ein Programm zur Verwaltung der persönlichen Finanzen, das von der KDE-Gemeinschaft entwickelt wird. KMyMoney hat das Ziel, wichtige Funktionen zu bieten, die auch von kommerziellen Programmen zur Verwaltung der persönlichen Finanzen unterstützt werden. Zudem zählen eine einfache Bedienung sowie korrekte doppelte Buchführung zu den herausragenden Fähigkeiten dieses Programms. KMyMoney unterstützt den Import von Datendateien im Format Quicken (QIF), kann Investionen verfolgen, unterstützt verschiedene Währungen und bietet umfangreiche Reportmöglichkeiten.

Um das Paket KMyMoney zu installieren, geben Sie folgendes ein:

```
# pkg install kmymoney-kde4
```

Sollte das Paket nicht verfügbar sein, benutzen Sie die Ports-Sammlung:

```
# cd /usr/ports/finance/kmymoney2-kde4
# make install clean
```

Kapitel 8. Multimedia

Überarbeitet von Ross Lippert.

8.1. Übersicht

FreeBSD unterstützt viele unterschiedliche Soundkarten, die Benutzern den Genuss von Highfidelity-Klängen auf dem Computer ermöglichen. Dazu gehört unter anderem die Möglichkeit, Tonquellen in den Formaten MPEG Audio Layer 3 (MP3), Waveform Audio File (WAV), Ogg Vorbis und vielen weiteren Formaten aufzunehmen und wiederzugeben. Darüber hinaus enthält die FreeBSD Ports-Sammlung Anwendungen, die das Bearbeiten von aufgenommenen Tonspuren, das Hinzufügen von Klangeffekten und die Kontrolle der angeschlossenen MIDI-Geräte erlauben.

FreeBSD unterstützt auch die Wiedergabe von Videos und DVDs. Die FreeBSD Ports-Sammlung enthält Anwendungen, um verschiedene Video-Medien wiederzugeben, zu kodieren und zu konvertieren.

Dieses Kapitel beschreibt die Einrichtung von Soundkarten, Video-Wiedergabe, TV-Tuner Karten und Scannern unter FreeBSD. Es werden auch einige Anwendungen beschrieben, die für die Verwendung dieser Geräte zur Verfügung stehen.

Dieses Kapitel behandelt die folgenden Punkte:

- Konfiguration einer Soundkarte in FreeBSD.
- Fehlersuche bei Sound Einstellungen.
- Wiedergabe und Kodierung von MP3s und anderen Audio-Formaten.
- Vorbereitung des Systems für die Wiedergabe von Videos.
- Wiedergabe von DVDs, `.mpg`- und `.avi`-Dateien.
- Rippen von CDs und DVDs.
- Konfiguration von TV-Karten.
- Wie Sie CDs und DVDs in Dateien rippen.
- Installation und Konfiguration von MythTV.
- Einrichtung von Scannern.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- Wissen, wie Sie einen neuen Kernel konfigurieren und installieren (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).

Warnung

Audio CDs besitzen eine andere Kodierung als normale ISO-Dateisysteme. Deswegen sollten sie nicht mit mount(8) eingehangen werden.

8.2. Soundkarten einrichten

Von Moses Moore.
Aktualisiert von Marc Fonvieille.

Übersetzt von Benedikt Köhler und Uwe Pierau.

8.2.1. Den Soundtreiber einrichten

Bevor Sie die Konfiguration beginnen, sollten Sie in Erfahrung bringen welches Soundkartenmodell und welcher Chip benutzt wird. FreeBSD unterstützt eine Reihe Soundkarten. Die Hardware-Notes zählen alle unterstützten Karten und deren Treiber für FreeBSD auf.

Um die Soundkarte benutzen zu können, muss der richtige Gerätetreiber geladen werden. Dies kann auf zwei Arten erreicht werden. Am einfachsten ist es, das Kernelmodul für die Soundkarte mit kldload(8) zu laden. Dieses Beispiel lädt den Treiber für eine Creative SoundBlaster® Live! Soundkarte:

```
# kldload snd_emu10k1
```

Um den Treiber automatisch beim Systemstart zu laden, fügen Sie folgende Zeile in `/boot/loader.conf` ein:

```
snd_emu10k1_load="YES"
```

Weitere ladbare Soundmodule sind in `/boot/defaults/loader.conf` aufgeführt. Wenn Sie nicht sicher sind, welchen Gerätetreiber Sie laden müssen, laden Sie das Modul `snd_driver`:

```
# kldload snd_driver
```

Der Treiber `snd_driver` ist ein Meta-Treiber, der alle gebräuchlichen Treiber lädt und die Suche nach dem richtigen Treiber vereinfacht. Durch Hinzufügen des Meta-Treibers in `/boot/loader.conf` können alternativ alle Audio-Treiber geladen werden.

Um zu ermitteln, welcher Treiber für die Soundkarte vom Metatreiber `snd_driver` geladen wurde, geben Sie `cat /dev/sndstat` ein.

Alternativ kann die Unterstützung für die Soundkarte direkt in den Kernel einkompiliert werden. Diese Methode wird im nächsten Abschnitt beschrieben. Weitere Informationen über den Bau eines Kernels finden Sie im Kapitel 9, *Konfiguration des FreeBSD-Kernels*.

8.2.1.1. Soundkarten in der Kernelkonfiguration einrichten

Bei der Verwendung eines eigenen Kernels müssen Sie sicherstellen, dass der Treiber für das Audio-Framework in der Kernelkonfigurationsdatei vorhanden ist:

```
device sound
```

Als Nächstes muss die Unterstützung für die Soundkarte hinzugefügt werden. Dafür müssen Sie wissen, welcher Treiber für die Karte benötigt wird. Um das Beispiel mit der Creative SoundBlaster® Live! Soundkarte aus dem vorherigen Abschnitt fortzusetzen, verwenden Sie die folgende Zeile in der Kernelkonfigurationsdatei:

```
device snd_emu10k1
```

Die richtige Syntax für die Zeile lesen Sie bitte in der Hilfeseite des entsprechenden Treibers nach. Die korrekte Syntax für alle unterstützten Treiber finden Sie außerdem in `/usr/src/sys/conf/NOTES` .

Nicht PnP-fähige ISA-Soundkarten benötigen eventuell Einstellungen, wie IRQ und I/O-Port in `/boot/device.hints` . Während des Systemstarts liest der loader(8) diese Datei und reicht die Einstellungen an den Kernel weiter. Für eine alte Creative SoundBlaster® 16 ISA-Karte, die sowohl den snd_sbc(4)- als auch den `snd_sb16`-Treiber benötigt, müssen die folgenden Zeilen in die Kernelkonfigurationsdatei eingetragen werden:

```
device snd_sbc
device snd_sb16
```

Wenn die Karte den I/O-Port `0x220` und IRQ `5` benutzt, müssen folgende Zeilen zusätzlich in `/boot/device.hints` hinzugefügt werden:

```
hint.sbc.0.at="isa"
hint.sbc.0.port="0x220"
hint.sbc.0.irq="5"
hint.sbc.0.drq="1"
hint.sbc.0.flags="0x15"
```

Die Syntax für `/boot/device.hints` wird in sound(4), sowie in der Manualpage des jeweiligen Treibers beschrieben.

Das Beispiel verwendet die vorgegebenen Werte. Falls die Karteneinstellungen andere Werte vorgeben, müssen die Werte in der Kernelkonfiguration angepasst werden. Weitere Informationen zu dieser Soundkarte finden Sie in snd_sbc(4).

8.2.2. Die Soundkarte testen

Nachdem Sie den neuen Kernel gestartet oder das erforderliche Modul geladen haben, sollte die Soundkarte in den Systemmeldungen auftauchen. Starten Sie dmesg(8) und suchen Sie nach Meldungen wie diesen:

```
pcm0: <Intel ICH3 (82801CA)> port 0xdc80-0xdcbf,0xd800-0xd8ff irq 5 at device 31.5 on ↺
pci0
pcm0: [GIANT-LOCKED]
pcm0: <Cirrus Logic CS4205 AC97 Codec>
```

Der Status der Karte kann auch mit diesem Kommando geprüft werden:

```
# cat /dev/sndstat
FreeBSD Audio Driver (newpcm)
Installed devices:
pcm0: <Intel ICH3 (82801CA)> at io 0xd800, 0xdc80 irq 5 bufsz
16384
kld snd_ich (1p/2r/0v channels duplex default)
```

Die Ausgabe kann auf jedem System anders aussehen. Wenn das Gerät `pcm` nicht erscheint, prüfen Sie die Kernelkonfigurationsdatei und stellen Sie sicher, dass der richtige Treiber gewählt wurde. Abschnitt 8.2.2.1, „Häufige Probleme“ beschreibt häufig auftretende Probleme.

Wenn alles glatt lief, sollte die Soundkarte jetzt unter FreeBSD funktionieren. Wenn ein CD-ROM oder DVD-ROM-Laufwerk an die Soundkarte angeschlossen ist, können Sie jetzt mit cdcontrol(1) eine CD abspielen:

```
% cdcontrol -f /dev/acd0 play 1
```

Es gibt viele Anwendungen, wie audio/workman, die eine bessere Benutzerschnittstelle besitzen. Zur Wiedergabe von MP3-Audiodateien kann audio/mpg123 installiert werden.

Eine weitere schnelle Möglichkeit die Karte zu prüfen, ist es, Daten an das Gerät `/dev/dsp` zu senden:

```
% cat Datei > /dev/dsp
```

Für *`Datei`* kann eine beliebige Datei verwendet werden. Wenn Sie einige Geräusche hören, funktioniert die Soundkarte.

Anmerkung

Die Gerätedateien `/dev/dsp*` werden automatisch erzeugt, wenn sie das erste Mal benötigt werden. Werden sie nicht verwendet, sind sie hingegen nicht vorhanden und tauchen daher auch nicht in der Ausgabe von ls(1) auf.

Die Einstellungen des Mixers können mit mixer(8) verändert werden. Weitere Details finden Sie in mixer(8).

8.2.2.1. Häufige Probleme

Fehler	Lösung
sb_dspwr(XX) timed out	Der I/O Port ist nicht korrekt angegeben.
bad irq XX	Der IRQ ist falsch angegeben. Stellen Sie sicher, dass der angegebene IRQ mit dem Sound IRQ übereinstimmt.
xxx: gus pcm not attached, out of memory	Es ist nicht genug Speicher verfügbar, um das Gerät zu betreiben.
xxx: can't open /dev/dsp!	Überprüfen Sie mit `fstat | grep dsp` ob eine andere Anwendung das Gerät geöffnet hat. Häufige Störenfriede sind esound oder die Sound-Unterstützung von KDE.

Ein weiterer Fall ist der, dass moderne Grafikkarten oft auch ihre eigenen Soundtreiber mit sich führen, um HDMI oder ähnliches zu verwenden. Diese Audiogeräte werden manchmal vor der eigentlichen, separaten Soundkarte aufgeführt und dadurch nicht als das Standardgerät zum Abspielen von Tönen benutzt. Um zu prüfen, ob das bei Ihnen der Fall ist, führen Sie dmesg aus und suchen Sie nach der Zeichenfolge `pcm`. Die Ausgabe sieht in etwa so aus:

```
...
hdac0: HDA Driver Revision: 20100226_0142
hdac1: HDA Driver Revision: 20100226_0142
hdac0: HDA Codec #0: NVidia (Unknown)
hdac0: HDA Codec #1: NVidia (Unknown)
hdac0: HDA Codec #2: NVidia (Unknown)
hdac0: HDA Codec #3: NVidia (Unknown)
pcm0: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 0 nid 1 on hdac0
pcm1: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 1 nid 1 on hdac0
pcm2: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 2 nid 1 on hdac0
pcm3: <HDA NVidia (Unknown) PCM #0 DisplayPort> at cad 3 nid 1 on hdac0
hdac1: HDA Codec #2: Realtek ALC889
pcm4: <HDA Realtek ALC889 PCM #0 Analog> at cad 2 nid 1 on hdac1
pcm5: <HDA Realtek ALC889 PCM #1 Analog> at cad 2 nid 1 on hdac1
pcm6: <HDA Realtek ALC889 PCM #2 Digital> at cad 2 nid 1 on hdac1
pcm7: <HDA Realtek ALC889 PCM #3 Digital> at cad 2 nid 1 on hdac1
...
```

Hier wurde die Grafikkarte (`NVidia`) vor der Soundkarte (`Realtek ALC889`) aufgeführt. Um die Soundkarte als Standardabspielgerät einzusetzen, ändern Sie `hw.snd.default_unit` auf die Einheit, welche für das Abspielen benutzt werden soll:

```
# sysctl hw.snd.default_unit= n
```

Hier repräsentiert `n` die Nummer der Soundkarte, die verwendet werden soll, in diesem Beispiel also `4`. Sie können diese Änderung dauerhaft machen, indem Sie die folgende Zeile zu `/etc/sysctl.conf` hinzufügen:

```
hw.snd.default_unit=4
```

8.2.3. Mehrere Tonquellen abspielen

Beigetragen von Munish Chopra.

Oft sollen mehrere Tonquellen gleichzeitig abgespielt werden. FreeBSD verwendet dazu *virtuelle Tonkanäle*, die mit sysctl(8) aktiviert werden können. Virtuelle Kanäle mischen die Tonquellen im Kernel, sodass mehrere Kanäle benutzt werden können, als von der Hardware unterstützt werden.

Die Anzahl der virtuellen Kanäle können über drei sysctl(8) Optionen eingestellt werden:

```
# sysctl dev.pcm.0.play.vchans=4
# sysctl dev.pcm.0.rec.vchans=4
# sysctl hw.snd.maxautovchans=4
```

Im Beispiel werden vier virtuelle Kanäle eingerichtet, eine im Normalfall ausreichende Anzahl. Sowohl `dev.pcm.0.play.vchans=4` und `dev.pcm.0.rec.vchans=4` sind die Anzahl der virtuellen Kanäle des Geräts `pcm0`, die fürs Abspielen und Aufnehmen verwendet werden und sie können konfiguriert werden, sobald das Gerät existiert. `hw.snd.maxautovchans` ist die Anzahl der virtuellen Kanäle, die einem Gerät zugewiesen werden, wenn es durch kldload(8) eingerichtet wird. Da das Modul `pcm` unabhängig von den Hardware-Treibern geladen werden kann, gibt `hw.snd.maxautovchans` die Anzahl der virtuellen Kanäle an, die später eingerichtete Geräte erhalten. Lesen Sie dazu pcm(4) für weitere Informationen.

Anmerkung

Die Anzahl der virtuellen Kanäle kann nicht geändert werden, solange das Gerät genutzt wird. Schließen Sie daher zuerst alle Programme wie Musikabspielprogramme oder Sound-Daemonen, die auf dieses Gerät zugreifen.

Die korrekte `pcm`-Gerätedatei wird automatisch zugeteilt, wenn ein Programm das Gerät `/dev/dsp0` anfordert.

8.2.4. Den Mixer einstellen

Beigetragen von Josef El-Rayes.

Die Voreinstellungen des Mixers sind im Treiber pcm(4) fest kodiert. Es gibt zwar viele Anwendungen und Dienste, die den Mixer einstellen können und die eingestellten Werte bei jedem Start wieder setzen, am einfachsten ist es allerdings, die Standardwerte für den Mixer direkt im Treiber einzustellen. Der Mixer kann mit den entsprechenden Werten in `/boot/device.hints` eingestellt werden:

```
hint.pcm.0.vol="50"
```

Die Zeile setzt die Lautstärke des Mixers beim Laden des Moduls pcm(4) auf den Wert `50`.

8.3. MP3-Audio

Ein Beitrag von Chern Lee.
Übersetzt von Benedikt Köhler.

Dieser Abschnitt beschreibt einige unter FreeBSD verfügbare MP3-Player. Zudem wird beschrieben, wie Audio-CDs gerippt und MP3s kodiert und dekodiert werden.

8.3.1. MP3-Player

XMMS ist ein beliebter graphischer MP3-Player. WinAmp-Skins können auch mit XMMS genutzt werden, da die Benutzerschnittstelle fast identisch mit der von Nullsofts WinAmp ist. Daneben unterstützt XMMS auch eigene Plugins.

XMMS kann als multimedia/xmms Port oder Paket installiert werden.

Die Benutzerschnittstelle von XMMS ist leicht zu erlernen und enthält eine Playlist, einen graphischen Equalizer und vieles mehr. Diejenigen, die mit WinAmp vertraut sind, werden XMMS sehr leicht zu benutzen finden.

Der Port audio/mpg123 ist ein alternativer, kommandozeilenorientierter MP3-Player.

mpg123 kann ausgeführt werden, indem das zu benutzende Audiogerät und die abzuspielende MP3-Datei auf der Kommandozeile angegeben wird. Wenn das Audiogerät beispielsweise `/dev/dsp1.0` ist und die MP3-Datei *`Foobar-GreatestHits.mp3`* heißt, geben Sie Folgendes ein um die Datei abzuspielen:

```
# mpg123 -a /dev/dsp1.0  Foobar-GreatestHits.mp3
```

```
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2 and 3.
Version 0.59r (1999/Jun/15).  Written and copyrights by Michael Hipp.
Uses code from various people.  See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN RISK!




Playing MPEG stream from Foobar-GreatestHits.mp3 ...
MPEG 1.0 layer III, 128 kbit/s, 44100 Hz joint-stereo
```

8.3.2. CD-Audio Tracks rippen

Bevor eine ganze CD oder einen CD-Track in das MP3-Format umgewandelt werden kann, müssen die Audiodaten von der CD auf die Festplatte gerippt werden. Dabei werden die CDDA (CD Digital Audio) Rohdaten in WAV-Dateien kopiert.

Die Anwendung `cdda2wav`, die im sysutils/cdrtools Paket enthalten ist, kann zum Rippen der Audiodaten und anderen Informationen von CDs genutzt werden.

Wenn die Audio CD in dem Laufwerk liegt, kann der folgende Befehl als `root` ausgeführt werden, um eine ganze CD in einzelne WAV-Dateien (eine Datei für jeden Track) zu rippen:

```
# cdda2wav -D 0,1,0 -B
```

Der Schalter `-D 0,1,0` bezieht sich auf das SCSI Device `0,1,0`, das sich aus dem Ergebnis des Befehls `cdrecord -scanbus` ergibt.

cdda2wav unterstützt auch ATAPI (IDE) CD-ROM-Laufwerke. Um von einem IDE-Laufwerk zu rippen, übergeben Sie auf der Kommandozeile statt der SCSI-IDs den Gerätenamen. Das folgende Kommando rippt den 7. Track:

```
# cdda2wav -D /dev/acd0 -t 7
```

Um einzelne Tracks zu rippen, benutzen Sie `-t` wie folgt:

```
# cdda2wav -D 0,1,0 -t 7
```

Dieses Beispiel rippt den siebten Track der Audio CD-ROM. Um mehrere Tracks zu rippen, zum Beispiel die Tracks eins bis sieben, können Sie wie folgt einen Bereich angeben:

```
# cdda2wav -D 0,1,0 -t 1+7
```

Mit dd(1) können ebenfalls Audio-Stücke von ATAPI-Laufwerken kopiert werden. Dies wird im Abschnitt 18.6.5, „Kopieren von Audio-CDs“ erläutert.

8.3.3. MP3-Dateien kodieren

Lame ist ein weitverbreiteter MP3-Encoder, der als Port audio/lame installiert werden kann. Aus lizenzrechtlichen Gründen ist kein Paket verfügbar.

Der folgende Befehl konvertiert die gerippte WAV-Datei *audio01.wav* in *audio01.mp3* um:

```
# lame -h -b 128 \
--tt "Foo Liedtitel " \
--ta "FooBar Künstler " \
--tl "FooBar Album " \
--ty "2001" \
--tc "Geripped und kodiert von Foo " \
--tg "Musikrichtung " ↺
\
audio01.wav audio01.mp3
```

128 kbits ist die gewöhnliche MP3-Bitrate. Die Bitraten 160 oder 192 kbits bieten eine höhere Qualität. Je höher die Bitrate ist, desto mehr Speicherplatz benötigt die resultierende MP3-Datei. `-h` verwendet den „higher quality but a little slower“ (höhere Qualität, aber etwas langsamer) Modus. Die Schalter, die mit `--t` beginnen, sind ID3-Tags, die in der Regel Informationen über das Lied enthalten und in die MP3-Datei eingebettet sind. Weitere Optionen können in der Manualpage von lame nachgelesen werden.

8.3.4. MP3-Dateien dekodieren

Um aus MP3-Dateien eine Audio CD zu erstellen, müssen diese zuerst in ein nicht komprimiertes WAV-Format umgewandelt werden. Sowohl XMMS als auch mpg123 unterstützen die Ausgabe der MP3-Dateien in unkomprimierte Dateiformate.

Dekodieren mit XMMS:

1. Starten Sie XMMS.
2. Klicken Sie mit der rechten Maustaste, um das XMMS-Menu zu öffnen.
3. Wählen Sie `Preferences` im Untermenü `Options`.
4. Ändern Sie das Output-Plugin in „Disk Writer Plugin“.
5. Drücken Sie `Configure`.
6. Geben Sie ein Verzeichnis ein, in das Sie die unkomprimierte Datei schreiben wollen.
7. Laden Sie die MP3-Datei wie gewohnt in XMMS mit einer Lautstärke von 100% und einem abgeschalteten EQ.
8. Drücken Sie `Play` und es wird so aussehen, als spiele XMMS die MP3-Datei ab, aber keine Musik ist zu hören. Der Player überspielt die MP3-Datei in eine Datei.
9. Vergessen Sie nicht, das Output-Plugin wieder in den Ausgangszustand zurückzusetzen um wieder MP3-Dateien anhören zu können.

Mit mpg123 nach stdout schreiben:

- Geben Sie `mpg123 -s` *`audio01.mp3`* `>` *`audio01.pcm`* ein.

XMMS schreibt die Datei im WAV-Format aus während mpg123 die MP3-Datei in rohe PCM-Audiodaten umwandelt. cdrecord kann mit beiden Formaten Audio-CDs erstellen, burncd(8) kann nur rohe PCM-Audiodaten verarbeiten. Der Dateikopf von WAV-Dateien erzeugt am Anfang des Stücks ein Knacken. Sie können den Dateikopf mit dem Werkzeug SoX, das sich als Paket oder aus dem Port audio/sox installieren lässt, entfernen:

```
% sox -t wav -r 44100 -s -w -c 2  track.wav track.raw
```

Lesen Sie Abschnitt 18.6, „Erstellen und Verwenden von CDs“, um mehr Informationen zur Benutzung von CD-Brennern mit FreeBSD zu erhalten.

8.4. Videos wiedergeben

Beigetragen von Ross Lippert.

Bevor Sie beginnen, sollten Sie das Modell der Videokarte und den benutzten Chip kennen. Obwohl Xorg viele Videokarten unterstützt, können nur einige Karten Videos schnell genug wiedergeben. Eine Liste der Erweiterungen, die der Xorg-Server für eine Videokarte unterstützt, erhalten Sie unter laufendem X11 mit xdpyinfo(1).

Halten Sie eine kurze MPEG-Datei bereit, mit der Sie Wiedergabeprogramme und deren Optionen testen können. Da einige DVD-Spieler in der Voreinstellung das DVD-Gerät mit `/dev/dvd` ansprechen oder diesen Namen fest einkodiert haben, ist es vielleicht hilfreich symbolische Links auf die richtigen Geräte anzulegen:

```
# ln -sf /dev/acd0 /dev/dvd
# ln -sf /dev/acd0 /dev/rdvd
```

Aufgrund der Beschaffenheit devfs(5) gehen gesondert angelegte Links wie diese bei einem Neustart des Systems verloren. Damit die symbolischen Links automatisch beim Neustart des Systems angelegt werden, fügen Sie die folgenden Zeilen in `/etc/devfs.conf` ein:

```
link acd0 dvd
link acd0 rdvd
```

Das Entschlüsseln von DVDs erfordert den Aufruf bestimmter DVD-ROM-Funktionen, sowie Schreibzugriff auf die DVD-Geräte.

Xorg benutzt Shared-Memory und es wird empfohlen, die nachstehenden sysctl(8)-Variablen auf die gezeigten Werte zu erhöhen:

```
kern.ipc.shmmax=67108864
kern.ipc.shmall=32768
```

8.4.1. Video-Schnittstellen

Es gibt einige Möglichkeiten, Videos unter Xorg abzuspielen. Welche Möglichkeit funktioniert, hängt stark von der verwendeten Hardware ab.

Gebräuchliche Video-Schnittstellen sind:

1. Xorg: normale Ausgabe über Shared-Memory.
2. XVideo: Eine Erweiterung der Xorg-Schnittstelle, die Videos in jedem X11-Drawable anzeigen kann.
3. SDL: Simple DirectMedia Layer.
4. DGA: Direct Graphics Access.
5. SVGAlib: Eine Schnittstelle zur Grafikausgabe auf der Konsole.

8.4.1.1. XVideo

Die Erweiterung *XVideo* (auch Xvideo, Xv oder xv) von Xorg erlaubt die beschleunigte Wiedergabe von Videos in jedem Drawable. Diese Erweiterung liefert auch auf weniger leistungsfähigen Systemen eine gute Wiedergabe.

Ob die Erweiterung läuft, entnehmen Sie der Ausgabe von `xvinfo`:

```
% xvinfo
```

XVideo wird untertsützt, wenn die Ausgabe wie folgt aussieht:

```
X-Video Extension version 2.2
screen #0
  Adaptor #0: "Savage Streams Engine"
    number of ports: 1
    port base: 43
    operations supported: PutImage
    supported visuals:
      depth 16, visualID 0x22
      depth 16, visualID 0x23
    number of attributes: 5
      "XV_COLORKEY" (range 0 to 16777215)
              client settable attribute
              client gettable attribute (current value is 2110)
      "XV_BRIGHTNESS" (range -128 to 127)
              client settable attribute
```

```
              client gettable attribute (current value is 0)
      "XV_CONTRAST" (range 0 to 255)
              client settable attribute
              client gettable attribute (current value is 128)
      "XV_SATURATION" (range 0 to 255)
              client settable attribute
              client gettable attribute (current value is 128)
      "XV_HUE" (range -180 to 180)
              client settable attribute
              client gettable attribute (current value is 0)
    maximum XvImage size: 1024 x 1024
    Number of image formats: 7
      id: 0x32595559 (YUY2)
        guid: 59555932-0000-0010-8000-00aa00389b71
        bits per pixel: 16
        number of planes: 1
        type: YUV (packed)
      id: 0x32315659 (YV12)
        guid: 59563132-0000-0010-8000-00aa00389b71
        bits per pixel: 12
        number of planes: 3
        type: YUV (planar)
      id: 0x30323449 (I420)
        guid: 49343230-0000-0010-8000-00aa00389b71
        bits per pixel: 12
        number of planes: 3
        type: YUV (planar)
      id: 0x36315652 (RV16)
        guid: 52563135-0000-0000-0000-000000000000
        bits per pixel: 16
        number of planes: 1
        type: RGB (packed)
        depth: 0
        red, green, blue masks: 0x1f, 0x3e0, 0x7c00
      id: 0x35315652 (RV15)
        guid: 52563136-0000-0000-0000-000000000000
        bits per pixel: 16
        number of planes: 1
        type: RGB (packed)
        depth: 0
        red, green, blue masks: 0x1f, 0x7e0, 0xf800
      id: 0x31313259 (Y211)
        guid: 59323131-0000-0010-8000-00aa00389b71
        bits per pixel: 6
        number of planes: 3
        type: YUV (packed)
      id: 0x0
        guid: 00000000-0000-0000-0000-000000000000
        bits per pixel: 0
        number of planes: 0
        type: RGB (packed)
        depth: 1
        red, green, blue masks: 0x0, 0x0, 0x0
```

Einige der aufgeführten Formate, wie YUV2 oder YUV12 existieren in machen XVideo-Implementierungen nicht. Dies kann zu Problemen mit einigen Spielern führen.

XVideo wird wahrscheinlich von der Karte nicht unterstützt, wenn die Ausgabe wie folgt aussieht:

```
X-Video Extension version 2.2
screen #0
no adaptors present
```

Wenn die XVideo-Erweiterung auf der Karte nicht läuft, wird es nur etwas schwieriger, die Anforderungen für die Wiedergabe von Videos zu erfüllen. Abhängig von der Videokarte und dem Prozessor können Sie dennoch zufriedenstellende Ergebnisse erzielen.

8.4.1.2. Simple DirectMedia Layer

Der Simple DirectMedia Layer, SDL, ist eine portable Schnittstelle für verschiedene Betriebssysteme, mit denen Anwendungen plattformübergreifend und effizient Ton und Grafik benutzen können. SDL bietet eine hardwarenahe Schnittstelle, die manchmal schneller als die Xorg-Schnittstelle sein kann.

SDL kann als Paket oder Port devel/sdl12installiert werden.

8.4.1.3. Direct Graphics Access

Direct Graphics Access (DGA) ist eine Xorg-Erweiterung die es Anwendungen erlaubt, am Xorg-Server vorbei direkt in den Framebuffer zu schreiben. Da die Anwendung und der Xorg-Server auf gemeinsame Speicherbereiche zugreifen, müssen die Anwendungen unter dem Benutzer `root` laufen.

Die DGA-Erweiterung kann mit dga(1) getestet werden. `dga` wechselt jedes Mal, wenn eine Taste gedrückt wird, die Farben der Anzeige. Um das Programm zu beenden, drücken Sie q.

8.4.2. Video-Anwendungen

Dieser Abschnitt behandelt Anwendungen aus der FreeBSD-Ports-Sammlung, die für die Wiedergabe von Videos genutzt werden können.

Viele unter FreeBSD laufende Videoanwendungen wurden unter Linux® entwickelt und befinden sich noch im Beta-Status. Der Betrieb dieser Anwendungen unter FreeBSD stößt vielleicht auf einige der nachstehenden Probleme:

1. Eine Anwendung kann eine Datei einer anderen Anwendung nicht abspielen.
2. Eine Anwendung kann eine selbst produzierte Datei nicht abspielen.
3. Wenn dieselbe Anwendung auf unterschiedlichen Maschinen gebaut wird, wird ein Video unterschiedlich wiedergegeben.
4. Ein vergleichsweise einfacher Filter, wie die Skalierung eines Bildes, führt zu deutlichen Artefakten in der Darstellung.
5. Eine Anwendung stürzt häufig ab.
6. Die Dokumentation wird bei der Installation des Ports nicht installiert. Sie befindet sich entweder auf dem Internet oder im Verzeichnis `work` des Ports.

Viele Anwendungen sind zudem sehr „Linux®-lastig“. Probleme entstehen durch die Implementierung von Standard-Bibliotheken in Linux®-Distributionen oder dadurch, dass die Anwendung bestimmte Linux®-Kernelfunktionen voraussetzt. Diese Probleme werden nicht immer vom Betreuer eines Ports bemerkt und umgangen. In der Praxis können dadurch folgende Probleme entstehen:

1. Eigenschaften des Prozessors werden über `/proc/cpuinfo` ermittelt.
2. Die falsche Anwendung von Threads führt dazu, dass sich ein Programm aufhängt statt sich zu beenden.
3. Die Anwendung hängt von anderen Anwendungen ab, die sich noch nicht in der FreeBSD Ports-Sammlung befinden.

8.4.2.1. MPlayer

MPlayer ist ein auf Geschwindigkeit und Flexibilität ausgelegter Video-Spieler für die Kommandozeile mit optionaler graphischer Oberfläche. Diese Anwendung, sowie weitere graphische Oberflächen stehen in der FreeBSD Ports-Sammlung zur Verfügung.

8.4.2.1.1. MPlayer bauen

MPlayer ist als Paket oder Port multimedia/mplayer verfügbar. Der Bau von MPlayer berücksichtigt die vorhandene Hardware und es können zahlreiche Optionen ausgewählt werden. Aus diesen Gründen ziehen es manche Benutzer vor, den Port zu übersetzen, anstatt das Paket zu installieren.

```
# cd /usr/ports/multimedia/mplayer
# make
```

Die Optionen sollten überprüft werden, um dem Umfang der Unterstützung, mit dem der Port gebaut wird, zu bestimmen. Wenn eine Option nicht ausgewählt wird, ist MPlayer nicht in der Lage, diese Art von Video-Format wiederzugeben. Mit den Pfeiltasten und der Leertaste können die erforderlichen Formate ausgewählt werden. Wenn Sie fertig sind, drücken Sie Enter, um den Bau und die Installation fortzusetzen.

In der Voreinstellung wird das Paket oder der Port das `mplayer`-Kommandozeilenprogramm und das graphische Programm `gmplayer` bauen. Um Videos zu dekodieren, installieren Sie den Port multimedia/mencoder. Aus lizenzrechtlichen Gründen steht ein Paket von `MEncoder` nicht zur Verfügung.

8.4.2.1.2. MPlayer benutzen

MPlayer erstellt beim ersten Start `~/.mplayer` im Heimatverzeichnis des Benutzers. Dieses Verzeichnis enthält die voreingestellten Konfigurationseinstellungen für den Benutzer.

Dieser Abschnitt beschreibt nur ein paar wenige Anwendungsmöglichkeiten. Eine vollständige Beschreibung der zahlreichen Möglichkeiten finden Sie in der Manualpage von mplayer(1).

Um die Datei *`testfile.avi`* abzuspielen, geben Sie die Video-Schnittstelle mit `-vo` an:

```
% mplayer -vo xv testfile.avi
```

```
% mplayer -vo sdl testfile.avi
```

```
% mplayer -vo x11 testfile.avi
```

```
# mplayer -vo dga testfile.avi
```

```
# mplayer -vo 'sdl:dga' testfile.avi
```

Es lohnt sich, alle Option zu testen. Die erzielte Geschwindigkeit hängt von vielen Faktoren ab und variiert beträchtlich je nach eingesetzter Hardware.

Wenn Sie eine DVD abspielen wollen, ersetzen Sie *`testfile.avi`* durch `-dvd://`*`N Gerät`*`.` *`N`* ist die Nummer des Stücks, das Sie abspielen wollen und *`Gerät`* gibt den Gerätenamen des DVD-ROMs an. Das nachstehende Kommando spielt das dritte Stück von `/dev/dvd`:

```
# mplayer -vo dga -dvd://3 /dev/dvd
```

Anmerkung

Das standardmäßig verwendete DVD-Laufwerk kann beim Bau des MPlayer-Ports mit der Option `WITH_DVD_DEVICE=/pfad/zum/gerät` festgelegt werden. Die Voreinstellung verwendet das Gerät `/dev/acd0`. Weitere Details finden Sie in `Makefile.options` des Ports.

Die Tastenkombinationen zum Abbrechen, Anhalten und Weiterführen der Wiedergabe entnehmen Sie der Ausgabe von `mplayer -h` oder der Manualpage.

Weitere nützliche Optionen für die Wiedergabe sind `-fs -zoom` zur Wiedergabe im Vollbild-Modus und `-framedrop` zur Steigerung der Geschwindigkeit.

Jeder Benutzer kann häufig verwendete Optionen in seine `~/.mplayer/config` eintragen:

```
vo=xv
```

```
fs=yes
zoom=yes
```

`mplayer` kann verwendet werden, um DVD-Stücke in `.vob`-Dateien zu rippen. Das zweite Stück einer DVD wandeln Sie wie folgt in eine Datei um:

```
# mplayer -dumpstream -dumpfile out.vob -dvd://2 /dev/dvd
```

Die Ausgabedatei `out.vob` wird im MPEG-Format abgespeichert und kann mit anderen Werkzeugen aus diesem Abschnitt bearbeitet werden.

Jeder Benutzer, der mehr Informationen über Video unter UNIX® sammeln möchte, sollte mplayerhq.hu/DOCS konsultieren, da es technisch sehr informativ ist. Diese Dokumentation sollte ebenfalls studiert werden, bevor Fehlerberichte eingereicht werden.

8.4.2.1.3. MEncoder

Vor der Verwendung von `mencoder` ist es hilfreich, sich mit den auf mplayerhq.hu/DOCS/HTML/en/mencoder.html beschriebenen Optionen vertraut zu machen. Es gibt unzählige Möglichkeiten die Qualität zu verbessern, die Bitrate zu verringern und Formate zu konvertieren. Einige davon haben erhebliche Auswirkungen auf die Geschwindigkeit. Falsche Kombinationen von Kommandozeilenparametern ergeben eventuell Dateien, die selbst `mplayer` nicht mehr wiedergeben kann.

Hier ist ein Beispiel für eine einfache Kopie:

```
% mencoder input.avi -oac copy -ovc copy -o output.avi
```

Wenn Sie in eine Datei rippen, benutzen Sie die Option `-dumpfile` von `mplayer`.

Um *`input.avi`* nach MPEG4 mit MPEG3 für den Ton zu konvertieren, muss zunächst der Port audio/lame installiert werden. Aus lizenzrechtlichen Gründen ist ein Paket nicht verfügbar. Wenn der Port installiert ist, geben Sie ein:

```
% mencoder input.avi -oac mp3lame -lameopts br=192 \
    -ovc lavc -lavcopts vcodec=mpeg4:vhq -o  output.avi
```

Die Ausgabedatei lässt sich mit Anwendungen wie `mplayer` oder `xine` abspielen.

`input.avi` kann durch `-dvd://1 /dev/dvd` ersetzt und das Kommando als `root` ausgeführt werden, um ein DVD-Stück direkt zu konvertieren. Da vielleicht ein paar Versuche nötig sind, um das gewünschte Ergebnis zu erhalten, empfiehlt es sich das Stück zuerst in eine Datei zu schreiben und anschließend die Datei weiter zu bearbeiten.

8.4.2.2. Der Video-Spieler xine

xine ist ein Video-Spieler mit einer wiederverwendbaren Bibliothek und ein Programm, das durch Plugins erweitert werden kann. Es kann als Paket oder Port multimedia/xine installiert werden.

Für einen reibungslosen Betrieb benötigt xine entweder eine schnelle CPU mit einer schnellen Grafikkarte, oder die XVideo-Erweiterung. Am schnellsten läuft xine mit der XVideo-Erweiterung.

In der Voreinstellung startet xine eine grafische Benutzeroberfläche. Über die Menüs können dann bestimmte Dateien geöffnet werden.

Alternativ kann xine auch über die Kommandozeile aufgerufen werden, um Dateien direkt wiederzugeben:

```
% xine -g -p mymovie.avi
```

Weitere Informationen und Tipps zur Fehlerbehebung finden Sie unter xine-project.org/faq.

8.4.2.3. Die transcode-Werkzeuge

transcode ist eine Sammlung von Werkzeugen zur Umwandlung von Video- und Audio-Dateien. transcode mischt Video-Dateien und kann kaputte Video-Dateien reparieren. Die Werkzeuge werden als Filter verwendet, das heißt die Ein- und Ausgaben verwenden `stdin/stdout` .

transcode kann als Paket oder Port multimedia/transcode installiert werden. Viele Benutzer bevorzugen es den Port zu bauen, da er ein Menü bereitstellt, wo die entsprechenden Formate für den Bau ausgewählt werden können. Mit den Pfeiltasten und der Leertaste können die erforderlichen Formate ausgewählt werden. Wenn Sie fertig sind, drücken Sie Enter, um den Bau und die Installation fortzusetzen.

Dieses Beispiel zeigt, wie eine DivX-Datei in eine PAL MPEG-1-Datei konvertiert wird:

```
% transcode -i input.avi -V --export_prof vcd-pal -o output_vcd
% mplex -f 1 -o output_vcd.mpg output_vcd.m1v output_vcd.mpa
```

Die daraus resultierende MPEG-Datei, `output_vcd.mpg`, kann beispielsweise mit MPlayer abgespielt werden. Die Datei kann auch mit einem Programm wie multimedia/vcdimager oder sysutils/cdrdao als Video-CD auf eine CD-R gebrannt werden.

Zusätzlich zu der Manualpage von `transcode`, sollten Sie auch die Informationen und Beispiele im transcode-Wiki lesen.

8.5. TV-Karten einrichten

Beigetragen von Josef El-Rayes.
Überarbeitet von Marc Fonvieille.

8.5.1. Einführung

Mit TV-Karten können Sie mit dem Rechner über Kabel oder Antenne fernsehen. Die meisten Karten besitzen einen RCA- oder S-Video-Eingang. Einige Karten haben auch einen FM-Radio-Empfänger.

Der bktr(4)-Treiber von FreeBSD unterstützt PCI-TV-Karten mit einem Brooktree Bt848/849/878/879 oder einem Conexant CN-878/Fusion 878a Chip. Die Karte sollte einen der unterstützten Empfänger besitzen, die in bktr(4) aufgeführt sind.

8.5.2. Den Treiber laden

Um die Karte benutzen zu können, muss der bktr(4)-Treiber geladen werden. Damit dies beim Systemstart automatisch erfolgt, muss die folgende Zeile in `/boot/loader.conf` hinzugefügt werden:

```
bktr_load="YES"
```

Alternativ kann der Treiber für die TV-Karte auch fest in den Kernel kompiliert werden. In diesem Fall müssen folgende Zeilen in die Kernelkonfigurationsdatei aufgenommen werden:

```
device  bktr
device iicbus
device iicbb
device smbus
```

Die zusätzlichen Treiber werden benötigt, da die Komponenten der Karte über einen I2C-Bus verbunden sind. Bauen und installieren Sie dann den neuen Kernel.

Um den Treiber zu testen, muss das System neu gestartet werden. Während des Neustarts sollte die TV-Karte erkannt werden:

```
bktr0: <BrookTree 848A> mem 0xd7000000-0xd7000fff irq 10 at device 10.0 on pci0
iicbb0: <I2C bit-banging driver> on bti2c0
iicbus0: <Philips I2C bus> on iicbb0 master-only
iicbus1: <Philips I2C bus> on iicbb0 master-only
smbus0: <System Management Bus> on bti2c0
bktr0: Pinnacle/Miro TV, Philips SECAM tuner.
```

Abhängig von der verwendeten Hardware können die Meldungen natürlich anders aussehen. Sie sollten aber prüfen, dass der Empfänger richtig erkannt wird. Die entdeckten Geräte lassen sich mit sysctl(8) oder in der Kernel-

konfigurationsdatei überschreiben. Wenn Sie beispielsweise einen Philips-SECAM-Empfänger erzwingen wollen, fügen Sie die folgende Zeile zur Kernelkonfigurationsdatei hinzu:

```
options OVERRIDE_TUNER=6
```

Alternativ können Sie sysctl(8) benutzen:

```
# sysctl hw.bt848.tuner=6
```

Weitere Informationen zu den verschiedenen Optionen finden Sie in bktr(4) sowie in `/usr/src/sys/conf/NOTES` .

8.5.3. Nützliche Anwendungen

Um die TV-Karte zu benutzen, installieren Sie eine der nachstehenden Anwendungen:

- multimedia/fxtv lässt das Fernsehprogramm in einem Fenster laufen und kann Bilder, Audio und Video aufzeichnen.
- multimedia/xawtv eine weitere TV-Anwendung mit vergleichbaren Funktionen.
- Mit audio/xmradio lässt sich der FM-Radio-Empfänger, der sich auf TV-Karten befindet, benutzen.

Weitere Anwendungen finden Sie in der FreeBSD Ports-Sammlung.

8.5.4. Fehlersuche

Wenn Sie Probleme mit der TV-Karte haben, prüfen Sie zuerst, ob der Video-Capture-Chip und der Empfänger vom bktr(4)-Treiber unterstützt werden und ob Sie die richtigen Optionen verwenden. Weitere Hilfe zu unterstützten TV-Karten finden Sie auf der Mailingliste freebsd-multimedia und in deren Archiven.

8.6. MythTV

MythTV ist eine beliebte Open Source PVR-Anwendung. Dieser Abschnitt beschreibt die Installation und Konfiguration von MythTV unter FreeBSD. Weitere Informationen zur Benutzung von MythTV finden Sie unter mythtv.org/wiki.

MythTV benötigt ein Frontend und ein Backend. Diese Komponenten können entweder auf dem gleichen System, oder auf unterschiedlichen Maschinen installiert werden.

Für das Frontend wird multimedia/mythtv-frontend, sowie ein X-Server benötigt, welcher in x11/xorg zu finden ist. Idealerweise besitzt der Frontend-Computer auch eine Videokarte, die XvMC unterstützt, sowie optional eine LIRC-kompatible Fernbedienung.

Für das Backend wird multimedia/mythtv benötigt, ebenso wie eine MySQL™-Datenbank, sowie zusätzlich einen Empfänger und Speicherplatz für Aufzeichnungen. Das MySQL™-Paket sollte automatisch als Abhängigkeit mitinstalliert werden, wenn multimedia/mythtv installiert wird.

8.6.1. Hardware

MythTV wurde entwickelt, um V4L zu verwenden, so dass auf Videoeingabegeräte wie Kodierer und Empfänger zugegriffen werden kann. Aktuell funktioniert MythTV am besten mit USB DVB-S/C/T Karten, die von multimedia/webcamd unterstützt werden, da dies eine V4L-Anwendung zur Verfügung stellt, die als Benutzerprogramm läuft. Jede DVB-Karte, die von webcamd unterstützt wird, sollte mit MythTV funktionieren, jedoch gibt es eine Liste von Karten, die hier abgerufen werden kann. Es existieren auch Treiber für Hauppauge-Karten in den folgenden Paketen: multimedia/pvr250 und multimedia/pvrxxx, allerdings liefern diese nur eine Treiberschnittstelle, die nicht dem Standard entspricht und die nicht mit MythTV-Versionen grösser als 0.23 funktionieren. Aus lizenzrechtlichen Gründen ist ein Paket nicht verfügbar, sodass die beiden Ports übersetzt werden müssen.

Die HTPC Wiki Seite enthält eine Liste von allen verfügbaren DVB-Treibern.

8.6.2. MythTV einrichten

Geben Sie folgendes ein, um MythTV aus der Ports-Sammlung zu installieren:

```
# cd /usr/ports/multimedia/mythtv
# make install
```

Richten Sie anschließend die MythTV-Datenbank ein:

```
# mysql -uroot -p < /usr/local/share/mythtv/database/mc.sql
```

Konfigurieren Sie dann das Backend:

```
# mythtv-setup
```

Zum Schluss starten Sie das Backend:

```
# echo 'mythbackend_enable="YES"' >> /etc/rc.conf
# service mythbackend start
```

8.7. Scanner

Beigetragen von Marc Fonvieille.

Unter FreeBSD stellt SANE (Scanner Access Now Easy) aus der Ports-Sammlung eine einheitliche Schnittstelle (API) für den Zugriff auf Scanner bereit. SANE wiederum greift auf Scanner mithilfe einiger FreeBSD-Treiber zu.

FreeBSD unterstützt sowohl SCSI- als auch USB-Scanner. Prüfen Sie vor der Konfiguration mithilfe der Liste der unterstützten Geräte ob der Scanner von SANE unterstützt wird.

8.7.1. Den Kernel für Scanner einrichten

Da sowohl SCSI- als auch USB-Scanner unterstützt werden, werden abhängig von der Schnittstelle unterschiedliche Treiber benötigt.

8.7.1.1. USB-Scanner

Im `GENERIC` -Kernel sind schon alle, für einen USB-Scanner notwendigen, Treiber enthalten. Benutzer mit einem angepassten Kernel sollten sicherstellen, dass die Kernelkonfiguration die nachstehenden Zeilen enthält:

```
device usb
device uhci
device ohci
device ehci
```

Schließen Sie den USB-Scanner an. Prüfen Sie mit dmesg(8), ob der Scanner in den Systemmeldungen erscheint:

```
ugen0.2: <EPSON> at usbus0
```

Diese Meldung besagt, dass der Scanner entweder die Gerätedatei `/dev/ugen0.2` , oder `/dev/uscanner0` benutzt. Dies ist abhängig von der verwendeten FreeBSD Version. In diesem Beispiel wurde ein EPSON Perfection® 1650 USB-Scanner verwendet.

8.7.1.2. SCSI-Scanner

Wenn der Scanner eine SCSI-Schnittstelle besitzt, ist die Kernelkonfiguration abhängig vom verwendeten SCSI-Controller. Der `GENERIC` -Kernel unterstützt die gebräuchlichen SCSI-Controller. Den richtigen Treiber finden Sie in `/usr/src/sys/conf/NOTES` . Neben dem Treiber muss die Kernelkonfiguration noch die nachstehenden Zeilen enthalten:

```
device scbus
```

```
device pass
```

Nachdem Sie einen Kernel gebaut und installiert haben, sollte der Scanner beim Neustart in den Systemmeldungen erscheinen:

```
pass2 at aic0 bus 0 target 2 lun 0
pass2: <AGFA SNAPSCAN 600 1.10> Fixed Scanner SCSI-2 device
pass2: 3.300MB/s transfers
```

Wenn der Scanner während des Systemstarts ausgeschaltet war, können Sie die Geräteerkennung erzwingen, indem Sie den SCSI-Bus erneut absuchen. Verwenden Sie dazu das Kommando camcontrol(8):

```
# camcontrol rescan all
Re-scan of bus 0 was successful
Re-scan of bus 1 was successful
Re-scan of bus 2 was successful
Re-scan of bus 3 was successful
```

Der Scanner sollte jetzt in der SCSI-Geräteliste erscheinen:

```
# camcontrol devlist
<IBM DDRS-34560 S97B>              at scbus0 target 5 lun 0 (pass0,da0)
<IBM DDRS-34560 S97B>              at scbus0 target 6 lun 0 (pass1,da1)
<AGFA SNAPSCAN 600 1.10>           at scbus1 target 2 lun 0 (pass3)
<PHILIPS CDD3610 CD-R/RW 1.00>     at scbus2 target 0 lun 0 (pass2,cd0)
```

Weitere Informationen über SCSI-Geräte unter FreeBSD finden Sie in scsi(4) und camcontrol(8).

8.7.2. SANE konfigurieren

SANE besteht aus zwei Teilen, den Backends (graphics/sane-backends) und den Frontends (graphics/sane-frontends). Das Backend greift auf den Scanner zu. Welches Backend welchen Scanner unterstützt, entnehmen Sie der Liste der unterstützten Geräte. Der Betrieb eines Scanners ist nur mit dem richtigen Backend möglich. Die Frontends sind die Anwendungen, mit denen gescannt wird (xscanimage).

Installieren Sie zuerst den Port oder das Paket graphics/sane-backends. Anschließend können Sie mit dem Befehl `sane-find-scanner` prüfen, ob SANE den Scanner erkennt:

```
# sane-find-scanner -q
found SCSI scanner "AGFA SNAPSCAN 600 1.10" at /dev/pass3
```

Die Ausgabe zeigt die Schnittstelle und die verwendete Gerätedatei des Scanners. Der Hersteller und das Modell können in der Ausgabe fehlen.

Anmerkung

Bei einigen USB-Scannern muss die Firmware geladen werden. Lesen Sie sane-find-scanner(1) und sane(7) für weitere Details.

Als nächstes müssen Sie prüfen, ob der Scanner vom Frontend erkannt wird. Die SANE-Backends werden mit dem Kommandozeilenwerkzeug scanimage(1) geliefert. Mit diesem Werkzeug können Sie sich Scanner anzeigen lassen und den Scan-Prozess von der Kommandozeile starten. Die Option `-L` zeigt die Scanner an:

```
# scanimage -L
device `snapscan:/dev/pass3' is a AGFA SNAPSCAN 600 flatbed scanner
```

Hier ist die Ausgabe für das Beispiel mit dem USB-Scanner in Abschnitt 8.7.1.1, „USB-Scanner“:

```
# scanimage -L
device 'epson2:libusb:/dev/usb:/dev/ugen0.2' is a Epson GT-8200 flatbed scanner
```

Die Zeile `'epson2:libusb:/dev/usb:/dev/ugen0.2'` nennt das Backend (`epson2`) und die Gerätedatei (`/dev/ugen0.2`), die der Scanner verwendet.

Anmerkung

Erscheint die Meldung, dass kein Scanner gefunden wurde oder wird gar keine Ausgabe erzeugt, konnte scanimage(1) keinen Scanner erkennen. In diesem Fall müssen Sie in der Konfigurationsdatei des Backends das zu benutzende Gerät eintragen. Die Konfigurationsdateien der Backends befinden sich in `/usr/local/etc/sane.d/` .

In dem obigen Beispiel wurde der Scanner erkannt und ist funktionsfähig.

Um festzustellen, ob der Scanner richtig erkannt wurde:

```
# scanimage -L

No scanners were identified. If you were expecting something different,
check that the scanner is plugged in, turned on and detected by the
sane-find-scanner tool (if appropriate). Please read the documentation
which came with this software (README, FAQ, manpages).
```

Da der Scanner nicht erkannt wurde, muss `/usr/local/etc/sane.d/epson2.conf` editiert werden. Der verwendete Scanner in diesem Beispiel war ein EPSON Perfection® 1650, daher wird das `epson2` -Backend benutzt. Lesen Sie bitte alle Kommentare in der Konfigurationsdatei des Backends. Die durchzuführenden Änderungen sind einfach. Kommentieren Sie zunächst alle Zeilen mit der falschen Schnittstelle aus. Da der Scanner eine USB-Schnittstelle besitzt, wurden im Beispiel alle Zeilen, die mit `scsi` anfingen, auskommentiert. Fügen Sie dann die Schnittstelle und den Gerätenamen am Ende der Datei ein. In diesem Beispiel wurde die nachstehende Zeile eingefügt:

```
usb /dev/uscanner0
```

Speichern Sie die Änderungen und prüfen Sie, ob der Scanner richtig erkannt wird:

```
# scanimage -L
device `epson:/dev/uscanner0' is a Epson GT-8200 flatbed scanner
```

Das Feld `` `epson:/dev/uscanner0' `` zeigt jetzt das richtige Backend und den richtigen Gerätenamen an.

Wenn `scanimage -L` den Scanner erkannt hat, ist der Scanner eingerichtet und bereit, zu scannen.

Obwohl scanimage(1) von der Kommandozeile scannen kann, ist eine graphische Anwendung zum Scannen besser geeignet. SANE bietet ein einfaches und effizientes Werkzeug: xscanimage.

Xsane, das über den Port oder das Paket graphics/xsane installiert wird, ist eine weitere beliebte graphische Anwendung. Dieses Frontend besitzt erweiterte Funktionen wie den Scan-Modus, eine Farbkorrektur und Batch-Scans. Beide Anwendungen lassen sich als GIMP-Plugin verwenden.

8.7.3. Den Scanner für Benutzerkonten freigeben

Wenn andere Benutzer den Scanner benutzen sollen, müssen sie Lese- und Schreibrechte auf die Gerätedatei des Scanners besitzen. Im vorherigen Beispiel wird die Datei `/dev/ugen0.2` verwendet, die faktisch nur ein Symlink auf die echte Gerätedatei, `/dev/usb/0.2.0` genannt, darstellt. Sowohl der Symlink als auch die Gerätedatei sind jeweils im Besitz der Gruppen `wheel` und `operator`. Damit ein Benutzer den Scanner benutzen kann, muss er Mitglied in einer der beiden Gruppen sein. Allerdings sollte aus Sicherheitsgründen genau überlegt werden, welche Benutzer zu welcher Gruppe hinzugefügt werden, besonders bei der Gruppe `wheel`. Eine bessere Lösung ist es, eine

spezielle Gruppe für den Zugriff auf USB-Geräte anzulegen und den Scanner für Mitglieder dieser Gruppe zugänglich zu machen.

Dieses Beispiel nutzt pw(8), um eine Gruppe namens *usb* zu erstellen:

```
# pw groupadd usb
```

Anschliessend muss der `/dev/ugen0.2` -Symlink und der Gerätename `/dev/usb/0.2.0` für die `usb`-Gruppe mit den Schreibrechten (`0660` oder `0664`) ausgestattet werden. All dies kann durch das Hinzufügen der folgenden Zeile in `/etc/devfs.rules` erreicht werden:

```
[system=5]
add path ugen0.2 mode 0660 group usb
add path usb/0.2.0 mode 0660 group usb
```

Jetzt müssen nur noch Benutzer zur Gruppe *usb* hinzugefügt werden, um ihnen den Zugriff auf den Scanner zu erlauben:

```
#pw groupmod usb -m joe
```

Weitere Details finden Sie in pw(8).

Kapitel 9. Konfiguration des FreeBSD-Kernels

9.1. Übersicht

Der Kernel ist das Herz des FreeBSD-Betriebssystems. Er ist verantwortlich für die Speicherverwaltung, das Durchsetzen von Sicherheitsdirektiven, Netzwerkfähigkeit, Festplattenzugriffen und vieles mehr. Obwohl FreeBSD es ermöglicht, dynamisch konfiguriert zu werden, ist es ab und an notwendig, einen angepassten Kernel zu konfigurieren und zu kompilieren.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie Folgendes wissen:

- Wann Sie einen angepassten Kernel kompilieren sollten.
- Wie Sie eine Hardware-Inventur durchführen.
- Wie Sie eine Kernelkonfigurationsdatei verändern.
- Wie Sie mit der Konfigurationsdatei einen neuen Kernel kompilieren.
- Wie Sie den neuen Kernel installieren.
- Was zu tun ist, falls etwas schiefgeht.

Alle Kommandos, aus den Beispielen dieses Kapitels, müssen mit `root`-Rechten ausgeführt werden.

9.2. Wieso einen eigenen Kernel bauen?

Traditionell besaß FreeBSD einen monolithischen Kernel. Der Kernel war ein einziges großes Programm, das eine bestimmte Auswahl an Hardware unterstützte. Um das Kernelverhalten zu ändern, musste man einen neuen Kernel kompilieren und dann den neuen Kernel booten.

Heutzutage befinden sich die meisten Funktionen des FreeBSD-Kernels in Modulen, die je nach Bedarf dynamisch geladen und entladen werden können. Dies erlaubt es, einen laufenden Kernel anzupassen, um sofort neue Hardware oder neue Funktionen zu unterstützen. Dies ist als modularer Kernel bekannt.

Gelegentlich ist es noch notwendig, eine statische Kernelkonfigurationen durchzuführen. In einigen Fällen ist die Funktion zu systemnah, um durch ein Modul realisiert zu werden. Andere Umgebungen verhindern vielleicht das Laden und Entladen von Kernelmodulen und erfordern, dass nur die benötigte Funktionalität statisch in den Kernel kompiliert wird.

Das Erstellen eines angepassten Kernels ist eines der Rituale für erfahrene BSD-Benutzer. Obwohl dieser Prozess recht viel Zeit in Anspruch nimmt, kann er doch viele Vorteile für das FreeBSD-System bringen. Im Gegensatz zum `GENERIC` -Kernel, der eine Vielzahl von Hardware unterstützen muss, kann ein angepasster Kernel so eingeschränkt werden, dass er nur noch die Hardware des Rechners unterstützt. Dies hat einige Vorteile:

- Schnellerer Bootvorgang. Da der Kernel nur nach der Hardware des Systems sucht, kann sich die Zeit für einen Systemstart verkürzen.
- Geringerer Speicherbedarf. Ein eigener Kernel benötigt in der Regel weniger Speicher als ein `GENERIC` -Kernel durch das Entfernen von Funktionen und Gerätetreibern. Das ist vorteilhaft, denn der Kernel verweilt immer im RAM und verhindert dadurch, dass dieser Speicher von Anwendungen genutzt wird. Deshalb ist ein angepasster Kernel auf einem System mit wenig RAM sinnvoll.

- Zusätzliche Hardwareunterstützung. Ein angepasster Kernel kann Unterstützung für Geräte bieten, die im `GENERIC` -Kernel nicht enthalten sind.

Bevor Sie einen angepassten Kernel erstellen, überlegen Sie sich bitte, warum Sie dies tun wollen. Wenn Sie lediglich eine bestimmte Hardwareunterstützung benötigen, existiert diese vielleicht schon als Kernelmodul.

Kernelmodule existieren in `/boot/kernel` und können mit kldload(8) dynamisch in den laufenden Kernel geladen werden. Die meisten Kerneltreiber verfügen über ein ladbares Modul und eine Manualpage. Der drahtlose Ethernet-Treiber ath(4) hat die folgenden Informationen in seiner Manualpage:

```
Alternatively, to load the driver as a module at boot time, place the
following line in loader.conf(5):

    if_ath_load="YES"
```

Durch das Hinzufügen von `if_ath_load="YES"` in `/boot/loader.conf` wird das Modul dynamisch beim Systemstart geladen.

In manchen Fällen gibt es kein entsprechendes Modul in `/boot/kernel` . Dies gilt insbesondere für bestimmte Subsysteme.

9.3. Informationen über die vorhandene Hardware beschaffen

Bevor die Kernelkonfigurationsdatei bearbeitet wird, ist es empfehlenswert eine Bestandsaufnahme der Hardware des Systems durchzuführen. Auf einem Dual-Boot-System können diese Informationen aus dem anderen Betriebssystem ermittelt werden. Microsoft®s Gerätemanager enthält beispielsweise Informationen über die installierte Hardware.

Anmerkung

Einige Versionen von Microsoft® Windows® verfügen über ein System-Icon auf dem Desktop, über das Sie den Gerätemanager direkt aufrufen können.

Wenn FreeBSD das einzige installierte Betriebssystem ist, dann listet dmesg(8) die Hardware auf, die während des Systemstarts gefunden wurde. Die meisten FreeBSD-Gerätetreiber haben eine eigene Manualpage, die Informationen über die unterstützte Hardware enthält. Die folgenden Zeilen zeigen beispielsweise an, dass der psm(4)-Treiber eine angeschlossene Maus gefunden hat:

```
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: [ITHREAD]
psm0: model Generic PS/2 mouse, device ID 0
```

Da diese Hardware vorhanden ist, sollte dieser Treiber nicht aus einer angepassten Kernelkonfigurationsdatei entfernt werden.

Wenn `dmesg` keine Informationen zur gefundenen Hardware anzeigt, können diese Informationen auch aus `/var/run/dmesg.boot` entnommen werden.

Ein weiteres Werkzeug für die Suche nach Hardware ist pciconf(8), das ausführliche Informationen bereitstellt. Ein Beispiel:

```
% pciconf -lv
ath0@pci0:3:0:0:        class=0x020000 card=0x058a1014 chip=0x1014168c rev=0x01 hdr=0x00
    vendor     = 'Atheros Communications Inc.'
    device     = 'AR5212 Atheros AR5212 802.11abg wireless'
    class      = network
    subclass   = ethernet
```

Die Ausgabe zeigt, dass der Treiber `ath` eine drahtlose Ethernetkarte gefunden hat.

Die Option `-k` von man(1) kann verwendet werden, um nützliche Informationen zu erhalten. Um beispielsweise eine Liste von Manualpages zu erhalten, welche ein spezifisches Wort enthalten:

```
# man -k Atheros
```

```
ath(4)                  - Atheros IEEE 802.11 wireless network driver
ath_hal(4)              - Atheros Hardware Access Layer (HAL)
```

Mit einer Inventarliste der Hardware können Sie dann sicherstellen, dass Sie die Treiber der installierten Hardware nicht versehentlich entfernen, wenn Sie die Kernelkonfigurationsdatei bearbeiten.

9.4. Die Kernelkonfigurationsdatei

Bevor eine angepasste Kernelkonfigurationsdatei erstellt werden kann, muss zuerst der vollständige FreeBSD Quellcodebaum installiert werden.

Falls `/usr/src/` nicht existiert oder leer ist, sind die Kernelquellen nicht installiert. Die Quellen können mit Subversion und der Anleitung im Abschnitt A.5, „Benutzen von Subversion“ installiert werden.

Sobald die Quellen installiert sind, können Sie sich einen Überblick über `/usr/src/sys` verschaffen. Dieses Verzeichnis enthält eine Reihe von Unterverzeichnissen, einschließlich Verzeichnisse für die unterstützten Architekturen `amd64`, `i386`, `ia64`, `pc98`, `powerpc` und `sparc64`. Alles in diesen Verzeichnissen ist nur für die jeweilige Architektur relevant. Der Rest des Codes ist machinenunabhängig und für alle Architekturen gleich. Jede unterstützte Architektur hat ein Unterverzeichnis `conf`, das die `GENERIC` Kernelkonfigurationsdatei für diese Architektur enthält.

Bearbeiten Sie `GENERIC` nicht direkt. Kopieren Sie stattdessen die Datei unter einem anderen Namen und machen dann die Änderungen an dieser Kopie. Traditionell besteht der Name des Kernels immer aus Großbuchstaben. Wenn Sie mehrere FreeBSD-Maschinen mit unterschiedlicher Hardware betreuen, ist es eine gute Idee, die Konfigurationsdatei nach den Hostnamen der Maschinen zu benennen. In diesem Beispiel wird eine Kopie der `GENERIC` Kernelkonfigurationsdatei, namens `MYKERNEL`, für die `amd64`-Architektur erstellt:

```
# cd /usr/src/sys/amd64/conf
# cp GENERIC MYKERNEL
```

`MYKERNEL` kann jetzt mit einem Texteditor bearbeitet werden. Der Standard-Editor ist vi, jedoch steht mit ee ein weiterer, einfach zu bedienender Editor bereit.

Das Format der Konfigurationsdatei ist einfach. Jede Zeile enthält ein Schlüsselwort, das ein Gerät oder ein Subsystem repräsentiert, ein Argument und eine kurze Beschreibung. Jeder Text, der hinter einem `#` steht, gilt als Kommentar und wird ignoriert. Um die Kernel-Unterstützung für ein Gerät oder Subsystem zu entfernen, muss ein `#` an den Anfang der Zeile, die dieses Gerät oder Subsystem repräsentiert, gesetzt werden. Verändern Sie keine Zeilen, die Sie nicht genau verstehen.

Neben den Kurzbeschreibungen in dieser Datei, finden Sie zusätzliche Erklärungen in `NOTES`, die sich in demselben Verzeichnis wie `GENERIC` für die jeweilige Architektur befindet. Von der Architektur unabhängige Optionen sind in `/usr/src/sys/conf/NOTES` aufgeführt.

Tipp

Wenn Sie die Kernelkonfigurationsdatei fertig bearbeitet haben, sollten Sie eine Sicherungskopie außerhalb von `/usr/src` speichern

Alternativ kann die Kernelkonfigurationsdatei an anderer Stelle gespeichert, und ein symbolischer Link auf die Datei erstellt werden:

```
# cd /usr/src/sys/amd64/conf
# mkdir /root/kernels
# cp GENERIC /root/kernels/MYKERNEL
# ln -s /root/kernels/MYKERNEL
```

Es ist möglich, eine `include`-Anweisung in die Kernelkonfigurationsdatei aufzunehmen. Diese erlaubt das lokale Einfügen von anderen Konfigurationsdateien in die aktuelle, was es einfacher macht, kleinere Änderungen an einer existierenden Datei zu vollziehen. Wenn Sie einen `GENERIC` -Kernel mit nur einer kleinen Anzahl von zusätzlichen Optionen und Treibern benötigen, brauchen Sie mit den folgenden Zeilen nur ein kleines Delta im Vergleich zu GENERIC anpassen, wie in diesem Beispiel zu sehen:

```
include GENERIC
ident MYKERNEL

options         IPFIREWALL
options         DUMMYNET
options         IPFIREWALL_DEFAULT_TO_ACCEPT
options         IPDIVERT
```

Diese Methode zeigt die Unterschiede der lokalen Konfigurationsdatei zu einem `GENERIC` -Kernel an. Sobald Aktualisierungen durchgeführt werden, können neue Eigenschaften, die zu `GENERIC` hinzugefügt werden, auch dem lokalen Kernel angehängt werden, es sei denn, es wird durch `nooptions` oder `nodevice` unterbunden. Eine umfassende Liste von Konfigurationseinstellungen und deren Beschreibungen finden Sie in config(5).

Anmerkung

Um einen Kernel mit allen möglichen Optionen zu bauen, führen Sie als `root` die folgenden Befehle aus:

```
# cd /usr/src/sys/ arch /conf && make LINT
```

9.5. Einen angepassten Kernel bauen und installieren

Nachdem die Änderungen an der angepassten Kernelkonfigurationsdatei gespeichert sind, kann der Quellcode für den Kernel mit den folgenden Schritten übersetzt werden:

Prozedur 9.1. Einen Kernel bauen

1. Wechseln Sie das Verzeichnis:

   ```
   # cd /usr/src
   ```

2. Bauen Sie den Kernel, indem Sie den Namen der Kernelkonfigurationsdatei angeben:

   ```
   # make buildkernel KERNCONF= MYKERNEL
   ```

3. Installieren Sie den neuen Kernel. Dieser Befehl wird den neuen Kernel nach `/boot/kernel/kernel` kopieren, und den alten Kernel nach `/boot/kernel.old/kernel` speichern:

   ```
   # make installkernel KERNCONF= MYKERNEL
   ```

4. Fahren Sie das System herunter und starten Sie den neuen Kernel. Wenn etwas nicht funktioniert, lesen Sie Der Kernel bootet nicht:.

In der Voreinstellung werden beim Bau eines angepassten Kernels stets alle Kernelmodule neu gebaut. Um einen Kernel schneller zu bauen, oder um nur bestimmte Module zu bauen, bearbeiten Sie `/etc/make.conf` , bevor Sie den Kernel neu bauen.

In diesem Beispiel werden über eine Variable nur die Kernelmodule definiert, die auch tatsächlich gebaut werden sollen. In der Voreinstellung werden alle Module gebaut:

```
MODULES_OVERRIDE = linux acpi
```

Alternativ kann auch eine Variable verwendet werden, die bestimmte Kernelmodule vom Bauprozess ausschließt:

```
WITHOUT_MODULES = linux acpi sound
```

Weitere Variablen und deren Beschreibung finden Sie in make.conf(5).

9.6. Wenn etwas schiefgeht

Es gibt vier Hauptfehlerquellen beim Erstellen eines angepassten Kernels:

`config` verursacht Fehler:
: Wenn `config` fehlschlägt, zeigt es die Nummer der Zeile an, die das Problem verursacht. Bei der folgenden Fehlermeldung sollten Sie die angegebene Zeile mit `GENERIC` oder `NOTES` vergleichen und sicherstellen, dass das Schlüsselwort in Zeile 17 richtig geschrieben ist:

  ```
  config: line 17: syntax error
  ```

`make` verursacht Fehler:
: Wenn `make` fehlschlägt, liegen meistens Fehler in der Konfigurationsdatei vor, die aber nicht schwerwiegend genug für `config` waren. Überprüfen Sie die Konfiguration und wenn Sie keinen Fehler entdecken können, schicken Sie eine E-Mail mit der Kernelkonfigurationsdatei an die Mailingliste 'Fragen und Antworten zu FreeBSD' <de-bsd-questions@de.FreeBSD.org>.

Der Kernel bootet nicht:
: Wenn der neue Kernel nicht bootet oder die Geräte nicht erkannt werden, ist das noch kein Grund zur Panik. Glücklicherweise besitzt FreeBSD einen exzellenten Mechanismus zur Wiederherstellung nach dem Einsatz inkompatibler Kernel. Wählen Sie einfach den zu bootenden Kernel im FreeBSD Bootloader aus. Dazu wählen Sie im Bootmenü die Option „Escape to a loader prompt“. Danach geben Sie am Prompt `boot` *`kernel.old`* oder den Namen eines anderen Kernels ein, der sauber bootet.

  Nun kann die Konfiguration noch einmal überprüft und der Kernel neu kompiliert werden. Dazu ist `/var/log/messages` sehr nützlich, da hier sämtliche Kernelmeldungen von jedem erfolgreichen Bootvorgang gespeichert werden. dmesg(8) gibt die Kernelmeldungen vom letzten Bootvorgang aus.

Anmerkung

Wenn Sie Probleme beim Kernelbau bekommen, heben Sie sich immer eine Kopie von `GENERIC` oder einen anderen Kernel, der garantiert bootet, auf. Dies ist sehr wichtig, weil jedes Mal, wenn ein neuer Kernel installiert wird, `kernel.old` mit dem zuletzt installierten Kernel überschrieben wird und dieser möglicherweise nicht bootfähig ist. Verschieben Sie daher den funktionierenden Kernel so schnell wie möglich, indem Sie das Verzeichnis mit dem funktionierenden Kernel umbenennen:

```
# mv /boot/kernel  /boot/kernel.bad
# mv /boot/kernel.good  /boot/kernel
```

Der Kernel funktioniert, aber `ps` nicht

Wenn Sie eine andere Version des Kernels installiert haben als die, mit der Ihre Systemwerkzeuge gebaut wurden, beispielsweise einen Kernel aus den -CURRENT-Quellen auf einem -RELEASE-System, werden Programme wie ps(1) und vmstat(8) nicht mehr funktionieren. Um dies zu beheben, sollten Sie das komplette System neu bauen und installieren. Achten Sie darauf, dass die Quellen, aus denen das System gebaut wird, zum installierten Kernel passt. Man sollte niemals einen Kernel, der nicht zur Systemversion passt, benutzen.

Kapitel 10. Drucken

Beigetragen von Sean Kelly.
Restrukturiert und aktualisiert durch Jim Mock.
Übersetzt von Stefan Bethke.

10.1. Übersicht

Mit FreeBSD können Sie viele unterschiedliche Drucker benutzen, von den ältesten Nadeldruckern bis zu den neuesten Laserdruckern, und allen möglichen Geräten dazwischen. Auf diese Weise können Sie hochwertige Ausdrucke mit Ihren Programmen erzeugen.

Sie können FreeBSD auch so konfigurieren, dass es Druckaufträge von anderen Computern über Ihr lokales Netzwerk entgegennimmt, seien es Windows®-, Mac OS®- oder andere FreeBSD-Computer. FreeBSD stellt sicher, dass die Druckaufträge in der richtigen Reihenfolge bearbeitet werden und kann optional ein Deckblatt mit dem Namen des Auftraggebers eines Druckauftrags aufgeben. FreeBSD kann auch Statistiken über die Computer und Benutzer führen, die Ausdrucke in Auftrag geben.

In diesem Kapitel erfahren Sie, wie Sie:

- FreeBSD-Druckerwarteschlangen einrichten.
- Druckfilter installieren, die Druckaufträge je nach Bedarf besonders behandeln und z.B. Dokumente automatisch in eine Form umwandeln, die Ihr Drucker versteht.
- Druckaufträge mit einem Deckblatt versehen können.
- Mit einem Drucker drucken können, der an einen anderen Computer angeschlossen ist.
- Mit einem Drucker drucken können, der direkt an das Netzwerk angeschlossen ist.
- die Größe von Druckaufträgen beschränken können, oder bestimmte Benutzer von den Benutzung des Drucksystems ausschließen können.
- Statistiken aufzeichnen und die Benutzung des Drucksystems nach Benutzern und Computern aufschlüsseln können.
- Probleme beim Drucken diagnostizieren und beheben können.

Bevor Sie dieses Kapitel lesen:

- Machen Sie sich mit der Konfiguration und Installation eines neuen Kernels vertraut (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).

10.2. Einführung

Um einen Drucker mit FreeBSD zu benutzen, können Sie das Berkeley Line Printer Spooling System, das auch als LPD-Drucksystem oder nur als LPD bekannt ist, verwenden. Dieses System zur Verwaltung von Druckaufträgen ist das Standardsystem in FreeBSD. Dieses Kapitel führt Sie in LPD und dessen Konfiguration ein.

Wenn Sie bereits mit LPD oder einem anderen Drucksystem vertraut sind, können Sie direkt im Abschnitt Einfache Drucker-Konfiguration weiterlesen.

LPD steuert alle Aspekte rund um die Drucker, die an den Computer angeschlossen sind. Es ist verantwortlich für:

- Die Zugriffskontrolle für direkt und über das Netzwerk angeschlossene Drucker.

- Die Entgegennahme von Dateien, die gedruckt werden sollen; eine so an das Drucksystem übergebene Datei wird als *Druckauftrag* bezeichnet.
- Den gleichzeitigen Zugriff von mehreren Benutzern auf einen Drucker. Alle Druckaufträge werden in einer *Druckerwarteschlange* gesammelt, und nacheinander abgearbeitet.
- Den Druck von *Deckblättern* (auch als *Banner-* oder *Burst*-Seiten bezeichnet), damit Benutzer ihre Druckaufträge schnell innerhalb eines Stapels von ausgedruckten Dokumenten finden können.
- Das Einstellen der korrekten Kommunikations-Parameter für Drucker, die seriell angeschlossen sind.
- Das Senden von Druckaufträgen an ein LPD-System auf einem anderen Computer.
- Das Ausführen von speziellen Filtern, um Druckaufträge in die unterschiedlichen Seitenbeschreibungssprachen umzusetzen oder an die Fähigkeiten eines Druckers anzupassen.
- Das Erfassen von Verrechnungsdaten für Druckaufträge.

Sie können LPD alle diese Funktionen, oder auch nur einen Teil davon, ausführen lassen, indem Sie die Konfigurationsdatei (`/etc/printcap`) anpassen, und indem Sie spezielle Filterprogramme bereitstellen.

10.2.1. Vorteile des Drucksystems

Wenn Sie der einzige Benutzer sind, der mit Ihrem Computer arbeitet, fragen Sie sich vielleicht, warum Sie die Konfigurationsarbeit für das Drucksystem auf sich nehmen sollten, wenn Sie Deckblätter, Abrechnungsdaten oder Zugriffskontrolle nicht benötigen. Obwohl Sie direkt auf den Drucker zugreifen können, bietet LPD eine Reihe von Vorteilen:

- LPD druckt im Hintergrund; Sie müssen nicht erst darauf warten, dass Ihr Druckauftrag an den Drucker übermittelt worden ist.
- LPD kann Druckaufträge mit Kopf- oder Fußzeilen versehen, oder ein spezielles Dateiformat, wie DVI von TeX, automatisch in ein für den Drucker verständliches Format umwandeln; Sie müssen diese Schritte nicht manuell ausführen.
- Viele freie und kommerzielle Programme, mit denen Sie drucken können, erwarten, mithilfe des LPD-Drucksystems zu drucken. Wenn Sie das Drucksystem konfiguriert haben, können Sie einfacher mit neuer oder auch vorhandener Software drucken.

10.3. Grund-Konfiguration

Um einen Drucker mit dem LPD-Drucksystem benutzen zu können, müssen Sie sowohl Ihren Drucker und die Drucker-Schnittstelle als auch das LPD-Drucksystem konfigurieren. Dieser Abschnitt beschreibt zwei Konfigurationen:

- Abschnitt Einfache Drucker-Konfiguration beschreibt, wie Sie einen Drucker an Ihren Computer anschließen und LPD so konfigurieren, dass Sie Textdateien zum Drucker senden können.
- Abschnitt Erweiterte Drucker-Konfiguration beschreibt, wie Sie mit speziellen Dateiformaten umgehen können, wie Sie Deckblätter drucken können, wie Sie den Zugriff auf Drucker einschränken können, und wie Sie Verrechnungsdaten aufzeichnen können.

10.3.1. Einfache Drucker-Konfiguration

Dieser Abschnitt beschreibt, wie Sie die LPD-Software konfigurieren, um Ihren Drucker zu benutzen. Diese Grundlagen werden erklärt:

- Abschnitt Hardware-Konfiguration erläutert, wie Sie Ihren Drucker an Ihren Computer anschließen können.
- Abschnitt Software-Konfiguration erklärt, wie Sie die LPD-Konfigurationsdatei (`/etc/printcap`) anpassen.

Wenn Sie einen Drucker einrichten möchten, der über das Netzwerk angeschlossen ist (anstatt über die serielle oder parallele Schnittstelle), lesen Sie bitte Abschnitt Drucker mit direkter TCP-Schnittstelle.

Obwohl dieser Abschnitt „Grund-Konfiguration“ heißt, ist die Konfiguration relativ komplex. Es ist vergleichsweise schwierig, einen Drucker mit Ihrem Computer und dem LPD-Drucksystem zu verbinden. Die weiteren Optionen, wie Kopfzeilen oder Deckblätter, sind einfach zu konfigurieren, sobald die Grund-Konfiguration erfolgreich abgeschlossen ist.

10.3.1.1. Hardware-Konfiguration

Dieser Abschnitt beschreibt, über welche Schnittstellen Sie einen Drucker mit Ihrem Computer verbinden können. Er behandelt sowohl die Schnittstellen und Kabel, als auch die Kerneloptionen, die Sie benötigen, um FreeBSD mit Ihrem Drucker kommunizieren zu lassen.

Wenn Sie Ihren Drucker bereits erfolgreich mit einem anderen Betriebssystem auf Ihrem PC eingesetzt haben, können Sie wahrscheinlich mit dem Abschnitt Software-Konfiguration fortfahren.

10.3.1.1.1. Schnittstellen und Kabel

Praktisch alle Drucker unterstützen mindestens eine dieser Schnittstellen:

- *Seriell* angeschlossene Drucker werden über eine serielle Schnittstelle (auch RS-232 oder COM-Schnittstelle genannt) mit Ihrem Computer verbunden. Diese Schnittstelle wird von vielen unterschiedlichen Systemen verwendet. Serielle Kabel sind leicht erhältlich und können auch einfach selbst hergestellt werden. Einige Drucker erfordern möglicherweise ein spezielles Kabel oder besondere Kommunikationseinstellungen. Die meisten seriellen Schnittstellen von PCs besitzen eine maximale Datenübertragungsrate von 115200 bps; zum Ausdruck großer Grafiken sind serielle Drucker daher ungeeignet.

- *Parallel* angeschlossene Drucker werden über eine parallele Schnittstelle mit Ihrem Computer verbunden. Diese Schnittstelle wird hauptsächlich von PCs und Workstations benutzt. Die Schnittstelle bietet eine höhere Datenübertragungsrate als serielle Schnittstellen. Kabel sind leicht erhältlich, sind aber vergleichsweise schwer selbst herzustellen. Üblicherweise brauchen keine Kommunikationsparameter festgelegt zu werden; dies macht die Einrichtung sehr einfach.

  Die parallele Schnittstelle wird auch als „Centronics“-Schnittstelle bezeichnet, nach dem Namen des Steckverbinders, der hier häufig zum Einsatz kommt.

- USB-Schnittstelle (Universal Serial Bus) bieten noch höhere Geschwindigkeiten als parallele Schnittstellen oder serielle RS-232-Schnittstellen. USB-Kabel sind einfach und billig. Zum Drucken ist die USB-Schnittstelle besser geeignet als serielle oder parallele Schnittstellen, auf vielen UNIX® Systemen werden USB-Schnittstellen jedoch nur unzureichend unterstützt. Um Probleme zu vermeiden, sollten Sie sich einen Drucker anschaffen, der sowohl eine USB-Schnittstelle als auch eine parallele Schnittstelle besitzt (viele Drucker besitzen heute beide Schnittstellen).

Im Allgemeinen versenden parallele Schnittstellen Daten nur in eine Richtung (vom Computer zum Drucker), serielle Schnittstellen und USB-Schnittstellen versenden Daten in beide Richtungen. Moderne parallele Schnittstellen (EPP and ECP) übertragen Daten bi-direktional nach dem Standard IEEE 1284.

Ein Drucker kann auf zwei Arten bi-direktional angesprochen werden. Die erste Methode benutzt einen Druckertreiber, der die herstellerspezifische Sprache des Druckers beherrscht. Diese Methode wird oft mit Tintenstrahl-Druckern eingesetzt und dazu benutzt, den Füllstand der Tintenpatronen und andere Status-Informationen auszugeben. Die zweite Methode wird benutzt, wenn der Drucker PostScript® beherrscht.

Da ein PostScript®-Druckauftrag ein komplettes Programm ist, kann es auch Daten an den Computer zurückliefern, ohne überhaupt eine Seite Papier zu bedrucken. Auf diesem Wege werden auch Probleme wie z.B. ein Papierstau vom Drucker an den Computer übermittelt. Darüberhinaus ist dies die effektivste Methode, um die tatsächlich gedruckte Anzahl an Seiten vom Drucker abzufragen: ein PostScript®-Programm ermittelt jeweils vor und direkt nach einem Druckauftrag den Seitenzähler des Druckers, und vergleicht die beiden Zählerwerte.

10.3.1.1.2. Parallele Schnittstellen

Um einen Drucker mit paralleler Schnittstelle an Ihren Computer anzuschließen, verbinden Sie den Drucker mit einer parallelen Schnittstelle Ihres Computers. Die Dokumentation zu Ihrem Drucker oder Computer sollte Ihnen hier weiterhelfen.

Notieren Sie sich, mit welcher parallelen Schnittstelle des Computers Sie den Drucker verbunden haben. Die meisten Computer haben lediglich eine parallele Schnittstelle. Der FreeBSD-Gerätename der ersten Schnittstelle lautet `ppc0`, der der zweiten `ppc1`, und so weiter. Der Gerätename für den Drucker an der ersten parallelen Schnittstelle folgt dem selben Schema und lautet `/dev/lpt0` , usw.

10.3.1.1.3. Serielle Schnittstellen

Um einen Drucker mit serieller Schnittstelle an Ihren Computer anzuschließen, verbinden Sie den Drucker mit einer seriellen Schnittstelle Ihres Computers. Die Dokumentation zu Ihrem Drucker oder Computer sollte Ihnen hier weiterhelfen.

Sollten Sie sich nicht sicher sein, welches das „richtige Kabel“ ist, können Sie eine dieser Alternativen ausprobieren:

- Ein *Modemkabel* verbindet alle Anschlüsse an einem Ende des Kabels eins-zu-eins mit den Anschlüssen am anderen Ende des Kabels. Ein solches Kabel wird auch als (engl.) „DTE-to-DCE-“ oder „DEE-zu-DÜE-“Kabel bezeichnet.
- Ein *Nullmodemkabel* verbindet einige Signale eins-zu-eins, andere über Kreuz (z.B. Sende- und Empfangsleitung), und verbindet einige weitere direkt im Stecker miteinander. Ein solches Kabel wird auch als (engl.) „DTE-to-DTE-“ oder „DEE-zu-DEE-“Kabel bezeichnet.
- Ein *Serielles Druckerkabel* schließlich, das für einige spezielle Drucker benötigt wird, verbindet zusätzliche Signale miteinander, anstatt sie im Stecker zurückzuführen.

Sie sollten auch die Kommunikationsparameter am Drucker einstellen; üblicherweise gibt es dazu DIP-Schalter, oder eine Option in der Menüführung am Drucker. Wählen Sie die höchste *Bitrate* (auch als `bps` Bits pro Sekunde oder *Baudrate* bezeichnet), die sowohl Ihr Drucker als auch Ihr Computer unterstützen. Wählen Sie 7 oder 8 Bits, gerade, ungerade oder keine Parität, und ein oder zwei Stoppbits. Wählen Sie die Art der Flusssteuerung: keine, XON/XOFF (auch als „in-band-“ oder „Software“-Flusssteuerung bezeichnet), oder Hardware. Notieren Sie sich diese Einstellungen, damit Sie sie später bei der Software-Konfiguration zur Verfügung haben.

10.3.1.2. Software-Konfiguration

Dieser Abschnitt beschreibt die notwendigen Konfigurationsschritte, damit Sie mit dem FreeBSD-LPD-System drucken können.

Diese Schritte müssen Sie ausführen:

1. Konfigurieren Sie Ihren Kernel, soweit notwendig, um die Schnittstelle benutzen zu können, an die Ihr Drucker angeschlossen ist. Abschnitt Kernel-Konfiguration erklärt, welche Optionen Sie benötigen.
2. Konfigurieren Sie die Kommunikationseinstellungen für die parallele Schnittstelle, sofern Sie sie benutzen. Abschnitt Kommunikationseinstellungen für die parallele Schnittstelle enthält die Details.
3. Prüfen Sie, ob Sie Daten an den Drucker senden können. Abschnitt Prüfen der Drucker-Kommunikation führt eine Reihe von Möglichkeiten auf.
4. Konfigurieren Sie LPD für Ihren Drucker, indem Sie die Konfigurationsdatei `/etc/printcap` anpassen. Details dazu finden Sie im Abschnitt LPD aktivieren: die `/etc/printcap` -Datei.

10.3.1.2.1. Kernel-Konfiguration

Der Betriebssystem-Kernel ist für eine bestimmte Kombination aus Geräten kompiliert. Dies schließt Ihre seriellen oder parallelen Schnittstellen mit ein. Dementsprechend kann es notwendig sein, die Kernelkonfiguration um weitere Schnittstellen zu erweitern.

So können Sie prüfen, ob Ihr Kernel die serielle Schnittstelle unterstützt, an die Sie den Drucker angeschlossen haben:

```
# grep sioN /var/run/dmesg.boot
```

Ersetzen Sie *N* durch die Nummer der seriellen Schnittstelle, beginnend bei Null. Wenn Sie eine Ausgabe ähnlich der folgenden erhalten, unterstützt ihr Kernel diese Schnittstelle:

```
sio2 at port 0x3e8-0x3ef irq 5 on isa
sio2: type 16550A
```

Erhalten Sie keine Ausgabe, oder eine Fehlermeldung, wird die Schnittstelle nicht korrekt unterstützt.

So können Sie prüfen, ob Ihr Kernel die parallele Schnittstelle unterstützt, an die Sie den Drucker angeschlossen haben:

```
# grep ppcN /var/run/dmesg.boot
```

Ersetzen Sie *N* durch die Nummer der parallelen Schnittstelle, beginnend bei Null. Wenn Sie eine Ausgabe ähnlich der folgenden erhalten, unterstützt ihr Kernel diese Schnittstelle:

```
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
ppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/8 bytes threshold
```

Erhalten Sie keine Ausgabe, oder eine Fehlermeldung, wird die Schnittstelle nicht korrekt unterstützt.

Gegebenenfalls müssen Sie Ihren Kernel umkonfigurieren und neu kompilieren, damit die von Ihnen gewählte Schnittstelle unterstützt wird.

Um Unterstützung für eine serielle Schnittstelle hinzuzufügen, lesen Sie bitte Kapitel Konfiguration des FreeBSD Kernels. Um eine parallele Schnittstelle hinzuzufügen, lesen Sie bitte ebenfalls jenes Kapitel als auch den folgenden Abschnitt Kommunikationseinstellungen für die parallele Schnittstelle.

10.3.1.3. Kommunikationseinstellungen für die parallele Schnittstelle

Wenn Sie die parallele Schnittstelle zur Kommunikation mit Ihrem Drucker benutzen, haben Sie die Wahl zwischen Interrupt-gesteuerter oder Polling-Datenübertragung. Der generische Druckergerätetreiber lpt(4) in FreeBSD benutzt das ppbus(4)-System, das die parallele Schnittstelle mithilfe des ppc(4)-Treibers steuert.

- Die *Interrupt-gesteuerte* Datenübertragung ist die Voreinstellung im GENERIC-Kernel. Der Treiber benutzt eine IRQ-Leitung, um zu erfahren, wann der Drucker weitere Daten empfangen kann.
- Bei der *Polling-Methode* prüft der Treiber in regelmäßigen Abständen, ob weitere Daten übertragen werden können.

Die Interrupt-gesteuerte Methode ist überlicherweise schneller und verbraucht weniger Rechenzeit als die Polling-Methode, es wird jedoch eine eigene IRQ-Leitung für die Schnittstelle benötigt. Darüberhinaus kann es mit einigen Druckermodellen zu Problemen kommen, wenn die Interrupt-gesteuerte Übertragung zum Einsatz kommt.

Die können die Kommunikationseinstellung entweder in der Kernel-Konfiguration wählen, oder mittels des lptcontrol(8)-Programms zur Laufzeit einstellen.

So legen Sie die Kommunikationseinstellung in der Kernel-Konfiguration fest:

1. Ändern Sie Ihre Kernel-Konfigurationsdatei. Finden Sie die Zeile, die mit `device ppc0` beginnt. Wenn Sie die zweite parallele Schnittstelle konfigurieren möchten, suchen Sie nach `device ppc1` , für die dritte Schnittstelle `ppc2`, usw.

   - Um die Interrupt-Steuerung zu aktivieren, passen Sie die folgende Zeile an:

```
hint.ppc.0.irq="N"
```

Ersetzen Sie *N* durch die Nummer der IRQ-Leitung, die dieser parallelen Schnittstelle zugewiesen ist. Stellen Sie sicher, dass Ihre Kernel-Konfigurationsdatei den ppc(4)-Treiber enthält:

```
device ppc
```

- Wenn Sie den Polling-Modus verwenden möchten, entfernen Sie die folgende Zeile aus `/boot/device.hints`:

```
hint.ppc.0.irq="N"
```

Sollte der Treiber die Schnittstelle dennoch im Interrupt-Modus betreiben, könnte dies an der Aktivierung durch das acpi(4)-System in FreeBSD liegen. Bitte prüfen Sie die ACPI- und die BIOS-Konfiguration.

2. Wenn Sie Ihre Kernel-Konfigurationsdatei angepasst haben, kompilieren und installieren Sie nun einen neuen Kernel. Das Kapitel Konfiguration des FreeBSD-Kernels enthält weitere Details dazu.

*So können Sie die Kommunikationseinstellung mit*lptcontrol(8) *ändern:*

1. Um die Interrupt-Steuerung für die Schnittstelle *N* zu aktivieren, geben Sie ein:

```
# lptcontrol -i -d /dev/lptN
```

2. Um den Polling-Modus für die Schnittstelle *N* zu aktivieren, geben Sie ein:

```
# lptcontrol -p -d /dev/lptN
```

Sie können diesen Befehl in `/etc/rc.local` aufnehmen, damit er bei jedem Systemstart automatisch ausgeführt wird. lptcontrol(8) enthält weitere Informationen.

10.3.1.4. Kommunikation mit den Drucker prüfen

Bevor Sie mit der Konfiguration des LPD-Drucksystems fortfahren, sollten Sie sicherstellen, das Sie erfolgreich Daten an Ihren Drucker senden können. Es ist deutlich einfacher, Kommunikations- und Konfigurationsprobleme unabhängig voneinander zu lösen.

Der Drucker kann mit einem Probeausdruck getestet werden. Für alle Drucker, die normalen Text unmittelbar drucken können, bietet sich das Programm lptest(1) an: es produziert alle 96 druckbaren ASCII-Zeichen auf 96 Zeilen.

Für einen PostScript®-Drucker (oder andere Drucker, die eine Seitenbeschreibungssprache verwenden) muss ein passendes Programm an den Drucker gesendet werden, z.B. dieses:

```
%!PS
100 100 moveto 300 300 lineto stroke
310 310 moveto /Helvetica findfont 12 scalefont setfont
(Funktioniert dieser Drucker?) show
showpage
```

Sie können dieses ProstScript-Programm in einer Datei speichern, und mit den Beispielen in den folgenden Abschnitt verwenden.

Anmerkung

Nicht alle Drucker, die eine Seitenbeschreibungssprache verwenden, benötigen ein Test-Programm: z.B. HPs PCL (das auch in vielen kompatiblen Druckern zum Einsatz zu kommt), versteht normalen Text. Besondere Escape-Sequenzen werden benutzt, um die erweiterten

> Möglichkeiten aufzurufen. PostScript®-Drucker können in der Regel keinen normalen Text direkt verarbeiten, weil sie ein PostScript®-Programm erwarten, das eine Seite produziert.

10.3.1.4.1. Einen Paralleldrucker prüfen

Dieser Abschnitt führt vor, wie Sie die Kommunikation mit Ihrem Drucker über die parallele Schnittstelle prüfen können.

So testen Sie einen Drucker an einer parallelen Schnittstelle:

1. Werden Sie `root` mithilfe des su(1)-Befehls.
2. Senden Sie Testdaten an den Drucker.
   - Wenn Ihr Drucker reinen Text direkt drucken kann, verwenden Sie lptest(1):

     ```
     # lptest > /dev/lptN
     ```

     Ersetzen Sie *`N`* durch die Nummer der parallelen Schnittstelle, an die der Drucker angeschlossen ist (angefangen bei Null).
   - Wenn Ihr Drucker PostScript® (oder eine andere Seitenbeschreibungssprache) versteht, senden Sie ein passendes Testprogramm an den Drucker. Geben Sie folgenden Befehl ein:

     ```
     # cat file > /dev/lptN
     ```

     Ersetzen Sie *`N`* durch die Nummer der parallelen Schnittstelle, an die der Drucker angeschlossen ist (angefangen bei Null). Geben Sie nun das Testprogramm ein, Zeile für Zeile. Kontrollieren Sie jede Zeile, bevor Sie die Eingabetaste drücken: Sie können die Zeile später nicht mehr ändern. Zum Schluss tippen Sie Ctrl+D. Wenn Sie ein anderes Zeichen nutzen, um das Ende der Datei anzuzeigen, müssen Sie natürlich die entsprechende Tastenkombination für dieses Zeichen betätigen.

     Sie können das Testprogramm auch in einer Datei speichern, und dann diesen Befehl aufrufen:

     ```
     # cat Testprogramm > /dev/lptN
     ```

     Ersetzen Sie *`Testprogramm`* durch den Dateinamen, unter dem Sie das Testprogramm gespeichert haben.

Der Drucker sollte einige Zeilen oder eine Seite drucken. Machen Sie sich keine Sorgen über falsche Formatierungen: die Software-Konfiguration enthält Informationen zum Umformatieren von Druckaufträgen.

10.3.1.4.2. Einen seriellen Drucker prüfen

Dieser Abschnitt führt vor, wie Sie die Kommunikation mit Ihrem Drucker über die parallele Schnittstelle prüfen können.

So testen Sie einen Drucker an einer seriellen Schnittstelle:

1. Werden Sie `root` mithilfe des su(1)-Befehls.
2. Ändern Sie die Datei `/etc/remote`. Fügen Sie den folgenden Eintrag hinzu:

   ```
   printer:dv=/dev/port :br#bps-rate :pa=parity
   ```

   Ersetzen Sie *`Gerät`* durch den Gerätenamen der seriellen Schnittstelle (`ttyu0` für die erste, `ttyu1` für die zweite, usw.), *`Baudrate`* ist die Geschwindigkeit und *`Parität`* die Parität (`even` für gerade, `odd` für ungerade oder `none` für keine), die Sie am Drucker eingestellt haben.

   Hier ein Beispieleintrag für einen Drucker, der über die dritte serielle Schnittstelle angeschlossen ist, mit 19.200 Baud kommuniziert und keine Parität verwendet:

```
printer:dv=/dev/ttyu2 :br#19200:pa=none
```

3. Verbinden Sie sich mit dem Drucker über den Befehl tip(1):

```
# tip printer
```

Kommt es hierbei zu einer Fehlermeldung, ändern Sie den Eintrag in `/dev/cuaaN` und verwenden Sie `/dev/cuaaN` statt `/dev/ttyuN` .

4. Senden Sie Testdaten an den Drucker.

- Wenn Ihr Drucker reinen Text direkt drucken kann, verwenden Sie lptest(1):

```
% $lptest
```

- Wenn Ihr Drucker PostScript® (oder eine andere Seitenbeschreibungssprache) versteht, senden Sie ein passendes Testprogramm an den Drucker. Geben Sie das Testprogramm ein, Zeile für Zeile. Kontrollieren Sie jede Zeile, bevor Sie die Eingabetaste drücken: Sie können die Zeile später nicht mehr ändern. Zum Schluss tippen Sie `Control-D`.

  Sie können das Testprogramm auch in einer Datei speichern, und dann diesen Befehl aufrufen:

```
% >Testprogramm
```

  Ersetzen Sie *`Testprogramm`* durch den Dateinamen, unter dem Sie das Testprogramm gespeichert haben. Nachdem tip(1) die Datei gesendet hat, tippen Sie `Control-D`.

Der Drucker sollte einige Zeilen oder eine Seite drucken. Machen Sie sich keine Sorgen über falsche Formatierungen: die Software-Konfiguration enthält Informationen zum Umformatieren von Druckaufträgen.

10.3.1.5. LPD aktivieren: die `/etc/printcap`-Datei

Nachdem Sie Ihren Drucker angeschlossen haben, Ihren Kernel richtig konfiguriert haben und erfolgreich einen Testausdruck produziert haben, können Sie nun das LPD-System konfigurieren.

Sie konfigurieren LPD, indem Sie die Datei `/etc/printcap` anpassen. Da LPD die Datei jedes Mal liest, wenn eine Aktion durchgeführt wird, werden Änderungen an der Konfiguration sofort aktiv.

Die printcap(5)-Datei ist einfach aufgebaut. Sie können `/etc/printcap` mit Ihrem bevorzugten Texteditor bearbeiten. Sie verwendet dasselbe Format wie auch `/usr/share/misc/termcap` oder `/etc/remote` . Informationen zum Format finden Sie in cgetent(3).

Die Grund-Konfiguration des LPD-Systems beinhaltet diese Schritte:

1. Wählen Sie einen Namen (und einige praktische Abkürzungen) für die Druckerwarteschlage, und tragen Sie ihn in die `/etc/printcap` -Datei ein. Abschnitt Einen Namen wählen enthält weitere Informationen.

2. Schalten Sie den Druck von Deckblättern aus (dies ist standardmäßig eingeschaltet), indem Sie das Attribut `sh` setzen. Abschnitt Den Druck von Deckblättern ausschalten erklärt, wie Sie dies tun können.

3. Legen Sie ein Pufferverzeichnis für die Warteschlange an, und geben Sie den Pfad mittels des `sd`-Attributs an: siehe Abschnitt Das Pufferverzeichnis anlegen.

4. Geben Sie den Gerätenamen für Ihren Drucker mittels des `lp`-Attributs an: siehe Abschnitt Festlegen der Drucker-Gerätedatei. Ist Ihr Drucker über eine serielle Schnittstelle angeschlossen, benutzen Sie das Attribut `ms#`, wie dies in Abschnitt Festlegen der Kommunikationsparameter beschrieben ist.

5. Installieren Sie einen Filter für reinen Text: siehe Abschnitt Den Textfilter installieren.

6. Testen Sie die Konfiguration, indem Sie etwas mit dem lpr(1)-Befehl drucken. Die Abschnitte Die Konfiguration testen und Fehlersuche und Problembehebung enthalten weitere Informationen.

Anmerkung

Drucker, die eine Seitenbeschreibungssprache wie PostScript® verwenden, können keinen reinen Text drucken. Es wird deshalb angenommen, dass Sie nur solche Dateien drucken, die Ihr Drucker verarbeiten kann.

Viele Anwender erwarten, dass sie normalen Text auf allen Druckern drucken können. Viele Programme, die mit LPD zusammenarbeiten, gehen ebenfalls von dieser Annahme aus. Wenn Sie einen PostScript®-Drucker installieren, und Sie sowohl PostScript®- als auch Textdateien drucken möchten, sollten Sie einen weiteren Konfigurationsschritt ausführen und einen Text-zu-PostScript®-Filter installieren. Der Abschnitt Drucken von reinen Textdateien auf einem PostScript®-Drucker erklärt, wie Sie dies tun können.

10.3.1.5.1. Einen Namen wählen

Der erste einfache Schritt ist, einen Namen für Ihren Drucker zu wählen. Sie können diesen Namen frei wählen, Sie sollten allerdings keine Sonderzeichen oder Umlaute verwenden. Sie können mehrere Alias-Namen vergeben.

Ein Drucker in `/etc/printcap` sollte den Alias `lp` haben. Dieser Name wird standardmäßig von allen Druckbefehlen verwendet, wenn auf der Befehlszeile oder in der `PRINTER`-Umgebungsvariablen kein anderer Drucker angegeben ist.

Ebenso ist es üblich, eine ausführliche Beschreibung des Druckermodells als letzten Alias-Namen zu verwenden.

Sobald Sie einen Namen und einige einfache Alias-Namen ausgewählt haben, tragen Sie sie in die Datei `/etc/printcap` ein. Beginnen Sie die Zeile mit dem Namen des Druckers und fügen Sie alle Alias-Namen an. Trennen Sie die Namen durch den senkrechten Strich `|`. Fügen Sie an das Ende der Zeile einen Doppelpunkt `:` an.

Das folgende Beispiel definiert zwei Drucker, einen Diablo 630 Zeilendrucker, und einen Panasonic KX-P4455 PostScript®-Laserdrucker:

```
#
#  /etc/printcap for host rose
#
rattan|line|diablo|lp|Diablo 630 Line Printer:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:
```

Der erste Drucker hat den Namen `rattan`, und hat die Alias-Namen `line`, `diablo`, `lp` und `Diablo 630 Line Printer`. Da er den Alias-Namen `lp` trägt, wird er standardmäßig von den Druckprogrammen verwendet. Der zweite Drucker heißt `bamboo`, und hat die Alias-Namen `ps`, `PS`, `S`, `panasonic` und `Panasonic KX-P4455 PostScript v51.4` .

10.3.1.5.2. Keine Deckblätter drucken

Standardmäßig druckt das LPD-System ein *Deckblatt* vor jedem Druckauftrag, die den Namen des Druckauftrags, den Benutzer und den Computer angibt. Während der Einrichtung des Systems und beim Testen stört das Deckblatt allerdings, weshalb Sie sie zunächst deaktivieren sollten.

Um den Druck von Deckblättern zu deaktivieren, fügen Sie das Attribut `sh` zur Druckerdefinition in `/etc/printcap` hinzu. Hier ein Beispiel:

```
#
#  /etc/printcap for host rose - no header pages anywhere
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:
```

Beachten Sie die korrekte Formatierung: die beiden Definitionen beginnen auf einer Zeile; weitere Zeilen der Definition sind mit einem Tab-Zeichen eingerückt, und alle Zeilen einer Definition, bis auf die letzte, enden mit dem Backslash \.

10.3.1.5.3. Das Pufferverzeichnis anlegen

Der nächste Schritt ist, das *Pufferverzeichnis* anzulegen. In diesem Verzeichnis werden Druckaufträge zwischengespeichert, während sie gedruckt werden. Gleichzeitig werden hier auch einige Verwaltungsdateien des Systems abgelegt.

Da sich die Dateien in diesem Verzeichnis häufig ändern, ist es üblich, das Verzeichnis unter `/var/spool` anzulegen. Es ist nicht notwendig, Sicherungskopien der Dateien herzustellen; das Verzeichnis kann nötigenfalls leicht mit mkdir(1) wieder angelegt werden.

Es ist auch üblich, dem Verzeichnis denselben Namen wie dem Drucker zu geben:

```
# mkdir /var/spool/printer-name
```

Wenn Sie viele Drucker verwenden, ist es am besten, wenn Sie für die Pufferverzeichnisse ein eigenes Unterverzeichnis in `/var/spool` anlegen, wie dies hier für die beiden Beispieldrucker `rattan` und `bamboo` gezeigt wird:

```
# mkdir /var/spool/lpd
# mkdir /var/spool/lpd/rattan
# mkdir /var/spool/lpd/bamboo
```

Anmerkung

Um zu verhindern, dass alle Benutzer den Inhalt aller Druckaufträge einsehen können, sollten Sie die Rechte auf den Pufferverzeichnissen einschränken. Die Verzeichnisse sollten dem Benutzer `daemon` und der Gruppe `daemon` gehören, und auch nur vom Benutzer und der Gruppe les-, schreib- und durchsuchbar sein. Für unsere Beispieldrucker:

```
# chown daemon:daemon /var/spool/lpd/rattan
# chown daemon:daemon /var/spool/lpd/bamboo
# chmod 770 /var/spool/lpd/rattan
# chmod 770 /var/spool/lpd/bamboo
```

Schließlich müssen Sie dem LPD-System noch mitteilen, wo Sie die Pufferverzeichnisse angelegt haben. Dazu geben Sie in der Definition das Attribut `sd` an:

```
#
#  /etc/printcap for host rose - added spooling directories
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :
```

Beachten Sie, dass der Druckername in der ersten Spalte beginnt, und dass alle Folgezeilen mit einem `Tab` eingerückt sind.

Wenn Sie das `sd`-Attribut nicht angeben, verwendet das System `/var/spool/lpd` als Verzeichnis.

10.3.1.5.4. Festlegen der Drucker-Gerätedatei

Nachdem Sie die korrekte Gerätedatei für die Schnittstelle im Abschnitt Hardware-Konfiguration identifiziert und angelegt haben, müssen Sie dem LPD-System mitteilen, welche Gerätedatei im Verzeichnis `/dev` es für die Datenübertragung zum Drucker verwenden soll.

Geben Sie die Gerätedatei durch das Attribut `lp` in `/etc/printcap` an.

Wenn `rattan` an die erste parallele Schnittstelle angeschlossen ist, und `bamboo` an die sechste serielle, dann sieht `/etc/printcap` so aus:

```
#
#  /etc/printcap for host rose - identified what devices to use
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :\
        :lp=/dev/lpt0 :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        sh:sd=/var/spool/lpd/bamboo :\
        :lp=/dev/ttyu5 :
```

Wenn Sie `lp` nicht angeben, versucht LPD die Gerätedatei `/dev/lp` zu verwenden. `/dev/lp` ist zurzeit in FreeBSD nicht definiert.

Wenn Ihr Drucker über eine parallele Schnittstelle angeschlossen ist, können Sie mit dem Abschnitt Den Textfilter installieren fortfahren. Verwenden Sie eine serielle Schnittstelle, beachten Sie bitte den folgenden Abschnitt.

10.3.1.5.5. Kommunikationsparameter festlegen

Für seriell angeschlossene Drucker kann LPD die Geschwindigkeit, Parität und weitere Kommunikationsparameter einstellen. Dies hat folgende Vorteile:

- Sie können die Parameter einfach in `/etc/printcap` ändern, ohne das Ausgabe-Filterprogramm anpassen zu müssen.
- Dasselbe Ausgabe-Filterprogramm kann für unterschiedliche Drucker verwendet werden, auch wenn diese unterschiedliche Kommunikationseinstellungen benötigen.

Die folgenden Attribute legen die seriellen Kommunikationsparameter fest:

`br#Baudrate`
: Setzt die Übertragungsgeschwindigkeit auf *`Baudrate`*. *`Baudrate`* kann üblicherweise 50, 75, 110, 134.5, 150, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, oder 115200 Bit pro Sekunde betragen.

`ms#stty-Modi`
: Setzt die Eigenschaften für das Gerät, nachdem es geöffnet wurde. Die verfügbaren Eigenschaften sind in stty(1) aufgeführt.

Wenn LPD das mit `lp` angegebene Gerät öffnet, setzt es die mit `ms#` angegebenen Eigenschaften. Von besonderem Interesse sind hier die Modi `parenb`, `parodd`, `cs5`, `cs6`, `cs7`, `cs8`, `cstopb`, `crtscts` und `ixon`, die in der stty(1)-Handbuchseite erläutert werden.

Für den über die sechste serielle Schnittstelle angeschlossenen Laserdrucker beträgt die Geschwindigkeit 38.400 Baud, und es werden diese Kommunikationseinstellungen verwendet: keine Parität (`-parenb`), 8-Bit-Zeichen (`cs8`), keine Modemsteuerung (`clocal`) und Hardware-Flusssteuerung (`crtscts`):

```
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :\
        :lp=/dev/ttyd5 :ms#-parenb cs8 clocal crtscts:
```

10.3.1.5.6. Den Textfilter installieren

Ein *Textfilter*, auch als *Eingangsfilter* bezeichnet, ist ein Programm, das von LPD aufgerufen wird, wenn ein Druckauftrag verarbeitet wird. Dabei wird die Standardeingabe des Programms mit der zu druckenden Datei verbunden, und die Standardausgabe mit dem im `lp`-Attribut angegebenen Gerät. Das Programm sollte nun die Datei einlesen, alle Übersetzungen durchführen, die für den Drucker notwendig sind, und das Ergebnis über die Standardausgabe an den Drucker senden. Textfilter werden im Abschnitt Filter genauer erläutert.

Um einen einfachen Test durchzuführen, reicht ein kleines Filterprogramm, dass schlicht `/bin/cat` aufruft, um die Daten unverändert und den Drucker zu schicken. FreeBSD verfügt über das Programm `lpf`, das Unterstreichung und Fettdruck für solche Drucker ermöglicht, die ansonsten dazu nicht in der Lage wären. Darüberhinaus gibt es viele andere Filter, die Sie einsetzen können. `lpf` wird im Abschnitt lpf: ein Textfilter ausführlich beschrieben.

Legen Sie zunächst das folgende Shell-Skript als `/usr/local/libexec/if-simple` mit Ihrem bevorzugten Texteditor an:

```
#!/bin/sh
#
# if-simple - Simple text input filter for lpd
# Installed in /usr/local/libexec/if-simple
#
# Simply copies stdin to stdout.  Ignores all filter arguments.

/bin/cat && exit 0
exit 2
```

Machen Sie die Datei ausführbar:

```
# chmod 555 /usr/local/libexec/if-simple
```

Konfigurieren Sie nun den Textfilter für Ihren Drucker in `/etc/printcap`, indem Sie das `if`-Attribut hinzufügen. Hier die Konfiguration unserer beiden Beispieldrucker:

```
#
#  /etc/printcap for host rose - added text filter
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :\ :lp=/dev/lpt0 :\
        :if=/usr/local/libexec/if-simple  :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :\
        :lp=/dev/ttyu5 :ms#-parenb cs8 clocal crtscts:\
        :if=/usr/local/libexec/if-simple  :
```

Anmerkung

Das Shell-Skript `if-simple` steht im Verzeichnis `/usr/share/examples/printing`.

10.3.1.5.7. LPD aktivieren

lpd(8) wird von `/etc/rc` gestartet, wenn die rc.conf(5)-Variable `lpd_enable` auf `YES` gesetzt ist. Fügen Sie dazu diese Zeile in `/etc/rc.conf` hinzu:

```
lpd_enable="YES"
```

Starten Sie Ihren Computer neu, oder starten Sie lpd(8) von Hand:

```
# lpd
```

10.3.1.5.8. Die Konfiguration testen

Damit ist die einfache Konfiguration abgeschlossen. Noch muss die Konfiguration aber getestet werden und etwaige Probleme müssen behoben werden. Um die Konfiguration zu testen, sollten Sie einen Probeausdruck mithilfe des Programms lpr(1) produzieren. lpr(1) übergibt Druckaufträge an das LPD-System.

Sie können lpr(1) mit lptest(1) kombinieren, um Testdaten zu drucken. lptest(1) wurde im Abschnitt Kommunikation mit den Drucker prüfen vorgestellt.

So testen Sie die einfache LPD-Konfiguration:

```
# lptest 20 5 | lpr -PDruckername
```

Ersetzen Sie *Druckername* durch den Namen des Druckers, den Sie testen möchten. Wenn Sie den Standard-Drucker testen möchten, rufen Sie lpr(1) ohne die Option `-P` auf. Wenn Sie einen Drucker testen möchten, der nur PostScript® versteht, müssen Sie ein PostScript®-Testprogramm an lpr(1) übergeben. Ein Testprogramm, das Sie in einer Datei gespeichert haben, können Sie mit dem Befehl `lpr Dateiname` an das LPD-System übergeben.

Bei einem PostScript®-Drucker hängt das Ergebnis naturgemäß vom Testprogramm ab. Wenn Sie lptest(1) verwenden, sollte das Ergebnis ungefähr so aussehen:

```
!"#$%&'()*+,-./01234
"#$%&'()*+,-./012345
#$%&'()*+,-./0123456
$%&'()*+,-./01234567
%&'()*+,-./012345678
```

Um sicherzustellen, dass alles richtig funktioniert, sollten Sie jetzt ein größeres PostScript®-Programm senden. Mit lptest(1) können Sie größere Datenmengen z.B. mit dem Befehl `lptest 80 60` erzeugen: lptest(1) produziert 60 Zeilen mit je 80 Zeichen.

Wenn Sie nicht erfolgreich drucken können, finden Sie im Abschnitt Fehlersuche und Problembehebung weitere Informationen.

10.4. Erweiterte Drucker-Konfiguration

Übersetzt von Johann Kois.

Dieser Abschnitt beschreibt den Einsatz von Filtern für das Drucken speziell formatierter Seiten oder von Deckblättern, das Drucken über ein Netzwerk sowie die Beschränkung und Verrechnung der Druckernutzung.

10.4.1. Filter

Obwohl LPD Netzwerkprotokolle, Warteschlangen, Zugriffskontrollen und andere für das Drucken wichtige Aspekte prinzipiell unterstützt, passiert ein Großteil der *wirklichen* Arbeit in den sogenannten *Filtern*. Dabei handelt es sich um Programme, die direkt mit einem Drucker kommunizieren und deren Gerätespezifika und spezielle Anforderungen erfüllen. Im einfachsten Fall installiert man nur einen reinen Textfilter, der mit beinahe allen Druckern funktionieren sollte. (Lesen Sie dazu auch den Abschnitt Den Text-Filter installieren.)

Um die erweiterten Fähigkeiten von Druckern auch einsetzen zu können, sollten Sie verstehen, wie Filter arbeiten, da diese für die Bereitstellung dieser Funktionen zuständig sind. Die schlechte Nachricht ist, dass *Sie* diese Filter bereitstellen müssen. Die gute Nachricht ist allerdings, dass diese in der Regel bereits vorhanden sind. Ist dies nicht der Fall, können Sie einen Filter meist relativ einfach selbst erstellen.

Der Filter `/usr/libexec/lpr/lpf` wird bereits mit FreeBSD geliefert. Er kümmert sich um die korrekte Behandlung von gelöschten Zeichen (das sogenannte *Backspacing*), um im Text enthaltene Tabulatoren, sowie um die Verrechnung von Druckaufträgen. Das ist aber auch alles, was dieser Filter kann. Zusätzliche Filter und für die Funktion von Filtern nötige Komponenten finden sich aber in der FreeBSD Ports-Sammlung.

Dieser Abschnitt behandelt folgende Themen:

- Der Abschnitt Die Funktionsweise von Filtern versucht, einen Überblick über die Rolle von Filtern innerhalb des Druckprozesses zu geben. Sie sollten diesen Abschnitt lesen, damit Sie verstehen, was „unter der Haube“ passiert, wenn LPD einen Filter verwendet. Dieses Wissen wird Ihnen dabei helfen, Probleme, die bei Installation von Filtern für verschiedene Drucker entstehen können, vorauszusehen und zu beheben.
- LPD geht davon aus, dass jeder Drucker in der Lage ist, normalen Text zu drucken. Für PostScript®- (oder andere sprachbasierte) Drucker stellt dies allerdings ein Problem dar, da diese nicht in der Lage sind, normalen Text

direkt zu drucken. Der Abschnitt Normalen Text auf PostScript®-Druckern drucken beschreibt, wie Sie dieses Problem lösen können. Besitzen Sie einen PostScript®-Drucker, sollten Sie diesen Abschnitt lesen.

- PostScript® ist ein populäres Ausgabeformat, das von vielen Programmen unterstützt wird. Es ist sogar möglich, PostScript®-Code direkt zu schreiben. Leider sind PostScript®-Drucker in der Regel relativ teuer. Der Abschnitt PostScript® auf Nicht-PostScript®-Druckern emulieren beschreibt, wie Sie einen Textfilter anpassen müssen, um PostScript®-Daten auf einem *nicht-PostScript®-fähigen Drucker* auszugeben. Haben Sie keinen PostScript®-Drucker, sollten Sie insbesondere diesen Abschnitt lesen.

- Der Abschnitt Konvertierungsfilter beschreibt eine Möglichkeit zur automatischen Konvertierung verschiedener Dateiformate in ein von Ihrem Drucker unterstütztes Format. Nachdem Sie diesen Abschnitt gelesen haben, werden Sie in der Lage sein, Ihren Drucker so zu konfigurieren, dass Sie durch die Eingabe von `lpr -t` troff-Daten, von `lpr -d` TeX-DVI-Daten, oder von `lpr -v` Rasterbilddaten drucken können. Daher sollten Sie diesen Abschnitt auf jeden Fall lesen.

- Im Abschnitt Ausgabefilter wird eine nur selten genutzte Eigenschaft von LPD, die sogenannten Ausgabefilter, beschrieben. Wenn Sie keine Deckblätter drucken müssen, können Sie diesen Abschnitt überspringen.

- Der Abschnitt lpf: Ein Textfilter beschreibt `lpf`, einen kompletten, wenn auch einfachen Textfilter für Zeilendrucker (oder auch Laserdrucker, die sich analog verhalten), der bereits mit FreeBSD geliefert wird. Wenn Sie nur am Ausdruck von reinem Text interessiert sind, oder wenn Ihr Drucker nur „Schrott“ produziert, wenn er auf Backspace-Zeichen trifft, sollten Sie sich `lpf` näher ansehen.

Anmerkung

Eine Kopie der verschiedenen Skripte finden Sie im Verzeichnis `/usr/share/examples/printing`.

10.4.1.1. Die Funktionsweise von Filtern

Bei einem Filter handelt es sich um ein ausführbares Programm, das von LPD gestartet wird, um den geräteabhängigen Teil der Kommunikation mit einem Drucker zu übernehmen.

Wenn LPD eine Datei über einen Druckauftrag drucken will, startet es ein Filterprogramm. Danach setzt es die Standardeingabe des Filters auf die zu druckende Datei, die Standardausgabe auf den Drucker und die Standardfehlerausgabe auf `/dev/console` (Voreinstellung) oder auf die über die Option `lf` in `/etc/printcap` festgelegte Datei.

Welcher Filter von LPD mit welchen Argumenten geladen wird, wird in der Datei `/etc/printcap` oder durch die Argumente, die der Anwender lpr(1) auf der Kommandozeile übergibt, festgelegt. Gibt der Anwender beispielsweise `lpr -t` ein, startet LPD über die `tf`-Fähigkeit den troff-Filter für den gewünschten Drucker. Wollen Sie hingegen normalen Text drucken, wird der `if`-Filter gestartet. (Für Ausnahmen von diesem Vorgehen lesen Sie bitte den Abschnitt Ausgabefilter.)

Es gibt drei Arten von Filtern, die Sie in `/etc/printcap` angeben können:

- *Textfilter* (die in der LPD-Dokumentation als *input filter* bezeichnet werden) sind für den Druck von normalem Text zuständig. Es handelt sich dabei um eine Art Standardfilter, da LPD von jedem Drucker erwartet, dass er normalen Text drucken kann. Aufgabe des Textfilters ist es, sicherzustellen, dass gelöschte Zeichen (*Backspaces*), Tabulatoren und andere Sonderzeichen Ihren Drucker nicht verwirren. Falls Sie für die Nutzung eines Druckers bezahlen müssen, kann der Textfilter über die Anzahl der gedruckten Zeilen auch die Anzahl der von Ihnen gedruckten Seiten ermitteln. Der Textfilter wird mit folgenden Argumenten gestartet:

  `filter-name` [-c] -w *`width`* -l *`length`* -i *`indent`* -n *`login`* -h *`host acct-file`*

  Die einzelnen Argumente haben folgende Bedeutung:

`-c`
: Notwendig, wenn `lpr -l` verwendet wird.

`width`
: Der Wert der in `/etc/printcap` festgelegten Option `pw` (*page width*). In der Voreinstellung ist dieser Wert auf 132 gesetzt.

`length`
: Der Wert der `pl`-Fähigkeit (*page length*), Voreinstellung 66.

`indent`
: Der durch `lpr -i` festgelegte Einzug, Voreinstellung 0.

`login`
: Der Name des Benutzers, der die Datei druckt.

`host`
: Der Rechner, auf dem der Druckauftrag gestartet wurde.

`acct-file`
: Der Name der Verrechnungsdatei, in der die Ergebnisse der `af`-Fähigkeit gespeichert werden.

- Ein *Konvertierungsfilter* konvertiert verschiedene Dateiformate in ein Format, das Ihr Drucker auf Papier ausgeben kann. So kann etwa der ditroff-Schriftsatz nicht direkt gedruckt werden, daher müssen Sie einen Konvertierungsfilter installieren, um diese Daten in ein Format zu bringen, das Ihr Drucker verarbeiten und drucken kann. Der Abschnitt Konvertierungsfilter enthält ausführliche Informationen zu diesen Filtern. Konvertierungsfilter können auch zur Verrechnung verwendet werden. Sie werden mit folgenden Argumenten gestartet:

  `filter-name` -x *`pixel-width`* -y *`pixel-height`* -n *`login`* -h *`host acct-file`*

  `pixel-width` ist der Wert der `px`-Fähigkeit (Voreinstellung 0), während *`pixel-height`* dem Wert der `py`-Fähigkeit (Voreinstellung ebenfalls 0) entspricht.

- *Ausgabefilter* werden nur verwendet, wenn keine Textfilter vorhanden sind oder wenn Deckblätter benötigt werden. Der Abschnitt Ausgabefilter enthält weitere Informationen. Ausgabefilter unterstützen nur zwei Argumente:

  `filter-name` -w *`width`* -l *`length`*

  Beide Argumente entsprechen den Optionen `-w` und `-l` der Textfilter.

Alle Filter sollten mit folgenden Rückgabewerten (Exitcodes) *beendet* werden:

exit 0
: Der Filter hat die Datei erfolgreich gedruckt.

exit 1
: Der Filter war nicht in der Lage, die Datei zu drucken und meldet diesen Exitcode an LPD, um die Datei erneut zu drucken. LPD startet daraufhin den Filter erneut.

exit 2
: Der Filter war nicht in der Lage, die Datei zu drucken. Bei diesem Exitcode soll LPD aber nicht versuchen, die Datei erneut zu drucken, sondern den Druckauftrag verwerfen.

`/usr/libexec/lpr/lpf` , der mit FreeBSD gelieferte Textfilter, nutzt die Argumente *page width* und *page length*, um festzulegen, wann ein Seitenumbruch (*form feed*) gesendet werden soll sowie zur Verrechnung von Druckaufträgen. Dazu werden der Benutzername, der für den Druckauftrag verwendete Rechner sowie die Verrechnungsdatei ausgewertet, um die entsprechenden Einträge zu erstellen.

Wenn Sie auf der Suche nach Filtern sind, achten Sie darauf, dass diese LPD-kompatibel sind. Dazu müssen diese die oben beschriebenen Argumente unterstützen. Wenn Sie planen, Ihre Filter selbst zu erstellen, müssen diese ebenfalls die gleichen Argumente und Exitcodes unterstützen.

10.4.1.2. Normalen Text auf PostScript®-Druckern drucken

Sie sind der alleinige Benutzer Ihres Computers und Ihres PostScript®-Druckers und Sie sind sich sicher, dass Sie niemals normalen Text an Ihren Drucker senden werden? Außerdem werden Sie niemals ein Programm verwenden, um normalen Text auszudrucken? Nur wenn dies alles zutrifft, können Sie diesen Abschnitt überspringen.

Wollen Sie allerdings sowohl PostScript® als auch normalen Text drucken, müssen Sie Ihren Drucker zuvor entsprechend konfigurieren. Dazu muss Ihr Textfilter in der Lage sein, zu unterscheiden, ob es sich bei einem ankommenden Druckauftrag um normalen Text oder um PostScript®-Daten handelt. Jeder PostScript®-Druckauftrag muss mit den Zeichen `%!` beginnen (sehen Sie in Ihrem Druckerhandbuch nach, ob Ihr Drucker weitere Sprachen unterstützt). Sind dies die beiden ersten Zeichen eines Druckauftrages, so handelt es sich um PostScript®-Daten, die direkt gedruckt werden können. Fehlen diese Zeichen allerdings, muss der Textfilter den Inhalt der Datei nach PostScript® konvertieren, bevor die Datei gedruckt werden kann.

Wie funktioniert diese Unterscheidung?

Haben Sie einen seriellen Drucker, können Sie `lprps` installieren. `lprps` ist ein PostScript®-Druckerfilter, der eine Zweiwegekommunikation mit einem Drucker ermöglicht. Er aktualisiert die Druckerstatusdatei mit Protokollinformationen des Druckers. Dadurch sind Anwender und Administratoren in der Lage, den genauen Zustand des Druckers zu prüfen (durch Meldungen wie toner low oder paper jam). Wichtiger ist allerdings, dass `lprps psif` enthält, ein Programm, das feststellen kann, ob ein ankommender Druckauftrag normalen Text enthält. Ist dies der Fall, wird `textps` (das ebenfalls mit `lprps` geliefert wird) aufgerufen und die Datei nach PostScript® konvertiert. Danach kann `lprps` die Datei an den Drucker senden.

`lprps` ist in der FreeBSD Ports-Sammlung enthalten. Je nach der von Ihnen verwendeten Papiergröße installieren Sie dazu den Port print/lprps-a4 oder print/lprps-letter. Nach der Installation müssen Sie nur noch den Pfad zum Programm `psif` angeben, das als Teil von `lprps` installiert wird. Haben Sie `lprps` über die Ports-Sammlung installiert, fügen Sie folgende Zeile in den Eintrag Ihres PostScript®-Druckers in `/etc/printcap` ein:

```
:if=/usr/local/libexec/psif :
```

Zusätzlich sollten Sie die `rw`-Fähigkeit aktivieren, um LPD im Schreib- und Lesemodus zu öffnen.

Haben Sie hingegen einen parallelen PostScript®-Drucker, was eine Zweiwegekommunikation mit Ihrem Drucker (auf die `lprps` angewiesen ist) unmöglich macht, können Sie das folgende Shell-Skript verwenden:

```
#!/bin/sh
#
#  psif - Drucke PostScript oder normalen Text auf einem PostScript-Drucker
#  Script-Version; das ist NICHT die mit lprps gelieferte Version!
#  Installiert unter:  /usr/local/libexec/psif
#

IFS="" read -r first_line
first_two_chars=`expr "$first_line" : '\(..\)'`

if [ "$first_two_chars" = "%!" -]; then
    #
    #  PostScript - einfach drucken.
    #
    echo "$first_line" && cat && printf "\004" && exit 0
    exit 2
else
    #
    #  Normaler Text - zuerst konvertieren, dann drucken.
    #
    ( echo "$first_line"; cat ) | /usr/local/bin/textps && printf "\004" && exit 0
```

```
    exit 2
fi
```

Für dieses Skript wurde `textps` als seperates Programm installiert, um normalen Text nach PostScript® zu konvertieren. Sie können aber auch jeden anderen Text-nach-PostScript®-Konverter verwenden. Die FreeBSD Ports-Sammlung enthält mit `a2ps` ein umfangreiches Programm zur Konvertierung von normalem Text nach PostScript®.

10.4.1.3. PostScript® auf Nicht-PostScript®-Druckern emulieren

Bei PostScript® handelt es sich um den *de facto*-Standard für hochwertigen Schriftsatz und Druck. Leider ist PostScript® aber auch ein *teurer* Standard. Glücklicherweise hat Aladdin Enterprises daher eine freie PostScript®-ähnliche Implementierung namens Ghostscript entwickelt, die auch unter FreeBSD lauffähig ist. Ghostscript kann fast jede PostScript®-Datei lesen und auf den verschiedensten Geräten ausgeben, darunter auch auf vielen Nicht-PostScript®-Druckern. Durch die Installation von Ghostscript und die Nutzung eines speziellen Textfilters erreichen Sie, dass sich Ihr Nicht-PostScript®-Drucker wie ein echter PostScript®-Drucker verhält.

Ghostscript ist in verschiedenen Versionen in der FreeBSD Ports-Sammlung enhalten, die am häfigsten verwendete Version ist print/ghostscript-gpl.

Um PostScript® zu emulieren, muss der Textfilter erkennen, ob er eine PostScript®-Datei drucken soll. Ist dies nicht der Fall, wird die Datei direkt an den Drucker geschickt. Anderenfalls wird die Datei an Ghostscript übergeben, das die Datei in ein Format konvertiert, das Ihr Drucker versteht.

Dazu ein Beispiel. Das folgende Skript ist ein Textfilter für den Drucker DeskJet 500 von Hewlett Packard. Nutzen Sie einen anderen Drucker, müssen Sie die Option `-sDEVICE` beim Aufruf von `gs` (Ghostscript) entsprechend anpassen. Eine Liste der von Ghostscript unterstützten Geräte erhalten Sie durch die Eingabe von `gs -h` auf der Kommandozeile.

```
#!/bin/sh
#
#  ifhp - Ghostscript-emuliertes PostScript auf einem HP DeskJet 500 drucken
#  Installiert unter:  /usr/local/libexec/ifhp

#
#  LF als CR+LF behandeln (um einen "Treppeneffekt" auf HP/PCL-Drucker
#  zu vermeiden)
#
printf "\033&k2G" || exit 2

#
#  Lies die ersten zwei Zeichen der Datei
#
IFS="" read -r first_line
first_two_chars=`expr "$first_line" : '\(..\)'`

if [ "$first_two_chars" = "%!" -]; then
    #
    #  Oh.  Es ist PostScript; mit Ghostscript konvertieren, danach drucken.
    #
    /usr/local/bin/gs -dSAFER -dNOPAUSE -q -sDEVICE=djet500 \
      -sOutputFile=- - && exit 0
else
    #
    #  Normaler Text oder HP/PCL, einfach direkt drucken.  Ans Ende setzen wir
    #  einen Seitenumbruch (also ein Form Feed), damit auch die letzte Seite
    #  ausgeworfen wird.
    #
    echo "$first_line" && cat && printf "\033&l0H" &&
exit 0
fi

exit 2
```

Zuletzt müssen Sie LPD noch durch die `if`-Fähigkeit über den neuen Filter informieren:

```
:if=/usr/local/libexec/ifhp :
```

Das ist alles. Ab sofort sollte sowohl ein `lpr normaler.text` als auch ein `lpr wasauchimmer.ps` funktionieren und beide Dateien sollten problemlos gedruckt werden.

10.4.1.4. Konvertierungsfilter

Nachdem Sie Ihren Drucker wie unter Einfache Drucker-Konfiguration eingerichtet haben, wollen Sie wahrscheinlich einige Konvertierungsfilter installieren, damit Sie (abgesehen von ASCII-Text) auch Ihre Lieblings-Dateiformate drucken können.

10.4.1.4.1. Warum sollte ich einen Konvertierungsfilter installieren?

Konvertierungsfilter erleichtern das Drucken von verschiedenen Dateiformaten. Nehmen wir beispielsweise an, dass Sie sehr viel mit dem TeX-Satzsystem arbeiten und über einen PostScript®-Drucker verfügen. Eine vom TeX-System erzeugte DVI-Datei kann erst dann gedruckt werden, nachdem diese nach PostScript® konvertiert wurde. Dazu geben Sie Folgendes ein:

```
% dvips seaweed-analysis.dvi
% lpr seaweed-analysis.ps
```

Haben Sie einen Konvertierungsfilter für DVI-Dateien installiert, können Sie die manuelle Konvertierung überspringen, da dies nun LPD für Sie erledigt. Wollen Sie eine DVI-Datei drucken, geben Sie nur noch den folgenden Befehl ein:

```
% lpr -d seaweed-analysis.dvi
```

Durch die Verwendung der Option `-d` wurde LPD angewiesen, unsere DVI-Datei vor dem Druck zu konvertieren. Der Abschnitt Formatierungs- und Konvertierungsoptionen beschreibt die dabei möglichen Optionen.

Für jede Konvertierungsoption, die Ihr Drucker unterstützen soll, müssen Sie einen eigenen *Konvertierungsfilter* installieren und dessen Pfad in der Datei `/etc/printcap` angeben. Ein Konvertierungsfilter verhält sich im Prinzip wie ein Textfilter bei einer einfachen Druckerkonfiguration (lesen Sie dazu auch den Abschnitt Den Textfilter installieren), allerdings konvertiert er die Datei in ein Format, das Ihr Drucker versteht, anstatt normalen Text zu drucken.

10.4.1.4.2. Welche Konvertierungsfilter sollte ich installieren?

Sie sollten nur Filter installieren, die Sie auch benötigen. Wenn Sie sehr viele DVI-Dateien drucken, sollten Sie auch einen DVI-Konvertierungsfilter installieren. Müssen Sie viele troff-Daten drucken, ist ein troff-Filter hilfreich.

Die folgende Tabelle listet die von LPD unterstützten Filter sowie die Einträge in `/etc/printcap` auf, mit denen Sie diese Fähigkeiten aktivieren. Zusätzlich wird angegeben, wie Sie `lpr` jeweils aufrufen müssen:

Dateityp	**/etc/printcap**-Fähigkeit	**lpr**-Option
cifplot	`cf`	`-c`
DVI	`df`	`-d`
plot	`gf`	`-g`
ditroff	`nf`	`-n`
FORTRAN-Text	`rf`	`-f`
troff	`tf`	`-f`
Rasterdaten	`vf`	`-v`
Normaler Text	`if`	keine, `-p`, or `-l`

Wollen Sie also `lpr -d` verwenden, muss die `df`-Fähigkeit in `/etc/printcap` aktiviert sein.

Obwohl manche Leute etwas anderes behaupten, sind Formate wie FORTRAN-Text und -Plot inzwischen nahezu obsolet. Wenn Sie diese Formate dennoch benötigen, installieren Sie einfach einen angepassten Filter. Wollen Sie beispielsweise zwar Printerleaf-Dateien (also Dateien des Desktop Publishing-Programms von Interleaf), aber keine Plotdateien drucken, so können Sie einen Printerleaf-Konvertierungsfilter installieren, der es durch die Aktivierung der `gf`-Fähigkeit erlaubt, diese Dateien direkt zu drucken. Nun müssen Sie Ihren Mitarbeitern nur noch mitteilen, dass `lpr -g` nun für „drucke Printerleaf-Dateien“ steht.

10.4.1.4.3. Konvertierungsfilter installieren

Da Konvertierungsfilter nicht zum Basissystem von FreeBSD gehören, sollten diese unter `/usr/local` installiert werden. Häufig wird das Verzeichnis `/usr/local/libexec` verwendet, da es sich bei Konvertierungsfiltern um spezielle Programme handelt, die nur von LPD, aber nicht von einem normalen Benutzer gestartet werden.

Um einen Konvertierungsfilter zu aktivieren, müssen Sie dessen Pfad zusätzlich zur benötigten Fähigkeit in der Datei `/etc/printcap` eintragen.

In unserem Beispiel wollen wir einen DVI-Konvertierungsfilter für den Drucker `bamboo` installieren. Unsere bereits bekannte `/etc/printcap` wurde allerdings um die `df`-Fähigkeit für den Drucker `bamboo` erweitert:

```
#
#  /etc/printcap des Rechners rose - neuer df-Filter für bamboo
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :\
        :lp=/dev/lpt0 :\
        :if=/usr/local/libexec/if-simple  :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :\
        :lp=/dev/ttyu5 :ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif :\
        :df=/usr/local/libexec/psdf :
```

Beim DVI-Filter handelt es sich um ein Shell-Skript namens `/usr/local/libexec/psdf :`

```
#!/bin/sh
#
#  psdf - DVI-nach-PostScript Druckerfilter
#  Installiert unter:  /usr/local/libexec/psdf
#
#  Wird von lpd aktiviert, wenn der Nutzer lpr -d eingibt.
#
exec /usr/local/bin/dvips -f | /usr/local/libexec/lprps "$@"
```

Dieses Skript startet `dvips` im Filtermodus (durch das Argument `-f` wird der Druckauftrag über die Standardeingabe entgegengenommen). Danach wird der PostScript®-Druckerfilter `lprps` (lesen Sie dazu auch den Abschnitt Drucken von reinen Textdateien auf einem PostScript®-Drucker) mit den von LPD übergebenen Argumenten gestartet. Das `lprps`-Werkzeug wiederum nutzt diese Argumente, um die gedruckten Seiten zu verrechnen.

10.4.1.4.4. Beispiele für Konvertierungsfilter

Da es keine verbindliche Prozedur zur Installation eines Druckerfilters gibt, folgen nun weitere Beispiele in diesem Abschnitt. Verwenden Sie diese, um Ihre eigenen Filter zu erstellen. Falls ein Filter Ihren Anforderungen bereits entspricht, können Sie ihn auch direkt verwenden.

Das erste Beispiel beschreibt einen Konvertierungsfilter für GIF-Dateien für den Drucker LaserJet III-Si von Hewlett Packard:

```
#!/bin/sh
#
#  hpvf - Konvertiert GIF-Dateien nach HP/PCL, danach wird gedruckt.
#  Installiert unter:  /usr/local/libexec/hpvf
```

```
PATH=/usr/X11R6/bin:$PATH; export PATH
giftopnm | ppmtopgm | pgmtopbm | pbmtolj -resolution 300 \
    && exit 0 \
    || exit 2
```

Dieser Filter konvertiert eine GIF-Datei in eine portable Anymap, diese in ein portables Graustufenbild, dieses wiederum in eine portable Bitmap, die schließlich in ein LaserJet/PCL-kompatibles Format umgewandelt wird.

`/etc/printcap` muss für einen Drucker, der diesen Filter nutzen will, folgenden Eintrag enthalten:

```
#
#  /etc/printcap des Rechners orchid
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0 :sh:sd=/var/spool/lpd/teak :mx#0:\
        :if=/usr/local/libexec/hpif :\
        :vf=/usr/local/libexec/hpvf :
```

Das folgende Skript ist ein Konvertierungsfilter, der das Drucken von troff-Daten des groff-Textsatzsystems auf dem PostScript®-Drucker `bamboo` ermöglicht:

```
#!/bin/sh
#
#  pstf - Konvertiert groff's troff-Daten nach PS, dann wird gedruckt.
#  Installiert unter:  /usr/local/libexec/pstf
#
exec grops | /usr/local/libexec/lprps "$@"
```

Dieses Skript nutzt wiederum `lprps`, um mit dem Drucker zu kommunizieren. Wäre der Drucker an einem parallelen Port angeschlossen, würde das Skript so aussehen:

```
#!/bin/sh
#
#  pstf - Konvertiert groff's troff-Daten nach PS, danach wird gedruckt.
#  Installiert unter:  /usr/local/libexec/pstf
#
exec grops
```

Das ist alles. Um den Filter verwenden zu können, müssen Sie ihn allerdings noch in `/etc/printcap` aktivieren:

```
:tf=/usr/local/libexec/pstf :
```

Das nächste Skript ist ein FORTRAN-Textfilter für jeden Drucker, der normalen Text direkt drucken kann und der hier für den Drucker `teak` installiert wird:

```
#!/bin/sh
#
# hprf - FORTRAN-Textfilter für den Drucker LaserJet 3si:
# Installiert unter:  /usr/local/libexec/hprf
#

printf "\033&k2G" && fpr && printf "\033&l0H" &&
 exit 0
exit 2
```

Zusätzlich benötigen wir wiederum einen Eintrag in `/etc/printcap`, um diesen Filter für den Drucker `teak` zu aktivieren:

```
:rf=/usr/local/libexec/hprf :
```

Das letzte Beispiel ist etwas komplexer. Es soll ein DVI-Filter für den bereits erwähnten LaserJet-Drucker `teak` installiert werden. Der erste Teil ist einfach: Sie müssen den Pfad des DVI-Filters in `/etc/printcap` eintragen:

```
:df=/usr/local/libexec/hpdf :
```

Nun kommt der schwierige Teil: Sie müssen den Filter funktionsfähig machen. Dazu benötigen Sie einen DVI-nach-LaserJet/PCL-Konverter. Glücklicherweise enthält die FreeBSD Ports-Sammlung mit print/dvi2xx ein solches Programm. Nach der Installation des Pakets verfügen wir über das Programm `dvilj2p`, das zur Konvertierung von DVI-Daten in zu den Druckern LaserJet IIp, LaserJet III, sowie LaserJet 2000 kompatible Codes benötigt wird.

Durch den Einsatz von `dvilj2p` wird der Filter `hpdf` relativ komplex, da `dvilj2p` nicht von der Standardeingabe lesen kann, sondern als Eingabe einen Dateinamen erwartet. Zusätzlich muss der Dateiname auf `.dvi` enden, daher ist die Verwendung von `/dev/fd/0` als Standardeingabe problematisch. Wir können diese Problem aber umgehen, indem wir einen temporären Dateinamen symbolisch nach `/dev/fd/0` linken. Dadurch wird `dvilj2p` gezwungen, dennoch von der Standardeingabe zu lesen.

Das letzte Problem, das wir noch lösen müssen, ist, dass wir `/tmp` nicht als temporären Link verwenden können. Symbolische Links gehören dem User sowie der Gruppe `bin`. Der Filter läuft aber als User `daemon`. Außerdem ist `/tmp` durch ein Sticky-Bit gesichert. Daher kann der Filter den Link zwar erzeugen, ein Aufräumen ist aber nicht mehr möglich, weil sich die Eigentümer des Filters und des temporären Verzeichnisses unterscheiden.

Daher legt der Filter den symbolischen Link im Arbeitsverzeichnis an, das gleichzeitig als Spooling-Verzeichnis dient (festgelegt durch die Aktivierung der `sd`-Fähigkeit in `/etc/printcap`). Das Arbeitsverzeichnis ist ein idealer Ort für den Filter, insbesondere da dieses (manchmal) sogar über mehr freien Speicherplatz als `/tmp` verfügt.

Mit diesen Informationen sind wir nun in der Lage, den Filter zu entwickeln:

```
#!/bin/sh
#
#  hpdf - DVI-Daten auf einen HP/PCL-Drucker drucken
#  Installiert unter:  /usr/local/libexec/hpdf

PATH=/usr/local/bin:$PATH; export PATH

#
#  Eine Funktion zum Aufräumen unserer temporären Dateien.
#  Diese finden sich im Arbeitsverzeichnis, das wir auch als
#  Spooling-Verzeichnis für unseren Drucker verwenden werden.
#
cleanup() {
   rm -f hpdf$$.dvi
}

#
#  Eine Funktion, um fatale Fehler zu behandeln.  Dazu die Meldung
#  ausgeben, danach ein exit 2.  Dadurch weiß LPD, dass es
#  den Auftrag nicht noch einmal drucken soll.
#
fatal() {
    echo "$@" 1>&2
    cleanup
    exit 2
}

#
#  Wenn ein Anwender den Auftrag entfernt, sendet LPD ein SIGINT, daher
#  wollen wir SIGINT und einige andere Signale abfangen (trappen), um
#  nach der Konvertierung aufräumen zu können.
#
trap cleanup 1 2 15

#
#  Bevor wir anfangen, räumen wir noch auf.  Sicher ist sicher.
#
cleanup

#
#  Die DVI-Eingabedatei auf die Standardeingabe linken (die zu druckende
#  Datei).
```

```
#
ln -s /dev/fd/0 hpdf$$.dvi || fatal "Konnte Symlink nicht anlegen!"

#
#  Umwandeln: LF = CR+LF
#
printf "\033&k2G" || fatal "Konnte Drucker nicht initialisieren!"

#
#  Konvertieren und drucken.  Da der Rückgabewert von dvilj2p
#  unzuverlässig ist, ignorieren wir ihn einfach.
#
dvilj2p -M1 -q -e- dfhp$$.dvi

#
#  Aufräumen und beenden.
#
cleanup
exit 0
```

10.4.1.4.5. Automatische Konvertierung: Eine Alternative zu Konvertierungsfiltern

Alle in diesem Abschnitt besprochenen Konvertierungsfilter sind zwar sehr hilfreich, allerdings müssen Sie nach wie vor bei jedem Aufruf von lpr(1) angeben, welchen Filter sie verwenden wollen, was mit der Zeit sicher nervend wird. Schlimmer ist allerdings, dass die Auswahl eines unpassenden Filters dazu führen kann, dass Sie Hunderte Seiten Papier ausdrucken.

Statt also Konvertierungsfilter zu installieren, könnten Sie den Textfilter (der ohnehin der Standardfilter ist) verwenden, um den zu druckenden Dateityp zu erkennen und anschließend den korrekten Konvertierungsfilter auszuwählen. Um den Dateityp zu bestimmen, können Sie beispielsweise `file` verwenden. Leider ist es bei *einigen* Dateitypen problematisch, diese zu unterscheiden. Daher könnten Sie für diese Dateitypen dennoch einen Konvertierungsfilter installieren.

Die FreeBSD Ports-Sammlung enthält mit `apsfilter` (print/apsfilter) einen Textfilter, der diese automatische Konvertierung durchführen kann. Er ist in der Lage, normalen Text, PostScript®, DVI und beinahe jede Art von Datei zu erkennen, diese zu konvertieren und auf Ihren Drucker auszugeben.

10.4.1.5. Ausgabefilter

LPD unterstützt noch eine weitere Filterart, die sogenannten Ausgabefilter. Diese sind – analog zu einem Textfilter – für den Druck von normalem Text ausgelegt, allerdings verfügen sie im Vergleich zu diesen nur über sehr eingeschränkte Fähigkeiten. Wenn Sie einen Ausgabefilter (aber keinen Textfilter) verwenden, dann

- startet LPD nur einen Ausgabefilter für den kompletten Druckauftrag, statt für jede Datei des Auftrags einen eigenen Filter zu starten.
- kümmert sich LPD nicht darum, den Beginn oder das Ende einer Datei innerhalb des Druckauftrages zu finden.
- übergibt LPD weder den Benutzer- noch den Rechnernamen desjenigen, der den Druckauftrag erteilt hat, an den Ausgabefilter, was eine Verrechnung von Druckaufträgen unmöglich macht. Ausgabefilter unterstützen insgesamt nur zwei Argumente:

  `filter-name` -w *`width`* -l *`length`*

  `width` basiert auf der `pw`-Fähigkeit, *`length`* hingegen auf der `pl`-Fähigkeit des gewählten Druckers.

Lassen Sie sich von dieser angeblichen Einfachheit eines Ausgabefilters nicht täuschen. Ausgabefilter sind beispielsweise *nicht dazu in der Lage*, jede Datei eines Druckauftrages auf einer neuen Seite zu drucken. Dazu benötigen Sie einen Textfilter (die im Abschnitt Den Textfilter installieren beschrieben werden). Außerdem sind Ausgabefilter in Wirklichkeit *komplexer*, da sie den gesendeten Bytestrom nicht nur auf Sonderzeichen hin untersuchen müssen, sondern auch die Übertragung von Signalen für LPD übernehmen müssen.

Sie *benötigen* Ausgabefilter aber dann, wenn Sie Deckblätter drucken wollen, da dazu Escape-Sequenzen und Initialisierungsstrings erforderlich sind. (Es ist allerdings *nicht möglich*, den Druck dieser Deckblätter zu verrechnen, da LPD keine Benutzer- oder Rechnerinformationen an den Ausgabefilter übergibt.)

LPD kann für den gleichen Drucker sowohl Ausgabefilter als auch Textfilter verwenden. In solchen Fällen verwendet LPD den Ausgabefilter nur für den Druck von Deckblättern (die im Abschnitt Deckblätter näher beschrieben werden). Nach dem Druck des Deckblattes erwartet LPD, dass sich der Ausgabefilter *selbst beendet*. Dazu werden zwei Bytes an den Ausgabefilter gesendet: ASCII 031, gefolgt von ASCII 001. Wenn ein Ausgabefilter diese zwei Bytes (031, 001) empfängt, sendet er das Signal `SIGSTOP` an sich selbst. Nachdem LPD den Rest des Druckauftrages erledigt hat, wird der Ausgabefilter erneut gestartet, indem ein `SIGCONT` an den Ausgabefilter gesendet wird.

Haben Sie nur einen Ausgabefilter, aber *keinen* Textfilter installiert, dann verwendet LPD den Ausgabefilter auch für den Druck von normalem Text. Wie bereits erwähnt, werden dabei allerdings alle Dateien des Druckauftrags unmittelbar hintereinander gedruckt, Seitenumbrüche oder ein zusätzlicher Papiervorschub sind also nicht möglich. Da dieses Verhalten von Ihnen wahrscheinlich *nicht* gewünscht wird, werden Sie in fast allen Fällen einen zusätzlichen Textfilter benötigen.

Der weiter oben beschriebene Textfilter `lpf` kann auch als Ausgabefilter verwendet werden. Wenn Sie nur einen funktionierenden Ausgabefilter benötigen, aber nicht den dafür benötigten Code (zur Zeichenerkennung und zum Senden von Signalen) schreiben wollen, sollten Sie sich `lpf` näher ansehen. Sie können `lpf` auch in ein Shell-Skript einbinden, um von Ihrem Drucker benötigte Initialisierungscodes zu verarbeiten.

10.4.1.6. `lpf`: Ein Textfilter

Der Textfilter (Eingabefilter) `/usr/libexec/lpr/lpf` wird bereits mit FreeBSD geliefert. Er erlaubt das Einrücken der Ausgabe (über `lpr -i`), die Übergabe von Zeichen-Literalen (über `lpr -l`), das Anpassen der Druckposition bei gelöschten Zeichen (*Backspaces*) oder Tabulatoren, sowie die Verrechnung gedruckter Seiten. Zusätzlich kann dieser Textfilter auch als Ausgabefilter arbeiten.

`lpf` ist für viele verschiedene Druckumgebungen geeignet. Zwar ist dieser Textfilter nicht in der Lage, Initialisierungssequenzen an einen Drucker zu senden, dieses Problem kann allerdings durch das Schreiben und Ausführen eines Shell-Skripts (das diese Funktion übernimmt) und das anschließende Aufrufen von `lpf` gelöst werden.

Damit Sie `lpf` für die Verrechnung von Druckaufträgen einsetzen können, müssen Sie die korrekten Werte für die `pw`- und `pl`-Fähigkeiten in `/etc/printcap` eintragen. `lpf` verwendet diese Werte, um festzustellen, wieviel Text auf eine Seite passt und wieviele Seiten im Druckauftrag enthalten sind. Weitere Informationen zur Verrechnung der Druckernutzung enthält der Abschnitt Die Druckernutzung verrechnen.

10.4.2. Deckblätter

Wenn Sie *viele* Benutzer mit verschiedenen Druckern verwalten müssen, sollten Sie *Deckblätter* als notwendiges Übel akzeptieren.

Deckblätter (manchmal auch als *Bannerseiten* oder *burst pages* bezeichnet) geben an, wem die Ausgabe eines Druckauftrags gehört. Sie werden normalerweise in großen fetten Buchstaben gedruckt, manchmal sogar mit zusätzlicher Umrandung, damit man sie leichter von den tatsächlichen Seiten eines Druckauftrages unterscheiden kann. Der Nachteil von Deckblättern ist allerdings, dass es sich dabei um eine zusätzliche zu druckende Seite handelt, die in der Regel bereits nach wenigen Minuten wieder im Papierkorb landet. Da aber für jeden Druckauftrag nur ein einziges Deckblatt gedruckt wird, ist der Papierverbrauch in den meisten Fällen tolerierbar.

Das LPD-System kann Deckblätter automatisch erzeugen, *wenn* Ihr Drucker normalen Text direkt drucken kann. Haben Sie hingegen einen PostScript®-Drucker, benötigen Sie ein externes Programm, um die Deckblätter zu generieren (Lesen Sie dazu auch den Abschnitt Deckblätter auf PostScript®-Druckern erzeugen.).

10.4.2.1. Deckblätter aktivieren

Im Abschnitt Einfache Drucker-Konfiguration haben wir die Ausgabe von Deckblättern durch die die Angabe der Option `sh` (*suppress header*) in `/etc/printcap` deaktiviert. Um die Ausgabe von Deckblättern wieder zu aktivieren, müssen Sie daher die `sh`-Fähigkeit wieder entfernen.

Das klingt zu einfach? Wo ist der Haken?

Sie haben recht. Es ist *möglich*, dass Sie einen Ausgabefilter verwenden müssen, um die nötigen Initialisierungsstrings an den Drucker zu senden. Das folgende Beispiel beschreibt einen Ausgabefilter für PCL-kompatible Drucker von Hewlett Packard:

```
#!/bin/sh
#
#  hpof - Ausgabefilter für PCL-kompatible Drucker von Hewlett Packard
#  Installiert unter:  /usr/local/libexec/hpof

printf "\033&k2G" || exit 2
exec /usr/libexec/lpr/lpf
```

Geben Sie den Pfad des Ausgabefilters über die `of`-Fähigkeit an. Weitere Informationen finden Sie im Abschnitt Ausgabefilter.

Das nächste Beispiel beschreibt die Datei `/etc/printcap` des bereits erwähnten Druckers `teak`. Allerdings sind nun die Ausgabe von Deckblättern sowie der vorhin beschriebene Ausgabefilter enthalten:

```
#
#  /etc/printcap für den Rechner orchid
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0 :sd=/var/spool/lpd/teak :mx#0:\
        :if=/usr/local/libexec/hpif :\
        :vf=/usr/local/libexec/hpvf :\
        :of=/usr/local/libexec/hpof :
```

Wenn ein Anwender nun einen Druckauftrag an den Drucker `teak` schickt, wird für jeden Druckauftrag ein Deckblatt erstellt. Benötigt ein Anwender keine Deckblätter, kann er die Ausgabe dieser Seiten durch die Verwendung von `lpr -h` unterdrücken. Weitere, für die Ausgabe von Deckblättern interessante lpr(1)-Optionen finden Sie im Abschnitt Deckblattoptionen.

Anmerkung

LPD verwendet ein *Form Feed*, um das Deckblatt abzuschließen. Wenn Ihr Drucker ein anderes Zeichen verwendet, um eine Seite auszuwerfen, geben Sie dieses über die `ff`-Fähigkeit in `/etc/printcap` an.

10.4.2.2. Deckblätter kontrollieren

Haben Sie die Ausgabe von Deckblättern aktiviert, gibt LPD eine ganze Seite in großen Buchstaben aus, die den Anwender, den verwendeten Rechner sowie den Druckauftrag beschreiben. Das folgende Beispiel ist ein Deckblatt für den Druckauftrag „outline“, der von `kelly` auf dem Rechner `rose` erstellt wurde:

```
     k                  ll       ll
     k                   l        l
     k                   l        l
     k   k     eeee      l        l     y    y
     k  k     e    e     l        l     y    y
     k k      eeeeee     l        l     y    y
     kk k     e          l        l     y    y
     k   k    e    e     l        l     y   yy
     k    k    eeee     lll      lll     yyy y
                                           y
                                          y    y
                                           yyyy
```

```
                              ll
                    t          l        i
                    t          l
     oooo    u    u   ttttt    l       ii     n nnn     eeee
    o    o   u    u     t      l        i     nn   n   e    e
    o    o   u    u     t      l        i     n    n   eeeeee
    o    o   u    u     t      l        i     n    n   e
    o    o   u   uu     t  t    l        i     n    n   e    e
     oooo     uuu u      tt     lll      iii    n    n    eeee




    r rrr     oooo     ssss     eeee
    rr   r   o    o   s    s   e    e
    r        o    o    ss      eeeeee
    r        o    o      ss    e
    r        o    o   s    s   e    e
    r         oooo     ssss     eeee




                                        Job:  outline
                                        Date: Sun Sep 17 11:04:58 1995
```

LPD fügt ein *Form Feed* an diesen Text an, damit der eigentliche Druckauftrag auf einer neuen Seite gestartet wird (es sei denn, Sie haben die `sf`-Fähigkeit (*suppress form feeds*) des jeweiligen Druckers in `/etc/printcap` aktiviert).

Wenn Sie dies wüschen, kann LPD auch nur ein *kurzes Deckblatt* ausgeben. Dazu verwenden Sie die Option `sb` (*short banner*) in `/etc/printcap`. Dadurch erhalten Sie ein Deckblatt ähnlich dem folgenden:

```
rose:kelly  Job: outline  Date: Sun Sep 17 11:07:51 1995
```

In der Voreinstellung druckt LPD zuerst das Deckblatt und danach den eigentlichen Druckauftrag. Um diese Reihenfolge umzukehren, geben Sie die Option `hl` (*header last*) in `/etc/printcap` ans.

10.4.2.3. Deckblätter verrechnen

Wenn Sie die in LPD eingebaute Funktion zur Erstellung von Deckblättern verwenden, werden Sie auf folgendes Paradigma stoßen: Deckblätter müssen *kostenlos* sein.

Warum ist das so?

Weil der Ausgabefilter das einzige externe Programm ist, das zum Zeitpunkt der Erstellung des Deckblatts eine Verrechnung durchführen könnte. Da Ausgabefilter aber weder über *Benutzer- noch über Rechnerinformationen* verfügen, ist es nicht möglich, einen Druckauftrag einem bestimmten Benutzer zuzuordnen. Da ein Benutzer die Ausgabe von Deckblättern über `lpr -h` unterdrücken kann, ist es auch nicht möglich, die Vorgabe „verrechne eine zusätzliche Seite“ in den Text- oder Konvertierungsfilter (die über die zur Verrechnung nötigen Benutzer- und Rechnerinformationen verfügen) aufzunehmen, weil Benutzer sonst für Deckblätter bezahlen müssten, die sie nicht gedruckt haben.

Es ist *ebenfalls nicht ausreichend*, jeden Filter eigene Deckblätter erzeugen zu lassen (und sie dadurch verrechnen zu können). Wollte ein Benutzer durch ein `lpr -h` die Ausgabe eines Deckblattes unterdrücken, würde dieses nun trotzdem verrechnet werden, da LPD keine Informationen über die Verwendung der Option `-h` an einen Filter weitergibt.

Welche Möglichkeiten habe ich nun?

Sie können:

- Das Paradigma von LPD einfach akzeptieren und die Deckblätter gratis abgeben.
- Eine alternatives Drucksystem wie LPRng installieren. Der Abschnitt Alternativen zum Standard-Drucksystem beschreibt verschiedene Drucksysteme, die LPD ersetzen können.
- Schreiben Sie einen *intelligenten* Ausgabefilter. Normalerweise kümmert sich ein Ausgabefilter nur um die Initialisierung des Druckers oder um eine einfache Zeichenkonvertierung. Außerdem eignet er sich für die Ausgabe von Deckblättern und normalem Text, wenn Sie keinen Text- oder Eingabefilter installiert haben. Haben Sie allerdings einen Textfilter installiert, verwendet LPD Ausgabefilter nur für die Ausgabe von Deckblättern. Ein Ausgabefilter kann den Text des von LPD erzeugten Deckblattes untersuchen, um festzustellen, welcher Benutzer und welcher Rechner den Druckauftrag gestartet hat. Leider weiß der Ausgabefilter auch mit dieser Methode nicht, welche Datei er zur Verrechnung verwenden soll (da der Name dieser Datei durch die `af`-Fähigkeit übergeben wird). Wenn Sie eine Standard-Verrechnungsdatei verwenden, können Sie diese in den Ausgabefilter einbauen. Um den Text des Deckblattes zu untersuchen, verwenden Sie die `sh`-Fähigkeit (*short header*) in `/etc/printcap`. Falls Ihnen das zuviel Aufwand ist, freuen sich Ihre Benutzer sicher darüber, wenn Sie ihnen den kostenlosen Druck von Deckblättern erlauben.

10.4.2.4. Deckblätter auf PostScript®-Druckern ausgeben

In der Regel erzeugt LPD ein Deckblatt mit normalem Text, das für viele verschiedene Drucker geeignet ist. Da PostScript®-Drucker normalen Text aber nicht drucken können, ist die LPD-Funktion zur Erstellung von Deckblättern auf diesen Drucker relativ sinnlos.

Es sei denn, jeder Text- und Konvertierungsfilter erzeugt über den Benutzer- und Rechnernamen sein eigenes, für den jeweiligen Drucker geeignetes Deckblatt. Das Problem dieser Methode ist allerdings, dass ein Anwender auch dann ein Deckblatt erhält, wenn er dies über `lpr -h` verhindern wollte.

Das folgende Skript benötigt drei Argumente (den Loginnamen des Benutzers, den Rechnernamen und den Namen des Druckauftrages), um daraus ein einfaches PostScript®-Deckblatt zu erzeugen:

```
#!/bin/sh
#
#  make-ps-header - ein PostScript-Deckblatt auf stdout ausgeben
#  Installiert unter:  /sr/local/libexec/make-ps-header
#

#
#  Die folgenden Werte sind PostScript-Einheiten (72 pro Zoll).
#  Passen Sie diese Werte für A4 oder die von Ihnen verwendete
#  Papiergröße an:
#
page_width=612
page_height=792
border=72

#
#  Argumente prüfen
#
if [ $# -ne 3 -]; then
    echo "Usage: `basename $0` <user> <host> <job>" 1>&2
    exit 1
fi

#
#  Diese Werte in Variablen speichern, damit der PostScript-Code
#  übersichtlicher wird.
#
user=$1
host=$2
```

```
job=$3
date=`date`

#
#  Sende den PostScript-Code an stdout.
#
exec cat <<EOF
%!PS

%
%  Sicherstellen, dass es keine unerwünschten Wechselwirkungen mit
%  dem folgenden Druckauftrag gibt.
%
save

%
%  Ziehe eine fette Umrandung.
%
$border $border moveto
$page_width $border 2 mul sub 0 rlineto
0 $page_height $border 2 mul sub rlineto
currentscreen 3 -1 roll pop 100 3 1 roll setscreen
$border 2 mul $page_width sub 0 rlineto closepath
0.8 setgray 10 setlinewidth stroke 0 setgray

%
%  Zeige den Benutzernamen groß und fett an.
%
/Helvetica-Bold findfont 64 scalefont setfont
$page_width ($user) stringwidth pop sub 2 div $page_height 200 sub moveto
($user) show

%
%  Und nun zeige noch die Einzelheiten an.
%
/Helvetica findfont 14 scalefont setfont
/y 200 def
[ (Job:) (Host:) (Date:) -] {
200 y moveto show /y y 18 sub def }
forall

/Helvetica-Bold findfont 14 scalefont setfont
/y 200 def
[ ($job) ($host) ($date) -] {
        270 y moveto show /y y 18 sub def
} forall

%
% Das wars.
%
restore
showpage
EOF
```

Nun kann jeder Konvertierungs- oder Textfilter dieses Skript aufrufen, um zuerst das Deckblatt zu erzeugen und danach den Druckauftrag zu drucken. Das nächste Beispiel enthält den bereits beschriebenen DVI-Konvertierungsfilter, der hier um die Funktion zur Erzeugung eines Deckblatts erweitert wurde:

```
#!/bin/sh
#
#  psdf - DVI-nach-PostScript - Druckerfilter
#  Installiert unter:  /usr/local/libexec/psdf
#
#  Wird von lpd aufgerufen, wenn der Benutzer lpr -d verwendet.
#
```

```
orig_args="$@"

fail() {
    echo "$@" 1>&2
    exit 2
}

while getopts "x:y:n:h:" option; do
    case $option in
        x|y)  -;; # Ignore
        n)    login=$OPTARG -;;
        h)    host=$OPTARG -;;
        *)    echo "LPD started `basename $0` wrong." 1>&2
              exit 2
              -;;
    esac
done

[ "$login" -] || fail "No login name"
[ "$host" -] || fail "No host name"

( /usr/local/libexec/make-ps-header $login $host "DVI File"
  /usr/local/bin/dvips -f ) | eval /usr/local/libexec/lprps $orig_args
```

Beachten Sie, dass der Filter die Liste der Argumente überprüft, um den Benutzer- und den Rechnernamen zu ermitteln. Dieser Vorgang ist prinzipiell für alle Filter identisch. Der Textfilter benötigt allerdings etwas andere Argumente, die im Abschnitt Die Funktionsweise von Filtern beschrieben werden.

Wie bereits erwähnt, deaktiviert diese Methode leider die „suppress header page“-Option (also die Option `-h`) von `lpr`. Benutzer können danach den Ausdruck eines Deckblattes nicht mehr verhindern, da der angepasste Filter zu jedem Druckauftrag automatisch ein Deckblatt erstellt.

Damit ein Benutzer bei Bedarf den Ausdruck eines Deckblatts dennoch unterbinden kann, müssen Sie auch hier den im Abschnitt Deckblätter verrechnen beschriebenen Trick einsetzen: Schreiben Sie einen Ausgabefilter, der das von LPD erzeugte Deckblatt untersucht und daraus eine PostScript®-Version erzeugt. Wenn der Benutzer den Druckauftrag mit `lpr -h` verschickt, erzeugt LPD kein Deckblatt, was in weiterer Folge auch für Ihren Ausgabefilter gilt. Soll hingegen ein Deckblatt erzeugt werden, liest der Ausgabefilter den von LPD übergebenen Text und erzeugt daraus ein für Ihren PostScript®-Drucker geeignetes Deckblatt.

Haben Sie Ihren PostScript®-Drucker über eine serielle Verbindung angeschlossen, können Sie auch `lprps` verwenden. In diesem Paket ist mit `psof` auch ein Ausgabefilter enthalten, der die eben beschriebenen Funktionen übernehmen kann. Beachten Sie aber, dass Sie mit `psof` keine Deckblätter verrechnen können.

10.4.3. Drucken über ein Netzwerk

FreeBSD unterstützt das Drucken über ein Netzwerk, also den Versand von Druckaufträgen an einen entfernten Drucker. Man unterscheidet dabei zwei Möglichkeiten:

- Den Zugriff auf einen an einem entfernten Rechner angeschlossenen Drucker. Sie konfigurieren dabei auf Ihrem System einen Drucker, der über eine konventionelle serielle oder parallele Verbindung an einem anderen Rechner angeschlossen ist. Danach richten Sie LPD auf dem entfernten System so ein, dass andere Drucker über das Netzwerk auf diesen Drucker zugreifen können. Der Abschnitt Auf entfernten Rechnern installierte Drucker beschreibt, wie Sie dazu vorgehen müssen.
- Den Zugriff auf einen direkt an ein Netzwerk angeschlossenen Drucker. Ein solcher Drucker verfügt anstelle (oder zusätzlich zu) einer parallelen oder seriellen Schnittstelle über eine Netzwerkschnittstelle. Ein solcher Drucker kann sich auf zwei Arten verhalten:
  - Er kann das LPD-Protokoll direkt unterstützen und sogar Druckjobs von entfernten Rechner verwalten. In diesem Fall verhält sich der Drucker wie ein normaler Rechner, auf dem LPD läuft. Lesen Sie den Abschnitt Auf entfernten Rechnern installierte Drucker, um einen solchen Drucker einzurichten.

- Er könnte Verbindungen über ein Netzwerk unterstützen. In diesem Fall „verbinden“ Sie den Drucker mit einem Rechner Ihres Netzwerks, der danach für die Verwaltung von Druckaufträgen sowie den tatsächlichen Druck verantwortlich ist. Der Abschnitt Drucker mit direkter TCP-Schnittstelle enthält Hinweise zur Installation derartiger Drucker.

10.4.3.1. Auf entfernten Rechnern installierte Drucker

Das LPD-Drucksystem unterstützt den Versand von Druckaufträgen an andere Rechner, auf denen entweder LPD läuft oder die zu LPD kompatibel sind. Dadurch können Sie einen Drucker auf einem Rechner installieren und von anderen Rechnern des Netzwerks darauf zugreifen. Außerdem werden Drucker mit direkter TCP-Schnittstelle unterstützt, wenn diese das LPD-Protokoll unterstützen.

Um diese Art des Druckens über ein Netzwerk zu aktivieren, installieren Sie zuerst Ihren Drucker auf einem Rechner Ihres Netzwerks, dem sogenannten *printer host*. Die dazu nötigen Schritte werden im Abschnitt Einfache Drucker-Konfiguration beschrieben. Falls Sie eine erweiterte Druckerkonfiguration benötigen, sollten Sie auch den Abschnitt Erweiterte Drucker-Konfiguration lesen. Danach testen Sie, ob der Drucker alle von Ihnen aktivierten LPD-Fähigkeiten unterstützt. Stellen Sie auch sicher, dass Ihr *lokales System* berechtigt ist, den LPD-Dienst auf dem *entfernten System* zu nutzen (lesen Sie dazu den Abschnitt Druckaufträge auf entfernten Druckern beschränken).

Wenn Sie einen Drucker mit einer zu LPD kompatiblen Netzwerkschnittstelle verwenden, handelt es sich beim *printer host* um den Drucker selbst, und der *Druckername* ist der von Ihnen für diesen Drucker vorgegebene Name. Lesen Sie die Dokumentation Ihres Druckers und/oder der Netzwerkschnittstelle Ihres Druckers, um dies zu klären.

Tipp

Wenn Sie einen Hewlett Packard Laserjet-Drucker verwenden, sorgt der Druckername `text` für eine automatische LF-zu-CRLF-Konvertierung. In diesem Fall wird das `hpif`-Skript nicht benötigt.

Danach müssen Sie auf jedem Rechner, der auf diesen Drucker zugreifen soll, einen entsprechenden Eintrag in deren `/etc/printcap` aufnehmen. Dazu werden folgende Informationen benötigt:

1. Der Name des Eintrags. Entspricht in der Regel dem Eintrag auf dem *printer host*.
2. Lassen Sie den Eintrag für die `lp`-Fähigkeit leer, schreiben Sie also `:lp=:`.
3. Erzeugen Sie ein Spooling-Verzeichnis und geben Sie dessen Pfad über die `sd`-Fähigkeit an. LPD speichert Ihre Druckaufträge in diesem Verzeichnis, bevor sie an den Drucker geschickt werden.
4. Geben Sie den Namen des *printer hosts* über die `rm`-Fähigkeit an.
5. Geben Sie den Namen des Druckers (auf dem *printer host*) über die `rp`-Fähigkeit an.

Das ist alles. Sie benötigen weder Konvertierungsfilter, noch Seitengrößen oder sonstige Angaben in Ihrer lokalen `/etc/printcap`.

Dazu ein Beispiel. Der Rechner `rose` verfügt über zwei Drucker, `bamboo` und `rattan`. Wir wollen nun allen Benutzern des Rechners `orchid` erlauben, diese Drucker zu verwenden. Es folgt nun wieder die bereits aus dem Abschnitt Deckblätter verwenden bekannte `/etc/printcap` für den Rechner `orchid`. Diese enthielt bereits einen Eintrag für den Drucker `teak`. Zusätzlich tragen wir nun die zwei Drucker des Rechners `rose` ein:

```
#
#  /etc/printcap für den Rechner orchid - mit zusätzlichen
#  Einträgen für die (entfernten) Drucker auf dem Rechner rose
#
```

```
#
#  teak ist ein lokaler Drucker und direkt mit orchid verbunden:
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0 :sd=/var/spool/lpd/teak :mx#0:\
        :if=/usr/local/libexec/ifhp :\
        :vf=/usr/local/libexec/vfhp :\
        :of=/usr/local/libexec/ofhp :

#
#  rattan ist mit rose verbunden, Druckaufträge für rattan gehen daher
#  an den Rechner rose:
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :lp=:rm=rose:rp=rattan:sd=/var/spool/lpd/rattan :

#
#  bamboo ist ebenfalls mit rose verbunden:
#
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :lp=:rm=rose:rp=bamboo:sd=/var/spool/lpd/bamboo :
```

Nun müssen wir nur noch die Spooling-Verzeichnisse auf dem Rechner `orchid` erzeugen:

```
# mkdir -p /var/spool/lpd/rattan /var/spool/lpd/bamboo
# chmod 770 /var/spool/lpd/rattan /var/spool/lpd/bamboo
# chown daemon:daemon /var/spool/lpd/rattan /var/spool/lpd/bamboo
```

Damit können Benutzer des Rechners `orchid` die Drucker `rattan` und `bamboo` verwenden. Ein Benutzer gibt auf `orchid` beispielsweise ein:

```
% lpr -P bamboo -d sushi-review.dvi
```

Die Anwendung LPD auf dem Rechner `orchid` kopiert daraufhin den Druckauftrag in das Spooling-Verzeichnis `/var/spool/lpd/bamboo` und stellt fest, dass es sich um einen DVI-Auftrag handelt. Sobald `rose` über genug freien Platz im `bamboo`-Spooling-Verzeichnis verfügt, würden die beiden LPD die Datei auf den Rechner `rose` transferieren. Diese Datei verbleibt danach in der Druckerwarteschlange des Rechners `rose`, bis der Ausdruck der Datei abgeschlossen ist. Vor dem Ausdruck würde die Datei noch von DVI nach PostScript® konvertiert werden, da es sich bei `bamboo` um einen an den Rechner `rose` angeschlossenen PostScript®-Drucker handelt.

10.4.3.2. Drucker mit direkter TCP-Schnittstelle

Wenn Sie eine Netzwerkkarte für Ihren Drucker kaufen, können Sie zwei verschiedene Versionen wählen: Eine Version, die ein Drucksystem emuliert (die teure Version), oder eine Version, die sich verhält, als wäre der Drucker an eine serielle oder parallele Schnittstelle angeschlossen (die billige Version). Dieser Abschnitt beschreibt die billige Variante. Bevorzugen Sie die teure Variante, sollten Sie den Abschnitt Auf entfernten Rechnern installierte Drucker nochmals lesen.

Das Format der Datei `/etc/printcap` erlaubt es Ihnen, anzugeben, welche serielle oder parallele Schnittstelle verwendet werden soll und (falls Sie eine serielle Schnittstelle verwenden) welche Parameter (Baudrate, Flußkontrolle, Behandlung von Tabulatoren, Konvertierung von neuen Zeilen und andere mehr) Sie verwenden wollen. Es gibt allerdings keine Möglichkeit, eine Verbindung zu einem Drucker zu definieren, der einen TCP/IP- oder einem anderem Netzwerkport auf Druckaufträge hin abfragt.

Um Daten an einen Netzwerkdrucker zu schicken, müssen Sie daher ein Kommunikationsprogramm entwickeln, das von Text- und Konvertierungsfiltern aufgerufen werden kann. Dazu ein Beispiel. Das Skript `netprint` übernimmt alle Daten von der Standardeingabe und schickt sie an einen Netzwerkdrucker. `netprint` erwartet zwei Argumente: Als erstes Argument wird der Hostname des Druckers und als zweites Argument der Port, über den die Verbindung erfolgen soll, übergeben. Dabei handelt sich allerdings um eine Ein-Wege-Kommunikation (von FreeBSD zum Drucker). Viele Netzwerkdrucker unterstützen aber auch eine Zwei-Wege-Kommunikation, deren Vorteile (Abfrage des Druckerstatus, die Verrechnung von Druckaufträgen und andere mehr) Sie vielleicht nutzen wollen.

```
#!/usr/bin/perl
#
#  netprint - Textfilter für einen Netzwerkdrucker
#  Installiert unter:  /usr/local/libexec/netprint
#
$#ARGV eq 1 || die "Usage: $0 <printer-hostname> <port-number>";

$printer_host = $ARGV[0];
$printer_port = $ARGV[1];

require 'sys/socket.ph';

($ignore, $ignore, $protocol) = getprotobyname('tcp');
($ignore, $ignore, $ignore, $ignore, $address)
    = gethostbyname($printer_host);

$sockaddr = pack('S n a4 x8', &AF_INET, $printer_port, $address);

socket(PRINTER, &PF_INET, &SOCK_STREAM, $protocol)
    || die "Can't create TCP/IP stream socket: $!";
connect(PRINTER, $sockaddr) || die "Can't contact $printer_host: $!";
while (<STDIN>) { print PRINTER; }
exit 0;
```

Dieses Skript kann für verschiedene Filter eingesetzt werden. Das folgende Beispiel verwendet den an ein Netzwerk angeschlossenen Zeilendrucker Diablo 750-N. Dieser Drucker empfängt zu druckende Daten auf dem Port 5100. Der Hostname des Druckers lautet `scrivener`. Daher sieht der Textfilter für diesen Drucker wie folgt aus:

```
#!/bin/sh
#
#  diablo-if-net - Textfilter für den Diablo-Drucker `scrivener'.
#  Drucker lauscht auf Port 5100.
#  Installiert unter:  /usr/local/libexec/diablo-if-net
#
exec /usr/libexec/lpr/lpf "$@" | /usr/local/libexec/netprint scrivener 5100
```

10.4.4. Den Druckerzugriff beschränken

Dieser Abschnitt beschreibt, wie Sie den Druckerzugriff beschränken können. Das LPD-Drucksystem erlaubt Ihnen die Kontrolle darüber, wer lokal oder über ein Netzwerk auf einen Drucker zugreifen darf, ob mehrere Kopien erstellt werden dürfen und wie groß Druckaufträge und Druckerwarteschlangen werden dürfen.

10.4.4.1. Den Ausdruck von mehreren Kopien verhindern

Das LPD-System macht es dem einzelnen Benutzer einfach, mehrere Kopien einer Datei zu drucken. So werden mit `lpr -#5` beispielsweise fünf Kopien jeder Datei des Druckauftrags erstellt. Ob dies gut oder schlecht ist, müssen Sie selbst entscheiden.

Wenn Sie der Meinung sind, dass multiple Kopien eine unnötige Beanspruchung Ihres Druckers darstellen, sollten Sie die `-#`-Opion von lpr(1) deaktivieren, indem Sie die `sc`-Fähigkeit in Ihre `/etc/printcap` aufnehmen. Verwendet ein Benutzer dennoch die Option `-#`, erhält er daraufhin folgende Meldung:

```
lpr: multiple copies are not allowed
```

Wenn Sie den Zugriff auf einen entfernten Drucker (wie in Abschnitt Auf entfernten Rechnern installierte Drucker beschrieben) konfiguriert haben, müssen Sie die `sc`-Fähigkeit auch auf den entfernten Rechnern, die auf Ihren Drucker zugreifen dürfen, in `/etc/printcap` eintragen, damit Benutzer diese Vorgabe nicht durch den Wechsel auf einen anderen Rechner umgehen können.

Dazu ein Beispiel. Es handelt sich dabei um die Datei `/etc/printcap` auf dem Rechner `rose`. Der Drucker `rattan` soll multiple Kopien zulassen, auf dem Laserdrucker `bamboo` sollen multiple Kopien hingegen nicht erlaubt sein, daher müssen wir für diesen Drucker die `sc`-Fähigkeit aktivieren:

```
#
#  /etc/printcap für den Rechner rose - multiple Kopien auf bamboo verbieten
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :\
        :lp=/dev/lpt0 :\
        :if=/usr/local/libexec/if-simple  :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :sc:\
        :lp=/dev/ttyu5 :ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif :\
        :df=/usr/local/libexec/psdf :
```

Außerdem müssen wir noch die `sc`-Fähigkeit in der Datei `/etc/printcap` des Rechners `orchid` aktivieren. Parallel dazu untersagen wir das Erstellen von multiplen Kopien auf dem Drucker `teak`:

```
#
#  /etc/printcap für den Rechner orchid - lokal machen wir keine multiplen Kopien
#  Lokaler Drucker teak oder entfernter Drucker bamboo:
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0 :sd=/var/spool/lpd/teak :mx#0:sc:\
        :if=/usr/local/libexec/ifhp :\
        :vf=/usr/local/libexec/vfhp :\
        :of=/usr/local/libexec/ofhp :

rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :lp=:rm=rose:rp=rattan:sd=/var/spool/lpd/rattan :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :lp=:rm=rose:rp=bamboo:sd=/var/spool/lpd/bamboo :sc:
```

Durch die Verwendung der `sc`-Fähigkeit ist zwar die Verwendung von `lpr -#` nicht mehr möglich, ein Benutzer kann aber weiterhin lpr(1) mehrmals hintereinander aufrufen oder eine Datei mehrfach in den gleichen Druckauftrag aufnehmen:

```
% lpr forsale.sign forsale.sign forsale.sign forsale.sign forsale.sign
```

Auch dieser Mißbrauch Ihres Druckers kann verhindert werden, falls Sie dies wünschen. Diese Maßnahmen werden in diesem Abschnitt allerdings nicht behandelt.

10.4.4.2. Den Zugriff auf bestimmte Drucker beschränken

Sie können angeben, wer auf welchem Drucker drucken darf, wenn Sie den Gruppenmechanismus von UNIX® in Kombination mit der `rg`-Fähigkeit von `/etc/printcap` einsetzen. Weisen Sie dazu alle Benutzer, die auf einen Drucker zugreifen dürfen, einer gemeinsamen Gruppe zu und geben Sie diese Gruppe über die `rg`-Fähigkeit an.

Wenn Benutzer, die dieser Gruppe nicht angehören (dies gilt auch für `root`), werden diese durch die Meldung begrüsst, wenn Sie diesen Drucker verwenden wollen.

```
lpr: Not a member of the restricted group
```

Analog zur `sc`-Fähigkeit (*suppress multiple copies*) müssen Sie die `rg`-Fähigkeit auch auf allen entfernten Rechnern aktivieren, die auf Ihren Drucker zugreifen dürfen (lesen Sie dazu auch den Abschnitt Auf entfernten Rechnern installierte Drucker).

Wollen wir beispielsweise allen Benutzern die Verwendung des Druckers `rattan`, aber nur Mitgliedern der Gruppe `artists` die Verwendung des Druckers `bamboo` erlauben, passen wir die bereits bekannte `/etc/printcap` des Rechners `rose` entsprechend an:

```
#
#  /etc/printcap des Rechners rose - Zugriffsbeschränkung für bamboo
#
```

```
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :\
        :lp=/dev/lpt0 :\
        :if=/usr/local/libexec/if-simple  :

bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :sc:rg=artists:\
        :lp=/dev/ttyu5 :ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif :\
        :df=/usr/local/libexec/psdf :
```

Die Datei `/etc/printcap` des Rechners `orchid` wird dadurch nicht beeinflusst. Jeder Benutzer des Rechners `orchid` kann also weiterhin den Drucker `bamboo` verwenden.

Anmerkung

Für jeden Drucker kann nur eine einzige priviligierte Gruppe erstellt werden.

10.4.4.3. Die Größe von Druckaufträgen kontrollieren

Wenn Sie viele Benutzer haben, die Ihre Drucker verwenden dürfen, werden Sie wahrscheinlich eine Obergrenze für Dateien angeben wollen, die Benutzer an Ihren Drucker senden dürfen. Dies ist sinnvoll, weil Speicherplatz für Spooling-Verzeichnisse nur begrenzt verfügbar ist und Sie stets sicherstellen müssen, dass auch die Druckaufträge anderer Benutzer verarbeitet werden können.

LPD verwendet die `mx`-Fähigkeit, um die maximal erlaubte Größe von Dateien eines Druckauftrags anzugeben. Dieser Wert wird in 1.024 Bytes großen `BUFSIZ`-Blöcken angegeben. Setzen Sie diesen Wert auf Null, gibt es keine Größenbeschränkung. Existiert die `mx`-Fähigkeit hingegen überhaupt nicht, so gilt ein Limit von 1.000 Blöcken.

Anmerkung

Diese Limits gelten nur für die Größe von *Dateien* innerhalb eines Druckauftrages, *nicht aber* für die Gesamtgröße des Druckauftrags.

LPD lehnt eine Datei auch dann nicht ab, wenn sie das Limit des Druckers überschreitet. Vielmehr wird die Datei bis zum Erreichen des Limits in die Warteschlange geladen, danach wird der Druck gestartet. Der das Limit überschreitende Rest wird hingegen verworfen und nicht gedruckt!

Mit diesem Wissen können wir nun Limits für die Drucker `rattan` und `bamboo` definieren. Da PostScript®-Dateien der Gruppe `artists` in der Regel sehr groß sind, setzen wir ein Limit von fünf Megabytes. Für den Druck von normalen Text (auf dem Drucker `rattan`) setzen wir hingegen kein Limit:

```
#
#  /etc/printcap für den Rechner rose
#

#
#  Kein Größenlimit:
#
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:mx#0:sd=/var/spool/lpd/rattan :\
        :lp=/dev/lpt0 :\
        :if=/usr/local/libexec/if-simple  :

#
```

```
#  Ein Limit von 5 Megabyte:
#
bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
        :sh:sd=/var/spool/lpd/bamboo :sc:rg=artists:mx#5000:\
        :lp=/dev/ttyu5 :ms#-parenb cs8 clocal crtscts:rw:\
        :if=/usr/local/libexec/psif :\
        :df=/usr/local/libexec/psdf :
```

Auch diese Limits gelten nur für lokale Benutzer. Wenn Sie den Zugriff auf Ihren Drucker auch über ein Netzwerk erlauben wollen, unterliegen die Benutzer dieser Rechner diesen Limits nicht. Daher müssen Sie diese Limits über die `mx`-Fähigkeit auch in der `/etc/printcap` jedes Rechners definieren, der Ihren Drucker verwenden darf. Der Abschnitt Auf entfernten Rechnern installierte Drucker enthält weitere Informationen zum Drucken über ein Netzwerk.

Es gibt eine weitere Möglichkeit, um die Größe von Druckaufträgen von entfernten Rechnern zu beschränken. Lesen Sie dazu den Abschnitt Druckaufträge von entfernten Rechnern beschränken.

10.4.4.4. Druckaufträge von entfernten Rechnern beschränken

Das LPD-System bietet mehrere Möglichkeiten, um Druckaufträge zu beschränken, die auf entfernten Rechnern gestartet wurden:

Rechner beschränken
: Sie können festlegen, von welchen entfernten Rechnern ein lokaler LPD Druckaufträge annimmt, indem Sie die Dateien `/etc/hosts.equiv` sowie `/etc/hosts.lpd` entsprechend anpassen. LPD überprüft diese Dateien, um festzustellen, ob ein Druckauftrag von einem Rechner stammt, der in einer dieser Dateien aufgeführt ist. Ist dies nicht der Fall, lehnt LPD den Druckauftrag ab.

  Der Aufbau dieser Datei ist sehr einfach: Jede Zeile enthält einen einzigen Rechnernamen. Beachten Sie aber, dass `/etc/hosts.equiv` auch vom ruserok(3)-Protokoll benötigt wird und Änderungen dieser Datei auch Programme wie rsh(1) und rcp(1) beeinflussen können.

  Das folgende Beispiel beschreibt die Datei `/etc/hosts.lpd` auf dem Rechner `rose`:

  ```
  orchid
  violet
  madrigal.fishbaum.de
  ```

  Durch diese Vorgaben akzeptiert `rose` nur noch Druckaufträge von den Rechnern `orchid`, `violet`, und `madrigal.fishbaum.de`. Versucht ein anderer Rechner, auf den LPD von `rose` zuzugreifen, wird dieser Druckauftrag abgelehnt werden.

Größenbeschränkungen
: Sie können festlegen, wieviel Speicherplatz auf dem Dateisystem, in dem das Spooling-Verzeichnis liegt, mindestens frei sein muss. Dazu erzeugen Sie im Spooling-Verzeichnis Ihres lokalen Druckers die Datei `minfree`. In dieser Datei geben Sie an, wieviele 512 Byte große Blöcke auf Ihrer Platte frei sein müssen, damit ein Druckauftrag von einem entfernten Rechner akzeptiert wird.

  Durch diese Vorgabe können Sie sicherstellen, dass Benutzer von entfernten Rechnern Ihr Dateisystem nicht „zumüllen“. Außerdem können Sie damit lokale Benutzer bevorzugen, da diese auch dann noch Druckaufträge erteilen dürfen, wenn der verfügbare Plattenplatz unter das in der Datei `minfree` definierte Limit gefallen ist.

  Legen wir nun die Datei `minfree` für den Drucker `bamboo` an. Zuerst untersuchen wir `/etc/printcap`, um das Spooling-Verzeichnis für diesen Drucker zu finden. Das folgende Beispiel zeigt den Eintrag für den Drucker `bamboo`:

  ```
  bamboo|ps|PS|S|panasonic|Panasonic KX-P4455 PostScript v51.4:\
          :sh:sd=/var/spool/lpd/bamboo :sc:rg=artists:mx#5000:\
          :lp=/dev/ttyu5 :ms#-parenb cs8 clocal crtscts:rw:mx#5000:\
          :if=/usr/local/libexec/psif :\
  ```

```
        :df=/usr/local/libexec/psdf  :
```

Das Spooling-Verzeichnis wird über die `sd`-Fähigkeit festgelegt. Wir wollen, dass mindestens drei Megabyte (also 6144 Blöcke) freier Plattenplatz vorhanden sein müssen, damit LPD einen Druckauftrag von einem entfernten Rechner akzeptiert:

```
# echo 6144 > /var/spool/lpd/bamboo/minfree
```

Benutzer beschränken
: Sie können auch festlegen, welche entfernten Benutzer Ihren lokalen Drucker verwenden dürfen, indem Sie die `rs`-Fähigkeit in `/etc/printcap` definieren. Wenn für den Eintrag eines lokalen Druckers die `rs`-Fähigkeit definiert ist, akzeptiert LPD Druckaufträge von entfernten Rechnern nur dann, *wenn* der Benutzer, der den Druckauftrag gesendet hat, auch über ein gleichnamiges Benutzerkonto auf dem lokalen Rechner verfügt. Ist dies nicht der Fall, lehnt LPD den Druckauftrag ab.

  Diese Fähigkeit ist besonders in Umgebungen nützlich, in denen beispielsweise verschiedene Abteilungen ein gemeinsames Netzwerk teilen, wobei einige Benutzer zu mehreren Abteilungen gehören. Haben diese Benutzer auch ein Benutzerkonto auf Ihrem System, so können sie Ihren Drucker auch von ihrer eigenen Abteilung aus nutzen. Wollen Sie zwar den Zugriff auf Ihren Drucker, *nicht aber* den Zugriff auf Ihre übrigen Ressourcen erlauben, können Sie für diese Benutzer einen sogenannten „Token-Account“ ohne Heimatverzeichnis und mit einer nutzlosen Shell wie `/usr/bin/false` erstellen.

10.4.5. Die Druckernutzung verrechnen

Sie wollen die Nutzung Ihrer Drucker kostenpflichtig machen? Warum auch nicht? Papier und Tinte kosten Geld. Auch eine regelmäßige Wartung muss bezahlt werden. Nachdem Sie einen Preis festgelegt haben, den Sie für jede gedruckte Seite verrechnen wollen, stellt sich die Frage, wie Sie die Verrechnung der Druckkosten technisch umsetzen können.

Die schlechte Nachricht ist, dass das LPD-System dabei wenig hilfreich ist. Die Verrechnung von Druckaufträgen hängt stark vom verwendeten Drucker, den zu druckenden Dateiformaten und *Ihren* Anforderungen an die Verrechnung der Druckernutzung ab.

Um die Verrechnung der Druckernutzung zu implementieren, müssen Sie sowohl Ihre Textfilter (um den Druck von normalem Text abzurechnen) als auch Ihre Konvertierungsfilter (um den Druck sonstiger Formate abzurechnen) entsprechend anpassen, damit diese die Zahl der gedruckten Seiten ermitteln können. Leider können Sie dazu nicht einen einfachen Ausgabefilter verwenden, da diese die Verrechnung von Druckaufträgen nicht unterstützen. Weitere Informationen zu den verschiedenen Filterarten finden Sie im Abschnitt Filter.

Prinzipiell gibt es zwei Möglichkeiten, wie Sie diese Verrechnung umsetzen können:

- Die *periodische Verrechnung* wird häufiger verwendet, da sie einfacher zu implementieren ist. Wenn ein Druckauftrag ausgeführt wird, schreibt der Filter den Benutzer, den verwendeten Rechner sowie die Anzahl der gedruckten Seiten in eine Verrechnungsdatei. Nach einem zu definierenden Zeitraum werden diese Dateien ausgewertet, die Gesamtzahl der von einem Benutzer gedruckten Seiten bestimmt und dem jeweiligen Benutzer verrechnet. Danach werden alle Protokolldateien zurückgesetzt, und die Protokollierung beginnt von Neuem.

- Die *unmittelbare Verrechnung* wird nur selten eingesetzt, das sie schwieriger zu implementieren ist. Bei dieser Methode wird der Druckauftrag verrechnet, sobald der Drucker verwendet wird. Dadurch können Sie beispielsweise verhindern, dass ein Benutzer seine erlaubte „Druckquote“ überschreitet. Zusätzlich können Sie es Ihren Benutzern erlauben, deren Druckquote abzufragen oder anzupassen. Allerdings benötigen Sie eine Datenbank, um Benutzer und deren Quoten verwalten zu können.

Das LPD-Drucksystem unterstützt beide Methoden. Allerdings müssen Sie die benötigen Filter sowie den zur Verrechnung nötigen Code selbst bereitstellen. Der Vorteil dabei ist allerdings, dass Sie in der Wahl Ihrer Verrechnungsmethode äußerst flexibel sind. So können Sie sich etwa für die periodische oder die unmittelbare Verrechnung entscheiden. Sie können festlegen, welche Informationen Sie erfassen wollen: Benutzernamen, Rechnernamen, die Art der Druckaufträge, die Anzahl der gedruckten Seiten, den Papierverbrauch, den Zeitaufwand für die

Bearbeitung eines Druckauftrages und viele andere mehr. Dazu müssen Sie Ihre Filter entsprechend anpassen, damit diese Informationen erfassst und gespeichert werden.

10.4.5.1. Kurzanleitung für die Implementierung der Druckerverrechnung

FreeBSD bietet Ihnen zwei Programme, um eine periodische Verrechnung rasch zu implementieren. Dabei handelt es sich um den im Abschnitt lpf: Ein Textfilter behandelten Textfilter sowie um pac(8), ein Programm, mit dem Sie Einträge aus Verrechnungsdateien auslesen und aufsummieren können.

Wie bereits im Abschnitt Filter erwähnt, startet LPD den Text- oder Konvertierungsfilter mit dem Namen der Verrechnungsdatei als Argument. Dadurch weiß der Filter, in welche Datei er einen Verrechnungseintrag schreiben soll. Der Name dieser Datei wird über die `af`-Fähigkeit in `/etc/printcap` festgelegt. Falls die Datei nicht über einen absoluten Pfad angegeben wird, handelt es sich um einen Pfad relativ zum Spooling-Verzeichnis.

LPD startet `lpf` mit den Argumenten *page width und page length*, die über die `pw`- und `pl`-Fähigkeit definiert werden. Das Kommando `lpf` verwendet diese Argumente danach, um den Papierverbrauch zu bestimmen. Nachdem die Datei an den Drucker geschickt wurde, wird ein Verrechnungseintrag in die Verrechnungsdatei geschrieben. Ein solcher Eintrag sieht dabei ähnlich den folgenden aus:

```
2.00 rose:andy
3.00 rose:kelly
3.00 orchid:mary
5.00 orchid:mary
2.00 orchid:zhang
```

Sie sollten für jeden Drucker eine eigene Verrechnungsdatei verwenden, da `lpf` die Verrechnungsdatei nicht sperren kann. Sind also gleichzeitig zwei `lpf`-Instanzen aktiv, kann es dazu kommen, dass Ihre Verrechnungsdatei zerstört wird, wenn beide Instanzen gleichzeitig in die gleiche Datei schreiben. Damit für jeden Drucker eine eigene Verrechnungsdatei angelegt wird, fügen Sie den Eintrag `af=acct` in `/etc/printcap` ein. Dadurch wird für jeden Drucker eine separate Verrechnungsdatei mit dem Namen `acct` im Spooling-Verzeichnis des jeweiligen Druckers erzeugt.

Wenn Sie Ihre Daten erfasst haben und die entstandenen Kosten Ihren Benutzern verrechnen wollen, starten Sie pac(8). Dazu wechseln Sie in das Spooling-Verzeichnis des auszuwertenden Druckers und geben `pac` ein. Dadurch erhalten Sie eine Ausgabe ähnlich der folgenden:

```
  Login               pages/feet   runs    price
orchid:kelly               5.00    1   $  0.10
orchid:mary               31.00    3   $  0.62
orchid:zhang               9.00    1   $  0.18
rose:andy                  2.00    1   $  0.04
rose:kelly               177.00  104   $  3.54
rose:mary                 87.00   32   $  1.74
rose:root                 26.00   12   $  0.52

total                    337.00  154   $  6.74
```

Folgende Argumente können an pac(8) übergeben werden:

`-P`*`Drucker`*
: Gibt an, welcher *`Drucker`* ausgewertet werden soll. Diese Option setzt voraus, dass für die `af`-Fähigkeit in `/etc/printcap` ein absoluter Pfad angegeben wurde.

`-c`
: Sortiert die Ausgabe nach den verursachten Kosten anstelle einer alphabetischen Sortierung der Benutzernamen.

`-m`
: Ignoriert den Rechnernamen in Verrechnungsdateien. Ist diese Option gesetzt, ist der Benutzer `smith` auf dem Rechner `alpha` mit dem Benutzer `smith` auf dem Rechner `gamma` identisch. Ist diese Option nicht gesetzt, handelt es sich um unterschiedliche Benutzer.

`-p`*`Preis`*
: Berechnet die entstandenen Kosten aus dem *`Preis`* in Dollar pro Seite statt aus dem über die `pc`-Fähigkeit in `/etc/printcap` definierten Preis. In der Voreinstellung sind dies zwei Cent pro Seite. Sie können aber auch einen eigenen *`Preis`* in Form einer Gleitkommazahl angeben.

`-r`
: Die Sortierreihenfolge umkehren.

`-s`
: Die Verrechnungsdatei in einer neuen Datei aufsummieren und die originale Verrechnungsdatei zurücksetzen.

`name …`
: Verrechnungsinformationen nur für die angegebenen Benutzernamen ausgeben.

In der Voreinstellung gibt pac(8) aus, wieviele Seiten von welchem Benutzer auf welchem Rechner gedruckt wurden. Wenn Rechnernamen für Sie uninteressant sind (weil sich Benutzer beispielsweise auf jedem Rechner anmelden können), sollten Sie `pac -m` verwenden, um die folgende Ausgabe zu erhalten:

```
  Login               pages/feet   runs    price
andy                        2.00    1   $  0.04
kelly                     182.00  105   $  3.64
mary                      118.00   35   $  2.36
root                       26.00   12   $  0.52
zhang                       9.00    1   $  0.18

total                     337.00  154   $  6.74
```

Um den zu verrechnenden Betrag zu ermitteln, verwendet pac(8) die `pc`-Fähigkeit von `/etc/printcap` (Voreinstellung 200, dieser Wert entspricht 2 Cents). Geben Sie hier (als Hundertfaches des tatsächlichen Wertes) den Preis pro Seite an, den Sie verrechnen wollen. Sie können diesen Wert überschreiben, wenn Sie pac(8) mit der Option `-p` ausführen. Beachten Sie dabei aber, dass Sie in diesem Fall die Einheiten in Dollar angeben, und nicht als Hundertfaches des tatsächlichen Cent-Betrages. So steht

```
# pac -p1.50
```

beispielsweise für einen Preis von einem Dollar und fünfzig Cent pro Seite.

Der Aufruf von `pac -s` führt schließlich dazu, dass die aufsummierten Informationen in einer eigenen Auswertedatei gespeichert werden. Diese hat den gleichen Namen wie die Verrechnungsdatei, es wird lediglich ein `_sum` an den Dateinamen angehängt. Danach wird die Verrechnungsdatei zurückgesetzt. Wenn Sie pac(8) erneut aufrufen, wird die Auswertedatei eingelesen, um die Startbeträge zu erhalten, alle weiteren Informationen stammen danach aus der normalen Verrechnungsdatei.

10.4.5.2. Wie kann man die Anzahl der gedruckten Seiten ermitteln?

Um die Druckernutzung auch nur annähernd genau verrechnen zu können, müssen Sie ermitteln, wieviel Papier ein Druckauftrag verbraucht. Die Bestimmung dieses Wertes ist das zentrale Problem, das Sie lösen müssen, wenn Sie Druckaufträge kostenpflichtig machen wollen.

Normaler Text stellt in der Regel kein Problem dar: Sie zählen dazu nur die Zeilen des Druckauftrages und dividieren diesen Wert durch die Anzahl der Zeilen pro Seite, die Ihr Drucker bietet. Allerdings dürfen Sie dabei nicht vergessen, dass gelöschte Zeichen (*Backspaces*) Zeilen überschreiben. Außerdem können sich lange logische Zeilen (im Druckauftrag) über mehrere physikalische Zeilen (am Ausdruck) erstrecken.

Der im Abschnitt lpf: Ein Textfilter vorgestellte Textfilter `lpf` berücksichtigt diese Besonderheiten. Wenn Sie einen eigenen Textfilter für die Verrechnung der Druckernutzung schreiben wollen, sollten Sie sich daher den Quellcode von `lpf` näher ansehen.

Aber was ist mit anderen Dateiformaten?

Für die DVI-nach-LaserJet- oder für die DVI-nach-PostScript®-Konvertierung können Sie die Protokolldateien von `dvilj` oder `dvips` auslesen, um festzustellen, wieviele Seiten konvertiert wurden. Die gleiche Methode könnte auch mit anderen Dateitypen funktionieren.

Alle diese Methoden haben aber das Problem, dass ein Drucker möglicherweise nicht alle Seiten des Druckauftrages drucken kann. So könnte es etwa zu einem Papierstau kommen, der Toner könnte zu Ende gehen oder es könnte ein Druckerdefekt auftreten – trotzdem würden alle Seiten des Druckauftrages verrechnet werden.

Was kann man dagegen tun?

Es gibt nur eine einzige *sichere* Methode, um die Druckernutzung *exakt* zu bestimmen. Besorgen Sie sich einen Drucker, der das verbrauchte Papier protokolliert und verbinden Sie ihn über eine serielle oder eine Netzwerkverbindung. Nahezu alle PostScript®-Drucker, aber auch viele andere Modelle und Druckertypen (beispielsweise Laserdrucker von Imagen) sind dazu in der Lage. Passen Sie die Filter für diese Drucker entsprechend an, damit diese nach jedem Druckauftrag die Anzahl der gedruckten Seiten ermitteln und verrechnen Sie Druckaufträge *ausschließlich* über diesen Wert. Danach müssen Sie sich um die Anzahl der gedruckten Zeilen oder um mögliche Druckerprobleme nie mehr kümmern.

Sie können aber auch großzügig sein und alle Ausdrucke kostenlos abgeben.

10.5. Drucker verwenden

Übersetzt von Johann Kois.

Dieser Abschnitt beschreibt, wie Sie einen unter FreeBSD konfigurierten Drucker verwenden können. Die folgende Liste bietet einen Überblick über wichtige Anwenderbefehle:

lpr(1)
: Einen Druckauftrag drucken

lpq(1)
: Eine Druckerwarteschlange prüfen

lprm(1)
: Einen Druckauftrag aus einer Warteschlange entfernen (stornieren)

Zusätzlich existiert mit lpc(8) ein Befehl zur zur Steuerung von Druckern und Druckerwarteschlangen, der im Abschnitt Drucker verwalten näher beschrieben wird.

Jeder der drei Befehle lpr(1), lprm(1), sowie lpq(1) akzeptiert die Option `-P` *`printer-name`*, mit der Sie den zu verwendenden Drucker (der dazu in `/etc/printcap` definiert sein muss) festlegen. Dadurch sind Sie in der Lage, Druckaufträge zu erstellen, zu stornieren, oder den Status Ihrer Druckaufträge zu überprüfen. Verwenden Sie die Option `-P` nicht, wird der in der Umgebungsvariable `PRINTER` definierte Drucker verwendet. Existiert diese Variable nicht, greifen diese Befehle auf den Drucker `lp` zurück.

Im Folgenden steht der Begriff *Standarddrucker* daher für den über die Umgebungsvariable `PRINTER` definierten Drucker, oder, falls diese Variable nicht existiert, für den Drucker `lp`.

10.5.1. Druckaufträge erstellen

Um eine Datei zu drucken, geben Sie folgenden Befehl ein:

```
% lpr filename ...
```

Dadurch wird jede angegebene Datei an den Standarddrucker geschickt. Wenn Sie keine Datei angeben, liest lpr(1) die zu druckenden Daten von der Standardeingabe. Um beispielsweise einige wichtige Systemdateien zu drucken, geben Sie folgenden Befehl ein:

```
% lpr /etc/host.conf /etc/hosts.equiv
```

Um einen bestimmten Drucker auszuwählen, verwenden Sie:

```
% lpr -P printer-name filename ...
```

Das folgende Beispiel gibt eine ausführliche Liste aller im Arbeitsverzeichnis enthaltenen Dateien auf den Drucker `rattan` aus:

```
% ls -l | lpr -P rattan
```

Da keine Dateien an lpr(1) übergeben werden, liest `lpr` die zu druckenden Daten von der Standardeingabe, in unserem Fall also die Ausgabe des Befehls `ls -l`.

lpr(1) akzeptiert auch verschiedene Optionen zur Formatierung und Konvertierung von Dateien, zur Erzeugung von multiplen Ausdrucken und so weiter. Lesen Sie dazu den Abschnitt Druckoptionen.

10.5.2. Druckaufträge verwalten

Wenn Sie lpr(1) verwenden, werden alle zu druckenden Daten in ein Paket, den sogenannten „Druckauftrag“, gepackt und an LPD geschickt. Jeder Drucker verfügt über eine Druckerwarteschlange, in der Ihre Druckaufträge gemeinsam mit denen anderer Benutzer verbleiben, bis sie gedruckt werden können. Zuerst eintreffende Druckaufträge werden dabei auch zuerst gedruckt.

Um die Druckerwarteschlange des Standarddruckers anzuzeigen, verwenden Sie lpq(1). Wollen Sie einen anderen Drucker abfragen, müssen Sie die Option `-P` verwenden. Der Befehl

```
% lpq -P bamboo
```

zeigt so die Druckerwarteschlange des Druckers `bamboo` an. Dieser Befehl liefert eine Ausgabe ähnlich der folgenden:

```
bamboo is ready and printing
Rank   Owner    Job  Files                              Total Size
active kelly    9    /etc/host.conf, /etc/hosts.equiv   88 bytes
2nd    kelly    10   (standard input)                   1635 bytes
3rd    mary     11   ...                                78519 bytes
```

Derzeit enthält die Warteschlange von `bamboo` drei Druckaufträge. Dem ersten Auftrag, der vom Benutzer kelly erstellt wurde, wurde die „Auftragsnummer (job number)“ 9 zugewiesen. Analog erhält jeder Druckerauftrag eine eindeutige Nummer zugewiesen. Diese Nummern sind nur dann von Bedeutung, wenn Sie einen Druckauftrag stornieren wollen. Der Abschnitt Druckaufträge stornieren beschreibt, wie Sie dazu vorgehen.

Der Auftrag mit der Nummer 9 besteht aus zwei Dateien, mehrere an lpr(1) übergebene Dateien werden also als Teil eines (gemeinsamen) Druckauftrags betrachtet. Dieser Druckauftrag ist derzeit aktiv (beachten Sie den Status `active` in der Spalte „Rank“), wird also gerade gedruckt. Der zweite Auftrag besteht aus Daten, die von der Standardeingabe an lpr(1) übergeben wurden. Der dritte Auftrag wurde vom Benutzer `mary` erstellt. Er ist sehr viel größer als die anderen Aufträge. Da der Pfad der zu druckenden Datei aufgrund seiner Länge nicht in der Spalte „Files“ Platz hat, werden von lpq(1) nur drei Punkte angezeigt.

Die erste Zeile der Ausgabe von lpq(1) ist ebenfalls sehr nützlich: Sie beschreibt den momentanen Druckerstatus (oder zumindest, was LPD denkt, dass der Drucker gerade macht).

lpq(1) unterstützt auch die Option `-l` zur Erstellung einer ausführlicheren Ausgabe. Die Eingabe von `lpq -l` erzeugt für unser obiges Beispiel die folgende Ausgabe:

```
waiting for bamboo to become ready (offline ?)
kelly: 1st     [job 009rose]
       /etc/host.conf                  73 bytes
       /etc/hosts.equiv                15 bytes
```

```
kelly: 2nd     [job 010rose]
       (standard input)                 1635 bytes

mary: 3rd                               [job 011rose]
      /home/orchid/mary/research/venus/alpha-regio/mapping 78519 bytes
```

10.5.3. Druckaufträge stornieren

Mit lprm(1) können Sie einen Druckauftrag stornieren. Häufig ist lprm(1) auch noch in der Lage, einen bereits aktiven Auftrag abzubrechen, allerdings wird dabei in der Regel trotzdem ein Teil des Auftrages oder der gesamte Auftrag gedruckt.

Um einen Druckauftrag auf dem Standarddrucker zu stornieren, müssen Sie zuerst die Auftragsnummer über lpq(1) ermitteln. Danach geben Sie Folgendes ein:

```
% lprm Job-Nummer
```

Um einen Druckauftrag eines anderen Druckers zu stornieren, benötigen Sie wiederum die Option `-P`. Der folgende Befehl entfernt den Druckauftrag mit der Nummer 10 aus der Warteschlange des Druckers `bamboo`:

```
% lprm -P bamboo 10
```

lprm(1) unterstützt verschiedene Kurzbefehle:

lprm -
: Entfernt alle Druckaufträge (des Standarddruckers), die von Ihnen erstellt wurden.

lprm *`user`*
: Entfernt alle Druckaufträge (des Standarddruckers), die vom Benutzer *`user`* erstellt wurden. Der Superuser kann im Gegensatz zu einem normalen Benutzer auch Aufträge anderer Benutzer entfernen.

lprm
: Wenn Sie weder eine Auftragsnummer, einen Benutzernamen, noch die Option - angeben, entfernt lprm(1) den aktiven Druckauftrag auf dem Standarddrucker, falls dieser Auftrag von Ihnen erstellt wurde. Der Superuser kann hingegen jeden aktiven Druckauftrag abbrechen.

Verwenden Sie zusätzlich die Option `-P` zu den eben beschriebenen Kurzbefehlen, wenn Sie diese auf einen anderen Drucker als den Standarddrucker anwenden wollen. So entfernt der folgende Befehl beispielsweise alle Druckaufträge des aktuellen Benutzers aus der Druckerwarteschlange des Druckers `rattan`:

```
% lprm -P rattan -
```

Anmerkung

Wenn Sie in einer Netzwerkumgebung arbeiten, erlaubt es lprm(1) Ihnen nur, Druckaufträge auf dem Rechner zu stornieren, auf dem sie erstellt wurden. Dies gilt selbst dann, wenn der gleiche Drucker auch auf anderen Rechnern des Netzwerks verfügbar ist. Die folgende Befehlsfolge veranschaulicht diesen Umstand:

```
% lpr -P rattan myfile
% rlogin orchid
% lpq -P rattan
Rank   Owner   Job  Files                          Total Size
active seeyan   12 ...                           49123 bytes
2nd    kelly      13   myfile                          12 bytes
% lprm -P rattan 13
rose: Permission denied
% logout
% lprm -P rattan 13
dfA013rose dequeued
cfA013rose dequeued
```

10.5.4. Abseits von normalem Text: Druckoptionen

lpr(1) unterstützt verschiedene Optionen zur Formatierung von Text, zur Konvertierung von Grafik- und anderen Dateiformaten, zur Erzeugung von multiplen Kopien, zur Verwaltung von Druckaufträgen und andere mehr. Dieser Abschnitt beschreibt einige dieser Optionen.

10.5.4.1. Formatierungs- und Konvertierungsoptionen

Die folgenden lpr(1)-Optionen kontrollieren die Formatierung von in einem Druckauftrag enthaltenen Dateien. Verwenden Sie diese Optionen, wenn Ihr Druckauftrag keinen normalen Text enthält, oder wenn Sie normalen Text mit pr(1) formatieren wollen.

Der folgende Befehl druckt so beispielsweise eine DVI-Datei (des TeX-Satzsystems) namens `fish-report.dvi` auf dem Drucker `bamboo`:

```
% lpr -P bamboo -d fish-report.dvi
```

Diese Optionen gelten für jede Datei des Druckauftrags, daher ist es nicht möglich beispielsweise DVI- und ditroff-Dateien über den gleichen Druckauftrag zu drucken. Sie müssen diese Dateien vielmehr über getrennte Druckaufträge drucken, wobei Sie jeweils geeignete Konvertierungsoptionen verwenden.

Anmerkung

Alle Optionen mit Ausnahme von `-p` und `-T` setzen einen installierten und für den jeweiligen Drucker konfigurierten Konvertierungsfilter voraus. So benötigt die Option `-d` den DVI-Konvertierungsfilter. Diese Filter werden im Abschnitt Konvertierungsfilter ausführlich beschrieben.

`-c`
: Druckt cifplot-Dateien.

`-d`
: Druckt DVI-Dateien.

`-f`
: Druckt FORTRAN-Textdateien.

`-g`
: Druckt Plot-Daten.

`-i` *`anzahl`*
: Rückt die Ausgabe um *`anzahl`* Spalten ein, lassen Sie *`anzahl`* weg, wird der Text um 8 Spalten eingerückt. Beachten Sie aber, dass diese Option nicht mit allen Konvertierungsfiltern funktioniert.

Anmerkung

Zwischen der Option `-i` und der der Zahl darf dabei kein Leerzeichen stehen.

`-l`
: Druckt Text inklusive vorhandener Steuerzeichen.

`-n`
: Druckt ditroff-Dateien (geräteunabhängiges troff).

-p
: Formatiert normalen Text mit pr(1), bevor der Ausdruck erfolgt.

`-T titel`
: Verwende *`titel`* auf dem pr(1)-Deckblatt anstelle des Dateinamens. Diese Option ist nur wirksam, wenn sie gemeinsam mit der Option `-p` verwendet.

`-t`
: Druckt troff-Daten.

`-v`
: Druckt Rasterdaten.

Dazu ein Beispiel. Der folgende Befehl druckt eine formatierte Version der Manualpage zu ls(1) auf den Standarddrucker:

```
% zcat /usr/share/man/man1/ls.1.gz | troff -t -man | lpr -t
```

zcat(1) dekomprimiert den Quellcode der Manualpage ls(1) und reicht ihn an troff(1) weiter, das ihn formatiert und daraus GNU troff-Daten erzeugt. Diese werden wiederum an lpr(1) weitergereicht, das den Druckauftrag schließlich an LPD übergibt. Da die Option `-t` von lpr(1) verwendet wurde, konvertiert das Drucksystem die GNU troff-Daten zuvor in ein Format, das der Standarddrucker verstehen und ausgeben kann.

10.5.4.2. Druckaufträge verwalten

Die folgenden Optionen von lpr(1) weisen LPD an, den Druckauftrag auf verschiedene Art und Weise zu behandeln:

`-# anzahl`
: Erzeugt *`anzahl`* Ausdrucke jeder im Druckauftrag enthaltenen Datei anstelle eines einzigen Exemplars. Diese Option kann von einem Administrator deaktiviert werden, um die Beanspruchung des Druckers zu verringern. Lesen Sie den Abschnitt Den Ausdruck von mehreren Kopien verhindern, wenn Sie diese Funktion benötigen.

  Das folgende Beispiel druckt drei Kopien der Datei `parser.c`, gefolgt von drei Kopien von `parser.h` auf den Standarddrucker:

  ```
  % lpr -#3 parser.c parser.h
  ```

-m
: Verschickt eine E-Mail, nachdem der Druckauftrag beendet wurde. Verwenden Sie diese Option, sendet LPD Ihnen eine E-Mail, wenn es die Bearbeitung Ihres Druckauftrages abgeschlossen hat. Diese Nachricht enthält Informationen darüber, ob Ihr Auftrag erfolgreich erledigt wurde oder ob ein Fehler auftrat. Ist dies der Fall, wird meist noch angegeben, welcher Fehler auftrat.

-s
: Kopiert die Dateien nicht in das Spooling-Verzeichnis, sondern verlinkt stattdessen symbolisch auf diese Dateien.

  Wenn Sie einen umfangreichen Druckauftrag erstellen, werden Sie diese Option wahrscheinlich verwenden wollen. Einerseits sparen Sie dadurch Speicherplatz im Spooling-Verzeichnis (im schlimmsten Fall könnte Ihr Druckauftrag ansonsten das Dateisystem des Spooling-Verzeichnis zum Überlaufen bringen), andererseits sparen Sie dadurch auch Zeit, weil LPD die in Ihrem Druckauftrag enthaltenen Dateien nicht in das Spooling-Verzeichnis kopieren muss.

  Da LPD in diesem Fall die Originaldateien verwendet, muss sichergestellt sein, dass diese nicht verändert werden, bevor der Ausdruck abgeschlossen ist.

Anmerkung

Wenn Sie auf einen entfernten Drucker drucken, muss LPD die Dateien dennoch vom lokalen auf den entfernten Rechner kopieren. In diesem Fall spart die Option `-s` Speicherplatz lediglich im lokalen Spooling-Verzeichnis, nicht aber im entfernten. Dennoch ist diese Option auch in diesem Fall nützlich.

-r
: Löscht die im Druckauftrag enthaltenen Dateien, nachdem sie in das Spooling-Verzeichnis kopiert oder unter Verwendung der Option `-s` gedruckt werden. Verwenden Sie diese Option daher nur mit äußerster Vorsicht!

10.5.4.3. Deckblatt-Optionen

Die folgenden lpr(1)-Optionen passen den Text an, der auf einem Deckblatt eines Druckauftrages ausgegeben wird. Wird die Ausgabe von Deckblättern auf dem Zieldrucker unterdrückt, bleiben diese Optionen wirkungslos. Lesen Sie den Abschnitt Deckblätter, wenn Sie diese Funktion benötigen.

-C *`text`*
: Ersetzt den Rechnernamen auf dem Deckblatt durch *`text`*. Der Rechnername ist dabei in der Regel der Name des Rechners, auf dem der Druckauftrag erstellt wurde.

-J *`text`*
: Ersetzt den Namen des Druckauftrages auf dem Deckblatt durch *`text`*. Der Name des Druckauftrages entspricht in der Regel dem Namen der ersten Datei des Druckauftrages oder `stdin`, wenn Sie die Standardeingabe an den Drucker weiterleiten.

-h
: Verhindert den Ausdruck von Deckblättern.

Anmerkung

Ob diese Option funktioniert, hängt von der Art und Weise ab, wie Deckblätter auf Ihrem System erzeugt werden. Lesen Sie den Abschnitt Deckblätter für weitere Informationen.

10.5.5. Drucker verwalten

Als Administrator Ihres Systems ist es Ihre Aufgabe, Drucker zu installieren, zu konfigurieren und zu testen. Um mit Ihrem Drucker zu kommunizieren, können Sie lpc(8) verwenden. Dadurch sind Sie in der Lage,

- Ihre Drucker zu starten und zu beenden.
- Die Warteschlangen Ihrer Drucker zu aktivieren und zu deaktivieren.
- Die Reihenfolge der Druckaufträge zu ändern.

Am Anfang dieses Abschnitts steht die Erklärung einiger Begriffe. Wenn ein Drucker *beendet* ist, wird der Inhalt seiner Warteschlange nicht gedruckt. Druckaufträge können zwar weiterhin erstellt werden, diese verbleiben aber solange in der Warteschlange, bis der Drucker wieder *gestartet* oder die Warteschlange gelöscht wird.

Ist eine Warteschlange *deaktiviert*, kann (mit Ausnahme von `root`) kein Benutzer mehr einen Druckauftrag erteilen. Ist die Warteschlange hingegen *aktiviert*, können Druckaufträge erteilt werden. Ist ein Drucker zwar *gestartet*, die Warteschlange hingegen *deaktiviert*, werden dennoch alle noch in der Warteschlange vorhandenen Druckaufträge gedruckt.

Im Allgemeinen benötigen Sie `root`-Rechte, um lpc(8) einsetzen zu können. Als normaler Benutzer erlaubt es Ihnen lpc(8) lediglich, den Druckstatus abzufragen und einen hängenden Drucker neu zu starten.

Es folgt nun eine Zusammenfassung der Befehle von lpc(8). Die meisten dieser Befehle benötigen das Argument *`printer-name`*, mit dem Sie angeben, auf welchen Drucker der Befehl angewendet werden soll. Wenn Sie für *`printer-name`* `all` angeben, wird der Befehl auf alle in `/etc/printcap` definierten Drucker angewendet.

`abort printer-name`
: Bricht den aktuellen Druckauftrag ab und beendet den Drucker. Solange die Warteschlange aktiviert ist, können allerdings weiterhin Druckaufträge erteilt werden.

`clean printer-name`
: Entfernt veraltete Dateien aus dem Spooling-Verzeichnis des Druckers, da diese manchmal nicht vollständig von LPD entfernt werden können. Dies ist insbesondere dann der Fall, wenn während der Bearbeitung des Druckauftrages Fehler auftraten. Dieser Befehl sucht dabei nach Dateien, die nicht in das Spooling-Verzeichnis gehören und entfernt diese.

`disable printer-name`
: Deaktiviert die Annahme neuer Druckaufträge. Solange der Drucker nicht beendet wird, werden weiterhin alle in der Warteschlange enthaltenen Auftrage bearbeitet und gedruckt. `root` kann jederzeit Druckaufträge erstellen, selbst dann, wenn die Druckerwarteschlange deaktiviert ist.

  Dieser Befehl ist besonders nützlich, wenn Sie einen neuen Drucker testen müssen oder einen neuen Filter installiert haben. Dazu deaktivieren Sie die Warteschlange des Druckers und erstellen Ihre Druckaufträge als `root`. Andere Benutzer können erst dann einen Druckauftrag erstellen, wenn Sie Ihre Tests abgeschlossen haben und die Druckerwarteschlange mit `enable` wieder reaktivieren.

`down printer-name nachricht`
: Beendet einen Drucker. Äquivalent zu `disable`, gefolgt von `stop`. Die von Ihnen definierte *`nachricht`* wird als Druckerstatus angezeigt, wenn ein Benutzer die Warteschlange des Druckers mit lpq(1) oder mit `lpc status` abfragt.

`enable printer-name`
: Aktiviert die Warteschlange eines Druckers. Erteilte Druckaufträge können zwar erteilt werden, diese werden aber nur dann gedruckt, wenn der Drucker auch gestartet ist.

`help command-name`
: Ausgaben von hilfreichen Informationen zu *`command-name`*. Wird kein *`command-name`* angegeben, wird die Liste der verfügbaren Befehle ausgegeben.

`restart printer-name`
: Startet den Drucker. Normale Benutzer können diesen Befehl verwenden, um einen hängenden LPD zu reaktivieren, sie sind allerdings nicht berechtigt, einen Drucker zu starten, der mit `stop` oder `down` beendet wurde. Dieser Befehl ist äquivalent zu `abort`, gefolgt von `start`.

`start printer-name`
: Startet den Drucker, um die in der Warteschlange enthaltenen Aufträge zu drucken.

`stop printer-name`
: Beendet den Drucker. Der Drucker beendet den aktiven Druckauftrag noch, danach wird kein weiterer in der Warteschlange enthaltener Auftrag gedruckt. Obwohl der Drucker beendet wurde, können weiterhin Druckaufträge erteilt werden, solange die Warteschlange nicht deaktiviert wurde.

`topq printer-name job-or-username`
: Sortiert die Druckerwarteschlange des Druckers *`printer-name`* um, wobei der Auftrag mit der angegebenen *`Auftragsnummer`*, oder Druckaufträge, die von *`username`* erstellt wurden, an den Beginn der Warteschlange gesetzt werden. Für diesen Befehl kann die Option `all` nicht als *`printer-name`* verwendet werden.

`up printer-name`
: Startet einen Drucker. Das Gegenstück zu `down`. Äquivalent zu `start`, gefolgt von `enable`.

lpc(8) akzeptiert diese Befehle direkt auf der Kommandozeile. Geben Sie keinen Befehl ein, wird lpc(8) im interaktiven Modus gestartet. In diesem Modus können Sie solange Befehle eingeben, bis Sie `exit` oder `quit` eingeben.

10.6. Alternativen zum LPD-Drucksystem

Wenn Sie dieses Kapitel bis hierher gelesen haben, wissen Sie so gut wie alles über LPD, das Standarddrucksystem von FreeBSD. Wahrscheinlich sind Ihnen bereits einige Unzulänglichkeiten dieses Systems aufgefallen, und Sie fragen sich nun, welche anderen Drucksysteme es für FreeBSD gibt.

LPRng

: LPRng steht für „LPR: the Next Generation“. Dabei handelt es sich um eine von Grund auf neu geschriebene Version von PLP. LPRng wurde von Patrick Powell und Justin Mason, dem Hauptmaintainer von PLP, entwickelt. Die offizielle Webseite von LPRng ist unter `http://www.lprng.org/` zu finden.

CUPS
: CUPS, das *Common UNIX Printing System*, stellt eine portable Abstraktionsschicht dar, die das Drucken auf allen UNIX®-artigen Betriebsystemen ermöglicht. CUPS wurde von Easy Software entwickelt, um UNIX®-Herstellern und -Benutzern eine einheitliche Standardlösung für den Druck von Dokumenten zu bieten.

  CUPS verwendet das Internet Printing Protocol (IPP), um Druckaufträge und -warteschlangen zu verwalten. Zusätzlich werden die Protokolle *Line Printer Daemon* (LPD), *Server Message Block* (SMB), und *AppSocket/JetDirect*), unterstützt, wenn auch nur mit eingeschränkter Funktionalität. Ausserdem ermöglicht CUPS das Auffinden von Netzwerkdruckern sowie die Verwendung auf *PostScript Printer Description* (PPD) basierender Druckoptionen.

  Die offizielle Webseite von CUPS ist `http://www.cups.org/` .

HPLIP
: HPLIP, das HP Linux® Imaging and Printing System, ist eine von HP entwickelte Sammlung von Programmen, die Unterstützung für das drucken, scannen und faxen bei HP-Geräten bieten. Diese Programm-Sammlung verwendet CUPS als Grundlage für einige seiner Druck-Eigenschaften.

  Die Hauptseite für HPLIP ist `http://hplipopensource.com/hplip-web/index.html` .

10.7. Problembehandlung

Wenn Sie eine einfache Testseite mit lptest(1) gedruckt haben, könnte eines der folgenden Probleme aufgetreten sein:

Der Druck hat erst mit einer gewissen Verzögerung geklappt oder das bedruckte Blatt verblieb im Drucker, so als wäre der Druckvorgang noch nicht abgeschlossen.
: Die Testseite wurde zwar gedruckt, danach tat sich allerdings nichts mehr. Vielleicht mussten Sie sogar eine Taste Ihres Druckers, etwa PRINT REMAINING oder FORM FEED drücken, damit der Druckvorgang fortgesetzt wurde.

  Wenn das der Fall ist, hat der Drucker vermutlich vor dem eigentlichen Drucken gewartet, ob noch weitere Daten für Ihren Druckauftrag gesendet werden. Um dieses Problem zu beheben, können Sie den Textfilter anweisen, ein *Form Feed* -Zeichen (oder ein anderes entsprechendes Zeichen) an den Drucker zu senden. Dies reicht für gewöhnlich aus, um den Drucker zum Druck des noch im internen Puffer verbliebenen Textes zu bewegen. Dadurch kann auch sichergestellt werden, dass jeder neue Druckauftrag auf einer neuen Seite beginnt.

Der folgende Ersatz für das Shell-Skript `/usr/local/libexec/if-simple` gibt ein „Form Feed“ aus, nachdem der Auftrag an den Drucker geschickt wurde:

```
#!/bin/sh
#
# if-simple - Einfacher Eingabefilter für lpd
# Installiert unter /usr/local/libexec/if-simple
#
# Kopiert stdin einfach nach stdout. Ignoriert alle Filter-Argumente.
# Schreibt ein Form-Feed-Zeichen (\f) nach dem Ende des Druckauftrages.

/bin/cat && printf "\f" && exit 0
exit 2
```

Der Drucker erzeugte einen „Treppeneffekt“ (*staircase effect*).
Sie haben einen Ausdruck ähnlich dem folgenden erhalten:

```
!"#$%&'()*+,-./01234
               "#$%&'()*+,-./012345
                            #$%&'()*+,-./0123456
```

Sie sind zu einem weiteren Opfer des *Treppeneffekts* geworden. Verursacht wird dieser Effekt durch unterschiedliche Ansichten darüber, welche Zeichen den Beginn einer neuen Zeile anzeigen sollen. UNIX®-ähnliche Betriebssysteme verwenden dafür ein einzelnes Zeichen: ASCII-Code 10, auch als *Line Feed* (LF) bekannt. MS-DOS®, OS/2® und andere Betriebssysteme verwenden stattdessen ein Zeichenpaar: ASCII-Code 10 *und* ASCII-Code 13, *Carriage Return* (CR). Viele Drucker verwenden in der Voreinstellung die Konvention von MS-DOS®, um Zeilenumbrüche darzustellen.

Wenn Sie unter FreeBSD drucken, wird nur das Zeichen *Line Feed* verwendet. Der Drucker erkennt dieses Zeichen und erweitert den Druckbereich um eine Zeile, verbleibt zum Druck des nächsten Zeichens aber in derselben horizontalen Position. Das ist der Grund für die Verwendung des *Carriage Return*: Es setzt die Position für das folgende Zeichen auf den linken Rand der Seite.

FreeBSD erwartet von einem Drucker das folgende Verhalten:

Drucker empfängt CR	Drucker druckt CR
Drucker empfängt LF	Drucker druckt CR + LF

Es gibt mehrere Möglichkeiten, dieses Verhalten zu erreichen:

- Verändern Sie die Konfiguration Ihres Druckers, um die Interpretation dieser Zeichen zu verändern. Lesen Sie Ihr Druckerhandbuch, wenn Sie nicht wissen, was Sie dazu tun müssen.

Anmerkung

Wenn Sie auf Ihrem Rechner neben FreeBSD noch andere Betriebssysteme verwenden, müssen Sie Ihren Drucker möglicherweise anschließend *erneut konfigurieren*, damit die Zeichen CR und LF unter diesen Systemen korrekt interpretiert werden. Ist dies bei Ihnen der Fall, werden Sie wohl eine der folgenden Lösungen bevorzugen.

- Lassen Sie LF durch den Treiber der seriellen Schnittstelle automatisch in CR+LF konvertieren. Selbstverständlich funktioniert dies nur mit Druckern, die an einer seriellen Schnittstelle angeschlossen sind. Um diese Möglichkeit zu nutzen, müssen Sie die `ms#`-Fähigkeit verwenden und in `/etc/printcap` den `onlcr`-Modus für den Drucker aktivieren.

- Senden Sie eine *Escape-Sequenz* an den Drucker, damit das Zeichen LF zeitweilig anders behandelt wird. Suchen Sie im Handbuch Ihres Druckers nach den von Ihrem Drucker unterstützten Escape-Sequenzen.

Wenn Sie eine entsprechenden Escape-Sequenz finden, müssen Sie den Textfilter so anpassen, dass zuerst die Escape-Sequenz und anschließend der Druckauftrag gesendet wird.

Es folgt nun ein Bespieltextfilter für einen Drucker, der die Hewlett Packard PCL Escape-Sequenzen versteht. Dieser Filter veranlasst den Drucker, LF-Zeichen als Folgen von LF+CR aufzufassen. Anschließend wird der Druckauftrag gesendet. Als Abschluss wird ein *Form Feed* gesendet, um die letzte Seite des Druckauftrags auszuwerfen. Dieses Beispiel sollte mit nahezu allen Druckern von Hewlett Packard funktionieren.

```
#!/bin/sh
#
# hpif - Einfacher Text-Eingabefilter für lpd für auf HP-PCL basierende Drucker
# Installiert unter /usr/local/libexec/hpif
#
# Kopiert stdin einfach nach stdout.  Ignoriert alle Filterargumente.
# Weist den Drucker an LF als CR+LF zu interpretieren.
# Wirft die Seite nach dem Drucken aus.

printf "\033&k2G" && cat && printf "\033&l0H" && exit 0
exit 2
```

Das nächste Beispiel aus `/etc/printcap` beschreibt den Rechner `orchid`, an dessen Parallelport ein Drucker angeschlossen ist. Es handelt sich dabei um einen Hewlett Packard LaserJet 3Si, der den Namen `teak` verwendet. Als Textfilter wird das Skript aus dem letzten Beispiel verwendet:

```
#
#  /etc/printcap für den Rechner orchid
#
teak|hp|laserjet|Hewlett Packard LaserJet 3Si:\
        :lp=/dev/lpt0 :sh:sd=/var/spool/lpd/teak :mx#0:\
        :if=/usr/local/libexec/hpif  :
```

Alle Zeilen wurden in die gleiche Zeile gedruckt.
: Der Drucker hat niemals eine neue Zeile begonnen. Alle Zeilen des Textes wurden in eine einzige Zeile gedruckt.

  Dieses Problem ist das „Gegenteil“ des oben beschriebenen Treppeneffekts und kommt wesentlich seltener vor. Die von FreeBSD zum Abschluss einer Zeile benutzten LF-Zeichen werden als CR-Zeichen interpretiert. Dadurch wird die Druckposition zwar auf den linken Rand der Seite, aber nicht um eine Zeile nach unten gesetzt.

  Konfigurieren Sie Ihren Drucker, um die folgende Interpretation der Zeichen LF und CR zu erzwingen:

Drucker empfängt	Drucker druckt
CR	CR
LF	CR + LF

Manche Zeichen wurden nicht gedruckt.
: Der Drucker hat in jeder Zeile einige Zeichen nicht gedruckt. Vielleicht ist das Problem auch während des Druckens schlimmer geworden, und der Drucker hat immer mehr Zeichen nicht gedruckt.

  Dieses Problem entsteht, weil der Drucker mit der Geschwindigkeit, mit der die Daten über die serielle Schnittstelle (an einer parallelen Schnittstelle sollte das Problem nicht auftreten) eintreffen, nicht mithalten kann. Es gibt zwei Möglichkeiten, dieses Problem zu lösen:

  - Wenn der Drucker die Flusskontrolle mit XON/XOFF unterstützt, können Sie in der `ms#`-Fähigkeit den `ixon`-Modus aktivieren.

  - Unterstützt der Drucker die Anfrage zum Senden/Löschen des Sende-Hardware-Handshakes (allgemein bekannt als `RTS/CTS`, dann sollten Sie den `crtscts`-Modus in der `ms#`-Fähigkeit aktivieren. Stellen Sie aber sicher, dass das verwendete Druckerkabel auch für die Hardware-Flusskontrolle geeignet ist.

Es wurden nur wirre Zeichen gedruckt.
: Anstelle des gewünschten Textes wurden nur zufällige Zeichen gedruckt.

  Dieses Problem wird ebenfalls durch falsche Konfigurationsparameter im Zusammenhang mit einem seriellen Drucker verursacht. Kontrollieren Sie die bps-Rate in der `br`-Fähigkeit und die Paritätseinstellung (*Parity*) in der `ms#`-Fähigkeit. Überprüfen Sie außerdem, ob der Drucker auch tatsächlich die gleichen Einstellungen verwendet, die in `/etc/printcap` definiert wurden.

Der Drucker hat überhaupt nicht reagiert.
: Wenn gar nichts passiert ist, dann liegt das vermutlich an FreeBSD und nicht am Drucker. Aktivieren Sie die Protokollierung (`lf`-Fähigkeit) für den entsprechenden Drucker in der Datei `/etc/printcap`. Es folgt nun ein Beispieleintrag für den Drucker `rattan`, bei dem die `lf`-Fähigkeit aktiviert wurde.

```
rattan|line|diablo|lp|Diablo 630 Line Printer:\
        :sh:sd=/var/spool/lpd/rattan :\
        :lp=/dev/lpt0 :\
        :if=/usr/local/libexec/if-simple  :\
        :lf=/var/log/rattan.log
```

  Versuchen Sie jetzt noch einmal zu drucken. Überprüfen Sie die Protokolldatei (in unserem Beispiel `/var/log/rattan.log`) auf etwaige Fehlermeldungen. Versuchen Sie aufgrund dieser Meldungen, das Problem zu beheben.

  Wenn Sie keine Protokolldatei festlegen, verwendet LPD in der Voreinstellung `/dev/console` für die Ausgabe der Fehlermeldungen.

Kapitel 11. Linux-Binärkompatibilität

Restrukturiert und teilweise aktualisiert von Jim Mock.
Beigetragen von Brian N. Handy und Rich Murphey.
Übersetzt von Johann Kois.

11.1. Übersicht

FreeBSD bietet Binärkompatibilität zu verschiedenen anderen UNIX® Betriebssystemen, darunter auch Linux. Nun könnten Sie sich fragen, warum FreeBSD in der Lage sein muss, Linux-Binärprogramme auszuführen? Die Antwort auf diese Frage ist sehr einfach. Viele Unternehmen und Entwickler programmieren bzw. entwickeln nur für Linux, da es „das Neueste und Beste“ in der Computerwelt ist. Für uns FreeBSD-Anwender heißt dies, genau diese Unternehmen und Entwickler zu bitten, FreeBSD-Versionen ihrer Programme herauszubringen. Das Problem dabei ist nur, dass die meisten dieser Firmen trotzdem nicht erkennen, wie viele zusätzliche Anwender ihre Produkte benutzen würden, wenn es auch FreeBSD-Versionen gäbe, und daher weiterhin ausschließlich für Linux entwickeln. Was also kann ein FreeBSD-Anwender tun? Genau an diesem Punkt kommt die Linux- Binärkompatibilität ins Spiel.

Um es auf den Punkt zu bringen, genau diese Kompatibilität erlaubt es FreeBSD-Anwendern, etwa 90 % aller Linux-Anwendungen ohne Code-Änderungen zu verwenden. Dies schließt solche Anwendungen wie StarOffice™, Open Office, die Linux-Versionen von Adobe® Acrobat®, RealPlayer®, Oracle®, Doom, Quake und viele andere ein. Es wird sogar berichtet, dass diese Linux-Anwendungen in manchen Fällen unter FreeBSD eine bessere Leistung als unter Linux aufweisen.

Allerdings gibt es nach wie vor einige Linux-spezifische Betriebssystem-Eigenschaften, die unter FreeBSD nicht unterstützt werden. Linux-Anwendungen, die i386™-spezifische Aufrufe (wie die Aktivierung des virtuellen 8086-Modus) verwenden, funktionieren unter FreeBSD leider nicht.

Nach dem Lesen dieses Kapitels werden Sie

- wissen, wie Sie die Linux-Binärkompatibilität installieren bzw. aktivieren.
- Wissen, wie man zusätzliche Linux-Systembibliotheken unter FreeBSD installiert.
- Linux-Anwendungen unter FreeBSD installieren können.
- Wissen, wie die Linux-Binärkompatibilität unter FreeBSD verwirklicht wurde.

Bevor Sie dieses Kapitel lesen, sollten Sie

- wissen, wie man Software Dritter installiert (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

11.2. Installation

Die Linux-Binärkompatibilität ist per Voreinstellung nicht aktiviert. Der einfachste Weg, dies zu tun, ist das `Linux` KLD („Kernel LoaDable object“) zu laden. Dies erreichen Sie durch die Eingabe des folgenden Befehls:

```
# kldload linux
```

Wollen Sie die Linux-Binärkompatibilität dauerhaft aktivieren, sollten Sie die folgende Zeile in `/etc/rc.conf` einfügen:

```
linux_enable="YES"
```

Der kldstat(8)-Befehl kann benutzt werden, um festzustellen, ob KLD geladen wurde:

```
% kldstat
Id Refs Address    Size     Name
 1    2 0xc0100000 16bdb8   kernel
 7    1 0xc24db000 d000     linux.ko
```

Wenn Sie das KLD nicht laden können oder wollen, besteht auch die Möglichkeit, die Linux-Binärkompatibiltät statisch in den Kernel einzubinden. Dazu fügen Sie Ihrer Kernelkonfigurationsdatei den Eintrag `options COMPAT_LINUX` hinzu. Anschließend installieren Sie Ihren neuen Kernel wie in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben.

11.2.1. Linux-Laufzeitbibliotheken installieren

Dies kann auf zwei Arten geschehen, entweder über den linux_base-Port oder durch manuelle Installation der Bibliotheken.

11.2.1.1. Installation unter Verwendung des linux_base-Ports

Dies ist die einfachste Methode, um die Laufzeitbibliotheken zu installieren. Sie funktioniert genauso wie die Installation eines beliebigen anderen Ports aus der Ports-Sammlung. Dazu machen Sie einfach folgendes:

```
# cd /usr/ports/emulators/linux_base-f10
# make install distclean
```

Anmerkung

Bei FreeBSD-Systemen vor FreeBSD 8.0 müssen Sie den Port emulators/linux_base-fc4 anstatt emulators/linux_base-f10 installieren.

Sie sollten nun über eine funktionierende Linux-Binärkompatibilität verfügen. Einige Programme könnten sich zwar über falsche Unterversionsnummern der Systembibliotheken beschweren, dies ist im Allgemeinen aber kein Problem.

Anmerkung

Unter Umständen gibt es mehrere Versionen des Ports emulators/linux_base. Die Ports entsprechen unterschiedlichen Versionen verschiedener Linux-Distributionen Sie sollten den Port installieren, der am besten die Anforderungen der Linux-Anwendung erfüllt.

11.2.1.2. Manuelle Installation der Bibliotheken

Wenn Sie die „Ports“-Sammlung nicht installiert haben, können Sie die Bibliotheken auch manuell installieren. Dazu brauchen Sie die jeweiligen Linux-Systembibliotheken, die das zu installierende Programm verwendet sowie den Laufzeit-Linker. Zusätzlich müssen Sie auf Ihrem FreeBSD-System einen „virtuellen“ Verzeichnisbaum für die Linux-Bibliotheken einrichten. Alle unter FreeBSD gestarteten Linux-Programme suchen zuerst in diesem Verzeichnisbaum nach Systembibliotheken. Wenn also ein Linuxprogramm beispielsweise `/lib/libc.so` lädt, versucht FreeBSD zuerst, `/compat/linux/lib/libc.so` laden. Ist diese Datei nicht vorhanden, wird `/lib/libc.so` geladen. Systembibliotheken sollten daher besser in den „virtuellen“ Verzeichnisbaum `/compat/linux/lib` als in den vom Linux-`ld.so` vorgeschlagenen installiert werden.

Im Allgemeinen müssen Sie nur zu Beginn nach den Systembibliotheken suchen, die von Linuxprogrammen benötigt werden. Nach den ersten Installationen von Linuxprogrammen auf Ihrem FreeBSD-System verfügen Sie über eine Sammlung von Linux-Systembibliotheken, die es Ihnen ermöglichen wird, neue Linuxprogramme ohne Zusatzarbeit zu installieren.

11.2.1.3. Installation zusätzlicher Systembibliotheken

Was passiert, wenn Sie den `linux_base`-Port installieren, und Ihr Programm beschwert sich trotzdem über fehlende Systembibliotheken? Woher wissen Sie, welche Systembibliotheken von Linux-Binärprogrammen benötigt werden, und wo Sie diese finden? Grundsätzlich gibt es dafür zwei Möglichkeiten (um dieser Anleitung zu folgen, müssen Sie unter FreeBSD als Benutzer `root` angemeldet sein):

Wenn Sie Zugriff auf ein Linux-System haben, können Sie dort nachsehen, welche Systembibliotheken eine Anwendung benötigt, und diese auf Ihr FreeBSD-System kopieren. Dazu folgendes Beispiel:

Nehmen wir an, Sie haben FTP verwendet, um die Linux-Binärversion von Doom zu bekommen und haben sie auf Ihrem Linux-System installiert. Nun können Sie überprüfen, welche Systembibliotheken das Programm benötigt, indem Sie `ldd linuxdoom` eingeben. Das Resultat sieht dann so aus:

```
% ldd linuxdoom
libXt.so.3 (DLL Jump 3.1) => /usr/X11/lib/libXt.so.3.1.0
libX11.so.3 (DLL Jump 3.1) => /usr/X11/lib/libX11.so.3.1.0
libc.so.4 (DLL Jump 4.5pl26) => /lib/libc.so.4.6.29
```

Sie müssten nun alle Dateien aus der letzten Spalte kopieren und sie unter `/compat/linux` speichern, wobei die Namen der ersten Spalte als symbolische Links auf diese Dateien zeigen. Damit haben Sie schließlich folgende Dateien auf Ihrem FreeBSD-System:

```
/compat/linux/usr/X11/lib/libXt.so.3.1.0
/compat/linux/usr/X11/lib/libXt.so.3 -> libXt.so.3.1.0
/compat/linux/usr/X11/lib/libX11.so.3.1.0
/compat/linux/usr/X11/lib/libX11.so.3 -> libX11.so.3.1.0
/compat/linux/lib/libc.so.4.6.29
/compat/linux/lib/libc.so.4 -> libc.so.4.6.29
```

Anmerkung

Beachten Sie, dass wenn Sie bereits eine Linux-Systembibliothek einer zur ersten Spalte passenden Hauptversionsnummer (laut `ldd`-Ausgabe) besitzen, Sie die Datei aus der zweiten Spalte nicht mehr kopieren müssen, da die bereits vorhandene Version funktionieren sollte. Hat die Systembibliothek jedoch eine neuere Versionsnummer, sollten Sie sie dennoch kopieren. Sie können die alte Version löschen, solange Sie einen symbolischen Link auf die neue Version anlegen. Wenn Sie also folgende Bibliotheken auf Ihrem System installiert haben:

```
/compat/linux/lib/libc.so.4.6.27
/compat/linux/lib/libc.so.4 -> libc.so.4.6.27
```

und Sie haben eine neue Binärdatei, die laut `ldd` eine neuere Bibliothek benötigt:

```
libc.so.4 (DLL Jump 4.5pl26) -> libc.so.4.6.29
```

Wenn diese sich nur um ein oder zwei Stellen in der Unterversionsnummer unterscheiden, müssen Sie `/lib/libc.so.4.6.29` nicht auf Ihr System kopieren, da das Programm auch mit der etwas älteren Version ohne Probleme funktionieren sollte. Wenn Sie wollen, können Sie `libc.so` aber

dennoch ersetzen (das heißt aktualisieren), was dann zu folgender Ausgabe führt:

```
/compat/linux/lib/libc.so.4.6.29
/compat/linux/lib/libc.so.4 -> libc.so.4.6.29
```

Anmerkung

Der Mechanismus der symbolischen Links wird *nur* für Linux-Binärdateien benötigt. Der FreeBSD-Laufzeitlinker sucht sich die passenden Hauptversionsnummern selbst, das heißt Sie müssen sich nicht darum kümmern.

11.2.2. Linux ELF-Binärdateien installieren

ELF-Binärdateien benötigen manchmal eine zusätzliche „Kennzeichnung“. Wenn Sie versuchen, eine nicht gekennzeichnete ELF-Binärdatei auszuführen, werden Sie eine Fehlermeldung ähnlich der folgenden erhalten:

```
% ./my-linux-elf-binary
ELF binary type not known
Abort
```

Damit der FreeBSD-Kernel eine Linux-ELF-Datei von einer FreeBSD-ELF-Datei unterscheiden kann, gibt es das Werkzeug brandelf(1).

```
% brandelf -t Linux my-linux-elf-binary
```

Die GNU Werkzeuge schreiben nun automatisch die passende Kennzeichnungsinformation in die ELF-Binärdateien, so dass Sie diesen Schritt in Zukunft nur noch selten benötigen werden.

11.2.3. Installieren einer beliebigen RPM-basierten Linuxanwendung

FreeBSD besitzt seine eigene Paketdatenbank und diese wird dazu verwendet, um alle Ports (auch Linux®-Ports) zu verfolgen. Deshalb wird die Linux® RPM-Datenbank nicht benutzt (fehlende Unterstützung).

Falls Sie jedoch eine beliebige RPM-basierte Linux®-Anwendung installieren wollen, erreichen Sie das mittels:

```
# cd /compat/linux
# rpm2cpio -q < /path/to/linux.archive.rpm | cpio -id
```

Benutzen Sie dann brandelf auf die installierten ELF-Binärdateien (nicht die Bibliotheken!). Sie werden keine saubere Deinstallation hinbekommen, aber evtl. helfen ein paar Tests weiter.

11.2.4. Namensauflösung konfigurieren

Wenn DNS nicht funktioniert, oder Sie folgende Fehlermeldung erhalten:

```
resolv+: "bind" is an invalid keyword resolv+:
"hosts" is an invalid keyword
```

müssen sie `/compat/linux/etc/host.conf` wie folgt anlegen:

```
order hosts, bind
multi on
```

Diese Reihenfolge legt fest, dass zuerst `/etc/hosts` und anschließend DNS durchsucht werden. Wenn `/compat/linux/etc/host.conf` nicht vorhanden ist, finden Linux-Anwendungen FreeBSD's `/etc/host.conf` und beschweren sich über die inkompatible FreeBSD-Syntax. Wenn Sie keinen Nameserver (in `/etc/resolv.conf`) konfiguriert haben, sollten Sie den Eintrag `bind` entfernen.

11.3. Mathematica® installieren

Für Mathematica 5.x aktualisiert von Boris Hollas.

Dieses Dokument beschreibt die Installation der Linux-Version von Mathematica® 5.x auf einem FreeBSD-System.

Die Linux-Version von Mathematica® oder Mathematica® für Studenten kann direkt von Wolfram unter `http://www.wolfram.com/` bestellt werden.

11.3.1. Den Mathematica®-Installer starten

Zuerst müssen Sie FreeBSD mitteilen, dass die Linux-Binärversion von Mathematica® die Linux-ABI verwendet. Dies erreichen Sie am einfachsten, indem Sie die Standard-ELF-Kennzeichnung für alle ungekennzeichneten Binärdateien auf Linux festlegen:

```
# sysctl kern.fallback_elf_brand=3
```

Danach wird FreeBSD annehmen, dass alle ungekennzeichneten ELF-Binärdateien die Linux-ABI verwenden und es wäre nun möglich, das Installationsprogramm direkt von der CD-ROM zu starten.

Unter FreeBSD müssen allerdings die Datei `MathInstaller` in ein lokales Verzeichnis Ihrer Festplatte kopieren:

```
# mount /cdrom
# cp /cdrom/Unix/Installers/Linux/MathInstaller /LokalesVerzeichnis/
```

In dieser Datei ersetzen Sie in der ersten Zeile den Wert `/bin/sh` durch `/compat/linux/bin/sh`. Dadurch wird sichergestellt, dass der Installer von der Linux-Version von sh(1) aufgerufen wird. Danach ersetzen Sie durch das im nächsten Abschnitt zu findende Skript oder über einen Texteditor alle Vorkommen von `Linux)` durch `FreeBSD)`. Dadurch ist es dem Mathematica®-Installer möglich, durch den Einsatz von `uname -s` das Betriebssystem zu bestimmen. FreeBSD wird dabei als Linux-artiges Betriebssystem behandelt. Durch den Aufruf von `MathInstaller` kann Mathematica® anschließend installiert werden.

11.3.2. Die Mathematica®-Programmdateien anpassen

Das von Mathematica® während der Installation erzeugte Shell-Skript muss angepasst werden, bevor Sie es einsetzen können. Wenn Sie die Mathematica®-Programmdateien unter `/usr/local/bin` installieren, finden Sie in diesem Verzeichnis die symbolische Links `math`, `mathematica`, `Mathematica`, sowie `MathKernel`. In jeder dieser Dateien müssen Sie jedes Vorkommen von `Linux)` durch `FreeBSD)` ersetzen (entweder über einen Texteditor oder durch das folgende Shellskript):

```
#!/bin/sh
cd /usr/local/bin
for i in math mathematica Mathematica MathKernel
  do sed 's/Linux)/FreeBSD)/g' $i > $i.tmp
  sed 's/\/bin\/sh/\/compat\/linux\/bin\/sh/g' $i.tmp > $i
  rm $i.tmp
  chmod a+x $i
done
```

11.3.3. Ihr Mathematica®-Passwort anfordern

Wenn Sie Mathematica® das erste Mal starten, werden Sie nach einem Passwort gefragt. Haben Sie noch kein Passwort von Wolfram erhalten, müssen Sie zuerst im Installationsverzeichnis `mathinfo` aufrufen, um Ihre „Rech-

ner-ID“ zu bestimmen. Diese Rechner-ID basiert ausschließlich auf der MAC-Adresse Ihrer ersten Netzwerkkarte. Daher ist es nicht möglich, Ihre Mathematica®-Kopie auf verschiedenen Rechnern zu installieren.

Wenn Sie sich bei Wolfram registrieren (durch E-Mail, Telefon oder Fax), teilen Sie Ihre „Rechner-ID“ mit und erhalten dafür ein aus Zahlengruppen bestehendes Passwort.

11.3.4. Das Mathematica®-Frontend über ein Netzwerk ausführen

Mathematica® verwendet einige spezielle Schriftarten, um Zeichen anzuzeigen, die in den Standardzeichensätzen nicht vorhanden sind (z.B. Integrale, Summen, griechische Buchstaben). Das X-Protokoll verlangt allerdings, dass diese Schriftarten *lokal* installiert sind. Das bedeutet, dass Sie diese Schriftarten von der CD-ROM oder von einem Rechner, auf dem Mathematica® installiert ist, auf Ihren Rechner kopieren müssen. Diese Schriftarten befinden sich normalerweise in `/cdrom/Unix/Files/SystemFiles/Fonts` (Mathematica®-CD) oder in `/usr/local/mathematica/SystemFiles/Fonts` (Festplatte). Die aktuellen Schriftarten befinden sich dabei in den Unterverzeichnissen `Type1` und `X`. Um diese Schriftarten zu verwenden, gibt es mehrere Möglichkeiten, die nun beschrieben werden:

Die erste Möglichkeit besteht darin, die Schriftarten in eins der bereits existierenden Schriftartenverzeichnisse unter `/usr/X11R6/lib/X11/fonts` zu kopieren. Dies bedeutet, dass Sie `fonts.dir` editieren müssen, indem Sie die Schriftnamen hinzufügen und die Anzahl der Schriftarten in der ersten Zeile ändern. Alternativ ist es auch möglich, im Verzeichnis, in das Sie die Schriftarten kopiert haben, das Kommando mkfontdir(1) auszuführen.

Die zweite Möglichkeit, besteht darin, die Verzeichnisse nach `/usr/X11R6/lib/X11/fonts` zu kopieren:

```
# cd /usr/X11R6/lib/X11/fonts
# mkdir X
# mkdir MathType1
# cd /cdrom/Unix/Files/SystemFiles/Fonts
# cp X/* /usr/X11R6/lib/X11/fonts/X
# cp Type1/* /usr/X11R6/lib/X11/fonts/MathType1
# cd /usr/X11R6/lib/X11/fonts/X
# mkfontdir
# cd ../MathType1
# mkfontdir
```

Nun fügen Sie die neuen Schriftartenverzeichnisse in Ihren Pfad ein:

```
# xset fp+ /usr/X11R6/lib/X11/fonts/X
# xset fp+ /usr/X11R6/lib/X11/fonts/MathType1
# xset fp rehash
```

Wenn Sie den Xorg-Server verwenden, können Sie die Schriftarten-Verzeichnisse automatisch laden lassen, wenn Sie sie in Ihrer `xorg.conf` angeben.

Wenn Sie *noch kein* `/usr/X11R6/lib/X11/fonts/Type1` -Verzeichnis haben, können Sie das `MathType1`-Verzeichnis im vorherigen Beispiel in `Type1` umbenennen.

11.4. Maple™ installieren

Beigetragen von Aaron Kaplan.
Mit Unterstützung durch Robert Getschmann.

Maple™ ist ein mit Mathematica® vergleichbares kommerzielles Mathematikprogramm. Sie können dieses Programm unter `http://www.maplesoft.com/` kaufen und sich anschließend registrieren, um eine Lizenz zu erhalten. Um dieses Programm unter FreeBSD zu installieren, gehen Sie wie folgt vor:

1. Führen Sie das `INSTALL`-Shell-Skript der Softwaredistribution aus. Wählen Sie die „RedHat“-Option aus, wenn Sie das Installationsprogramm danach fragt. Ein typisches Installationsverzeichnis wäre z.B. `/usr/local/maple`.

2. Wenn Sie dies noch nicht gemacht haben, besorgen Sie sich nun eine Maple™-Lizenz von Maple Waterloo Software (`http://register.maplesoft.com`) und kopieren Sie diese nach `/usr/local/maple/license/license.dat`.

3. Installieren Sie den FLEXlm-Lizenz-Manager, indem Sie das `INSTALL_LIC`-Installations-Shellskript ausführen, das mit Maple™ ausgeliefert wird. Geben Sie Ihren primären Rechnernamen für den Lizenz-Server an.

4. Verändern Sie `/usr/local/maple/bin/maple.system.type` wie folgt:

```
   ----- snip ------------------
*** maple.system.type.orig      Sun Jul  8 16:35:33 2001
--- maple.system.type   Sun Jul  8 16:35:51 2001
***************
*** 72,77 ****
--- 72,78 ----
          # the IBM RS/6000 AIX case
          MAPLE_BIN="bin.IBM_RISC_UNIX"
          -;;
+     "FreeBSD"|\
      "Linux")
          # the Linux/x86 case
        # We have two Linux implementations, one for Red Hat and
   ----- snip end of patch -----
```

   Bitte beachten Sie, dass nach `"FreeBSD"|\` kein anderes Zeichen eingefügt werden darf.

   Dieser Patch weist Maple™ an, FreeBSD als „eine Art von Linux-System“ zu erkennen. Das Shell-Skript `bin/maple` ruft das Shell-Skript `bin/maple.system.type` auf, welches wiederum `uname -a` verwendet, um den Namen des Betriebssystems herauszufinden. Abhängig vom Betriebssystem weiß das System nun, welche Binärdateien verwendet werden sollen.

5. Starten Sie den Lizenz-Server.

   Das folgende, als `/usr/local/etc/rc.d/lmgrd` installierte Shell-Skript ist ein komfortabler Weg, um `lmgrd` zu starten:

```
   ----- snip ------------

#! /bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/X11R6/bin
PATH=${PATH}:/usr/local/maple/bin:/usr/local/maple/FLEXlm/UNIX/LINUX
export PATH

LICENSE_FILE=/usr/local/maple/license/license.dat
LOG=/var/log/lmgrd.log

case "$1" in
start)
 lmgrd -c ${LICENSE_FILE} 2>> ${LOG} 1>&2
 echo -n " lmgrd"
 ;;
stop)
 lmgrd -c ${LICENSE_FILE} -x lmdown 2>> ${LOG} 1>&2
 ;;
*)
 echo "Usage: `basename $0` {start|stop}" 1>&2
 exit 64
 ;;
esac

exit 0
   ----- snip ------------
```

6. Versuchen Sie, Maple™ zu starten:

```
% cd /usr/local/maple/bin
% ./xmaple
```

Nun sollte das Programm laufen und alles funktionieren. Falls ja, vergessen Sie nicht, an Maplesoft zu schreiben und sie wissen zu lassen, dass Sie gerne eine native FreeBSD-Version hätten.

11.4.1. Häufige Fehlerquellen

- Der FLEXlm-Lizenzmanager kann schwierig zu bedienen sein. Zusätzliche Dokumentation zu diesem Thema finden Sie unter `http://www.globetrotter.com/` .

- Es ist bekannt, dass `lmgrd` sehr pingelig ist, wenn es um die Lizenzdatei geht. Gibt es Probleme, führt dies zu einem Speicherauszug (*core dump*). Ein korrekte Lizenzdatei sollte ähnlich der folgenden aussehen:

```
# ====================================================
# License File for UNIX Installations ("Pointer File")
# ====================================================
SERVER chillig ANY
#USE_SERVER
VENDOR maplelmg

FEATURE Maple maplelmg 2000.0831 permanent 1 XXXXXXXXXXX \
         PLATFORMS=i86_r ISSUER="Waterloo Maple Inc." \
         ISSUED=11-may-2000 NOTICE=" Technische Universitat Wien" \
         SN=XXXXXXXXX
```

Anmerkung

Seriennummer und Schlüssel wurden durch mehrere `X` unkenntlich gemacht. `chillig` ist ein Rechnername.

Veränderungen an der Lizenzdatei sind möglich, solange Sie die `FEATURE`-Zeile nicht verändern (diese ist durch den Lizenzschlüssel geschützt).

11.5. MATLAB® installieren

Beigesteuert von Dan Pelleg.

Im Folgenden wird die Installation der Linux-Anwendung MATLAB® Version 6.5 auf FreeBSD beschrieben. Mit Ausnahme der Java Virtual Machine™ (siehe Abschnitt 11.5.3, „Einrichten der Java™-Laufzeitumgebung“) läuft die Anwendung auch ganz gut.

Die Linux-Version von MATLAB® können Sie direkt bei The MathWorks bestellen. Vergewissern Sie sich, dass Sie die Lizenz-Datei oder eine Anleitung zum Erstellen der Lizenz-Datei erhalten haben. Wenn Sie mit MathWorks in Kontakt stehen, weisen Sie bitte auf die fehlende FreeBSD-Version der Software hin.

11.5.1. Das MATLAB®-Installationsskript

Um MATLAB® zu installieren, gehen Sie wie folgt vor:

1. Hängen Sie die Installations-CD ein und wechseln Sie zu `root`, wie im Installations-Skript gefordert. Starten Sie die Installation mit dem folgenden Kommando:

```
# /compat/linux/bin/sh /cdrom/install
```

Tipp

Die Installation erfordert eine graphische Benutzeroberfläche. Wenn Sie die Fehlermeldung erhalten, dass das Display nicht geöffnet werden konnte, führen Sie das folgende Kommando aus:

```
# setenv HOME ~USER
```

Für *USER* setzen Sie den Benutzer ein, von dem aus Sie `root` geworden sind.

2. Beantworten Sie die Frage nach dem MATLAB®-Root-Verzeichnis mit: **`/compat/linux/usr/local/matlab`** .

Tipp

Den langen Pfad werden Sie noch öfter brauchen. Die Tipparbeit können Sie sich mit dem folgenden Befehl erleichtern:

```
# set MATLAB=/compat/linux/usr/local/matlab
```

3. Editieren Sie die Lizenz-Datei entsprechend der Anweisung, die Sie beim Erwerb der Lizenz erhalten haben.

Tipp

Sie können die Datei schon vorher mit Ihrem Lieblingseditor bearbeiten. Kopieren Sie die Lizenz-Datei nach `$MATLAB/license.dat` bevor das Installationsprogramm Sie auffordert, die Datei zu editieren.

4. Schließen Sie die Installation ab.

Die MATLAB®-Installation ist jetzt abgeschlossen. Die folgenden Schritte passen MATLAB® an FreeBSD an.

11.5.2. Den Lizenzmanager starten

1. Erstellen Sie symbolische Links zu den Startskripten des Lizenzmanagers:

```
# ln -s $MATLAB/etc/lmboot /usr/local/etc/lmboot_TMW
# ln -s $MATLAB/etc/lmdown /usr/local/etc/lmdown_TMW
```

2. Erstellen Sie das Startskript `/usr/local/etc/rc.d/flexlm` . Das folgende Beispiel ist eine geänderte Version des mitgelieferten Skripts `$MATLAB/etc/rc.lm.glnx86` . Angepasst wurden die Pfade zu den Dateien und der Start des Lizenzmanagers unter der Linux-Emulation.

```
#!/bin/sh
case "$1" in
  start)
        if [ -f /usr/local/etc/lmboot_TMW -]; then
              /compat/linux/bin/sh /usr/local/etc/lmboot_TMW -u username && echo
 'MATLAB_lmgrd'
```

```
        fi
        -;;
 stop)
 if [ -f /usr/local/etc/lmdown_TMW -]; then
           /compat/linux/bin/sh /usr/local/etc/lmdown_TMW  > /dev/null 2>&1
 fi
        -;;
 *)
 echo "Usage: $0 {start|stop}"
 exit 1
 ;;
esac

exit 0
```

Wichtig

Machen Sie Datei ausführbar:

```
# chmod +x /usr/local/etc/rc.d/flexlm
```

Ersetzen Sie im Skript *username* durch einen existierenden Benutzer Ihres Systems (bitte keinesfalls `root`).

3. Starten Sie den Lizenzmanager:

```
# service flexlm start
```

11.5.3. Einrichten der Java™-Laufzeitumgebung

Erstellen Sie einen symbolischen Link auf eine unter FreeBSD laufende Java™-Laufzeitumgebung (JRE):

```
# cd $MATLAB/sys/java/jre/glnx86/
# unlink jre; ln -s ./jre1.1.8 ./jre
```

11.5.4. Ein MATLAB®-Startskript erstellen

1. Kopieren Sie das folgende Skript nach `/usr/local/bin/matlab` :

```
#!/bin/sh
/compat/linux/bin/sh /compat/linux/usr/local/matlab/bin/matlab "$@"
```

2. Machen Sie das Skript ausführbar:

```
# chmod +x /usr/local/bin/matlab
```

Tipp

Abhängig von der Version des Ports emulators/linux_base kann das Skript auf Fehler laufen. Die Fehler können Sie vermeiden, indem Sie die Datei `/compat/linux/usr/local/matlab/bin/matlab` editieren. Ändern Sie die nachstehende Zeile

```
if [ `expr "$lscmd" : '.*->.*'` -ne 0 -]; then
```

(mit Version 13.0.1 in der Zeile 410) in die folgende um:

```
if test -L $newbase; then
```

11.5.5. Stopp-Skript für MATLAB® erstellen

Das nachstehende Skript beendet MATLAB® ordnungsgemäß.

1. Erstellen Sie die Datei `$MATLAB/toolbox/local/finish.m` mit dem nachstehenden Inhalt:

```
! $MATLAB/bin/finish.sh
```

> **Anmerkung**
>
> Übernehmen Sie die Zeichenkette `$MATLAB` unverändert.

> **Tipp**
>
> Im selben Verzeichnis befinden sich die Dateien `finishsav.m` und `finishdlg.m`. Die Dateien sichern die Einstellungen der Arbeitsfläche bevor MATLAB® beendet wird. Wenn Sie eine der beiden Dateien benutzen, fügen Sie die obige Zeile unmittelbar nach dem `save`-Kommando ein.

2. Erstellen Sie die Datei `$MATLAB/bin/finish.sh` mit nachstehendem Inhalt:

```
#!/compat/linux/bin/sh
(sleep 5; killall -1 matlab_helper) &
exit 0
```

3. Machen Sie die Datei ausführbar:

```
# chmod +x $MATLAB/bin/finish.sh
```

11.5.6. MATLAB® benutzen

Jetzt können Sie MATLAB® mit dem `matlab` starten.

11.6. Oracle® installieren

Beigetragen von Marcel Moolenaar.

11.6.1. Übersicht

Dieses Dokument beschreibt die Installation von Oracle® 8.0.5 und Oracle® 8.0.5.1 Enterprise Edition für Linux auf einem FreeBSD-Rechner.

11.6.2. Installation der Linux-Umgebung

Stellen Sie sicher, dass Sie sowohl emulators/linux_base und devel/linux_devtools aus der Ports-Sammlung installiert haben. Wenn Sie mit diesen Ports Schwierigkeiten haben, müssen Sie vielleicht ältere Versionen der Linux-Umgebung aus der Ports-Sammlung installieren.

Wenn Sie den Intelligent-Agent verwenden wollen, müssen Sie zusätzlich das RedHat Tcl-Paket installieren: `tcl-8.0.3-20.i386.rpm` . Zur Installation von RPM-Paketen wir der Port archivers/rpm benötigt. Ist der Port installiert, lassen sich RPM-Pakete anschließend mit dem nachstehenden Befehl installieren:

```
# rpm -i --ignoreos --root /compat/linux --dbpath /var/lib/rpm package
```

Die Installation der RPM-Pakete sollte ohne Fehlermeldung ablaufen.

11.6.3. Die Oracle®-Umgebung erzeugen

Bevor Sie Oracle® installieren können, müssen Sie eine entsprechende Umgebung erzeugen. Dieses Dokument beschreibt nur, was Sie *im Speziellen* tun müssen, um die Linux-Version von Oracle® unter FreeBSD zu installieren, nicht aber, was bereits in der Installationsanleitung von Oracle® beschrieben wird.

11.6.3.1. Kernel-Tuning

Wie in der Installationsanleitung von Oracle® beschrieben, müssen Sie die maximale Shared-Memory Größe festlegen. Verwenden Sie `SHMMAX` nicht unter FreeBSD. `SHMMAX` wird lediglich aus `SHMMAXPGS` und `PGSIZE` berechnet. Definieren Sie stattdessen `SHMMAXPGS`. Alle anderen Optionen können wie in der Anleitung beschrieben verwendet werden. Zum Beispiel:

```
options SHMMAXPGS=10000
options SHMMNI=100
options SHMSEG=10
options SEMMNS=200
options SEMMNI=70
options SEMMSL=61
```

Passen Sie diese Optionen entsprechend dem von Ihnen gewünschten Einsatzzweck von Oracle® an.

Stellen Sie außerdem sicher, dass Sie folgende Optionen in Ihren Kernel kompilieren:

```
options SYSVSHM #SysV shared memory
options SYSVSEM #SysV semaphores
options SYSVMSG #SysV interprocess communication
```

11.6.3.2. Oracle®-Benutzer anlegen

Legen Sie den Account `oracle` an. Der Account unterschiedet sich von normalen Accounts dadurch, dass er eine Linux-Shell zugeordnet bekommen muss. Fügen Sie `/compat/linux/bin/bash` in die Datei `/etc/shells` ein und setzen Sie die Shell für den `oracle`-Account auf `/compat/linux/bin/bash` .

11.6.3.3. Umgebung

Neben den normalen Oracle®-Variablen, wie z.B. `ORACLE_HOME` und `ORACLE_SID` müssen Sie die folgenden Variablen setzen:

Variable	Wert
`LD_LIBRARY_PATH`	`$ORACLE_HOME/lib`
`CLASSPATH`	`$ORACLE_HOME/jdbc/lib/classes111.zip`
`PATH`	`/compat/linux/bin /compat/linux/sbin /compat/linux/usr/bin /compat/linux/usr/sbin /bin /sbin /usr/bin /usr/sbin /usr/local/bin $ORACLE_HOME/bin`

Es ist empfehlenswert, alle Variablen in der Datei `.profile` zu setzen. Ein komplettes Beispiel sieht folgendermaßen aus:

```
ORACLE_BASE=/oracle; export ORACLE_BASE
ORACLE_HOME=/oracle; export ORACLE_HOME
LD_LIBRARY_PATH=$ORACLE_HOME/lib
```

```
export LD_LIBRARY_PATH
ORACLE_SID=ORCL; export ORACLE_SID
ORACLE_TERM=386x; export ORACLE_TERM
CLASSPATH=$ORACLE_HOME/jdbc/lib/classes111.zip
export CLASSPATH
PATH=/compat/linux/bin:/compat/linux/sbin:/compat/linux/usr/bin
PATH=$PATH:/compat/linux/usr/sbin:/bin:/sbin:/usr/bin:/usr/sbin
PATH=$PATH:/usr/local/bin:$ORACLE_HOME/bin
export PATH
```

11.6.4. Oracle® installieren

Auf Grund einer kleinen Unregelmäßigkeit im Linux-Emulator müssen Sie das Verzeichnis `.oracle` unter `/var/tmp` erzeugen, bevor Sie das Installationsprogramm starten. Das Verzeichnis muss dem Account `oracle` gehören. Sie sollten Oracle® nun ohne Probleme installieren können. Treten dennoch Probleme auf, überprüfen Sie zuerst Ihre Oracle®-Distribution und Ihre Konfiguration. Nachdem Sie Oracle® erfolgreich installiert haben, installieren Sie die Patches wie in den zwei folgenden Abschnitten beschrieben:

Ein häufiges Problem ist, dass der TCP Protokoll-Adapter nicht korrekt installiert wird. Daraus folgt, dass Sie keine TCP-Listener starten können. Dieses Problem kann durch folgende Schritte behoben werden:

```
# cd $ORACLE_HOME/network/lib
# make -f ins_network.mk ntcontab.o
# cd $ORACLE_HOME/lib
# ar r libnetwork.a ntcontab.o
# cd $ORACLE_HOME/network/lib
# make -f ins_network.mk install
```

Vergessen Sie nicht, `root.sh` nochmals auszuführen!

11.6.4.1. root.sh patchen

Während der Oracle®-Installation werden einige Aktionen, die als `root` ausgeführt werden müssen, in ein Shell-Skript mit dem Namen `root.sh` gespeichert. Dieses Skript befindet sich im Verzeichnis `orainst`. Verwenden Sie folgenden Patch für `root.sh`, damit es das richtige `chown` Kommando verwendet, oder lassen Sie das Skript alternativ unter einer Linux-Shell ablaufen:

```
*** orainst/root.sh.orig Tue Oct 6 21:57:33 1998
--- orainst/root.sh Mon Dec 28 15:58:53 1998
***************
*** 31,37 ****
# This is the default value for CHOWN
# It will redefined later in this script for those ports
# which have it conditionally defined in ss_install.h
! CHOWN=/bin/chown
#
# Define variables to be used in this script
--- 31,37 ----
# This is the default value for CHOWN
# It will redefined later in this script for those ports
# which have it conditionally defined in ss_install.h
! CHOWN=/usr/sbin/chown
#
# Define variables to be used in this script
```

Wenn Sie Oracle® nicht von CD-ROM installieren, können Sie Quelldatei für `root.sh` verändern. Sie heißt `rthd.sh` und befindet sich im `orainst` -Verzeichnis des Quellcodebaums.

11.6.4.2. genclntsh patchen

Das Skript `genclntsh` wird verwendet, um eine Shared-Library für Clients zu erzeugen. Diese wird bei der Erzeugung der Demos verwendet. Verwenden Sie folgenden Patch, um die Definition von `PATH` auszukommentieren:

```
*** bin/genclntsh.orig Wed Sep 30 07:37:19 1998
```

```
--- bin/genclntsh Tue Dec 22 15:36:49 1998
***************
*** 32,38 ****
#
# Explicit path to ensure that we're using the correct commands
#PATH=/usr/bin:/usr/ccs/bin export PATH
! PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin export PATH
#
# each product MUST provide a $PRODUCT/admin/shrept.lst
--- 32,38 ----
#
# Explicit path to ensure that we're using the correct commands
#PATH=/usr/bin:/usr/ccs/bin export PATH
! #PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin export PATH
#
# each product MUST provide a $PRODUCT/admin/shrept.lst
```

11.6.5. Oracle® starten

Wenn Sie den Anweisungen gefolgt sind, sollten Sie nun in der Lage sein, Oracle® zu starten, genau so, wie Sie dies auch unter Linux tun würden.

11.7. Weiterführende Themen

Wenn Sie sich fragen, wie die Linux-Binärkompatibilität unter FreeBSD realisiert wurde, sollten Sie diesen Abschnitt lesen. Der Großteil der folgenden Informationen stammt aus einer E-Mail, die von Terry Lambert (<tlambert@primenet.com>) an die FreeBSD-Chat-Mailingliste (<freebsd-chat@FreeBSD.org>) geschrieben wurde (Message ID: <199906020108.SAA07001@usr09.primenet.com>).

11.7.1. Wie funktioniert es?

FreeBSD verfügt über eine „execution class loader“ genannte Abstraktion. Dabei handelt es sich um einen Eingriff in den execve(2) Systemaufruf.

FreeBSD verfügt über eine Liste von Ladern, anstelle eines einzigen, auf `#!` zurückgreifenden Laders, um Shell-Interpreter oder Shell-Skripte auszuführen.

Historisch gesehen untersuchte der einzige, auf UNIX-Plattformen vorhandene Lader die "magische Zahl" (in der Regel die ersten 4 oder 8 Bytes der Datei), um festzustellen, ob der Binärtyp dem System bekannt war. War dies der Fall, wurde der Binärlader aufgerufen.

Wenn es sich nicht um den zum System gehörigen Binärtyp handelte, gab execve(2) einen Fehler zurück, und die Shell versuchte stattdessen, die Datei als Shell-Befehl auszuführen.

Dabei wurde als Standardeinstellung „was auch immer die aktuelle Shell ist“ festgelegt.

Später wurde ein Hack in sh(1) eingefügt, der die zwei ersten Zeichen untersuchte. Wenn diese `:\n` entsprachen, wurde stattdessen die csh(1)-Shell aufgerufen (wir glauben, dass dies zuerst von SCO umgesetzt wurde).

FreeBSD versucht heute eine Liste von Ladern, unter denen sich ein allgemeiner Lader für Interpreter befindet. Der auszuführende Interpreter wird im ersten, durch Leerzeichen getrennten Feld, der `#!`-Zeile angegeben. Lässt sich der Interpreter nicht ermitteln, wird auf `/bin/sh` zurückgegriffen.

Für die Linux ABI-Unterstützung erkennt FreeBSD die magische Zahl als ELF-Binärdatei (Zu diesem Zeitpunkt wird nicht zwischen FreeBSD, Solaris™, Linux oder anderen Systemen unterschieden, die über ELF-Binärdateien verfügen.).

Der ELF-Lader sucht nach einer speziellen *Kennzeichnung*, die aus einem Kommentarabschnitt in der ELF-Datei besteht, und die in SVR4/Solaris™ ELF Binärdateien nicht vorhanden ist.

Damit Linux-Binärdateien (unter FreeBSD) funktionieren, müssen sie als `Linux` *gekennzeichnet* werden, und zwar durch brandelf(1):

```
# brandelf -t Linux file
```

Nachdem dies geschehen ist, erkennt der ELF-Lader die `Linux`-Kennzeichnung der Datei.

Wenn der ELF-Lader die `Linux`-Kennzeichnung sieht, wird ein Zeiger in der `proc`-Struktur ersetzt. Alle Systemaufrufe werden durch diesen Zeiger indiziert (in einem traditionellen UNIX® System wäre das ein `sysent[]`-Strukturfeld, das die Systemaufrufe enthält). Der Prozess wird weiterhin speziell gekennzeichnet, so dass der Trap-vector im Signal-trampoline-code eine spezielle Behandlung erfährt und das Linux-Kernelmodul verschiedene kleinere Korrekturen vornehmen kann.

Der Linux-Systemaufrufvektor enthält neben anderen Dingen eine Liste der `sysent[]`-Einträge, deren Adressen sich im Kernelmodul befinden.

Wenn ein Linux-Programm einen Systemaufruf ausführt, dereferenziert die Trap-Behandlungsroutine den Zeiger auf die Eintrittspunkte für die Systemaufrufe und erhält damit die Linux-Eintrittspunkte und nicht die FreeBSD-Eintrittspunkte.

Zusätzlich *verändert* der Linuxmodus die Systempfade dynamisch; genauso, wie dies die Option `union` beim Einbinden von Dateisystemen macht (Achtung: *nicht* das Dateisystem unionfs!). Zuerst wird die Datei im Verzeichnis `/compat/linux/Originalpfad` gesucht, *danach*, wenn sie dort nicht gefunden wurde, wird sie im FreeBSD-Verzeichnis `/Originalpfad` gesucht. Dadurch wird sichergestellt, dass Binärdateien, die zur Ausführung andere Binärdateien benötigen, ausgeführt werden können (so dass alle Linux-Werkzeuge unter der ABI laufen). Dies bedeutet auch, dass Linux-Binärdateien FreeBSD-Binärdateien laden und ausführen können, wenn keine passenden Linux-Binärdateien vorhanden sind. Ein in `/compat/linux` plaziertes uname(1) kann damit Linux-Programmen vorgaukeln, dass sie auf einem Linux-System laufen.

Im Endeffekt gibt es einen Linux-Kernel innerhalb des FreeBSD-Kernels. Die Sprungtabellen für Linux- beziehungsweise FreeBSD-Systemaufrufe verweisen allerdings auf dieselben Funktionen, die Kerneldienste wie Dateisystemoperationen, Operationen für den virtuellen Speicher, Signalübermittlung und System V IPC bereitstellen, Der einzige Unterschied ist, dass Binärdateien unter FreeBSD FreeBSD-*glue*-Funktionen verwenden. Linux-Binärdateien hingegen verwenden die Linux-*glue*-Funktionen. Die meisten älteren Betriebssysteme hatten ihre eigenen *glue*-Funktionen: Funktionsadressen in einem globalen, statischen `sysent[]` Strukturfeld an Stelle von Funktionsadressen, die durch einen dynamisch initialisierten Zeiger aus der `proc` Struktur, die den Aufruf gemacht hatte, dereferenziert wurden.

Welche ist die echte FreeBSD-ABI? Das spielt keine Rolle. Grundsätzlich ist der einzige Unterschied (zurzeit ist das so; dies könnte sich in zukünftigen Versionen leicht ändern und wird sich wahrscheinlich auch ändern), dass die FreeBSD-*glue*-Funktionen statisch in den Kernel gelinkt sind, und dass die Linux-*glue*-Funktionen statisch gelinkt oder über ein Modul eingebunden werden können.

Ja, aber ist das wirkliche eine Emulation? Nein. Es ist eine Implementierung eines ABIs, keine Emulation. Es ist kein Emulator (oder Simulator, um der nächsten Frage zuvorzukommen) beteiligt.

Warum wird es manchmal „Linux-Emulation“ genannt? Um es schwerer zu machen, FreeBSD zu verkaufen. Wirklich, das kommt daher, weil dies zu einer Zeit implemtiert wurde, in der es kein anderes Wort (als Emulation) gab, das beschrieb, was vor sich ging. Wenn der Kernel nicht entsprechend konfiguriert wurde oder das Modul geladen wurde, war es falsch zu behaupten, FreeBSD würde Linux-Binärprogramme ausführen. Man benötigte ein Wort, das beschrieb, was da geladen wurde – daher „Der Linux-Emulator“.

Teil III. Systemadministration

Die restlichen Kapitel behandeln alle Aspekte der FreeBSD Systemadministration. Am Anfang jedes Kapitels finden Sie eine Zusammenfassung, die beschreibt, was Sie nach dem Durcharbeiten des Kapitels gelernt haben. Weiterhin werden die Voraussetzungen beschrieben, die für das Durcharbeiten des Kapitels erforderlich sind.

Diese Kapitel sollten Sie lesen, wenn Sie die Informationen darin benötigen. Sie brauchen Sie nicht in einer bestimmten Reihenfolge zu lesen, noch müssen Sie die Kapitel lesen, bevor Sie anfangen, FreeBSD zu benutzen.

Inhaltsverzeichnis

Kapitel 12. Konfiguration und Tuning

Geschrieben von Chern Lee.
Nach einem Tutorium von Mike Smith.
Basiert ebenfalls auf tuning(7) von Matt Dillon.
Übersetzt von Martin Heinen.

12.1. Übersicht

Die richtige Systemkonfiguration ist einer der wichtigsten Aspekte unter FreeBSD. Dieses Kapitel beschreibt die Konfiguration von FreeBSD sowie Maßnahmen zur Leistungssteigerung von FreeBSD-Systemen.

Nachdem Sie dieses Kapitel durchgearbeitet haben, werden Sie Folgendes wissen:

- Die Grundlagen der Konfiguration von `rc.conf` und die Skripte zum Starten von Anwendungen in `/usr/local/etc/rc.d`.
- Wie Sie Netzwerkkarten konfigurieren und testen.
- Wie Sie virtuelle Hosts und Netzwerkgeräte konfigurieren.
- Wie Sie die verschiedenen Konfigurationsdateien in `/etc` benutzen.
- Wie Sie mit FreeBSD mit sysctl(8)-Variablen einstellen können.
- Wie Sie die Platten-Performance einstellen und Kernel-Parameter modifizieren können.

Bevor Sie dieses Kapitel lesen, sollten Sie

- die Grundlagen von UNIX® und FreeBSD (Kapitel 4, *Grundlagen des UNIX Betriebssystems*) verstehen.
- Damit vertraut sein, wie Sie einen Kernel konfigurieren und kompilieren (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).

12.2. Start von Diensten

Beigetragen von Tom Rhodes.

Viele Benutzer installieren Software Dritter auf FreeBSD mithilfe der Ports-Sammlung. Häufig soll die Software bei einem Systemstart mitgestartet werden. Beispielsweise sollen die Dienste mail/postfix oder www/apache22 nach einem Systemstart laufen. Dieser Abschnitt stellt die Startprozeduren für Software Dritter vor.

Unter FreeBSD werden die meisten der im System enthaltenen Dienste wie cron(8) mithilfe von Systemskripten gestartet.

12.2.1. Dienste über das `rc.d`-System starten

Mit `rc.d` lässt sich der Start von Anwendungen besser steuern und es sind mehr Funktionen verfügbar. Mit den in Abschnitt 12.4, „Dienste unter FreeBSD verwalten“ besprochenen Schlüsselwörtern können Anwendungen in einer bestimmten Reihenfolge gestartet werden und Optionen können in `rc.conf` statt fest im Startskript der Anwendung festgelegt werden. Ein einfaches Startskript sieht wie folgt aus:

```
#!/bin/sh
#
# PROVIDE: utility
# REQUIRE: DAEMON
# KEYWORD: shutdown

. /etc/rc.subr

name=utility
rcvar=utility_enable

command="/usr/local/sbin/utility"

load_rc_config $name

#
# DO NOT CHANGE THESE DEFAULT VALUES HERE
# SET THEM IN THE /etc/rc.conf FILE
#
utility_enable=${utility_enable-"NO"}
pidfile=${utility_pidfile-"/var/run/utility.pid"}

run_rc_command "$1"
```

Dieses Skript stellt sicher, dass `utility` nach den `DAEMON`-Pseudodiensten gestartet wird. Es stellt auch eine Methode bereit, die Prozess-ID (PID) der Anwendung in einer Datei zu speichern.

In `/etc/rc.conf` könnte für diese Anwendung die folgende Zeile stehen:

```
utility_enable="YES"
```

Die Methode erleichtert den Umgang mit Kommandozeilenargumenten, bindet Funktionen aus `/etc/rc.subr` ein, ist kompatibel zu rcorder(8) und lässt sich über `rc.conf` leichter konfigurieren.

12.2.2. Andere Arten, um Dienste zu starten

Andere Dienste können über inetd(8) gestartet werden. Die Konfiguration von inetd(8) wird in Abschnitt 29.2, „Der inetd „Super-Server““ ausführlich beschrieben.

Systemdienste können auch mit cron(8) gestartet werden. Dieser Ansatz hat einige Vorteile; nicht zuletzt, weil cron(8) die Prozesse unter dem Eigentümer der `crontab` startet, ist es möglich, dass Dienste von normalen Benutzern gestartet und gepflegt werden können.

Für die Zeitangabe in cron(8) kann `@reboot` eingesetzt werden. Damit wird das Kommando gestartet, wenn cron(8) kurz nach dem Systemboot gestartet wird.

12.3. cron(8) konfigurieren

Beigetragen von Tom Rhodes.

Ein sehr nützliches Werkzeug von FreeBSD ist cron. Dieses Programm läuft im Hintergrund und überprüft fortlaufend `/etc/crontab` und `/var/cron/tabs` . In diesen Dateien wird festgelegt, welche Programme zu welchem Zeitpunkt von cron ausgeführt werden sollen.

Das Werkzeug verwendet zwei verschiedene Konfigurationsdateien: die System-`crontab` und die Benutzer-`crontabs`. Der einzige Unterschied zwischen beiden Formaten ist das sechste Feld. In der System-`crontab` gibt das sechste Feld den Benutzer an, mit dem cron das Kommando ausführen wird. In einer Benutzer-`crontab` werden alle Kommandos unter dem Benutzer ausgeführt, welcher die `crontab` erstellt hat. Hier ist das sechste Feld das letzte Feld. Dies ist ein wichtiges Sicherheitsmerkmal. Das letzte Feld bezeichnet immer das Kommando, das ausgeführt werden soll.

Anmerkung

Benutzer können mit Benutzer-`crontabs` ohne `root`-Rechte Befehle terminieren. Die Kommandos in Benutzer-`crontabs` laufen unter dem Benutzer, der die `crontab` erstellt hat.

Der Benutzer `root` kann, wie jeder andere Benutzer, eine Benutzer-`crontab` besitzen. Die Benutzer-`crontab` von `root` ist nicht mit `/etc/crontab`, der System-`crontab`, zu verwechseln. Da die System-`crontab` die angegebenen Kommandos effektiv als `root`-Benutzer aufruft, besteht normalerweise keine Notwendigkeit eine eigene Benutzer-`crontab` für `root` zu erstellen.

Hier ist ein Beispieleintrag aus der System-crontab, `/etc/crontab`:

```
# /etc/crontab - root's crontab for FreeBSD
#
#$FreeBSD: head/de_DE.ISO8859-1/books/handbook/config/chapter.xml 48479 2016-03-27 ↺
18:06:07Z bhd $
# ❶
#
SHELL=/bin/sh
PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin ❷
#
#
#minute hour mday month wday who command ❸
#
*/5 * * * * root /usr/libexec/atrun ❹
```

❶ Das Zeichen # am Zeilenanfang leitet, wie in den meisten Konfigurationsdateien, einen Kommentar ein. Benutzen Sie Kommentare, um die Funktion eines Eintrags zu erläutern. Kommentare müssen in einer extra Zeile stehen. Sie können nicht in derselben Zeile wie ein Kommando stehen, da sie sonst Teil des Kommandos wären. Leerzeilen in dieser Datei werden ignoriert.

❷ Umgebungsvariablen werden mit dem Gleichheits-Zeichen (=) festgelegt. Im Beispiel werden die Variablen `SHELL`, `PATH` und `HOME` definiert. Wenn die Variable `SHELL` nicht definiert wird, benutzt cron die Bourne Shell. Wird die Variable `PATH` nicht gesetzt, müssen alle Pfadangaben absolut sein, da es keinen Vorgabewert für `PATH` gibt.

❸ In dieser Zeile werden sieben Felder beschrieben: `minute`, `hour`, `mday`, `month`, `wday`, `who` und `command`. Die ersten Felder legen den Zeitpunkt fest, an dem ein Kommando laufen soll. Das Feld `minute` legt die Minute fest, das Feld `hour` die Stunde, das Feld `mday` den Tag des Monats. Im Feld `month` wird der Monat und im Feld `wday` der Wochentag festgelegt. Alle Felder müssen numerische Werte enthalten und die Zeitangaben sind im 24-Stunden-Format. Das Zeichen * repräsentiert dabei alle möglichen Werte für dieses Feld. Das Feld `who` gibt es nur in der Datei `/etc/crontab` und gibt den Account an, unter dem das Kommando laufen soll. Im letzten Feld wird schließlich das auszuführende Kommando angegeben.

❹ Diese Zeile definiert die Zeitpunkte an denen `atrun` laufen soll. Dieses Beispiel verwendet die Zeichenfolge `*/5` gefolgt von mehreren *-Zeichen. Das Zeichen * ist ein Platzhalter und steht für *jede* mögliche Zeit. Diese Zeile führt `/usr/libexec/atrun` unter dem `root`-Account alle fünf Minuten aus.

Bei den Kommandos können beliebige Optionen angegeben werden. Wenn das Kommando zu lang ist und auf der nächsten Zeile fortgesetzt werden soll, muss am Ende der Zeile das Fortsetzungszeichen (\) angegeben werden.

Bis auf das sechste Feld, das den Account angibt, sieht jede crontab(5) so wie das Beispiel aus. Das sechste Feld existiert nur in der System-`crontab`. In den restlichen crontab(5)-Dateien fehlt dieses Feld.

12.3.1. crontab installieren

Wichtig

Die nachstehende Prozedur gilt nur für Benutzer-crontabs. Die System-crontab kann mit einem Editor bearbeitet werden. cron bemerkt, dass sich die Datei geändert hat und wird die neue Version benutzen. Lesen Sie bitte auch die FAQ zur Meldung root: not found.

Eine Benutzer-crontab, beispielsweise die Datei crontab-datei , können Sie mit jedem Editor erstellen. Die Benutzer-crontab installieren Sie mit dem nachstehenden Befehl:

```
# crontab crontab-datei
```

Das Argument zum Befehl crontab(5) ist die vorher erstellte crontab-datei .

Der Befehl crontab -l zeigt die installierte crontab-Datei an.

Benutzer, die eine eigene crontab-Datei ohne Vorlage erstellen wollen, können den Befehl crontab -e verwenden. Dieser Befehl ruft einen Editor auf und installiert beim Verlassen des Editors die crontab-Datei.

Wollen Sie die installierte Benutzer-crontab entfernen, rufen Sie den crontab -r.

12.4. Dienste unter FreeBSD verwalten

Beigetragen von Tom Rhodes.

FreeBSD verwendet die vom rc(8)-System bereit gestellten Startskripten beim Systemstart und für die Verwaltung von Diensten. Die Skripte sind in /etc/rc.d abgelegt und bieten grundlegende Dienste an, die über die Optionen start , stop und restart des service(8) Kommandos kontrolliert werden können. Beispielsweise kann sshd(8) mit dem nachstehenden Kommando neu gestartet werden:

```
# service sshd restart
```

Analog können Sie andere Dienste starten und stoppen. Normalerweise werden die Dienste beim Systemstart über Einträge in der Datei rc.conf(5) automatisch gestartet. natd(8) wird zum Beispiel mit dem folgenden Eintrag in /etc/rc.conf aktiviert:

```
natd_enable="YES"
```

Wenn dort bereits die Zeile natd_enable="NO" existiert, ändern Sie NO in YES. Die rc(8)-Skripten starten, wie unten beschrieben, auch abhängige Dienste.

Da das rc(8)-System primär zum automatischen Starten und Stoppen von Systemdiensten dient, funktionieren die Optionen start , stop und restart nur, wenn die entsprechenden Variablen in /etc/rc.conf gesetzt sind. Beispielsweise funktioniert sshd restart nur dann, wenn in /etc/rc.conf die Variable sshd_enable auf YES gesetzt wurde. Wenn Sie die Optionen start , stop oder restart unabhängig von den Einstellungen in /etc/rc.conf benutzen wollen, müssen Sie den Optionen mit dem Präfix „one“ verwenden. Um beispielsweise sshd unabhängig von den Einstellungen in /etc/rc.conf neu zu starten, benutzen Sie das nachstehende Kommando:

```
# service sshd onerestart
```

Ob ein Dienst in /etc/rc.conf aktiviert ist, können Sie herausfinden, indem Sie das entsprechende rc(8)-Skript mit der Option rcvar aufrufen. Dieses Beipiel prüft, ob der sshd-Dienst in /etc/rc.conf aktiviert ist:

```
# service sshd rcvar
# sshd
#
```

```
sshd_enable="YES"
#   (default: "")
```

Anmerkung

Die Zeile `# sshd` wird von dem Kommando ausgegeben; sie kennzeichnet nicht die Eingabeaufforderung von `root`.

Ob ein Dienst läuft, kann mit `status` abgefragt werden. Das folgende Kommando überprüft, ob `sshd` auch wirklich gestartet wurde:

```
# service sshd status
sshd is running as pid 433.
```

Einige Dienste können über die Option `reload` neu initialisiert werden. Dazu wird dem Dienst über ein Signal mitgeteilt, dass er seine Konfigurationsdateien neu einlesen soll. Oft wird dazu das Signal `SIGHUP` verwendet. Beachten Sie aber, dass nicht alle Dienste diese Option unterstützen.

Die meisten Systemdienste werden beim Systemstart vom rc(8)-System gestartet. Zum Beispiel aktiviert das Skript `/etc/rc.d/bgfsck` die Prüfung von Dateisystemen im Hintergrund. Das Skript gibt die folgende Meldung aus, wenn es gestartet wird:

```
Starting background file system checks in 60 seconds.
```

Dieses Skript wird während des Systemstarts ausgeführt und führt eine Überprüfung der Dateisysteme im Hintergrund durch.

Viele Systemdienste hängen von anderen Diensten ab. yp(8) und andere RPC-basierende Systeme hängen beispielsweise von dem `rpcbind`-Dienst ab. Im Kopf der Startskripten befinden sich die Informationen über Abhängigkeiten von anderen Diensten und weitere Metadaten.Mithilfe dieser Daten bestimmt das Programm rcorder(8) beim Systemstart die Startreihenfolge der Dienste.

Folgende Schlüsselwörter müssen im Kopf aller Startskripten verwendet werden, da sie von rc.subr(8) zum „Aktivieren“ des Startskripts benötigt werden:

- `PROVIDE`: Gibt die Namen der Dienste an, die mit dieser Datei zur Verfügung gestellt werden.

Die folgenden Schlüsselwörter können im Kopf des Startskripts angegeben werden. Sie sind zwar nicht unbedingt notwendig, sind aber hilfreich beim Umgang mit rcorder(8):

- `REQUIRE`: Gibt die Namen der Dienste an, von denen dieser Dienst abhängt. Ein Skript, das dieses Schlüsselwort enthält wird *nach* den angegebenen Diensten ausgeführt.
- `BEFORE`: Zählt Dienste auf, die auf diesen Dienst angewiesen sind. Ein Skript, dass dieses Schlüsselwort enthält wird *vor* den angegebenen Diensten ausgeführt.

Durch das Verwenden dieser Schlüsselwörter kann ein Administrator die Startreihenfolge von Systemdiensten feingranuliert steuern, ohne mit den Schwierigkeiten des „runlevel“-Systems anderer UNIX® Systeme kämpfen zu müssen.

Weitere Informationen über das rc(8)-System finden Sie in rc(8) und rc.subr(8). Wenn Sie eigene `rc.d`-Skripte schreiben wollen, sollten Sie diesen Artikel lesen.

12.4.1. Systemspezifische Konfiguration

Informationen zur Systemkonfiguration sind hauptsächlich in `/etc/rc.conf` , die meist beim Start des Systems verwendet wird, abgelegt. Sie enthält die Konfigurationen für die `rc*` Dateien.

In `rc.conf` werden die Vorgabewerte aus `/etc/defaults/rc.conf` überschrieben. Die Vorgabedatei sollte nicht editiert werden. Stattdessen sollten alle systemspezifischen Änderungen in `rc.conf` vorgenommen werden.

Um den administrativen Aufwand gering zu halten, existieren in geclusterten Anwendungen mehrere Strategien, globale Konfigurationen von systemspezifischen Konfigurationen zu trennen. Der empfohlene Weg hält die globale Konfiguration in einer separaten Datei z.B. `/etc/rc.conf.local`. Zum Beispiel so:

- `/etc/rc.conf`:

```
sshd_enable="YES"
keyrate="fast"
defaultrouter="10.1.1.254"
```

- `/etc/rc.conf.local`:

```
hostname="node1.example.org"
ifconfig_fxp0="inet 10.1.1.1/8"
```

`/etc/rc.conf` kann dann auf jedes System mit rsync oder puppet verteilt werden, während `/etc/rc.conf.local` dabei systemspezifisch bleibt.

Bei einem Upgrade des Systems mit sysinstall(8) oder `make world` wird `/etc/rc.conf` nicht überschrieben, so dass die Systemkonfiguration erhalten bleibt.

Tipp

`/etc/rc.conf` und `/etc/rc.conf.local` werden von sh(1) gelesen. Dies erlaubt es dem Systemadministrator, komplexe Konfigurationsszenarien zu erstellen. Lesen Sie rc.conf(5), um weitere Informationen zu diesem Thema zu erhalten.

12.5. Einrichten von Netzwerkkarten

Beigetragen von Marc Fonvieille.

Die Konfiguration einer Netzwerkkarte gehört zu den alltäglichen Aufgaben eines FreeBSD Administrators.

12.5.1. Bestimmen des richtigen Treibers

Ermitteln Sie zunächst das Modell der Netzwerkkarte und den darin verwendeten Chip. FreeBSD unterstützt eine Vielzahl von Netzwerkkarten. Prüfen Sie die Hardware-Kompatibilitätsliste für das FreeBSD Release, um zu sehen ob die Karte unterstützt wird.

Wenn die Karte unterstützt wird, müssen Sie den Treiber für die Karte bestimmen. `/usr/src/sys/conf/NOTES` und `/usr/src/sys/`*`arch`*`/conf/NOTES` enthalten eine Liste der verfügbaren Treiber mit Informationen zu den unterstützten Chipsätzen. Wenn Sie sich nicht sicher sind, ob Sie den richtigen Treiber ausgewählt haben, lesen Sie die Hilfeseite des Treibers. Sie enthält weitere Informationen über die unterstützten Geräte und bekannte Einschränkungen des Treibers.

Die Treiber für gebräuchliche Netzwerkkarten sind schon im `GENERIC` -Kernel enthalten, so dass die Karte während des Systemstarts erkannt werden sollte. In diesem Beispiel findet das System zwei Karten, die den dc(4)-Treiber benutzen:

```
dc0: <82c169 PNIC 10/100BaseTX> port 0xa000-0xa0ff mem 0xd3800000-0xd38
000ff irq 15 at device 11.0 on pci0
miibus0: <MII bus> on dc0
bmtphy0: <BCM5201 10/100baseTX PHY> PHY 1 on miibus0
```

```
bmtphy0:  10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
dc0: Ethernet address: 00:a0:cc:da:da:da
dc0: [ITHREAD]
dc1: <82c169 PNIC 10/100BaseTX> port 0x9800-0x98ff mem 0xd3000000-0xd30
000ff irq 11 at device 12.0 on pci0
miibus1: <MII bus> on dc1
bmtphy1: <BCM5201 10/100baseTX PHY> PHY 1 on miibus1
bmtphy1:  10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
dc1: Ethernet address: 00:a0:cc:da:da:db
dc1: [ITHREAD]
```

Ist der Treiber für die Netzwerkkarte nicht in `GENERIC` enthalten, muss zunächst ein Treiber geladen werden, um die Karte konfigurieren und benutzen zu können. Dafür gibt es zwei Methoden:

- Am einfachsten ist es, das Kernelmodul für die Karte mit kldload(8) zu laden. Um den Treiber automatisch beim Systemstart zu laden, fügen Sie die entsprechende Zeile in `/boot/loader.conf` ein. Es gibt nicht für alle Karten Kernelmodule.
- Alternativ kann der Treiber für die Karte fest in den Kernel eingebunden werden. Lesen Sie dazu `/usr/src/sys/conf/NOTES` , `/usr/src/sys/` *`arch`*`/conf/NOTES` und die Hilfeseite des Treibers, den Sie in den Kernel einbinden möchten, an. Die Übersetzung des Kernels wird in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben. Wenn die Karte während des Systemstarts vom Kernel erkannt wurde, muss der Kernel nicht neu übersetzt werden.

12.5.1.1. Windows®-NDIS-Treiber einsetzen

Leider stellen nach wie vor viele Unternehmen die Spezifikationen ihrer Treiber der Open Source Gemeinde nicht zur Verfügung, weil sie diese Informationen als Geschäftsgeheimnisse betrachten. Daher haben die Entwickler von FreeBSD und anderen Betriebssystemen nur zwei Möglichkeiten. Entweder versuchen sie in einem aufwändigen Prozess den Treiber durch *Reverse Engineering* nachzubauen, oder sie versuchen, die vorhandenen Binärtreiber der Microsoft® Windows®-Plattform zu verwenden.

FreeBSD bietet „native“ Unterstützung für die *Network Driver Interface Specification* (NDIS). ndisgen(8) wird benutzt, um einen Windows® XP-Treiber in ein Format zu konvertieren, das von FreeBSD verwendet werden kann. Da der ndis(4)-Treiber einen Windows® XP-Binärtreiber nutzt, kann er nur auf i386™- und amd64-Systemen verwendet werden. Unterstützt werden PCI, CardBus, PCMCIA und USB-Geräte.

Um den NDISulator zu verwenden, benötigen Sie drei Dinge:

1. Die FreeBSD Kernelquellen
2. Den Windows® XP-Binärtreiber mit der Erweiterung `.SYS`
3. Die Konfigurationsdatei des Windows® XP-Treibers mit der Erweiterung `.INF`

Laden Sie die `.SYS`- und `.INF`-Dateien für die Karte. Diese befinden sich meistens auf einer beigelegten CD-ROM, oder können von der Internetseite des Herstellers heruntergeladen werden. In den folgenden Beispielen werden die Dateien `W32DRIVER.SYS` und `W32DRIVER.INF` verwendet.

Die Architektur des Treibers muss zur jeweiligen Version von FreeBSD passen. Benutzen Sie einen Windows® 32-bit Treiber für FreeBSD/i386. Für FreeBSD/amd64 wird ein Windows® 64-bit Treiber benötigt.

Als Nächstes kompilieren Sie den binären Treiber, um ein Kernelmodul zu erzeugen. Dazu rufen Sie als `root` ndisgen(8) auf:

```
# ndisgen /path/to/W32DRIVER.INF  /path/to/W32DRIVER.SYS
```

Dieses Kommando arbeitet interaktiv, benötigt es weitere Informationen, so fragt es Sie danach. Das Ergebnis ist ein neu erzeugtes Kernelmodul im aktuellen Verzeichnis. Benutzen Sie kldload(8) um das neue Modul zu laden:

```
# kldload ./W32DRIVER.ko
```

Neben dem erzeugten Kernelmodul müssen auch die Kernelmodule `ndis.ko` und `if_ndis.ko` geladen werden. Dies passiert automatisch, wenn Sie ein von ndis(4) abhängiges Modul laden. Andernfalls können die Module mit den folgenden Kommandos manuell geladen werden:

```
# kldload ndis
# kldload if_ndis
```

Der erste Befehl lädt den ndis(4)-Miniport-Treiber, der zweite das tatsächliche Netzwerkgerät.

Überprüfen Sie die Ausgabe von dmesg(8) auf eventuelle Fehler während des Ladevorgangs. Gab es dabei keine Probleme, sollte die Ausgabe wie folgt aussehen:

```
ndis0: <Wireless-G PCI Adapter> mem 0xf4100000-0xf4101fff irq 3 at device 8.0 on pci1
ndis0: NDIS API version: 5.0
ndis0: Ethernet address: 0a:b1:2c:d3:4e:f5
ndis0: 11b rates: 1Mbps 2Mbps 5.5Mbps 11Mbps
ndis0: 11g rates: 6Mbps 9Mbps 12Mbps 18Mbps 36Mbps 48Mbps 54Mbps
```

Ab jetzt kann das Gerät `ndis0` wie jede andere Netzwerkkarte konfiguriert werden.

Um die ndis(4)-Module automatisch beim Systemstart zu laden, kopieren Sie das erzeugte Modul `W32DRIVER_SYS.ko` nach `/boot/modules`. Danach fügen Sie die folgende Zeile in `/boot/loader.conf` ein:

```
W32DRIVER_SYS_load="YES"
```

12.5.2. Konfiguration von Netzwerkkarten

Nachdem der richtige Treiber für die Karte geladen ist, muss die Karte konfiguriert werden. Unter Umständen ist die Karte schon während der Installation mit bsdinstall(8) konfiguriert worden.

Das nachstehende Kommando zeigt die Konfiguration der Netzwerkkarten an:

```
% ifconfig
dc0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80008<VLAN_MTU,LINKSTATE>
        ether 00:a0:cc:da:da:da
        inet 192.168.1.3 netmask 0xffffff00 broadcast 192.168.1.255
        media: Ethernet autoselect (100baseTX <full-duplex>)
        status: active
dc1: flags=8802<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80008<VLAN_MTU,LINKSTATE>
        ether 00:a0:cc:da:da:db
        inet 10.0.0.1 netmask 0xffffff00 broadcast 10.0.0.255
        media: Ethernet 10baseT/UTP
        status: no carrier
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
        options=3<RXCSUM,TXCSUM>
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x4
        inet6 ::1 prefixlen 128
        inet 127.0.0.1 netmask 0xff000000
        nd6 options=3<PERFORMNUD,ACCEPT_RTADV>
```

Im Beispiel werden Informationen zu den folgenden Geräten angezeigt:

- `dc0`: Der erste Ethernet-Adapter.
- `dc1`: Der zweite Ethernet-Adapter.
- `lo0`: Das Loopback-Gerät.

Der Name der Netzwerkkarte wird aus dem Namen des Treibers und einer Zahl zusammengesetzt. Die Zahl gibt die Reihenfolge an, in der die Geräte beim Systemstart erkannt wurden. Die dritte Karte, die den sis(4) Treiber benutzt, würde beispielsweise `sis2` heißen.

Der Adapter `dc0` aus dem Beispiel ist aktiv. Sie erkennen das an den folgenden Hinweisen:

1. `UP` bedeutet, dass die Karte konfiguriert und aktiv ist.
2. Der Karte wurde die Internet-Adresse (`inet`) `192.168.1.3` zugewiesen.
3. Die Subnetzmaske ist richtig (`0xffffff00` entspricht `255.255.255.0`).
4. Die Broadcast-Adresse `192.168.1.255` ist richtig.
5. Die MAC-Adresse der Karte (`ether`) lautet `00:a0:cc:da:da:da` .
6. Die automatische Medienerkennung ist aktiviert (`media: Ethernet autoselect (100baseTX <full-duplex>)`). Der Adapter `dc1` benutzt das Medium `10baseT/UTP` . Weitere Informationen über die einstellbaren Medien entnehmen Sie der Hilfeseite des Treibers.
7. Der Verbindungsstatus (`status`) ist `active` , das heißt es wurde ein Trägersignal entdeckt. Für `dc1` wird `status: no carrier` angezeigt. Das ist normal, wenn kein Kabel an der Karte angeschlossen ist.

Wäre die Karte nicht konfiguriert, würde die Ausgabe von ifconfig(8) so aussehen:

```
dc0: flags=8843<BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=80008<VLAN_MTU,LINKSTATE>
        ether 00:a0:cc:da:da:da
        media: Ethernet autoselect (100baseTX <full-duplex>)
        status: active
```

Die Karte muss als Benutzer `root` konfiguriert werden. Die Konfiguration kann auf der Kommandozeile mit ifconfig(8) erfolgen. Allerdings gehen diese Informationen bei einem Neustart verloren. Tragen Sie stattdessen die Konfiguration in `/etc/rc.conf` ein. Fügen Sie für jede Karte im System eine Zeile hinzu, wie in diesem Beispiel zu sehen:

```
ifconfig_dc0="inet 192.168.1.3 netmask 255.255.255.0"
ifconfig_dc1="inet 10.0.0.1 netmask 255.255.255.0 media 10baseT/UTP"
```

Ersetzen Sie `dc0` und `dc1` und die IP-Adressen durch die richtigen Werte für das System. Die Manualpages des Treibers und ifconfig(8) enthalten weitere Einzelheiten über verfügbare Optionen. Die Syntax von `/etc/rc.conf` wird in rc.conf(5) beschrieben.

Wenn das Netz während der Installation konfiguriert wurde, existieren vielleicht schon Einträge für die Netzwerkkarte(n). Überprüfen Sie `/etc/rc.conf` bevor Sie weitere Zeilen hinzufügen.

Wenn das Netzwerk kein DNS benutzt, können Sie in `/etc/hosts` die Namen und IP-Adressen der Rechner des LANs eintragen. Weitere Informationen entnehmen Sie hosts(5) und `/usr/share/examples/etc/hosts` .

Anmerkung

Falls kein DHCP-Server zur Verfügung steht, Sie aber Zugang zum Internet benötigen, müssen Sie das Standard-Gateway und die Nameserver manuell konfigurieren:

```
# echo 'defaultrouter="Ihr_Default_Gateway"' >> /etc/rc.conf
# echo 'nameserver Ihr_DNS_Server' >> /etc/resolv.conf
```

12.5.3. Test und Fehlersuche

Nachdem die notwendigen Änderungen in `/etc/rc.conf` gespeichert wurden, kann das System neu gestartet werden, um die Konfiguration zu testen und zu überprüfen, ob das System ohne Fehler neu gestartet wurde. Alternativ können Sie mit folgenden Befehl die Netzwerkeinstellungen neu initialisieren:

```
# service netif restart
```

Anmerkung

Falls in `/etc/rc.conf` ein Default-Gateway definiert wurde, müssen Sie auch den folgenden Befehl ausführen:

```
# service routing restart
```

Wenn das System gestartet ist, sollten Sie die Netzwerkkarten testen.

12.5.3.1. Test der Ethernet-Karte

Um zu prüfen, ob die Ethernet-Karte richtig konfiguriert ist, testen Sie zunächst mit ping(8) den Adapter selbst und sprechen Sie dann eine andere Maschine im LAN an.

Zuerst, der Test des Adapters:

```
% ping -c5 192.168.1.3
PING 192.168.1.3 (192.168.1.3): 56 data bytes
64 bytes from 192.168.1.3: icmp_seq=0 ttl=64 time=0.082 ms
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=0.074 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=0.076 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=0.108 ms
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=0.076 ms

--- 192.168.1.3 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.074/0.083/0.108/0.013 ms
```

```
% ping -c5 192.168.1.2
PING 192.168.1.2 (192.168.1.2): 56 data bytes
64 bytes from 192.168.1.2: icmp_seq=0 ttl=64 time=0.726 ms
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.766 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.700 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.747 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.704 ms

--- 192.168.1.2 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.700/0.729/0.766/0.025 ms
```

Um die Namensauflösung zu testen, verwenden Sie den Namen der Maschine anstelle der IP-Adresse. Wenn kein DNS-Server im Netzwerk vorhanden ist, muss `/etc/hosts` entsprechend eingerichtet sein.

12.5.3.2. Fehlersuche

Fehler zu beheben, ist immer sehr mühsam. Indem Sie die einfachen Sachen zuerst prüfen, erleichtern Sie sich die Aufgabe. Steckt das Netwerkkabel? Sind die Netzwerkdienste richtig konfiguriert? Funktioniert die Firewall? Wird die Netwerkkarte von FreeBSD unterstützt? Lesen Sie immer die Hardware-Informationen des Releases, bevor Sie einen Fehlerbericht einsenden. Aktualisieren Sie die FreeBSD-Version auf die neueste -STABLE Version. Suchen Sie in den Archiven der Mailinglisten und im Internet nach bekannten Lösungen.

Wenn die Karte funktioniert, die Verbindungen aber zu langsam sind, sollten Sie tuning(7) lesen. Prüfen Sie auch die Netzwerkkonfiguration, da falsche Einstellungen die Ursache für langsame Verbindungen sein können.

Wenn Sie viele device timeout Meldungen in den Systemprotokollen finden, prüfen Sie, dass es keinen Konflikt zwischen der Netzwerkkarte und anderen Geräten des Systems gibt. Überprüfen Sie nochmals die Verkabelung. Unter Umständen benötigen Sie eine andere Netzwerkkarte.

Bei watchdog timeout Fehlermeldungen, kontrollieren Sie zuerst die Verkabelung. Überprüfen Sie dann, ob der PCI-Steckplatz der Karte Bus Mastering unterstützt. Auf einigen älteren Motherboards ist das nur für einen Steckplatz (meistens Steckplatz 0) der Fall. Lesen Sie in der Dokumentation der Karte und des Motherboards nach, ob das vielleicht die Ursache des Problems sein könnte.

Die Meldung No route to host erscheint, wenn das System ein Paket nicht zustellen kann. Das kann vorkommen weil beispielsweise keine Default-Route gesetzt wurde oder das Netzwerkkabel nicht richtig steckt. Schauen Sie in der Ausgabe von `netstat -rn` nach, ob eine gültige Route zu dem Zielsystem existiert. Wenn nicht, lesen Sie Kapitel 31, *Weiterführende Netzwerkthemen*.

Die Meldung ping: sendto: Permission denied wird oft von einer falsch konfigurierten Firewall verursacht. Wenn keine Regeln definiert wurden, blockiert eine aktivierte Firewall alle Pakete, selbst einfache ping(8)-Pakete. Weitere Informationen erhalten Sie in Kapitel 30, *Firewalls*.

Falls die Leistung der Karte schlecht ist, setzen Sie die Medienerkennung von `autoselect` (automatisch) auf das richtige Medium. In vielen Fällen löst diese Maßnahme Leistungsprobleme. Wenn nicht, prüfen Sie nochmal die Netzwerkeinstellungen und lesen Sie tuning(7).

12.6. Virtual Hosts

Ein gebräuchlicher Zweck von FreeBSD ist das virtuelle Hosting, bei dem ein Server im Netzwerk wie mehrere Server aussieht. Dies wird dadurch erreicht, dass einem Netzwerkinterface mehrere Netzwerk-Adressen zugewiesen werden.

Ein Netzwerkinterface hat eine „echte“ Adresse und kann beliebig viele „alias“ Adressen haben. Die Aliase werden durch entsprechende alias Einträge in `/etc/rc.conf` festgelegt, wie in diesem Beispiel zu sehen ist:

```
ifconfig_fxp0_alias0="inet xxx.xxx.xxx.xxx netmask xxx.xxx.xxx.xxx"
```

Beachten Sie, dass die Alias-Einträge mit `alias0` anfangen müssen und weiter hochgezählt werden, das heißt `alias1`, `alias2`, und so weiter. Die Konfiguration der Aliase hört bei der ersten fehlenden Zahl auf.

Die Berechnung der Alias-Netzwerkmasken ist wichtig. Für jedes Interface muss es eine Adresse geben, die die Netzwerkmaske des Netzwerkes richtig beschreibt. Alle anderen Adressen in diesem Netzwerk haben dann eine Netzwerkmaske, die mit 1 gefüllt ist, also `255.255.255.255` oder hexadezimal `0xffffffff`.

Als Beispiel betrachten wir den Fall, in dem `fxp0` mit zwei Netzwerken verbunden ist: dem Netzwerk `10.1.1.0` mit der Netzwerkmaske `255.255.255.0` und dem Netzwerk `202.0.75.16` mit der Netzwerkmaske `255.255.255.240`. Das System soll die Adressen `10.1.1.1` bis `10.1.1.5` und `202.0.75.17` bis `202.0.75.20` belegen. Nur die erste Adresse in einem Netzwerk sollte die richtige Netzwerkmaske haben. Alle anderen Adressen (`10.1.1.2` bis `10.1.1.5` und `202.0.75.18` bis `202.0.75.20`) müssen die Maske `255.255.255.255` erhalten.

Die folgenden Einträge in `/etc/rc.conf` konfigurieren den Adapter entsprechend dem Beispiel:

```
ifconfig_fxp0="inet 10.1.1.1 netmask 255.255.255.0"
ifconfig_fxp0_alias0="inet 10.1.1.2 netmask 255.255.255.255"
ifconfig_fxp0_alias1="inet 10.1.1.3 netmask 255.255.255.255"
ifconfig_fxp0_alias2="inet 10.1.1.4 netmask 255.255.255.255"
ifconfig_fxp0_alias3="inet 10.1.1.5 netmask 255.255.255.255"
ifconfig_fxp0_alias4="inet 202.0.75.17 netmask 255.255.255.240"
ifconfig_fxp0_alias5="inet 202.0.75.18 netmask 255.255.255.255"
ifconfig_fxp0_alias6="inet 202.0.75.19 netmask 255.255.255.255"
ifconfig_fxp0_alias7="inet 202.0.75.20 netmask 255.255.255.255"
```

12.7. Konfiguration des syslogd Servers

Beigetragen von Niclas Zeising.

Das Aufzeichnen von Log-Meldungen ist ein wichtiger Aspekt der Systemadministration. Es wird nicht nur verwendet um Hard- und Softwarefehler ausfindig zu machen, auch zur Überwachung der Sicherheit und der Reaktion bei einem Zwischenfall spielen diese Aufzeichnungen eine wichtige Rolle. Systemdienste ohne kontrollierendes Terminal senden Meldungen in der Regel an einen Log-Server, oder schreiben sie in eine Logdatei.

Dieser Abschnitt beschreibt die Konfiguration und Verwendung des FreeBSD syslogd(8) Servers, und diskutiert auch die Log-Rotation und das Management von Logdateien mit newsyslog(8). Der Fokus wird hierbei auf die Einrichtung und Benutzung eines syslogd(8) auf dem lokalen Rechner gelegt. Für erweiterte Einstellungen und die Verwendung eines separaten Log-Servers lesen Sie bitte Abschnitt 29.12, „Protokollierung von anderen Hosts mit syslogd“.

12.7.1. Verwendung von `syslogd`

In der Standardkonfiguration von FreeBSD wird syslogd(8) beim Booten automatisch gestartet. Dieses Verhalten wird über die Variable `syslogd_enable` in `/etc/rc.conf` gesteuert. Dazu gibt es noch zahlreiche Argumente, die das Verhalten von syslogd(8) beeinflussen. Benutzen Sie zum verändern dieser Argumente `syslogd_flags` in `/etc/rc.conf`. Lesen Sie syslogd(8) für weitere Informationen über die Argumente, und rc.conf(5), Abschnitt 12.4.1, „Systemspezifische Konfiguration“ und Abschnitt 12.4, „Dienste unter FreeBSD verwalten“ wenn Sie mehr über `/etc/rc.conf` und das rc(8)-Subsystem wissen möchten.

12.7.2. Konfiguration von `syslogd`

Die Konfigurationsdatei `/etc/syslog.conf` steuert, was syslogd(8) mit Log-Meldungen macht, sobald sie empfangen werden. Es gibt verschiedene Parameter, die das Verhalten bei eingehenden Ereignissen kontrollieren. Zu den grundlegenden gehören *facility* und *level*. *facility* beschreibt das Subsystem, welches das Ereignis generiert hat. Beispielsweise der Kernel, oder ein Daemon. *level* hingegen beschreibt den Schweregrad des aufgetretenen Ereignisses. Dies macht es möglich, Meldungen in verschiedenen Logdateien zu protokollieren, oder Meldungen zu verwerfen, je nach Konfiguration von *facility* und *level*. Ebenfalls besteht die Möglichkeit auf Meldungen zu reagieren, die von einer bestimmten Anwendung stammen, oder von einem spezifischen Host erzeugt wurden.

Die Konfigurationsdatei von syslogd(8) enthält für jede Aktion eine Zeile. Die Syntax besteht aus einem Auswahlfeld, gefolgt von einem Aktionsfeld. Die Syntax für das Auswahlfeld ist *`facility.level`*. Dies entspricht Log-Meldungen von *`facility`* mit einem Level von *`level`* oder höher. Um noch präziser festzulegen was protokolliert wird, kann dem Level optional ein Vergleichsflag vorangestellt werden. Mehrere Auswahlen können, durch Semikolon (`;`) getrennt, für die gleiche Aktion verwendet werden. `*` wählt dabei alles aus. Das Aktionsfeld definiert, wohin die Log-Meldungen gesendet werden, beispielsweise in eine Datei oder zu einem entfernten Log-Server. Als Beispiel dient hier `/etc/syslog.conf` aus FreeBSD:

```
# $FreeBSD$
#
#       Spaces ARE valid field separators in this file. However,
#       other *nix-like systems still insist on using tabs as field
#       separators. If you are sharing this file between systems, you$
#       may want to use only tabs as field separators here.
#       Consult the syslog.conf(5) manpage.
*.err;kern.warning;auth.notice;mail.crit                /dev/console ❶
*.notice;authpriv.none;kern.debug;lpr.info;mail.crit;news.err   /var/log/messages
security.*                                      /var/log/security
auth.info;authpriv.info                         /var/log/auth.log
mail.info                                       /var/log/maillog ❷
lpr.info                                        /var/log/lpd-errs
ftp.info                                        /var/log/xferlog
cron.*                                          /var/log/cron
!-devd
*.=debug                                        /var/log/debug.log ❸
*.emerg                                         *
# uncomment this to log all writes to /dev/console to /var/log/console.log
#console.info                                   /var/log/console.log
# uncomment this to enable logging of all log messages to /var/log/all.log
# touch /var/log/all.log and chmod it to mode 600 before it will work
```

```
#*.*                                            /var/log/all.log
# uncomment this to enable logging to a remote loghost named loghost
#*.*                                            @loghost
# uncomment these if you're running inn
# news.crit                                     /var/log/news/news.crit
# news.err                                      /var/log/news/news.err
# news.notice                                   /var/log/news/news.notice
# Uncomment this if you wish to see messages produced by devd
# !devd
# *.>=info
!ppp ❹
*.*                                             /var/log/ppp.log
!*
```

❶ Selektiert alle Meldungen vom Level `err`, sowie `kern.warning` , `auth.notice` und `mail.crit` und schickt diese zur Konsole (`/dev/console`).

❷ Selektiert alle Meldungen von `mail` ab dem Level `info` oder höher und schreibt diese in `/var/log/maillog` .

❸ Diese Zeile benutzt das Vergleichsflag `=`, um nur Meldungen vom Level `debug` zu selektieren und schreibt diese in `/var/log/debug.log` .

❹ Hier ist ein Beispiel für die Nutzung einer *Programmspezifikation*. Die nachfolgenden Regeln sind dann nur für Programme gültig, welche der Programmspezifikation stehen. In diesem Fall landen alle Meldungen von `ppp` (und keinem anderen Programm) in `/var/log/ppp.log` .

Dieses Beispiel zeigt, dass es jede Menge Level und Subsysteme gibt. Die Level, beginnend mit den höchst kritischen, hin zu den weniger kritischen, sind: `emerg`, `alert`, `crit`, `err`, `warning`, `notice`, `info` und `debug`.

Die *facilities*, in beliebiger Reihenfolge, sind: `auth`, `authpriv`, `console`, `cron`, `daemon`, `ftp`, `kern`, `lpr`, `mail`, `mark`, `news`, `security`, `syslog`, `user`, `uucp`, sowie `local0` bis `local7`. Beachten Sie, dass andere Betriebssysteme hiervon abweichende *facilities* haben können.

Mit diesem Wissen ist es nun einfach, eine weitere Zeile in `/etc/syslog.conf` hinzuzufügen, welche alle Meldungen von den unterschiedlichsten Dämonen mit einem Level von `notice` und höher in `/var/log/daemon.log` . Fügen Sie einfach folgendes hinzu:

```
daemon.notice                                   /var/log/daemon.log
```

Für weitere Informationen zu verschiedenen Level und *faclilities*, lesen Sie syslog(3) und syslogd(8). Weitere Informationen zu `/etc/syslog.conf` , dessen Syntax und erweiterten Anwendungsbeispielen, finden Sie in syslog.conf(5) und Abschnitt 29.12, „Protokollierung von anderen Hosts mit `syslogd`“.

12.7.3. Log-Management und Rotation mit `newsyslog`

Log-Dateien können schnell wachsen, was viel Speicherplatz verbrauchen kann. Zudem wird es schwieriger, nützliche Informationen schnell zu finden. Log-Management versucht, diesen Effekt zu mildern. FreeBSD verwendet newsyslog(8) für die Verwaltung von Log-Dateien. Dieses Programm rotiert und komprimiert in regelmäßigen Abständen Log-Dateien. Optional kann es auch fehlende Log-Dateien erstellen und Programme benachrichtigen, wenn Log-Dateien verschoben wurden. Dabei müssen die Log-Dateien nicht unbedingt von syslogd(8) stammen, newsyslog(8) ist auch in der Lage, Nachrichten von anderen Programmen zu verarbeiten. Obwohl newsyslog(8) normalerweise von cron(8) aufgerufen wird, ist es kein Systemdämon. In der Standardkonfiguration wird dieser Job jede Stunde ausgeführt.

12.7.3.1. Konfiguration von `newsyslog`

Um zu wissen, welche Maßnahmen zu ergreifen sind, liest newsyslog(8) seine Konfigurationsdatei, standardmäßig `/etc/newsyslog.conf` . Diese Konfigurationsdatei enthält eine Zeile für jede Datei, die von newsyslog(8) verwaltet wird. Jede Zeile enthält Informationen über den Besitzer der Datei, die Dateiberechtigungen, wann die Datei rotiert wird, optionale *Flags*, welche die Log-Rotation beeinflussen (bspw. Komprimierung) und Programme, denen ein Signal geschickt wird, wenn Log-Dateien rotiert werden. Hier folgt die Standardkonfiguration in FreeBSD:

```
# configuration file for newsyslog
# $FreeBSD$
```

```
#
# Entries which do not specify the '/pid_file' field will cause the
# syslogd process to be signalled when that log file is rotated.  This
# action is only appropriate for log files which are written to by the
# syslogd process (ie, files listed in /etc/syslog.conf).  If there
# is no process which needs to be signalled when a given log file is
# rotated, then the entry for that file should include the 'N' flag.
#
# The 'flags' field is one or more of the letters: BCDGJNUXZ or a '-'.
#
# Note: some sites will want to select more restrictive protections than the
# defaults.  In particular, it may be desirable to switch many of the 644
# entries to 640 or 600.  For example, some sites will consider the
# contents of maillog, messages, and lpd-errs to be confidential.  In the
# future, these defaults may change to more conservative ones.
#
# logfilename          [owner:group]    mode count size when  flags [/pid_file] [sig_num]
/var/log/all.log                        600  7     *    @T00  J
/var/log/amd.log                        644  7     100  *     J
/var/log/auth.log                       600  7     100  @0101T JC
/var/log/console.log                    600  5     100  *     J
/var/log/cron                           600  3     100  *     JC
/var/log/daily.log                      640  7     *    @T00  JN
/var/log/debug.log                      600  7     100  *     JC
/var/log/kerberos.log                   600  7     100  *     J
/var/log/lpd-errs                       644  7     100  *     JC
/var/log/maillog                        640  7     *    @T00  JC
/var/log/messages                       644  5     100  @0101T JC
/var/log/monthly.log                    640  12    *    $M1D0 JN
/var/log/pflog                          600  3     100  *     JB    /var/run/pflogd.pid
/var/log/ppp.log        root:network    640  3     100  *     JC
/var/log/devd.log                       644  3     100  *     JC
/var/log/security                       600  10    100  *     JC
/var/log/sendmail.st                    640  10    *    168   B
/var/log/utx.log                        644  3     *    @01T05 B
/var/log/weekly.log                     640  5     1    $W6D0 JN
/var/log/xferlog                        600  7     100  *     JC
```

Jede Zeile beginnt mit dem Namen der Datei, die rotiert werden soll, optional gefolgt von Besitzer und Gruppe für rotierende, als auch für neu erstellte Dateien. Das Feld `mode` definiert die Zugriffsrechte der Datei. `count` gibt an, wie viele rotierte Dateien aufbewahrt werden sollen. Anhand der `size`- und `when`-*Flags* erkennt newsyslog, wann die Datei rotiert werden muss. Eine Logdatei wird rotiert, wenn ihre Größe den Wert von `size` überschreitet, oder wenn die Zeit im `when`-Feld abgelaufen ist. Ein `*` bedeutet, dass dieses Feld ignoriert wird. Das *`flags`*-Feld gibt newsyslog weitere Instruktionen, zum Beispiel wie eine Datei zu rotieren ist, oder eine Datei zu erstellen falls diese nicht existiert. Die letzten beiden Felder sind optional und bestimmen die PID-Datei sowie wenn die Datei rotiert wird. Weitere Informationen zu allen Feldern, gültigen *flags* und wie Sie die Rotationszeit angeben können, finden Sie in syslog.conf(5). Denken Sie daran, dass newsyslog von `cron` aufgerufen wird und somit Dateien auch nur dann rotiert, wenn es von cron(8) aufgerufen wird, und nicht häufiger.

12.8. Konfigurationsdateien

12.8.1. `/etc` Layout

Konfigurationsdateien finden sich in einigen Verzeichnissen unter anderem in:

`/etc`	Enthält generelle systemspezifische Konfigurationsinformationen.
`/etc/defaults`	Default Versionen der Konfigurationsdateien.
`/etc/mail`	Enthält die sendmail(8) Konfiguration und weitere MTA Konfigurationsdateien.
`/etc/ppp`	Hier findet sich die Konfiguration für die User- und Kernel-ppp Programme.

`/etc/namedb`	Das Vorgabeverzeichnis, in dem Daten von named(8) gehalten werden. Normalerweise werden hier `named.conf` und Zonendaten abgelegt.
`/usr/local/etc`	Installierte Anwendungen legen hier ihre Konfigurationsdateien ab. Dieses Verzeichnis kann Unterverzeichnisse für bestimmte Anwendungen enthalten.
`/usr/local/etc/rc.d`	rc(8)-Skripten installierter Anwendungen.
`/var/db`	Automatisch generierte systemspezifische Datenbanken, wie die Paket-Datenbank oder die locate(1)-Datenbank.

12.8.2. Hostnamen

12.8.2.1. `/etc/resolv.conf`

Wie ein FreeBSD-System auf das *Internet Domain Name System* (DNS) zugreift, wird in `/etc/resolv.conf` festgelegt.

Die gebräuchlichsten Einträge in `/etc/resolv.conf` sind:

`nameserver`	Die IP-Adresse eines Nameservers, den der Resolver abfragen soll. Bis zu drei Server werden in der Reihenfolge, in der sie aufgezählt sind, abgefragt.
`search`	Suchliste mit Domain-Namen zum Auflösen von Hostnamen. Die Liste wird normalerweise durch den Domain-Teil des lokalen Hostnamens festgelegt.
`domain`	Der lokale Domain-Name.

Beispiel für eine typische `/etc/resolv.conf`:

```
search example.com
nameserver 147.11.1.11
nameserver 147.11.100.30
```

Anmerkung

Nur eine der Anweisungen `search` oder `domain` sollte benutzt werden.

Wenn Sie DHCP benutzen, überschreibt dhclient(8) für gewöhnlich `/etc/resolv.conf` mit den Informationen vom DHCP-Server.

12.8.2.2. `/etc/hosts`

`/etc/hosts` ist eine einfache textbasierte Datenbank. Zusammen mit DNS und NIS stellt sie eine Abbildung zwischen Namen und IP-Adressen zur Verfügung. Anstatt named(8) zu konfigurieren, können hier lokale Rechner, die über ein LAN verbunden sind, eingetragen werden. Lokale Einträge für gebräuchliche Internet-Adressen in `/etc/hosts` verhindern die Abfrage eines externen Servers und beschleunigen die Namensauflösung.

```
# $FreeBSD$
#
#
# Host Database
#
# This file should contain the addresses and aliases for local hosts that
# share this file.  Replace 'my.domain' below with the domainname of your
# machine.
#
# In the presence of the domain name service or NIS, this file may
# not be consulted at all; see /etc/nsswitch.conf for the resolution order.
```

```
#
#
::1         localhost localhost.my.domain
127.0.0.1       localhost localhost.my.domain
#
# Imaginary network.
#10.0.0.2       myname.my.domain myname
#10.0.0.3       myfriend.my.domain myfriend
#
# According to RFC 1918, you can use the following IP networks for
# private nets which will never be connected to the Internet:
#
#   10.0.0.0    -   10.255.255.255
#   172.16.0.0  -   172.31.255.255
#   192.168.0.0 -   192.168.255.255
#
# In case you want to be able to connect to the Internet, you need
# real official assigned numbers.  Do not try to invent your own network
# numbers but instead get one from your network provider (if any) or
# from your regional registry (ARIN, APNIC, LACNIC, RIPE NCC, or AfriNIC.)
#
```

`/etc/hosts` hat das folgende Format:

```
[Internet Adresse] [Offizieller Hostname] [Alias1] [Alias2] ...
```

Zum Beispiel:

```
10.0.0.1 myRealHostname.example.com myRealHostname foobar1 foobar2
```

Weitere Informationen entnehmen Sie bitte hosts(5).

12.9. Einstellungen mit sysctl(8)

Mit sysctl(8) können Sie Änderungen an einem laufenden FreeBSD-System vornehmen. Unter anderem können Optionen des TCP/IP-Stacks oder des virtuellen Speichermanagements verändert werden. Unter der Hand eines erfahrenen Systemadministrators kann dies die Systemperformance erheblich verbessern. Über 500 Variablen können mit sysctl(8) gelesen und gesetzt werden.

Der Hauptzweck von sysctl(8) besteht darin, Systemeinstellungen zu lesen und zu verändern.

Alle auslesbaren Variablen werden wie folgt angezeigt:

```
% sysctl -a
```

Um eine spezielle Variable zu lesen, geben Sie den Namen an:

```
% sysctl kern.maxproc
kern.maxproc: 1044
```

Um eine Variable zu setzen, benutzen Sie die Syntax *`Variable`*=*`Wert`*:

```
# sysctl kern.maxfiles=5000
kern.maxfiles: 2088 -> 5000
```

Mit sysctl können Strings, Zahlen oder Boolean-Werte gesetzt werden. Bei Boolean-Werten steht `1` für wahr und `0` für falsch.

Um die Variablen automatisch während des Systemstarts zu setzen, fügen Sie sie in `/etc/sysctl.conf` ein. Weitere Informationen finden Sie in der Hilfeseite sysctl.conf(5) und in Abschnitt 12.9.1, „`sysctl.conf`“.

12.9.1. `sysctl.conf`

`/etc/sysctl.conf` sieht ähnlich wie `/etc/rc.conf` aus. Werte werden in der Form `Variable=Wert` gesetzt. Die angegebenen Werte werden gesetzt, nachdem sich das System bereits im Mehrbenutzermodus befindet. Allerdings lassen sich im Mehrbenutzermodus nicht alle Werte setzen.

Um das Protokollieren von fatalen Signalen abzustellen und Benutzer daran zu hindern, von anderen Benutzern gestartete Prozesse zu sehen, können Sie in `/etc/sysctl.conf` die folgenden Variablen setzen:

```
# Do not log fatal signal exits (e.g. sig 11)
kern.logsigexit=0

# Prevent users from seeing information about processes that
# are being run under another UID.
security.bsd.see_other_uids=0
```

12.9.2. Schreibgeschützte Variablen

Contributed by Tom Rhodes.

Wenn schreibgeschützte sysctl(8)-Variablen verändert werden, ist ein Neustart des Systems erforderlich.

Beispielsweise hat cardbus(4) auf einigen Laptops Schwierigkeiten, Speicherbereiche zu erkennen. Es treten dann Fehlermeldungen wie die folgende auf:

```
cbb0: Could not map register memory
device_probe_and_attach: cbb0 attach returned 12
```

Um dieses Problem zu lösen, muss eine schreibgeschützte sysctl(8)-Variable verändert werden. Fügen Sie `hw.pci.allow_unsupported_io_range=1` in `/boot/loader.conf` hinzu und starten Sie das System neu. Danach sollte cardbus(4) fehlerfrei funktionieren.

12.10. Tuning von Laufwerken

Der folgende Abschnitt beschreibt die verschiedenen Methoden zur Feinabstimmung der Laufwerke. Oft sind mechanische Teile in Laufwerken, wie SCSI-Laufwerke, verbaut. Diese können einen Flaschenhals bei der Gesamtleistung des Systems darstellen. Sie können zwar auch ein Laufwerk ohne mechanische Teile einbauen, wie z.B. ein *Solid-State-Drive*, aber Laufwerke mit mechanischen Teilen werden auch in naher Zukunft nicht vom Markt verschwinden. Bei der Feinabstimmung ist es ratsam, die Funktionen von iostat(8) zu verwenden, um verschiedene Änderungen zu testen und um nützliche IO-Informationen des Systems zu erhalten.

12.10.1. Sysctl Variablen

12.10.1.1. `vfs.vmiodirenable`

Die sysctl(8)-Variable `vfs.vmiodirenable` besitzt in der Voreinstellung den Wert `1`. Die Variable kann auf den Wert `0` (deaktiviert) oder `1` (aktiviert) gesetzt werden. Sie steuert, wie Verzeichnisse vom System zwischengespeichert werden. Die meisten Verzeichnisse sind klein und benutzen nur ein einzelnes Fragment, typischerweise 1 kB, im Dateisystem und 512 Bytes im Buffer-Cache. Ist die Variable deaktiviert, wird der Buffer-Cache nur eine limitierte Anzahl Verzeichnisse zwischenspeichern, auch wenn das System über sehr viel Speicher verfügt. Ist die Variable aktiviert, kann der Buffer-Cache den VM-Page-Cache benutzen, um Verzeichnisse zwischenzuspeichern. Der ganze Speicher steht damit zum Zwischenspeichern von Verzeichnissen zur Verfügung. Der Nachteil bei dieser Vorgehensweise ist, dass zum Zwischenspeichern eines Verzeichnisses mindestens eine physikalische Seite im Speicher, die normalerweise 4 kB groß ist, anstelle von 512 Bytes gebraucht wird. Es wird empfohlen, diese Option aktiviert zu lassen, wenn Sie Dienste zur Verfügung stellen, die viele Dateien manipulieren. Beispiele für solche Dienste sind Web-Caches, große Mail-Systeme oder Netnews. Die aktivierte Variable vermindert, trotz des verschwendeten Speichers, in aller Regel nicht die Leistung des Systems, obwohl Sie das nachprüfen sollten.

12.10.1.2. `vfs.write_behind`

In der Voreinstellung besitzt die sysctl(8)-Variable `vfs.write_behind` den Wert `1` (aktiviert). Mit dieser Einstellung schreibt das Dateisystem anfallende vollständige Cluster, die besonders beim sequentiellen Schreiben großer Dateien auftreten, direkt auf das Medium aus. Dies verhindert, dass sich im Buffer-Cache veränderte Puffer (*dirty buffers*) ansammeln, die die I/O-Verarbeitung nicht mehr beschleunigen würden. Unter bestimmten Umständen blockiert diese Funktion allerdings Prozesse. Setzen Sie in diesem Fall die Variable `vfs.write_behind` auf den Wert `0`.

12.10.1.3. `vfs.hirunningspace`

Die sysctl(8)-Variable `vfs.hirunningspace` bestimmt systemweit die Menge ausstehender Schreiboperationen, die dem Platten-Controller zu jedem beliebigen Zeitpunkt übergeben werden können. Normalerweise können Sie den Vorgabewert verwenden. Auf Systemen mit vielen Platten kann der Wert aber auf 4 bis 5 *Megabyte* erhöht werden. Ein zu hoher Wert (größer als der Schreib-Schwellwert des Buffer-Caches) kann zu Leistungverlusten führen. Setzen Sie den Wert daher nicht zu hoch! Hohe Werte können auch Leseoperationen verzögern, die gleichzeitig mit Schreiboperationen ausgeführt werden.

Es gibt weitere sysctl(8)-Variablen, mit denen Sie den Buffer-Cache und den VM-Page-Cache beeinflussen können. Es wird nicht empfohlen, diese Variablen zu verändern, da das VM-System den virtuellen Speicher selbst sehr gut verwaltet.

12.10.1.4. `vm.swap_idle_enabled`

Die sysctl(8)-Variable `vm.swap_idle_enabled` ist für große Mehrbenutzer-Systeme gedacht, auf denen sich viele Benutzer an- und abmelden und auf denen es viele Prozesse im Leerlauf (*idle*) gibt. Solche Systeme fragen kontinuierlich freien Speicher an. Wenn Sie die Variable `vm.swap_idle_enabled` aktivieren, können Sie die Auslagerungs-Hysterese von Seiten mit den Variablen `vm.swap_idle_threshold1` und `vm.swap_idle_threshold2` einstellen. Die Schwellwerte beider Variablen geben die Zeit in Sekunden an, in denen sich ein Prozess im Leerlauf befinden muss. Wenn die Werte so eingestellt sind, dass Seiten früher als nach dem normalen Algorithmus ausgelagert werden, verschafft das dem Auslagerungs-Prozess mehr Luft. Aktivieren Sie diese Funktion nur, wenn Sie sie wirklich benötigen: Die Speicherseiten werden eher früher als später ausgelagert. Der Platz im Swap-Bereich wird dadurch schneller verbraucht und die Plattenaktivitäten steigen an. Auf kleinen Systemen hat diese Funktion spürbare Auswirkungen. Auf großen Systemen, die sowieso schon Seiten auslagern müssen, können ganze Prozesse leichter in den Speicher geladen oder ausgelagert werden.

12.10.1.5. `hw.ata.wc`

Obwohl das Abstellen des IDE-Schreib-Zwischenspeichers die Bandbreite zum Schreiben auf die IDE-Festplatte verringert, kann es aus Gründen der Datenkonsistenz als notwendig angesehen werden. Das Problem ist, dass IDE-Platten keine zuverlässige Aussage über das Ende eines Schreibvorgangs treffen. Wenn der Schreib-Zwischenspeicher aktiviert ist, werden die Daten nicht in der Reihenfolge ihres Eintreffens geschrieben. Es kann sogar passieren, dass das Schreiben mancher Blöcke im Fall von starker Plattenaktivität auf unbefristete Zeit verzögert wird. Ein Absturz oder Stromausfall zu dieser Zeit kann die Dateisysteme erheblich beschädigen. Sie sollten den Wert der sysctl(8)-Variable `hw.ata.wc` auf dem System überprüfen. Wenn der Schreib-Zwischenspeicher abgestellt ist, können Sie ihn beim Systemstart aktivieren, indem Sie die Variable in `/boot/loader.conf` auf den Wert `1` setzen.

Weitere Informationen finden Sie in ata(4).

12.10.1.6. `SCSI_DELAY` (`kern.cam.scsi_delay`)

Mit der Kerneloption `SCSI_DELAY` kann die Dauer des Systemstarts verringert werden. Der Vorgabewert ist recht hoch und er verzögert den Systemstart um `15` oder mehr Sekunden. Normalerweise kann dieser Wert, insbesondere mit modernen Laufwerken, mit der sysctl(8)-Variable `kern.cam.scsi_delay` auf `5` Sekunden heruntergesetzt werden. Die Variable sowie die Kerneloption verwenden für die Zeitangabe Millisekunden und *nicht* Sekunden.

12.10.2. Soft Updates

Mit tunefs(8) lassen sich Feineinstellungen an Dateisystemen vornehmen. Das Programm hat verschiedene Optionen. Soft Updates werden wie folgt ein- und ausgeschaltet:

```
# tunefs -n enable /filesystem
# tunefs -n disable /filesystem
```

Ein eingehängtes Dateisystem kann nicht mit tunefs(8) modifiziert werden. Soft Updates werden am besten im Single-User Modus aktiviert, bevor Partitionen eingehangen sind.

Durch Einsatz eines Zwischenspeichers wird die Performance im Bereich der Metadaten, vorwiegend beim Anlegen und Löschen von Dateien, gesteigert. Es wird empfohlen, Soft Updates auf allen UFS-Dateisystemen zu aktivieren. Allerdings sollten Sie sich über die zwei Nachteile von Soft Updates bewusst sein: Erstens garantieren Soft Updates zwar die Konsistenz der Daten im Fall eines Absturzes, aber es kann passieren, dass das Dateisystem über mehrere Sekunden oder gar eine Minute nicht synchronisiert wurde. Nicht geschriebene Daten gehen dann vielleicht verloren. Zweitens verzögern Soft Updates die Freigabe von Datenblöcken. Eine größere Aktualisierung eines fast vollen Dateisystems, wie dem Root-Dateisystem, z.B. während eines `make installworld`, kann das Dateisystem vollaufen lassen. Dadurch würde die Aktualisierung fehlschlagen.

12.10.2.1. Details über Soft Updates

Bei einem Metadaten-Update werden die Inodes und Verzeichniseinträge aktualisiert auf die Platte zurückgeschrieben. Es gibt zwei klassische Ansätze, um die Metadaten des Dateisystems auf die Platte zu schreiben.

Das historisch übliche Verfahren waren synchrone Updates der Metadaten, d. h. wenn eine Änderung an einem Verzeichnis nötig war, wurde anschließend gewartet, bis diese Änderung tatsächlich auf die Platte zurückgeschrieben worden war. Der *Inhalt* der Dateien wurde im „Buffer Cache“ zwischengespeichert und später asynchron auf die Platte geschrieben. Der Vorteil dieser Implementierung ist, dass sie sicher funktioniert. Wenn während eines Updates ein Ausfall erfolgt, haben die Metadaten immer einen konsistenten Zustand. Eine Datei ist entweder komplett angelegt oder gar nicht. Wenn die Datenblöcke einer Datei im Fall eines Absturzes noch nicht den Weg aus dem „Buffer Cache“ auf die Platte gefunden haben, kann fsck(8) das Dateisystem reparieren, indem es die Dateilänge einfach auf `0` setzt. Außerdem ist die Implementierung einfach und überschaubar. Der Nachteil ist, dass Änderungen der Metadaten sehr langsam vor sich gehen. Ein `rm -r` beispielsweise fasst alle Dateien eines Verzeichnisses der Reihe nach an, aber jede dieser Änderungen am Verzeichnis (Löschen einer Datei) wird einzeln synchron auf die Platte geschrieben. Gleiches beim Auspacken großer Hierarchien mit `tar -x`.

Der zweite Ansatz sind asynchrone Metadaten-Updates. Das ist der Standard, wenn UFS-Dateisysteme mit `mount -o async` eingehängt werden. Man schickt die Updates der Metadaten einfach auch noch über den „Buffer Cache“, sie werden also zwischen die Updates der normalen Daten eingeschoben. Vorteil ist, dass man nun nicht mehr auf jeden Update warten muss, Operationen, die zahlreiche Metadaten ändern, werden also viel schneller. Auch hier ist die Implementierung sehr einfach und wenig anfällig für Fehler. Nachteil ist, dass keinerlei Konsistenz des Dateisystems mehr gesichert ist. Wenn mitten in einer Operation, die viele Metadaten ändert, ein Ausfall erfolgt (Stromausfall, drücken des Reset-Schalters), dann ist das Dateisystem anschließend in einem unbestimmten Zustand. Niemand kann genau sagen, was noch geschrieben worden ist und was nicht mehr; die Datenblöcke einer Datei können schon auf der Platte stehen, während die inode Tabelle oder das zugehörige Verzeichnis nicht mehr aktualisiert worden ist. Man kann praktisch kein fsck(8) mehr implementieren, das diesen Zustand wieder reparieren kann, da die dazu nötigen Informationen einfach auf der Platte fehlen. Wenn ein Dateisystem irreparabel beschädigt wurde, hat man nur noch die Möglichkeit es neu zu erzeugen und die Daten vom Backup zurückspielen.

Der Ausweg aus diesem Dilemma ist ein *dirty region logging*, was auch als *Journalling* bezeichnet wird. Man schreibt die Metadaten-Updates zwar synchron, aber nur in einen kleinen Plattenbereich, die *logging area*. Von da aus werden sie dann asynchron auf ihre eigentlichen Bereiche verteilt. Da die *logging area* ein kleines zusammenhängendes Stückchen ist, haben die Schreibköpfe der Platte bei massiven Operationen auf Metadaten keine allzu großen Wege zurückzulegen, so dass alles ein ganzes Stück schneller geht als bei klassischen synchronen Updates. Die Komplexität der Implementierung hält sich ebenfalls in Grenzen, somit auch die Anfälligkeit für Fehler. Als Nachteil ergibt sich, dass Metadaten zweimal auf die Platte geschrieben werden müssen (einmal in die *logging area*, einmal an die richtige Stelle), so dass das im Falle regulärer Arbeit (also keine gehäuften Metadatenoperationen) eine „Pessimisierung“ des Falls der synchronen Updates eintritt, es wird alles langsamer. Dafür hat man als Vorteil, dass im Falle eines Crashes der konsistente Zustand dadurch erzielbar ist, dass die angefangenen Operationen aus dem *dirty region log* entweder zu Ende ausgeführt oder komplett verworfen werden, wodurch das Dateisystem schnell wieder zur Verfügung steht.

Die Lösung von Kirk McKusick, dem Schöpfer von Berkeley FFS, waren *Soft Updates*: die notwendigen Updates der Metadaten werden im Speicher gehalten und dann sortiert auf die Platte geschrieben („ordered metadata updates“). Dadurch hat man den Effekt, dass im Falle massiver Metadaten-Änderungen spätere Operationen die vorhergehenden, noch nicht auf die Platte geschriebenen Updates desselben Elements im Speicher „einholen“. Alle Operationen, auf ein Verzeichnis beispielsweise, werden also in der Regel noch im Speicher abgewickelt, bevor der Update überhaupt auf die Platte geschrieben wird (die dazugehörigen Datenblöcke werden natürlich auch so sortiert, dass sie nicht vor ihren Metadaten auf der Platte sind). Im Fall eines Absturzes hat man ein implizites „log rewind“: alle Operationen, die noch nicht den Weg auf die Platte gefunden haben, sehen danach so aus, als hätten sie nie stattgefunden. Man hat so also den konsistenten Zustand von ca. 30 bis 60 Sekunden früher sichergestellt. Der verwendete Algorithmus garantiert dabei, dass alle tatsächlich benutzten Ressourcen auch in den entsprechenden Bitmaps (Block- und inode Tabellen) als belegt markiert sind. Der einzige Fehler, der auftreten kann, ist, dass Ressourcen noch als „belegt“ markiert sind, die tatsächlich „frei“ sind. fsck(8) erkennt dies und korrigiert diese nicht mehr belegten Ressourcen. Die Notwendigkeit eines Dateisystem-Checks darf aus diesem Grunde auch ignoriert und das Dateisystem mittels `mount -f` zwangsweise eingebunden werden. Um noch allozierte Ressourcen freizugeben muss später ein fsck(8) nachgeholt werden. Das ist dann auch die Idee des *background fsck*: beim Starten des Systems wird lediglich ein *Schnappschuss* des Filesystems gemacht, mit dem fsck(8) dann später arbeiten kann. Alle Dateisysteme dürfen „unsauber“ eingebunden werden und das System kann sofort in den Multiuser-Modus gehen. Danach wird ein Hintergrund-fsck(8) für die Dateisysteme gestartet, die dies benötigen, um möglicherweise irrtümlich belegte Ressourcen freizugeben. Dateisysteme ohne *Soft Updates* benötigen natürlich immer noch den üblichen Vordergrund-fsck(8), bevor sie eingebunden werden können.

Der Vorteil ist, dass die Metadaten-Operationen beinahe so schnell ablaufen wie im asynchronen Fall, also auch schneller als beim *logging*, das die Metadaten immer zweimal schreiben muss. Als Nachteil stehen dem die Komplexität des Codes, ein erhöhter Speicherverbrauch und einige spezielle Eigenheiten entgegen. Nach einem Absturz ist ein etwas „älterer“ Stand auf der Platte – statt einer leeren, aber bereits angelegten Datei, wie nach einem herkömmlichen fsck(8) Lauf, ist auf einem Dateisystem mit *Soft Updates* keine Spur der entsprechenden Datei mehr zu sehen, da weder die Metadaten noch der Dateiinhalt je auf die Platte geschrieben wurden. Weiterhin kann der Platz nach einem rm(1) nicht sofort wieder als verfügbar markiert werden, sondern erst dann, wenn der Update auch auf die Platte vermittelt worden ist. Dies kann besonders dann Probleme bereiten, wenn große Datenmengen in einem Dateisystem installiert werden, das nicht genügend Platz hat, um alle Dateien zweimal unterzubringen.

12.11. Einstellungen von Kernel Limits

12.11.1. Datei und Prozeß Limits

12.11.1.1. `kern.maxfiles`

Abhängig von den Anforderungen an das System kann die sysctl(8)-Variable `kern.maxfiles` erhöht oder gesenkt werden. Die Variable legt die maximale Anzahl von Dateideskriptoren auf dem System fest. Wenn die Dateideskriptoren aufgebraucht sind, werden Sie die Meldung file: table is full wiederholt im Puffer für Systemmeldungen sehen. Den Inhalt des Puffers können Sie sich mit dmesg(8) anzeigen lassen.

Jede offene Datei, jedes Socket und jede FIFO verbraucht einen Dateideskriptor. Auf „dicken“ Produktionsservern können leicht Tausende Dateideskriptoren benötigt werden, abhängig von der Art und Anzahl der gleichzeitig laufenden Dienste.

In älteren FreeBSD-Versionen wurde die Voreinstellung von `kern.maxfile` aus der Kernelkonfigurationsoption `maxusers` bestimmt. `kern.maxfiles` wächst proportional mit dem Wert von `maxusers` . Wenn Sie einen angepassten Kernel kompilieren, empfiehlt es sich diese Option entsprechend der maximalen Benutzerzahl des Systems einzustellen. Obwohl auf einer Produktionsmaschine vielleicht nicht 256 Benutzer gleichzeitig angemeldet sind, können die benötigten Ressourcen ähnlich hoch wie bei einem großen Webserver sein.

Die nur lesbare sysctl(8)-Variable `kern.maxusers` wird beim Systemstart automatisch aus dem zur Verfügung stehenden Hauptspeicher bestimmt. Im laufenden Betrieb kann dieser Wert aus `kern.maxusers` ermittelt werden.

Einige Systeme benötigen für diese Variable einen anderen Wert, wobei 64, 128 und 256 gewöhnliche Werte darstellen. Es wird nicht empfohlen, die Anzahl der Dateideskriptoren auf einen Wert größer 256 zu setzen, es sei denn, Sie benötigen wirklich eine riesige Anzahl von ihnen. Viele der von kern.maxusers auf einen Standardwert gesetzten Parameter können beim Systemstart oder im laufenden Betrieb in /boot/loader.conf angepasst werden. In loader.conf(5) und /boot/defaults/loader.conf finden Sie weitere Details und Hinweise.

Ältere FreeBSD-Versionen setzen diesen Wert selbst, wenn Sie in der Konfigurationsdatei den Wert 0 [1] angeben. Wenn Sie den Wert selbst bestimmen wollen, sollten Sie maxusers mindestens auf 4 setzen. Dies gilt insbesondere dann, wenn Sie beabsichtigen, Xorg zu benutzen oder Software zu kompilieren. Der wichtigste Wert, der durch maxusers bestimmt wird, die maximale Anzahl an Prozessen ist, die auf 20 + 16 * maxusers gesetzt wird. Wird maxusers auf 1 setzen, können gleichzeitig nur 36 Prozesse laufen, von denen ungefähr 18 schon beim Booten des Systems gestartet werden. Dazu kommen nochmals etwa 15 Prozesse beim Start von Xorg. Selbst eine einfache Aufgabe wie das Lesen einer Manualpage benötigt neun Prozesse zum Filtern, Dekomprimieren und Betrachten der Datei. Für die meisten Benutzer sollte es ausreichen, maxusers auf 64 zu setzen, womit 1044 gleichzeitige Prozesse zur Verfügung stehen. Wenn Sie allerdings den Fehler proc table full beim Start eines Programms oder auf einem Server mit einer großen Benutzerzahl sehen, dann sollten Sie den Wert nochmals erhöhen und den Kernel neu bauen.

Anmerkung

Die Anzahl der Benutzer, die sich auf einem Rechner anmelden kann, wird durch maxusers *nicht* begrenzt. Der Wert dieser Variablen legt neben der möglichen Anzahl der Prozesse eines Benutzers weitere sinnvolle Größen für bestimmte Systemtabellen fest.

12.11.1.2. kern.ipc.somaxconn

Die sysctl(8)-Variable kern.ipc.somaxconn beschränkt die Größe der Warteschlange (*Listen-Queue*) für neue TCP-Verbindungen. Der Vorgabewert von 128 ist normalerweise zu klein, um neue Verbindungen auf einem stark ausgelasteten Webserver zuverlässig zu handhaben. Auf solchen Servern sollte der Wert auf 1024 oder höher gesetzt werden. Dienste wie sendmail(8) oder Apache können die Größe der Queue selbst einschränken. Oft gibt es die Möglichkeit, die Größe der Listen-Queue in einer Konfigurationsdatei einzustellen. Eine große Listen-Queue übersteht vielleicht auch einen Denial of Service Angriff (DoS).

12.11.2. Netzwerk Limits

Die Kerneloption NMBCLUSTERS schreibt die Anzahl der Netzwerkpuffer (Mbufs) fest, die das System besitzt. Eine zu geringe Anzahl Mbufs auf einem Server mit viel Netzwerkverkehr verringert die Leistung von FreeBSD. Jeder Mbuf-Cluster nimmt ungefähr 2 kB Speicher in Anspruch, so dass ein Wert von 1024 insgesamt 2 Megabyte Speicher für Netzwerkpuffer im System reserviert. Wie viele Cluster benötigt werden, lässt sich durch eine einfache Berechnung herausfinden. Ein Webserver, der maximal 1000 gleichzeitige Verbindungen servieren soll, wobei jede der Verbindungen einen 6 kB großen Sendepuffer und einen 16 kB großen Empfangspuffer benötigt, braucht ungefähr 32 MB Speicher für Netzwerkpuffer. Als Daumenregel verdoppeln Sie diese Zahl, so dass sich für NMBCLUSTERS der Wert 2x32 MB / 2 kB= 64 MB / 2 kB= 32768 ergibt. Für Maschinen mit viel Speicher werden Werte zwischen 4096 und 32768 empfohlen. Unter keinen Umständen sollten Sie diesen Wert willkürlich erhöhen, da dies zu einem Absturz beim Systemstart führen kann. Verwenden Sie netstat(1) mit -m um den Gebrauch der Netzwerkpuffer zu kontrollieren.

Die Netzwerkpuffer können beim Systemstart mit der Loader-Variablen kern.ipc.nmbclusters eingestellt werden. Nur auf älteren FreeBSD-Systemen müssen Sie die Kerneloption NMBCLUSTERS verwenden.

Die Anzahl der sendfile(2) Puffer muss auf ausgelasteten Servern, die den Systemaufruf sendfile(2) oft verwenden, vielleicht erhöht werden. Dazu können Sie die Kerneloption NSFBUFS verwenden oder die Anzahl der Puffer

[1] Der verwendete Algorithmus setzt maxusers auf die Speichergröße des Systems. Der minimale Wert beträgt dabei 32, das Maximum ist 384.

in `/boot/loader.conf` (siehe loader(8)) setzen. Die Puffer sollten erhöht werden, wenn Sie Prozesse im Zustand `sfbufa` sehen. Die schreibgeschützte sysctl(8)-Variable `kern.ipc.nsfbufs` zeigt die Anzahl eingerichteten Puffer im Kernel. Der Wert dieser Variablen wird normalerweise von `kern.maxusers` bestimmt. Manchmal muss die Pufferanzahl jedoch manuell eingestellt werden.

Wichtig

Auch wenn ein Socket nicht blockierend angelegt wurde, kann der Aufruf von sendfile(2) blockieren, um auf freie `struct sf_buf` Puffer zu warten.

12.11.2.1. `net.inet.ip.portrange.*`

Die sysctl(8)-Variable `net.inet.ip.portrange.*` legt die Portnummern für TCP- und UDP-Sockets fest. Es gibt drei Bereiche: den niedrigen Bereich, den normalen Bereich und den hohen Bereich. Die meisten Netzprogramme benutzen den normalen Bereich. Dieser Bereich umfasst in der Voreinstellung die Portnummern `1024` bis `5000` und wird durch die Variablen `net.inet.ip.portrange.first` und `net.inet.ip.portrange.last` festgelegt. Die festgelegten Bereiche für Portnummern werden von ausgehenden Verbindungen benutzt. Unter bestimmten Umständen, beispielsweise auf stark ausgelasteten Proxy-Servern, sind alle Portnummern für ausgehende Verbindungen belegt. Bereiche für Portnummern spielen auf Servern keine Rolle, die hauptsächlich eingehende Verbindungen verarbeiten (wie ein normaler Webserver) oder nur eine begrenzte Anzahl ausgehender Verbindungen öffnen (beispielsweise ein Mail-Relay). Wenn keine freien Portnummern mehr vorhanden sind, sollte die Variable `net.inet.ip.portrange.last` langsam erhöht werden. Ein Wert von `10000`, `20000` oder `30000` ist angemessen. Beachten Sie auch eine vorhandene Firewall, wenn Sie die Bereiche für Portnummern ändern. Einige Firewalls sperren große Bereiche (normalerweise aus den kleinen Portnummern) und erwarten, dass hohe Portnummern für ausgehende Verbindungen verwendet werden. Daher kann es erforderlich sein, den Wert von `net.inet.ip.portrange.first` zu erhöhen.

12.11.2.2. TCP Bandwidth Delay Product Begrenzung

Die TCP Bandwidth Delay Product Begrenzung wird aktiviert, indem die sysctl(8)-Variable `net.inet.tcp.inflight.enable` auf den Wert `1` gesetzt wird. Das System wird dadurch angewiesen, für jede Verbindung, das Produkt aus der Übertragungsrate und der Verzögerungszeit zu bestimmen. Dieses Produkt begrenzt die Datenmenge, die für einen optimales Durchsatz zwischengespeichert werden muss.

Diese Begrenzung ist nützlich, wenn Sie Daten über Verbindungen mit einem hohen Produkt aus Übertragungsrate und Verzögerungszeit wie Modems, Gigabit-Ethernet oder schnellen WANs, zur Verfügung stellen. Insbesondere wirkt sich die Begrenzung aus, wenn die Verbindung die Option *Window-scaling* verwendet oder große Sende-Fenster (*send window*) benutzt. Schalten Sie die Debug-Meldungen aus, wenn Sie die Begrenzung aktiviert haben. Dazu setzen Sie die Variable `net.inet.tcp.inflight.debug` auf `0`. Auf Produktions-Systemen sollten Sie zudem die Variable `net.inet.tcp.inflight.min` mindestens auf den Wert `6144` setzen. Allerdings kann ein zu hoher Wert, abhängig von der Verbindung, die Begrenzungsfunktion unwirksam machen. Die Begrenzung reduziert die Datenmenge in den Queues von Routern und Switches, sowie die Datenmenge in der Queue der lokalen Netzwerkkarte. Die Verzögerungszeit (*Round Trip Time*) für interaktive Anwendungen sinkt, da weniger Pakete zwischengespeichert werden. Dies gilt besonders für Verbindungen über langsame Modems. Die Begrenzung wirkt sich allerdings nur auf das Versenden von Daten aus (Uploads, Server). Auf den Empfang von Daten (Downloads) hat die Begrenzung keine Auswirkungen.

Die Variable `net.inet.tcp.inflight.stab` sollte *nicht* angepasst werden. Der Vorgabewert der Variablen beträgt `20`, das heißt es werden maximal zwei Pakete zu dem Produkt aus Übertragungsrate und Verzögerungszeit addiert. Dies stabilisiert den Algorithmus und verbessert die Reaktionszeit auf Veränderungen. Bei langsamen Verbindungen können sich aber die Laufzeiten der Pakete erhöhen (ohne diesen Algorithmus wären sie allerdings noch höher). In solchen Fällen können Sie versuchen, den Wert der Variablen auf `15`, `10` oder `5` herabzusetzten. Gleichzeitig müssen Sie vielleicht auch `net.inet.tcp.inflight.min` auf einen kleineren Wert (beispielsweise `3500`) setzen. Ändern Sie diese Variablen nur ab, wenn Sie keine anderen Möglichkeiten mehr haben.

12.11.3. Virtueller Speicher (Virtual Memory)

12.11.3.1. `kern.maxvnodes`

Ein vnode ist die interne Darstellung einer Datei oder eines Verzeichnisses. Die Erhöhung der Anzahl der für das Betriebssystem verfügbaren vnodes verringert also die Schreib- und Lesezugriffe auf der Festplatte. vnodes werden im Normalfall vom Betriebssystem automatisch vergeben und müssen nicht manuell angepasst werden. In einigen Fällen stellt der Zugriff auf eine Platte allerdings einen Flaschenhals dar, daher sollten Sie in diesem Fall die Anzahl der möglichen vnodes erhöhen, um dieses Problem zu beheben. Beachten Sie dabei aber die Größe des inaktiven und freien Hauptspeichers.

Um die Anzahl der derzeit verwendeten vnodes zu sehen, geben Sie Folgendes ein:

```
# sysctl vfs.numvnodes
vfs.numvnodes: 91349
```

Die maximal mögliche Anzahl der vnodes erhalten Sie durch die Eingabe von:

```
# sysctl kern.maxvnodes
kern.maxvnodes: 100000
```

Wenn sich die Anzahl der genutzten vnodes dem maximal möglichen Wert nähert, sollten Sie den Wert `kern.maxvnodes` zuerst um etwa `1000` erhöhen. Beobachten Sie danach die Anzahl der vom System genutzten `vfs.numvnodes` . Nähert sich der Wert wiederum dem definierten Maximum, müssen Sie `kern.maxvnodes` nochmals erhöhen. Sie sollten nun eine Änderung des Speicherverbrauches über top(1) registrieren können und über mehr aktiven Speicher verfügen.

12.12. Hinzufügen von Swap-Bereichen

Manchmal benötigt ein System mehr Swap-Bereiche. Dieser Abschnitt beschreibt zwei Methoden, um Swap-Bereiche hinzuzufügen: auf einer bestehenden Partition oder auf einem neuen Laufwerk, und das Hinzufügen einer Swap-Datei auf einer existierenden Partition.

Für Informationen zur Verschlüsselung von Swap-Partitionen, zu den dabei möglichen Optionen sowie zu den Gründen für eine Verschlüsselung des Auslagerungsspeichers lesen Sie Abschnitt 18.14, „Den Auslagerungsspeicher verschlüsseln“.

12.12.1. Swap auf einer neuen oder existierenden Festplatte

Das Hinzufügen einer neuen Festplatte für Swap-Bereich bietet eine bessere Leistung, als das Hinzufügen einer Partition auf einem vorhandenem Laufwerk. Die Einrichtung von Partitionen und Laufwerken wird in Abschnitt 18.3, „Hinzufügen von Laufwerken“ beschrieben. Abschnitt 3.7.1, „Ein Partitionslayout entwerfen“ diskutiert Aspekte über die Anordnung und Größe von Swap-Bereichen.

Benutzen Sie swapon(8) um eine Swap-Partition zum System hinzuzufügen. Zum Beispiel:

```
# swapon
/dev/ada1s1b
```

Warnung

Sie können jede Partition verwenden, sofern sie nicht schon eingehangen ist. Das gilt auch dann, wenn die Partition bereits Daten enthält. Wird swapon(8) auf einer Partition ausgeführt die noch Daten enthält, werden die vorhandenen Daten überschrieben und sind unweigerlich verloren. Stellen Sie sicher, das die Partition, die Sie als Swap-Bereich hinzufügen möchten, wirklich die gewünschte Partition ist, bevor sie swapon(8) ausführen.

Um diese Swap-Partition automatisch beim Systemstart hinzuzufügen, fügen Sie einen Eintrag in `/etc/fstab` hinzu:

```
/dev/ada1s1b  none swap sw 0 0
```

Die einzelnen Einträge von `/etc/fstab` werden in fstab(5) erläutert.

12.12.2. Swap-Bereiche über NFS

Swap-Bereiche über NFS sollten Sie nur dann einsetzen, wenn Sie über keine lokale Platte verfügen, da es durch die zur Verfügung stehende Bandbreite limitiert wird und außerdem den NFS-Server zusätzlich belastet.

12.12.3. Swap-Dateien

Sie können eine Datei festgelegter Größe als Swap-Bereich nutzen. Im folgenden Beispiel wird eine 64 MB große Datei mit dem Namen `/usr/swap0` benutzt.

Beispiel 12.1. Erstellen einer Swap-Datei

1. Der `GENERIC` -Kernel unterstützt bereits RAM-Disks (md(4)), welche für diese Aktion benötigt werden. Wenn Sie einen eigenen Kernel erstellen, vergewissern Sie sich, dass die folgende Zeile in der Kernelkonfigurationsdatei enthalten ist:

   ```
   device   md
   ```

   Informationen, wie man einen eigenen Kernel erstellen kann, erhalten Sie in Kapitel 9, *Konfiguration des FreeBSD-Kernels*.

2. Legen Sie die Swap-Datei `/usr/swap0` an:

   ```
   # dd if=/dev/zero of=/usr/swap0 bs=1024k count=64
   ```

3. Setzen Sie die richtigen Berechtigungen für `/usr/swap0` :

   ```
   # chmod 0600 /usr/swap0
   ```

4. Aktivieren Sie die Swap-Datei `/etc/rc.conf` :

   ```
   swapfile="/usr/swap0"   # Set to name of swapfile if aux swapfile desired.
   ```

5. Um die Swap-Datei zu aktivieren, führen Sie entweder einen Neustart durch oder geben das folgende Kommando ein:

   ```
   # mdconfig -a -t vnode -f /usr/swap0 -u 0 && swapon /dev/md0
   ```

12.13. Energie- und Ressourcenverwaltung

Verfasst von Hiten Pandya und Tom Rhodes.

Es ist wichtig, Hardware effizient einzusetzen. Vor der Einführung des *Advanced Configuration and Power Interface* (ACPI) konnten Stromverbrauch und Wärmeabgabe eines Systems nur schlecht von Betriebssystemen gesteuert werden. Die Hardware wurde vom BIOS gesteuert, was die Kontrolle der Energieverwaltung für den Anwender erschwerte. Das *Advanced Power Management (APM)* erlaubte es lediglich, einige wenige Funktionen zu steuern, obwohl die Überwachung von Energie- und Ressourcenverbrauch zu den wichtigsten Aufgaben eines Betriebssys-

tems gehört, um auf verschiedene Ereignisse, beispielsweise einen unerwarteten Temperaturanstieg, reagieren können.

Dieser Abschnitt erklärt das Advanced Configuration and Power Interface (ACPI).

12.13.1. Was ist ACPI?

Advanced Configuration and Power Interface (ACPI) ist ein Standard verschiedener Hersteller, der die Verwaltung von Hardware und Energiesparfunktionen festlegt. Die ACPI-Funktionen können von einem Betriebssystem gesteuert werden. Der Vorgänger des ACPI, APM, erwies sich in modernen Systemen als unzureichend.

12.13.2. Mängel des Advanced Power Managements

Das APM steuert den Energieverbrauch eines Systems auf Basis der Systemaktivität. Das APM-BIOS wird von dem Hersteller des Systems zur Verfügung gestellt und ist auf die spezielle Hardware angepasst. Der APM-Treiber des Betriebssystems greift auf das *APM Software Interface* zu, das den Energieverbrauch regelt. APM findet sich in der Regel nur noch in Systemen, die vor 2001 produziert wurden.

Das APM hat hauptsächlich vier Probleme. Erstens läuft die Energieverwaltung unabhängig vom Betriebssystem in einem herstellerspezifischen BIOS. Beispielsweise kann das APM-BIOS die Festplatten nach einer konfigurierbaren Zeit ohne die Zustimmung des Betriebssystems herunterfahren. Zweitens befindet sich die ganze APM-Logik im BIOS; das Betriebssystem hat gar keine APM-Komponenten. Bei Problemen mit dem APM-BIOS muss das Flash-ROM aktualisiert werden. Diese Prozedur ist gefährlich, da sie im Fehlerfall das System unbrauchbar machen kann. Zum Dritten ist APM eine Technik, die herstellerspezifisch ist und nicht koordiniert wird. Fehler im BIOS eines Herstellers werden nicht unbedingt im BIOS anderer Hersteller korrigiert. Das letzte Problem ist, dass im APM-BIOS nicht genügend Platz vorhanden ist, um eine durchdachte oder eine auf den Zweck der Maschine zugeschnittene Energieverwaltung zu implementieren.

Das *Plug and Play BIOS (PNPBIOS)* war ebenfalls unzureichend. Das PNPBIOS verwendet eine 16-Bit-Technik. Damit das Betriebssystem das PNPBIOS ansprechen kann, muss es in einer 16-Bit-Emulation laufen.

Der APM-Treiber wird in apm(4) beschrieben.

12.13.3. Konfiguration des ACPI

Das Modul acpi(4) wird standardmäßig beim Systemstart vom loader(8) geladen und sollte daher *nicht* fest in den Kernel eingebunden werden. Dadurch kann ein Modul ohne einen Neubau des Kernels leichter ersetzt und getestet werden. Das ACPI-Modul im laufenden Betrieb zu laden, führt oft nicht zum gewünschten Ergebnis. Treten bei Ihrem System Probleme auf, kann ACPI auch komplett deaktiviert werden. Dazu definieren Sie die Variable `hint.acpi.0.disabled="1"` in der Datei `/boot/loader.conf`. Alternativ können Sie die Variable auch am loader(8)-Prompt eingeben. Das Modul kann im laufenden Betrieb nicht entfernt werden, da es zur Kommunikation mit der Hardware verwendet wird.

Anmerkung

ACPI und APM können nicht zusammen verwendet werden. Das zuletzt geladene Modul beendet sich, sobald es bemerkt, dass das andere Modul geladen ist.

Mit acpiconf(8) können Sie das System in einen Ruhemodus (*sleep mode*) versetzen. Es gibt verschiedene Modi (von `1` bis `5`), die Sie auf der Kommandozeile mit `-s` angeben können. Für die meisten Anwender sind die Modi `1` und `3` völlig ausreichend. Der Modus `5` schaltet das System aus (*Soft-off*) und entspricht dem folgenden Befehl:

```
# halt -p
```

Verschiedene Optionen können als sysctl(8)-Variablen gesetzt werden. Lesen Sie dazu acpi(4) sowie acpiconf(8).

12.14. ACPI-Fehlersuche

Verfasst von Nate Lawson.
Mit Beiträgen von Peter Schultz und Tom Rhodes.

ACPI ist ein gänzlich neuer Weg, um Geräte aufzufinden und deren Stromverbrauch zu regulieren. Weiterhin bietet ACPI einen einheitlichen Zugriff auf Geräte, die vorher vom BIOS verwaltet wurden. Es werden zwar Fortschritte gemacht, dass ACPI auf allen Systemen läuft, doch tauchen immer wieder Fehler auf: fehlerhafter Bytecode der ACPI-Machine-Language (AML) einiger Systemplatinen, ein unvollständiges FreeBSD-Kernel-Subsystem oder Fehler im ACPI-CA-Interpreter von Intel®.

Dieser Abschnitt hilft Benutzern, zusammen mit den Betreuern des FreeBSD-ACPI-Subsystems, Fehlerquellen zu finden und Fehler zu beseitigen.

12.14.1. Fehlerberichte einreichen

Anmerkung

Bevor Sie einen Fehlerbericht einreichen, stellen Sie bitte sicher, dass das BIOS und die Firmware des Controllers aktuell sind.

Wenn Sie einen Fehlerbericht einsenden wollen, schicken Sie bitte die folgenden Informationen an die Mailingliste freebsd-acpi:

- Beschreiben Sie den Fehler und alle Umstände, unter denen der Fehler auftritt. Geben Sie ebenfalls den Typ und das Modell Ihres Systems an. Wenn Sie einen neuen Fehler entdeckt haben, versuchen Sie möglichst genau zu beschreiben, wann der Fehler das erste Mal aufgetreten ist.
- Die Ausgabe von dmesg(8) nach der Eingabe von `boot -v`. Geben Sie auch alle Fehlermeldungen an, die erscheinen, wenn Sie den Fehler provozieren.
- Die Ausgabe von dmesg(8) nach der Eingabe von `boot -v` und mit deaktiviertem ACPI, wenn das Problem ohne ACPI nicht auftritt.
- Die Ausgabe von `sysctl hw.acpi`. Dieses Kommando zeigt die vom System unterstützten ACPI-Funktionen an.
- Die URL, unter der die ACPI-Source-Language (ASL) liegt. Schicken Sie bitte *nicht* die ASL an die Mailingliste, da die ASL sehr groß sein kann. Eine Kopie der ASL erstellen Sie mit dem nachstehenden Befehl:

  ```
  # acpidump -td > name-system.asl
  ```

  Setzen Sie für *`name`* den Namen des Kontos und für *`system`* den Hersteller und das Modell des Systems ein. Zum Beispiel: `njl-FooCo6000.asl`.

Obwohl die meisten Entwickler die Mailingliste freebsd-current lesen, sollten Sie Fehlerberichte an die Liste freebsd-acpi schicken. Seien Sie bitte geduldig; wir haben alle Arbeit außerhalb des Projekts. Wenn der Fehler nicht offensichtlich ist, bitten wir Sie vielleicht, einen offiziellen Fehlerbericht (PR) mit send-pr(1) einzusenden. Geben Sie im Fehlerbericht bitte dieselben Informationen wie oben an. Mithilfe der PRs verfolgen und lösen wir Probleme. Senden Sie bitte keinen PR ein, ohne vorher den Fehlerbericht an die Liste freebsd-acpi zu senden. Es kann sein, dass der Fehler schon von jemand anderem gemeldet wurde.

12.14.2. ACPI-Grundlagen

ACPI gibt es in allen modernen Rechnern der ia32- (x86), ia64- (Itanium) und amd64- (AMD) Architektur. Der vollständige Standard bietet Funktionen zur Steuerung und Verwaltung der CPU-Leistung, der Stromversorgung, von

Wärmebereichen, Batterien, eingebetteten Controllern und Bussen. Auf den meisten Systemen wird nicht der vollständige Standard implementiert. Arbeitsplatzrechner besitzen meist nur Funktionen zur Verwaltung der Busse, während Notebooks Funktionen zur Temperaturkontrolle und Ruhezustände besitzen.

Ein ACPI konformes System besitzt verschiedene Komponenten. Die BIOS- und Chipsatz-Hersteller stellen mehrere statische Tabellen bereit, zum Beispiel die Fixed-ACPI-Description-Table (FADT). Die Tabellen enthalten beispielsweise die mit SMP-Systemen benutzte APIC-Map, Konfigurationsregister und einfache Konfigurationen. Zusätzlich gibt es die Differentiated-System-Description-Table (DSDT), die Bytecode enthält. Die Tabelle ordnet Geräte und Methoden in einem baumartigen Namensraum an.

Ein ACPI-Treiber muss die statischen Tabellen einlesen, einen Interpreter für den Bytecode bereitstellen und die Gerätetreiber im Kernel so modifizieren, dass sie mit dem ACPI-Subsystem kommunizieren. Für FreeBSD, Linux® und NetBSD hat Intel® den Interpreter ACPI-CA, zur Verfügung gestellt. Der Quelltext zu ACPI-CA befindet sich im Verzeichnis `src/sys/contrib/dev/acpica` . Die Schnittstelle von ACPI-CA zu FreeBSD befindet sich unter `src/sys/dev/acpica/Osd` . Treiber, die verschiedene ACPI-Geräte implementieren, befinden sich im Verzeichnis `src/sys/dev/acpica` .

12.14.3. Häufige Probleme

Damit ACPI richtig funktioniert, müssen alle Teile funktionieren. Im Folgenden finden Sie eine Liste mit Problemen und möglichen Umgehungen oder Fehlerbehebungen. Die Liste ist nach der Häufigkeit, mit der die Probleme auftreten, sortiert.

12.14.3.1. Mausprobleme

Es kann vorkommen, dass die Maus nicht mehr funktioniert, wenn Sie nach einem Suspend weiterarbeiten wollen. Ist dies bei Ihnen der Fall, reicht es meistens aus, den Eintrag `hint.psm.0.flags="0x3000"` in `/boot/loader.conf` aufzunehmen. Besteht das Problem weiterhin, sollten Sie einen Fehlerbericht senden.

12.14.3.2. Suspend/Resume

ACPI kennt drei Suspend-to-RAM-Zustände (STR): `S1`-`S3`. Es gibt einen Suspend-to-Disk-Zustand (STD): `S4`. Der Zustand `S5` wird Soft-Off genannt. In diesem Zustand befindet sich ein Rechner, wenn die Stromversorgung angeschlossen ist, der Rechner aber nicht hochgefahren ist. Der Zustand `S4` kann auf zwei Arten implementiert werden: `S4`BIOS und `S4`OS. Im ersten Fall wird der Suspend-to-Disk-Zustand durch das BIOS hergestellt im zweiten Fall alleine durch das Betriebssystem.

Anmerkung

Die Suspend-Zustände sind Ruhezustände, in denen der Rechner weniger Energie als im Normalbetrieb benötigt. Resume bezeichnet die Rückkehr zum Normalbetrieb.

Die Suspend-Zustände können Sie mit dem Kommando `sysctl hw.acpi` ermitteln. Das Folgende könnte beispielsweise ausgegeben werden:

```
hw.acpi.supported_sleep_state: S3 S4 S5
hw.acpi.s4bios: 0
```

Diese Ausgabe besagt, dass mit dem Befehl `acpiconf -s` die Zustände `S3`, `S4`OS und `S5` eingestellt werden können. Hätte `s4bios` den Wert `1`, gäbe es den Zustand `S4`BIOS anstelle von `S4`OS.

Wenn Sie die Suspend- und Resume-Funktionen testen, fangen Sie mit dem `S1`-Zustand an, wenn er angeboten wird. Dieser Zustand wird am ehesten funktionieren, da der Zustand wenig Treiber-Unterstützung benötigt. Der Zustand `S2` ist ähnlich wie `S1`, allerdings hat ihn noch niemand implementiert. Als nächstes sollten Sie den Zustand `S3` ausprobieren. Dies ist der tiefste STR-Schlafzustand. Dieser Zustand ist auf massive Treiber-Unterstützung ange-

wiesen, um die Geräte wieder richtig zu initialisieren. Wenn Sie Probleme mit diesem Zustand haben, können Sie die Mailingliste freebsd-acpi anschreiben. Erwarten Sie allerdings nicht zu viel: Es gibt viele Treiber und Geräte, an denen noch gearbeitet und getestet wird.

Ein häufiges Problem mit Suspend/Resume ist, dass viele Gerätetreiber ihre Firmware, Register und Gerätespeicher nicht korrekt speichern, wiederherstellen und/oder reinitialisieren. Um dieses Problem zu lösen, sollten Sie zuerst die folgenden Befehle ausführen:

```
# sysctl debug.bootverbose=1
# sysctl debug.acpi.suspend_bounce=1
# acpiconf -s 3
```

Dieser Test emuliert einen Suspend/Resume-Zyklus für alle Geräte (ohne dass diese dabei wirklich in den Status `S3` wechseln). In vielen Fällen reicht dies bereits aus, um Probleme (beispielsweise verlorener Firmware-Status, Timeouts, hängende Geräte) zu entdecken. Beachten Sie dabei, dass das Gerät bei diesem Test nicht wirklich in den Status `S3` wechseln. Es kann also vorkommen, dass manche Geräte weiterhin mit Strom versorgt werden (dies wäre bei einem wirklichen Wechsel in den Status `S3` NICHT möglich. Andere Geräte werden normal weiterarbeiten, weil sie über keine Suspend/Resume-Funktionen verfügen.

Schwierigere Fälle können den Einsatz zusätzlicher Hardware (beispielsweise serielle Ports/Kabel für die Verbindung über eine serielle Konsole oder Firewire-Ports/Kabel für dcons(4)) sowie Kenntnisse im Bereich Kerneldebugging erforderlich machen.

Um das Problem einzugrenzen, entfernen Sie soviele Treiber wie möglich aus dem Kernel. Sie können das Problem isolieren, indem Sie einen Treiber nach dem anderen laden, bis der Fehler wieder auftritt. Typischerweise verursachen binäre Treiber wie `nvidia.ko`, Grafiktreiber und USB-Treiber die meisten Fehler, hingegen laufen Ethernet-Treiber für gewöhnlich sehr zuverlässig. Wenn ein Treiber zuverlässig geladen und entfernt werden kann, können Sie den Vorgang automatisieren, indem Sie die entsprechenden Kommandos in `/etc/rc.suspend` und `/etc/rc.resume` einfügen. In den Dateien finden Sie ein deaktiviertes Beispiel, das einen Treiber lädt und wieder entfernt. Ist die Bildschirmanzeige bei der Wiederaufnahme des Betriebs gestört, setzen Sie die Variable `hw.acpi.reset_video` auf `0`. Versuchen Sie auch, die Variable `hw.acpi.sleep_delay` auf kürzere Zeitspannen zu setzen.

Die Suspend- und Resume-Funktionen können Sie auch auf einer neuen Linux®-Distribution mit ACPI testen. Wenn es mit Linux® funktioniert, liegt das Problem wahrscheinlich bei einem FreeBSD-Treiber. Es hilft uns, das Problem zu lösen, wenn Sie feststellen können, welcher Treiber das Problem verursacht. Beachten Sie bitte, dass die ACPI-Entwickler normalerweise keine anderen Treiber pflegen (beispielsweise Sound- oder ATA-Treiber). Es ist wohl das beste, die Ergebnisse der Fehlersuche an die Mailingliste freebsd-current und den Entwickler des Treibers zu schicken. Erfahrene Benutzer können versuchen, den Fehler in der Resume-Funktion zu finden, indem sie einige printf(3)-Anweisungen in den Code des fehlerhaften Treibers einfügen.

Schließlich können Sie ACPI noch abschalten und stattdessen APM verwenden. Wenn die Suspend- und Resume-Funktionen mit APM funktionieren, sollten Sie besser APM verwenden (insbesondere mit alter Hardware von vor dem Jahr 2000). Die Hersteller benötigten einige Zeit, um ACPI korrekt zu implementieren, daher gibt es mit älterer Hardware oft ACPI-Probleme.

12.14.3.3. Systemhänger

Die meisten Systemhänger entstehen durch verlorene Interrupts oder einen Interrupt-Sturm. Probleme werden verursacht durch die Art, in der das BIOS Interrupts vor dem Systemstart konfiguriert, durch eine fehlerhafte APIC-Tabelle und durch die Zustellung des System-Control-Interrupts (SCI).

Anhand der Ausgabe des Befehls `vmstat -i` können Sie verlorene Interrupts von einem Interrupt-Sturm unterscheiden. Untersuchen Sie die Ausgabezeile, die `acpi0` enthält. Ein Interrupt-Sturm liegt vor, wenn der Zähler öfter als ein paar Mal pro Sekunde hochgezählt wird. Wenn sich das System aufgehangen hat, versuchen Sie mit der Tastenkombination Ctrl+Alt+Esc in den Debugger DDB zu gelangen. Geben Sie dort den Befehl `show interrupts` ein.

Wenn Sie Interrupt-Probleme haben, ist es vorerst wohl am besten, APIC zu deaktivieren. Tragen Sie dazu die Zeile `hint.apic.0.disabled="1"` in `/boot/loader.conf` ein.

12.14.3.4. Abstürze (Panics)

Panics werden so schnell wie möglich behoben; mit ACPI kommt es aber selten dazu. Zuerst sollten Sie die Panic reproduzieren und dann versuchen einen *backtrace* (eine Rückverfolgung der Funktionsaufrufe) zu erstellen. Richten Sie dazu den DDB über die serielle Schnittstelle (siehe Abschnitt 26.6.4.3, „DDB Debugger über die serielle Schnittstelle“) oder eine gesonderte dump(8)-Partition ein. In DDB können Sie den *backtrace* mit dem Kommando `tr` erstellen. Falls Sie den *backtrace* vom Bildschirm abschreiben müssen, schreiben Sie bitte mindestens die fünf ersten und die fünf letzten Zeile der Ausgabe auf.

Versuchen Sie anschließend, das Problem durch einen Neustart ohne ACPI zu beseitigen. Wenn das funktioniert hat, können Sie versuchen, das verantwortliche ACPI-Subsystem durch Setzen der Variablen `debug.acpi.disable` herauszufinden. Die Hilfeseite acpi(4) enthält dazu einige Beispiele.

12.14.3.5. Nach einem Suspend oder einem Stopp startet das System wieder

Setzen Sie zuerst `hw.acpi.disable_on_poweroff="0"` in `/boot/loader.conf 0`. Damit wird verhindert, dass ACPI während des Systemabschlusses die Bearbeitung verschiedener Ereignisse deaktiviert. Auf manchen Systemen muss die Variable den Wert `1` besitzen (die Voreinstellung). Normalerweise wird der unerwünschte Neustart des Systems durch Setzen dieser Variablen behoben.

12.14.3.6. Andere Probleme

Wenn Sie weitere Probleme mit ACPI haben (Umgang mit einer Docking-Station, nicht erkannte Geräte), schicken Sie bitte eine Beschreibung an die Mailingliste freebsd-acpi. Allerdings kann es sein, dass einige Probleme von noch unvollständigen Teilen des ACPI-Subsystems abhängen und es etwas dauern kann bis diese Teile fertig sind. Seien Sie geduldig und rechnen Sie damit, Fehlerbehebungen zu testen.

12.14.4. ASL, acpidump(8) und IASL

Einige BIOS-Hersteller liefern einen fehlerhaften Bytecode aus. Dies erkennen Sie an Kernelmeldungen wie diesen:

```
ACPI-1287: *** Error: Method execution failed [\\_SB_.PCI0.LPC0.FIGD._STA] \\
(Node 0xc3f6d160), AE_NOT_FOUND
```

Oft können Sie das Problem dadurch lösen, dass Sie eine aktuelle BIOS-Version einspielen. Die meisten Meldungen auf der Konsole sind harmlos, wenn aber beispielsweise der Batteriestatus falsch angezeigt wird, können Sie in den Meldungen nach Problemen mit der AML-Machine-Language (AML) suchen. Der Bytecode der AML wird aus der ACPI-Source-Language (ASL) übersetzt und in einer Tabelle, der DSDT, abgelegt. Eine Kopie der ASL können Sie mit dem Befehl acpidump(8) erstellen. Verwenden Sie mit diesem Befehl sowohl die Option `-t` (die Inhalte der statischen Tabellen anzeigen) als auch die Option `-d` (die AML in ASL zurückübersetzen). Ein Beispiel für die Syntax finden Sie in Abschnitt 12.14.1, „Fehlerberichte einreichen“.

Sie können einfach prüfen, ob sich die ASL übersetzen lässt. Für gewöhnlich können Sie Warnungen während des Übersetzens ignorieren. Fehlermeldungen führen normal dazu, dass ACPI fehlerhaft arbeitet. ASL übersetzen Sie mit dem nachstehenden Kommando:

```
# iasl ihre.asl
```

12.14.5. Die ASL reparieren

Es ist das Ziel von FreeBSD, dass ACPI ohne Eingriffe des Benutzers läuft. Zurzeit werden allerdings noch Umgehungen für Fehler der BIOS-Hersteller entwickelt. Der Microsoft®-Interpreter (`acpi.sys` und `acpiec.sys`) prüft die ASL nicht streng gegen den Standard. Daher reparieren BIOS-Hersteller, die ACPI nur unter Windows® testen, ihre ASL nicht. Die FreeBSD Entwickler hoffen, dass sie das vom Standard abweichende Verhalten des Microsoft®-Interpreters dokumentieren und in FreeBSD replizieren können. Dadurch müssen Benutzer ihre ASL nicht selbst reparieren. Sie können Ihre ASL selbst reparieren, wenn Sie ein Problem umgehen und uns helfen möchten. Senden Sie bitte die mit diff(1) erstellte Differenz zwischen alter und neuer ASL. Die Entwickler werden versuchen, den Interpreter ACPI-CA zu korrigieren.

Die nachfolgende Liste enthält häufige Fehlermeldungen, deren Ursache und eine Beschreibung, wie die Fehler korrigiert werden:

12.14.5.1. Abhängigkeiten vom Betriebssystem

Einige AMLs gehen davon aus, dass der Anwender eine Windows®-Versionen benutzt. Versuchen Sie das Betriebssystem, das Sie in der ASL finden, in `/boot/loader.conf` anzugeben: `hw.acpi.osname="Windows 2001"`.

12.14.5.2. Fehlende Return-Anweisungen

Einige Methoden verzichten auf die vom Standard vorgeschriebene Rückgabe eines Wertes. Obwohl der Interpreter ACPI-CA dies nicht beheben kann, besitzt FreeBSD die Möglichkeit, den Rückgabewert implizit zu setzen. Wenn Sie wissen, welcher Wert zurückgegeben werden muss, können Sie die fehlenden Return-Anweisungen selbst einsetzen. Die Option `-f` zwingt iasl(8), die ASL zu übersetzen.

12.14.5.3. Überschreiben der vorgegebenen AML

Nachdem Sie Ihre ASL in der Datei `ihre.asl` angepasst haben, übersetzen Sie die ASL wie folgt:

```
# iasl ihre.asl
```

Die Option `-f` erzwingt das Erstellen der AML auch dann, wenn während der Übersetzung Fehler auftreten. Einige Fehler, wie fehlende Return-Anweisungen, werden automatisch vom Interpreter umgangen.

In der Voreinstellung erstellt der Befehl iasl(8) die Ausgabedatei `DSDT.aml`. Wenn Sie diese Datei anstelle der fehlerhaften Kopie des BIOS laden wollen, editieren Sie `/boot/loader.conf` wie folgt:

```
acpi_dsdt_load="YES"
acpi_dsdt_name="/boot/DSDT.aml"
```

Stellen Sie bitte sicher, dass sich `DSDT.aml` in `/boot` befindet.

12.14.6. ACPI-Meldungen zur Fehlersuche erzeugen

Der ACPI-Treiber besitzt flexible Möglichkeiten zur Fehlersuche. Sie können sowohl die zu untersuchenden Subsysteme als auch die zu erzeugenden Ausgaben festlegen. Die zu untersuchenden Subsysteme werden als so genannte „layers“ angegeben. Die Subsysteme sind in ACPI-CA-Komponenten (`ACPI_ALL_COMPONENTS`) und ACPI-Hardware (`ACPI_ALL_DRIVERS`) aufgeteilt. Welche Meldungen ausgegeben werden, wird über „level“ gesteuert. „level“ reicht von `ACPI_LV_ERROR` (es werden nur Fehler ausgegeben) bis zu `ACPI_LV_VERBOSE` (alles wird ausgegeben). „level“ ist eine Bitmaske, sodass verschiedene Stufen auf einmal (durch Leerzeichen getrennt) angegeben werden können. Die erzeugte Ausgabemenge passt vielleicht nicht in den Konsolenpuffer. In diesem Fall sollte die Ausgabe mithilfe einer seriellen Konsole gesichert werden. Die möglichen Werte für „layers“ und „level“ werden in acpi(4) beschrieben.

Die Ausgaben zur Fehlersuche sind in der Voreinstellung nicht aktiviert. Wenn ACPI im Kernel enthalten ist, fügen Sie `options ACPI_DEBUG` zur Kernelkonfigurationsdatei hinzu. Sie können die Ausgaben zur Fehlersuche global aktivieren, indem Sie in der Datei `/etc/make.conf` die Zeile `ACPI_DEBUG=1` einfügen. Das Modul `acpi.ko` können Sie wie folgt neu übersetzen:

```
# cd /sys/modules/acpi/acpi
&& make clean &&
make ACPI_DEBUG=1
```

Installieren Sie anschließend `acpi.ko` im Verzeichnis `/boot/kernel`. In `/boot/loader.conf` stellen Sie „level“ und „layer“ ein. Das folgende Beispiel aktiviert die Ausgabe von Fehlern für alle ACPI-CA-Komponenten und alle ACPI-Hardwaretreiber (wie CPU, LID):

```
debug.acpi.layer="ACPI_ALL_COMPONENTS ACPI_ALL_DRIVERS"
debug.acpi.level="ACPI_LV_ERROR"
```

Wenn ein Problem durch ein bestimmtes Ereignis, beispielsweise den Start nach einem Ruhezustand, hervorgerufen wird, können Sie die Einstellungen für „level“ und „layer“ auch mit dem Kommando `sysctl` vornehmen. In diesem Fall müssen Sie `/boot/loader.conf` nicht editieren. Auf der Kommandozeile geben Sie über sysctl(8) dieselben Variablennamen wie in `/boot/loader.conf` an.

12.14.7. ACPI-Informationsquellen

Weitere Informationen zu ACPI erhalten Sie an den folgenden Stellen:

- die FreeBSD ACPI Mailingliste,
- die Archive der ACPI-Mailingliste: `http://lists.FreeBSD.org/pipermail/freebsd-acpi/` ,
- die alten Archive der ACPI-Mailingliste: `http://home.jp.FreeBSD.org/mail-list/acpi-jp/` ,
- die ACPI-Spezifikation (Version 2.0): `http://acpi.info/spec.htm` ,
- acpi(4), acpi_thermal(4), acpidump(8), iasl(8) und acpidb(8),
- DSDT debugging resource.

Kapitel 13. FreeBSDs Bootvorgang

Übersetzt von Hans-Christian Ebke.

13.1. Übersicht

Das Starten des Computers und das Laden des Betriebssystems wird im Allgemeinen als „Bootstrap-Vorgang“ bezeichnet, oder einfach als „Booten“. FreeBSDs Bootvorgang ermöglicht große Flexibilität, was das Anpassen dessen anbelangt, was passiert, wenn das System gestartet wird. Es kann zwischen verschiedenen Betriebssystemen, die auf demselben Computer installiert sind oder verschiedenen Versionen desselben Betriebssystems oder installierten Kernels gewählt werden.

Dieses Kapitel zeigt die zur Verfügung stehenden Konfigurationsmöglichkeiten und wie man den Bootvorgang anpasst. Dies schließt alles ein, bis der Kernel gestartet worden ist, der dann alle Geräte gefunden hat und init(8) gestartet hat. Dies passiert, wenn die Farbe des Textes während des Bootvorgangs von weiß zu grau wechselt.

Dieses Kapitel informiert über folgende Punkte:

- Die Komponenten des FreeBSD-Bootvorgangs und deren Interaktion.
- Die Optionen, mit denen der FreeBSD-Bootvorgang gesteuert werden kann.
- Wie Geräte mit device.hints(5) konfiguriert werden.

Anmerkung

Dieses Kapitel erklärt den Bootvorgang von FreeBSD auf Intel x86-Plattformen.

13.2. Das Problem des Bootens

Wenn der Computer eingeschaltet wird und das Betriebssystem gestartet werden soll, entsteht ein interessantes Dilemma, denn der Computer weiß per Definition nicht, wie er irgendetwas tut, bis das Betriebssystem gestartet wurde. Das schließt das Starten von Programmen, die sich auf der Festplatte befinden, ein. Wenn der Computer kein Programm von der Festplatte starten kann, sich das Betriebssystem aber genau dort befindet, wie wird es dann gestartet?

Dieses Problem ähnelt einer Geschichte des Barons von Münchhausen. Dort war eine Person in einen Sumpf gefallen und hat sich selbst an den Riemen seiner Stiefel (engl. *bootstrap*) herausgezogen. In den jungen Jahren des Computerzeitalters wurde mit dem Begriff Bootstrap dann die Technik das Betriebssystem zu laden bezeichnet und wurde hinterher mit booten abgekürzt.

Auf x86-Plattformen ist das BIOS (Basic Input/Output System) dafür verantwortlich, das Betriebssystem zu laden. Dazu liest das BIOS den Master Bootsektor (MBR; Master Boot Record) aus, der sich an einer bestimmten Stelle auf der Festplatte/Diskette befinden muss. Das BIOS kann den MBR selbstständig laden und ausführen und geht davon aus, dass dieser die restlichen Dinge, die für das Laden des Betriebssystems notwendig sind, selbst oder mit Hilfe des BIOS erledigen kann.

Der Code innerhalb des MBRs wird für gewöhnlich als *Boot-Manager* bezeichnet, insbesondere, wenn eine Interaktion mit dem Anwender stattfindet. Ist dies der Fall, verwaltet der Boot-Manager zusätzlichen Code im ersten *Track* der Platte oder in Dateisystemen anderer Betriebssysteme. (Boot-Manager werden manchmal auch als *Boot*

Loader bezeichnet, unter FreeBSD wird dieser Begriff aber für eine spätere Phase des Systemstarts verwendet.) Zu den bekanntesten Boot-Managern gehören boot0 (der auch als Boot Easy bekannte Standard-Boot-Manager von FreeBSD), Grub, GAG, sowie LILO. (Von diesen Boot-Managern hat nur boot0 innerhalb des MBRs Platz.)

Falls nur ein Betriebssystem installiert ist, ist der Standard MBR ausreichend. Dieser MBR sucht nach dem ersten bootbaren Slice (das dabei als *active* gekennzeichnet ist) auf dem Laufwerk und führt den dort vorhandenen Code aus, um das restliche Betriebssystem zu laden. Der von fdisk(8) in der Voreinstellung installierte MBR ist ein solcher MBR und basiert auf `/boot/mbr` .

Falls mehrere Betriebssysteme installiert sind, kann ein anderer Boot-Manager installiert werden, der eine Liste der verfügbaren Betriebssysteme anzeigt, so dass der Benutzer wählen kann, welches Betriebssystem er booten möchte. Der nächste Abschnitt beschreibt zwei Boot-Manager.

Das restliche FreeBSD-Bootstrap-System ist in drei Phasen unterteilt. Die erste Phase wird vom MBR durchgeführt, der gerade genug Funktionalität besitzt um den Computer in einen bestimmten Status zu verhelfen und die zweite Phase zu starten. Die zweite Phase führt ein wenig mehr Operationen durch und startet schließlich die dritte Phase, die das Laden des Betriebssystems abschließt. Der ganze Prozess wird in drei Phasen durchgeführt, weil PC Standards die Größe der Programme, die in Phase eins und zwei ausgeführt werden, limitiert. Durch das Verketten der durchzuführenden Aufgaben wird es FreeBSD möglich, ein sehr flexibles Ladeprogramm zu besitzen.

Als nächstes wird der Kernel gestartet, der zunächst nach Geräten sucht und sie für den Gebrauch initialisiert. Nach dem Booten des Kernels übergibt dieser die Kontrolle an den Benutzer Prozess init(8), der erst sicherstellt, dass alle Laufwerke benutzbar sind und die Ressourcen Konfiguration auf Benutzer Ebene startet. Diese wiederum mountet Dateisysteme, macht die Netzwerkkarten für die Kommunikation mit dem Netzwerk bereit und startet alle Prozesse, die konfiguriert wurden, um beim Hochfahren des FreeBSD-Systems gestartet zu werden.

13.3. Boot-Manager und Boot-Phasen

13.3.1. Der Boot-Manager

Der Code im MBR oder im Boot-Manager wird manchmal auch als *stage zero* des Boot-Prozesses bezeichnet. Dieser Abschnitt beschreibt zwei Boot-Manager: boot0 und LILO.

Der boot0 Boot-Manager: Der vom FreeBSD-Installationsprogramm oder boot0cfg(8) in der Voreinstelung installierte Master Boot Record (MBR) basiert auf `/boot/boot0` . Die Größe und Leistungsfähigkeit von boot0 ist auf 446 Bytes beschränkt, weil der restliche Platz für die Partitionstabelle sowie den `0x55AA` -Identifier am Ende des MBRs benötigt wird. Wenn boot0 und mehrere Betriebssysteme installiert sind, wird beim Starten des Computers eine Anzeige ähnlich der folgenden zu sehen sein:

Beispiel 13.1. **`boot0`**-Screenshot

```
F1 Win
F2 FreeBSD

Default: F2
```

Diverse Betriebssysteme, insbesondere Windows®, überschreiben den existierenden MBR, wenn sie nach FreeBSD installiert werden. Falls dies passiert, kann mit folgendem Kommando der momentane MBR durch den FreeBSD-MBR ersetzt werden:

```
# fdisk -B -b /boot/boot0  Gerät
```

Bei *Gerät* handelt es sich um das Gerät, von dem gebootet wird, also beispielsweise `ad0` für die erste IDE-Festplatte, `ad2` für die erste IDE-Festplatte am zweiten IDE-Controller, `da0` für die erste SCSI-Festplatte. Um eine angepasste Konfiguration des MBR zu erstellen, lesen Sie boot0cfg(8).

Der LILO-Boot-Manager: Damit dieser Boot-Manager auch FreeBSD booten kann, starten Sie zuerst Linux und fügen danach folgende Zeilen in die Konfigurationsdatei `/etc/lilo.conf` ein:

```
other=/dev/hdXY
table=/dev/hdX
loader=/boot/chain.b
label=FreeBSD
```

Dabei muss die primäre Partition von FreeBSD sowie dessen Platte im Linux-Format angeben werden. Dazu wird *X* durch die Linux-Bezeichnung der Platte und *Y* durch die von Linux verwendete Partitionsnummer ersetzt. Für ein SCSI-Laufwerk wird */dev/sd* anstelle von */dev/hd* verwendet. Die Zeile `loader=/boot/chain.b` kann weggelassen werden, wenn beide Betriebssysteme auf der gleichen Platte installiert sind. Geben Sie danach `/sbin/lilo -v` ein, um die Änderungen zu übernehmen. Achten Sie dabei besonders auf etwaige Fehlermeldungen.

13.3.2. Phase Eins, `/boot/boot1` und Phase Zwei, `/boot/boot2`

Im Prinzip sind die erste und die zweite Phase Teile desselben Programms, im selben Bereich auf der Festplatte. Aufgrund von Speicherplatz-Beschränkungen wurden sie in zwei Teile aufgeteilt, welche jedoch immer zusammen installiert werden. Beide werden entweder vom Installer oder von bsdlabel aus der kombinierten `/boot/boot` kopiert.

Beide Phasen befinden sich außerhalb des Dateisystems im Bootsektor des Boot-Slices, wo boot0 oder ein anderer Boot-Manager ein Programm erwarten, das den weiteren Bootvorgang durchführen kann. Die Anzahl der dabei verwendeten Sektoren wird durch die Größe von `/boot/boot` bestimmt.

`boot1` ist ein sehr einfaches Programm, da es nur 512 Bytes groß sein darf, und es besitzt gerade genug Funktionalität, um FreeBSDs *bsdlabel*, das Informationen über den Slice enthält, auszulesen, und um `boot2` zu finden und auszuführen.

`boot2` ist schon ein wenig umfangreicher und besitzt genügend Funktionalität, um Dateien in FreeBSDs Dateisystem zu finden. Außerdem hat es eine einfache Schnittstelle, die es ermöglicht, den zu ladenden Kernel oder Loader auszuwählen.

Da der Loader einen weitaus größeren Funktionsumfang hat und eine schöne und einfach zu bedienende Boot-Konfigurations-Schnittstelle zur Verfügung stellt, wird er gewöhnlich von `boot2` anstatt des Kernels gestartet. Früher war es jedoch dazu da den Kernel direkt zu starten.

Beispiel 13.2. **`boot2`**-Screenshot

```
>> FreeBSD/i386 BOOT
Default: 0:ad(0,a)/boot/loader
boot:
```

bsdlabel(8) kann dazu verwendet werden, dass installierte `boot1` und `boot2` zu ersetzen:

```
# bsdlabel -B diskslice
```

Wobei *diskslice* das Laufwerk und die Slice darstellt, von dem gebootet wird, beispielsweise `ad0s1` für die erste Slice auf der ersten IDE-Festplatte.

Dangerously Dedicated Mode

Wenn man nur den Festplatten-Namen, also z.B. `ad0` benutzt, wird bsdlabel(8) eine "dangerously dedicated disk" erstellen, ohne Slices. Das ist ein Zustand, den man meistens nicht hervorrufen möchte. Aus diesem Grund sollte man das *`diskslice`* von bsdlabel(8) noch einmal prüfen, bevor Return gedrückt wird.

13.3.3. Phase drei, /boot/loader

Der boot-loader ist der letzte von drei Schritten im Bootstrap-Prozess und kann im Dateisystem normalerweise unter `/boot/loader` gefunden werden.

Der Loader soll eine interaktive Konfigurations-Schnittstelle mit eingebauten Befehlssatz sein, ergänzt durch einen umfangreichen Interpreter mit einem komplexeren Befehlssatz.

13.3.3.1. Loader Ablauf

Der Loader sucht während seiner Initialisierung nach Konsolen und Laufwerken, findet heraus, von welchem Laufwerk er gerade bootet, und setzt dementsprechend bestimmte Variablen. Dann wird ein Interpreter gestartet, der Befehle interaktiv oder von einem Skript empfangen kann.

Danach liest der Loader die Datei `/boot/loader.rc` aus, welche ihn standardmäßig anweist `/boot/defaults/loader.conf` zu lesen, wo sinnvolle Standardeinstellungen für diverse Variablen festgelegt werden und wiederum `/boot/loader.conf` für lokale Änderungen an diesen Variablen ausgelesen wird. Anschließend arbeitet dann `loader.rc` entsprechend dieser Variablen und lädt die ausgewählten Module und den gewünschten Kernel.

In der Voreinstellung wartet der Loader 10 Sekunden lang auf eine Tastatureingabe und bootet den Kernel, falls keine Taste betätigt wurde. Falls doch eine Taste betätigt wurde wird dem Benutzer eine Eingabeaufforderung angezeigt. Sie nimmt einen Befehlssatz entgegen, der es dem Benutzer erlaubt, Änderungen an Variablen vorzunehmen, Module zu laden, alle Module zu entladen oder schließlich zu booten bzw. neu zu booten.

13.3.3.2. Die eingebauten Befehle des Loaders

Dies sind nur die gebräuchlichsten Befehle. Eine vollständige Beschreibung aller verfügbaren Befehle finden Sie in loader(8).

autoboot *Sekunden*
: Es wird mit dem Booten des Kernels fortgefahren, falls keine Taste in der gegebenen Zeitspanne betätigt wurde. In der gegebenen Zeitspanne, Vorgabe sind 10 Sekunden, wird ein Countdown angezeigt.

boot [-options] [Kernelname]
: Bewirkt das sofortige Booten des Kernels mit allen gegebenen Optionen, oder dem angegebenen Kernelnamen. Das übergeben eines Kernelnamens ist nur nach einem *unload*-Befehl anwendbar, andernfalls wird der zuvor verwendete Kernel benutzt.

boot-conf
: Bewirkt die automatische Konfiguration der Module, abhängig von den entsprechenden Variablen (üblicherweise `kernel`). Dies nur dann sinnvoll, wenn zuvor `unload` benutzt wurde.

help [Thema]
: Zeigt die Hilfe an, die zuvor aus der Datei `/boot/loader.help` gelesen wird. Falls `index` als Thema angegeben wird, wird die Liste der zur Verfügung stehenden Hilfe-Themen angezeigt.

include *`Dateiname`* ...
: Verarbeitet die angegebene Datei. Das Einlesen und Interpretieren geschieht Zeile für Zeile und wird im Falle eines Fehlers umgehend unterbrochen.

load [-t Typ] *`Dateiname`*
: Lädt den Kernel, das Kernel-Modul, oder die Datei des angegebenen Typs. Optionen, die auf *`Dateinamen`* folgen, werden der Datei übergeben.

ls [-l] [Pfad]
: Listet die Dateien im angegebenen Pfad auf, oder das root-Verzeichnis(/), falls kein Pfad angegeben wurde. Die Option `-l` bewirkt, dass die Dateigrößen ebenfalls angezeigt werden.

lsdev [-v]
: Listet alle Geräte auf, für die Module geladen werden können. Die Option `-v` bewirkt eine detailreichere Ausgabe.

lsmod [-v]
: Listet alle geladenen Module auf. Die Option `-v` bewirkt eine detailreichere Ausgabe.

more *`Dateiname`*
: Zeigt den Dateinhalt der angegebenen Datei an, wobei eine Pause alle `LINES` Zeilen gemacht wird.

reboot
: Bewirkt einen umgehenden Neustart des Systems.

set *`Variable`*, set *`Variable=Wert`*
: Setzt die Umgebungsvariablen des Loaders.

unload
: Entlädt sämtliche geladenen Module.

13.3.3.3. Beispiele für die Loader Bedienung

Hier ein paar praktische Beispiele für die Bedienung des Loaders.

- Um den gewöhnlichen Kernel im Single-User Modus zu starten:

```
boot -s
```

- Um alle gewöhnlichen Kernelmodule zu entladen und dann den alten, oder einen anderen Kernel zu laden:

```
unload
load kernel.old
```

  Verwenden Sie `kernel.GENERIC`, um den allgemeinen Kernel zu bezeichnen, der vorinstalliert wird. `kernel.old` bezeichnet den Kernel, der vor dem System-Upgrade installiert war.

Anmerkung

Der folgende Befehl lädt die gewöhnlichen Module mit einem anderen Kernel:

```
unload
set kernel="kernel.old"
boot-conf
```

- Folgendes lädt ein Kernelkonfigurations-Skript (ein automatisiertes Skript, dass dasselbe tut, was der Benutzer normalerweise von Hand an der Eingabeaufforderung durchführen würde):

```
load -t userconfig_script /boot/kernel.conf
```

13.3.3.4. Willkommmensbildschirme während des Bootvorgangs

Contributed by Joseph J. Barbish.

Übersetzt von Benedict Reuschling.

Der Willkommmensbildschirm erzeugt einen alternativen Boot-Bildschirm, der alle Bootmeldungen und Meldungen über startende Dienste bis zur Anzeige des Anmeldeprompts überdeckt.

Es existieren zwei grundlegende Umgebungen in FreeBSD. Die erste ist die altbekannte, auf virtuellen Konsolen basierte Kommandozeile. Nachdem das System den Bootvorgang abgeschlossen hat, wird ein Anmeldebildschirm auf der Konsole anzeigt. Die zweite Umgebung ist die graphische Umgebung von Xorg. In diesem Kapitel finden Sie weitere Informationen zur Installation und Konfiguration eines grafischen Display-Managers und Login-Managers.

13.3.3.4.1. Willkommensbildschirm-Funktionalität

Die Willkommensbildschirm-Funktionalität unterstützt 256-Farben in den Formaten Bitmap (`.bmp`), ZSoft PCX (`.pcx`) oder TheDraw (`.bin`). Die Willkommensbildschirm-Dateien dürfen eine Auflösung von 320 mal 200 Pixeln oder weniger besitzen, damit Standard-VGA Geräte damit arbeiten können.

Damit grössere Bilder bis zu einer maximalen Auflösung von 1024 mal 768 Pixeln verwendet werden können, muss das VESA Modul beim Systemstart geladen werden. Für einen angepassten Kernel muss die `VESA`-Kernelkonfigurationsoption eingefügt werden. VESA-Unterstützung ermöglicht es, einen Willkommensbildschirm als Vollbild auf dem gesamten Bildschirm anzuzeigen.

Wenn der Willkommensbildschirm beim Bootvorgang angezeigt wird, kann dieser jederzeit mit einem beliebigen Tastendruck ausgeschaltet werden.

Der Willkommensbildschirm ist standardmässig so eingestellt, dass er als Bildschirmschoner verwendet wird. Nach einer bestimmten Zeit der Untätigkeit wird der Willkommensbildschirm angezeigt und wechselt durch verschiedene Stufen der Intensität von hell zu einem sehr dunklen Bild und wieder zurück. Das Verhalten des Willkommensbildschirms kann durch hinzufügen einer `saver=`-Zeile in `/etc/rc.conf` geändert werden. Es gibt mehrere eingebaute Bildschirmschoner, die in splash(4) beschrieben werden. Die `saver=`-Option bezieht sich nur auf virtuelle Konsolen und hat keinen Effekt bei grafischen Display-Managern.

Ein paar Nachrichten des Bootloaders und ganz besonders das Menü mit den Bootoptionen und dem Warte-Countdown werden zur Bootzeit angezeigt, selbst wenn der Willkommensbildschirm aktiviert ist.

Dateien mit Beispiel-Willkommensbildschirmen können von der Galerie auf http://artwork.freebsdgr.org heruntergeladen werden. Durch die Installation des Ports sysutils/bsd-splash-changer können Willkommensbildschirme von einer zufällig ausgewählten Sammlung von Bildern bei jedem Neustart angezeigt werden.

13.3.3.4.2. Aktivieren der Willkommensbildschirm-Funktionalität

Die Willkommensbildschirm-Datei `.bmp`, `.pcx` oder `.bin` muss im Wurzelverzeichnis, z.B. `/boot` abgelegt werden.

Für die Standard-Auflösung von 256-Farben, 320 mal 200 Pixel oder weniger, bearbeiten Sie `/boot/loader.conf`, so dass diese die folgenden Zeilen enthält:

```
splash_bmp_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.bmp "
```

Für grössere Video-Auflösungen bis zum Maximum von 1024 mal 768 Pixeln ändern Sie die Datei `/boot/loader.conf`, damit diese die folgenden Zeilen enthält:

```
vesa_load="YES"
splash_bmp_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.bmp "
```

Dieses Beispiel geht davon aus, das `/boot/splash.bmp` als Willkommensbildschirm verwendet wird. Um eine PCX-Datei zu verwenden, benutzen Sie die folgenden Zeilen, inklusive der `vesa_load="YES"`-Zeile, abhängig von der Auflösung:

```
splash_pcx_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.pcx "
```

Beginnend mit FreeBSD 8.3 gibt es die Option, ASCII-Art im `TheDraw`-Format zu benutzen.

```
splash_txt="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.bin "
```

Der Dateiname ist nicht, wie im obigen Beispiel gezeigt, auf „splash“ beschränkt. Es darf ein beliebiger Name sein, solange die Datei von einem der unterstützten Typen ist, beispielsweise *`splash_640x400`*`.bmp` oder *`bluewave`*`.pcx`.

Weitere interessante Optionen für `loader.conf` sind:

`beastie_disable="YES"`
: Diese Option verhindert die Anzeige des Menüs mit den Bootoptionen, aber der Countdown ist immer noch aktiv. Selbst wenn das Bootmenü deaktiviert ist, kann während des Countdowns eine der korrespondierenden Optionen ausgewählt werden.

`loader_logo="beastie"`
: Dies ersetzt die Standardanzeige des Wortes „FreeBSD“. Stattdessen wird auf der rechten Seite des Bootmenüs das bunte Beastie-Logo angezeigt.

Weitere Informationen finden Sie in splash(4), loader.conf(5) und vga(4).

13.4. Kernel Interaktion während des Bootprozesses

Wenn der Kernel einmal geladen ist, entweder durch den Loader oder durch boot2 welches den Loader umgeht, dann überprüft er evtl. vorhandene Boot-Flags und passt sein Verhalten nach Bedarf an.

13.4.1. Kernel Boot-Flags

Es folgt eine Auflistung der gebräuchlichsten Boot-Flags:

`-a`
: Bewirkt, dass während der Kernel-Initialisierung gefragt wird, welches Gerät als Root-Dateisystem gemounted werden soll.

`-C`
: Es wird von CD-ROM gebootet.

`-c`
: Startet UserConfig, das Boot-Zeit Konfigurationsprogramm.

`-s`
: Bewirkt den Start des Single-User Modus.

`-v`
: Zeigt mehr Informationen während des Starten des Kernels an.

Anmerkung

Informationen zu den anderen Boot-Flags finden Sie in boot(8).

13.5. Konfiguration von Geräten

Beigetragen von Tom Rhodes.

Der Boot-Loader liest während des Systemstarts die Datei device.hints(5), die Variablen, auch „device hints“ genannt, zur Konfiguration von Geräten enthält.

Die Variablen können auch mit Kommandos in der Phase 3 des Boot-Loaders bearbeitet werden. Neue Variablen werden mit `set` gesetzt, `unset` löscht schon definierte Variablen und `show` zeigt Variablen an. Variablen aus `/boot/device.hints` können zu diesem Zeitpunkt überschrieben werden. Die hier durchgeführten Änderungen sind nicht permanent und beim nächsten Systemstart nicht mehr gültig.

Nach dem Systemstart können alle Variablen mit kenv(1) angezeigt werden.

Pro Zeile enthält `/boot/device.hints` eine Variable. Kommentare werden durch `#` eingeleitet. Die verwendete Syntax lautet:

```
hint.driver.unit.keyword="value"
```

Der Boot-Loader verwendet die nachstehende Syntax:

```
set hint.driver.unit.keyword= value
```

Der Gerätetreiber wird mit `driver`, die Nummer des Geräts mit `unit` angegeben. `keyword` ist eine Option aus der folgenden Liste:

- `at`: Gibt den Bus, auf dem sich das Gerät befindet, an.
- `port`: Die Startadresse des I/O-Bereichs.
- `irq`: Gibt die zu verwendende Unterbrechungsanforderung (IRQ) an.
- `drq`: Die Nummer des DMA Kanals.
- `maddr`: Die physikalische Speicheradresse des Geräts.
- `flags`: Setzt verschiedene gerätespezifische Optionen.
- `disabled`: Deaktiviert das Gerät, wenn der Wert auf `1` gesetzt wird.

Ein Gerätetreiber kann mehr Optionen, als die hier beschriebenen, besitzen oder benötigen. Es wird empfohlen, die Optionen in der Manualpage des Treibers nachzuschlagen. Weitere Informationen finden Sie in device.hints(5), kenv(1), loader.conf(5) und loader(8).

13.6. Init: Initialisierung der Prozess-Kontrolle

Nachdem der Kernel den Bootprozess abgeschlossen hat, übergibt er die Kontrolle an den Benutzer-Prozess init(8). Dieses Programm befindet sich in `/sbin/init` , oder dem Pfad, der durch die Variable `init_path` im `Loader` spezifiziert wird.

13.6.1. Der automatische Reboot-Vorgang

Der automatische Reboot-Vorgang stellt sicher, dass alle Dateisysteme des Systems konsistent sind. Falls dies nicht der Fall ist und die Inkonsistenz des UFS-Dateisystems nicht durch fsck(8) behebbar ist, schaltet init(8) das System in den Single-User Modus, damit der Systemadministrator sich des Problems annehmen kann.

13.6.2. Der Single-User Modus

Der Wechsel in den Single-User Modus kann durch den automatischen Reboot-Vorgang, das Booten mit der Option `-s`, oder das Setzen der `boot_single` Variable in `loader` erreicht werden.

Weiterhin kann der Single-User Modus aus dem Mehrbenutzermodus heraus durch den Aufruf von shutdown(8), ohne die Option `-r` oder `-h`, erreicht werden.

Falls die System-Konsole (`console`) in `/etc/ttys` auf `insecure` (dt.: unsicher) gesetzt ist, fordert das System zur Eingabe des Passworts von `root` auf, bevor es den Single-User Modus aktiviert.

Beispiel 13.3. Auf insecure gesetzte Konsole in **`/etc/ttys`**

```
# name  getty                           type    status          comments
#
# If console is marked "insecure", then init will ask for the root password
# when going to single-user mode.
console none                            unknown off insecure
```

Anmerkung

Eine Konsole sollte auf `insecure` gesetzt sein, wenn die physikalische Sicherheit der Konsole nicht gegeben ist und sichergestellt werden soll, dass nur Personen, die das Passwort von `root` kennen, den Single-User Modus benutzen können. Daher kann man `insecure` anstatt dem voreingestellten `secure` verwenden, um dieses Maß an Sicherheit zu ergänzen.

13.6.3. Mehrbenutzermodus

Stellt init(8) fest, dass das Dateisystem in Ordnung ist, oder der Benutzer den Single-User Modus beendet, schaltet das System in den Mehrbenutzermodus, in dem dann die Ressourcen Konfiguration des Systems gestartet wird.

13.6.3.1. Ressourcen Konfiguration, rc-Dateien

Das Ressourcen Konfigurationssystem (engl. *resource configuration*, rc) liest seine Standardkonfiguration von `/etc/defaults/rc.conf` und System-spezifische Details von `/etc/rc.conf`. Dann mountet es die Dateisysteme gemäß `/etc/fstab`, startet die Netzwerkdienste, diverse System Daemons und führt schließlich die Start-Skripten der lokal installierten Anwendungen aus.

Lesen Sie rc(8) und ebenso die Skripte an sich, um mehr über das Ressourcen Konfigurationssystem zu erfahren.

13.7. Der Shutdown-Vorgang

Im Falle eines regulären Herunterfahrens durch shutdown(8) führt init(8) `/etc/rc.shutdown` aus, sendet dann sämtlichen Prozessen ein `TERM` Signal und schließlich ein `KILL` Signal an alle Prozesse, die sich nicht rechtzeitig beendet haben.

FreeBSD-Systeme, die Energieverwaltungsfunktionen unterstützen, können mit `shutdown -p now` ausgeschaltet werden. Zum Neustart des Systems wird `shutdown -r now` benutzt. Das Kommando shutdown(8) kann nur von `root` oder Mitgliedern der Gruppe `operator` benutzt werden. Man kann auch halt(8) und reboot(8) verwenden. Weitere Informationen finden Sie in den Hilfeseiten der drei Kommandos.

Anmerkung

Die Energieverwaltungsfunktionen erfordern, dass die Unterstützung für acpi(4) als Modul geladen, oder statisch in einen angepassten Kernel kompiliert wird.

Kapitel 14. Sicherheit

Viel von diesem Kapitel stammt aus der security(7) Manualpage von Matthew Dillon.
Übersetzt von Martin Heinen.

14.1. Übersicht

Dieses Kapitel bietet eine Einführung in die Konzepte der Systemsicherheit. Des weiteren werden einige allgemeine Daumenregeln und einige fortgeschrittene Themen unter FreeBSD behandelt. Viele der hier besprochenen Punkte treffen sowohl auf die Systemsicherheit als auch auf die Internetsicherheit zu. Die Absicherung eines Systems ist unumgänglich, um Daten, geistiges Eigentum, Zeit und vieles mehr vor Hackern und dergleichen zu schützen.

FreeBSD besitzt eine Reihe von Werkzeugen und Mechanismen, um die Integrität und die Sicherheit des Systems und des Netzwerks zu schützen.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie:

- Grundlegende auf FreeBSD bezogene Sicherheitsaspekte kennen.
- Die verschiedenen Verschlüsselungsmechanismen von FreeBSD kennen.
- Wissen, wie Sie ein Einmalpasswörter zur Authentifizierung verwenden.
- TCP-Wrapper für inetd(8) einrichten können.
- Wissen, wie Sie Kerberos unter FreeBSD einrichten.
- Wissen, wie Sie IPsec konfigurieren und ein VPN einrichten.
- Wissen, wie Sie OpenSSH unter FreeBSD konfigurieren und benutzen.
- Wissen, wie Sie ACLs für Dateisysteme benutzen.
- Portaudit anwenden können, um Softwarepakete aus der Ports-Sammlung auf bekannte Sicherheitslücken hin zu überprüfen.
- Mit FreeBSD-Sicherheitshinweisen umgehen können.
- Eine Vorstellung davon haben, was Prozessüberwachung (*Process Accounting*) ist und wie Sie diese Funktion unter FreeBSD aktivieren können.
- Wissen, wie Sie die Ressourcen-Datenbank benutzt, um die Ressourcen für Benutzer zu steuern.

Bevor Sie dieses Kapitel lesen, sollten Sie

- Grundlegende Konzepte von FreeBSD und dem Internet verstehen.

Dieses Buch behandelt weitere Sicherheitsthemen. Beispielsweise werden verbindliche Zugriffskontrollen im Kapitel 16, *Verbindliche Zugriffskontrolle* und Firewalls im Kapitel 30, *Firewalls* besprochen.

14.2. Einführung

Sicherheit ist ein Konzept, das beim Systemadministrator anfängt und aufhört. Obwohl FreeBSD über Sicherheitsfunktionen verfügt, ist die Erstellung und Pflege von zusätzlichen Sicherheitsmechanismen wohl eine der größten Aufgaben eines Systemadministrators.

Zur Systemsicherheit gehört auch die Beschäftigung mit verschiedenen Arten von Angriffen, auch solchen, die versuchen, ein System still zu legen, oder sonst unbrauchbar zu machen ohne `root` zu kompromittieren. Sicherheitsaspekte lassen sich in mehrere Kategorien unterteilen:

1. Denial-of-Service Angriffe.
2. Kompromittierte Accounts.
3. Kompromittierter `root`-Account durch zugängliche Server.
4. Kompromittierter `root`-Account durch kompromittierte Accounts.
5. Einrichten von Hintertüren.

Ein Denial-of-Service DoS-Angriff entzieht einer Maschine Ressourcen, die sie zur Bereitstellung von Diensten benötigt. Meist versuchen DoS-Angriffe die Dienste oder den Netzwerkstack einer Maschine zu überlasten, um so die Maschine auszuschalten oder nicht nutzbar zu machen. Oft können Angriffe auf Dienste durch die Angabe von Optionen verhindert werden, die die Last, die ein Dienst auf das System unter widrigen Umständen ausüben kann, begrenzt. Angriffen auf das Netzwerk ist schwerer zu begegnen. Außer durch Trennen der Internetverbindung ist zum Beispiel einem Angriff mit gefälschten Paketen nicht zu begegnen. Diese Art von Angriff wird das System zwar nicht unbrauchbar machen, kann aber die Internetverbindung sättigen.

Kompromittierte Accounts kommen noch häufiger als DoS-Angriffe vor. Viele Systemadministratoren lassen immer noch unverschlüsselte Dienste laufen, was zur Folge hat, dass das Passwort von Benutzern, die sich von einem entfernten Standort anmelden, leicht ausgespäht werden kann. Ein aufmerksamer Systemadministrator wird die Logdateien über Anmeldungen von entfernten Systemen auf verdächtige Quelladressen, auch für erfolgreiche Anmeldungen, untersuchen.

Allerdings gibt der Zugriff auf einen Account auf einem gut gesicherten und gepflegten System nicht notwendig Zugriff auf den `root`-Account. Diese Unterscheidung ist wichtig, da ein Angreifer, der keinen Zugang zu `root` besitzt, seine Spuren nicht verwischen kann. Er kann höchstens die Dateien des betreffenden Benutzers verändern oder die Maschine stilllegen. Kompromittierte Accounts sind sehr häufig, da Benutzer meist nicht dieselben Vorsichtsmaßnahmen wie Administratoren treffen.

Es gibt viele Wege, Zugang zum `root`-Account eines Systems zu bekommen: Ein Angreifer kann das Passwort von `root` kennen, er kann einen Fehler in einem Server entdecken, der unter `root` läuft und dann über eine Netzwerkverbindung zu diesem Server einbrechen. Oder er kennt einen Fehler in einem SUID-`root` Programm, der es ihm erlaubt, `root` zu werden, wenn er einmal einen Account kompromittiert hat. Wenn ein Angreifer einen Weg gefunden hat, `root` zu werden, braucht er vielleicht keine Hintertür auf dem System installieren.

Sicherheitsmaßnahmen sollten immer in mehreren Schichten angelegt werden. Die Schichten können wie folgt eingeteilt werden:

1. Absichern von `root`-Accounts.
2. Absichern von unter `root` laufenden Servern und SUID/SGID Programmen.
3. Absichern von Benutzer-Accounts.
4. Absichern der Passwort-Datei.
5. Absichern des Kernels, der Geräte und von Dateisystemen.
6. Schnelles Aufdecken von unbefugten Veränderungen des Systems.
7. Paranoia.

Die einzelnen Punkte der obigen Liste werden im nächsten Abschnitt genauer behandelt.

14.3. Absichern von FreeBSD

Dieser Abschnitt behandelt die im letzten Abschnitt erwähnten Methoden zur Absicherung eines FreeBSD-Systems.

14.3.1. Absichern von `root` und Accounts

Auf den meisten Systemen ist `root` ein Passwort zugewiesen. Sie sollten *immer* davon ausgehen, dass dieses Passwort kompromittiert ist. Das heißt nicht, dass Sie das Passwort entfernen sollten, da es meist für den Konsolenzugriff notwendig ist. Vielmehr heißt es, dass Sie das Passwort nicht außerhalb der Konsole, auch nicht zusammen mit su(1), verwenden sollten. Stellen Sie sicher, dass die PTYs in `ttys` als `insecure` markiert sind und damit Anmeldungen von `root` verboten sind. In FreeBSD ist die Anmeldung für `root` über ssh(1) in der Voreinstellung deaktiviert, da in `/etc/ssh/sshd_config` `PermitRootLogin` auf `no` gesetzt ist. Beachten Sie jede Zugriffsmethode – Dienste wie FTP werden oft vergessen. Nur an der Systemkonsole sollte ein direktes Anmelden als `root` möglich sein.

Natürlich muss ein Systemadministrator `root`-Zugriff erlangen können. Dieser sollte aber durch zusätzliche Passwörter geschützt sein. Ein Weg, Zugang zu `root` zu ermöglichen, ist es, berechtigte Mitarbeiter in `/etc/group` in die Gruppe `wheel` aufzunehmen. Die Personen dieser Gruppe können mit `su` zu `root` wechseln. Nur die Mitarbeiter, die tatsächlich `root`-Zugriff benötigen, sollten in die Gruppe `wheel` aufgenommen werden. Wenn Sie Kerberos für die Authentifizierung benutzen, erstellen Sie `.k5login` im Heimatverzeichnis von `root`, damit su(1) verwendet werden kann, ohne jemanden in `wheel` aufnehmen zu müssen.

Um ein Konto komplett zu sperren, verwenden Sie pw(8):

```
#pw lock staff
```

Danach ist es diesem Benutzer nicht mehr möglich (auch nicht mit ssh(1)), sich anzumelden.

Eine weitere Möglichkeit, bestimmte Benutzer zu sperren, ist es, das verschlüsselte Passwort durch das Zeichen „*“ zu ersetzen. Da ein verschlüsseltes Passwort niemals diesem Zeichen entsprechen kann, kann sich der betroffene Benutzer ebenfalls nicht mehr anmelden. Beispielsweise müsste dazu das Konto

```
foobar:R9DT/Fa1/LV9U:1000:1000::0:0:Foo Bar:/home/foobar:/usr/local/bin/tcsh
```

mit vipw(8) wie folgt abgeändert werden:

```
foobar:*:1000:1000::0:0:Foo Bar:/home/foobar:/usr/local/bin/tcsh
```

Diese Änderung hindert den Benutzer `foobar` daran, sich auf konventionellem Wege am System anzumelden. Diese Maßnahmen greifen allerdings nicht, wenn das betroffene System auch eine Anmeldung über Kerberos oder ssh(1) erlaubt.

Diese Sicherheitsmechanismen setzen voraus, dass sich Benutzer von einer restriktiven Maschine auf einer weniger restriktiven Maschine anmelden. Wenn zum Beispiel auf dem Hauptrechner alle möglichen Arten von Servern laufen, so sollten auf der Workstation keine Server laufen. Um die Workstation vernünftig abzusichern, sollten darauf so wenig Server wie möglich bis hin zu keinem Server laufen. Sie sollten zudem über einen Bildschirmschoner verfügen, der mit einem Passwort gesichert ist. Natürlich kann ein Angreifer, der physikalischen Zugang zu einer Maschine hat, jede Art von Sicherheitsmechanismen umgehen. Beachten Sie, dass der Großteil der Einbrüche über das Netzwerk erfolgt und die Einbrecher keinen Zugang zu der Maschine besitzen.

Mit Kerberos können Sie das Passwort eines Mitarbeiters an einer Stelle ändern und alle Maschinen, auf denen der Mitarbeiter einen Account hat, beachten die Änderung sofort. Wird der Account eines Mitarbeiters einmal kompromittiert, so sollte die Fähigkeit, das Passwort mit einem Schlag auf allen Maschinen zu ändern, nicht unterschätzt werden. Mit einzelnen Passwörtern wird es schwierig, das Passwort auf N Maschinen zu ändern. Mit Kerberos können Sie auch Beschränkungen für Passwörter festlegen: Nicht nur das Ticket kann nach einiger Zeit ungültig werden, Sie können auch festlegen, dass ein Benutzer nach einer bestimmten Zeit das Passwort wechseln muss.

14.3.2. Absichern von unter `root` laufenden Servern und SUID/SGID Programmen

Ein kluger Systemadministrator lässt nur die wirklich benötigten Dienste laufen und ist sich darüber im Klaren, dass Server von Dritten oft die fehleranfälligsten sind. Lassen Sie keine Server laufen, die Sie vorher nicht genau

überprüft haben. Denken Sie zweimal darüber nach, bevor Sie einen Dienst als `root` laufen lassen. Viele Daemonen können unter einem separaten Dienstkonto, oder in einem Sandkasten ausgeführt werden. Aktivieren Sie keine unsicheren Dienste, wie telnetd(8) oder rlogind(8).

Ein weiteres potentielles Risiko sind SUID- und SGID-Programme. Die meisten dieser Programme, wie rlogin(1) stehen in `/bin`, `/sbin`, `/usr/bin`, oder `/usr/sbin` zur Verfügung. Obwohl nichts 100% sicher ist, können Sie davon ausgehen, dass die SUID- und SGID-Programme des Basissystems ausreichend sicher sind. Es wird empfohlen, den Zugriff auf SUID-Programme mit einer Gruppe, auf die nur Mitarbeiter zugreifen können, zu beschränken. SUID-Programme, die niemand benutzt, sollten gelöscht werden. SGID-Programme sind vergleichbar gefährlich. Wenn ein Einbrecher Zugriff auf SGID-`kmem` Programm erhält, kann er vielleicht `/dev/kmem` und damit die verschlüsselte Passwortdatei lesen. Dies kompromittiert unter Umständen jeden Account, der mit einem Passwort geschützt ist. Alternativ kann ein Einbrecher, der in die Gruppe `kmem` eingebrochen ist, die Tastendrücke auf PTYs verfolgen. Dies schließt auch PTYs mit ein, auf denen sich ein Benutzer mit sicheren Methoden anmeldet. Ein Einbrecher, der Zugriff auf die `tty` Gruppe hat, kann auf fast jeden Terminal anderer Benutzer schreiben. Wenn der Benutzer einen Terminal-Emulator benutzt, der über eine Tastatur-Simulation verfügt, könnte der Angreifer Daten generieren, die den Terminal veranlassen, ein Kommando unter diesem Benutzer laufen zu lassen.

14.3.3. Absichern von Benutzer-Accounts

Accounts sind für gewöhnlich sehr schwierig abzusichern. Seien Sie daher aufmerksam bei der Überwachung der Benutzerkonten. Die Verwendung von ssh(1) und Kerberos erfordert zwar zusätzliche Administration und technische Unterstützung, ist aber verglichen mit der verschlüsselten Passwort-Datei die bessere Lösung.

14.3.4. Absichern der Passwort-Datei

Der einzig sichere Weg ist, so viele Accounts wie möglich als ungültig zu markieren und ssh(1) oder Kerberos zu benutzen, um auf sie zuzugreifen. Obwohl die Datei `/etc/spwd.db`, die die verschlüsselten Passwörter enthält, nur von `root` gelesen werden kann, mag ein Angreifer lesenden Zugriff auf diese Datei erlangen, ohne die Fähigkeit sie auch zu beschreiben.

Überwachungsskripten sollten Änderungen an der Passwort-Datei melden. Dies wird in Überprüfen der Integrität von Dateien beschrieben.

14.3.5. Absichern des Kernels, der Geräte und von Dateisystemen

Die meisten modernen Kernel haben einen Gerätetreiber, der es erlaubt, Pakete abzuhören. Unter FreeBSD wird das Gerät `bpf` genannt. Dieses Gerät ist für DHCP erforderlich, kann aber in der Kernelkonfigurationsdatei entfernt werden, wenn das System kein DHCP anbietet.

Auch wenn `bpf` deaktiviert ist, müssen Sie sich immer noch um `/dev/mem` und `/dev/kmem` sorgen. Außerdem kann der Angreifer immer noch auf die rohen Geräte (*raw devices*) schreiben. Ein Angreifer könnte kldload(8) benutzen, um sein eigenes `bpf` oder ein anderes zum Abhören geeignetes Gerät in den laufenden Kernel einzubringen. Um diese Probleme zu vermeiden, lassen Sie den Kernel auf einem höheren Sicherheitslevel laufen, mindestens auf securelevel 1.

Das Securelevel des Kernels kann auf verschiedene Wege gesetzt werden. Der einfachste Weg, den Securelevel des laufenden Kernels zu erhöhen, ist das Setzen von `kern.securelevel`:

```
# sysctl kern.securelevel=1
```

In der Voreinstellung bootet der FreeBSD Kernel mit einem Securelevel von -1. Der Securelevel wird solange bei -1 bleiben, bis er entweder durch den Administrator oder von init(8) durch einen Eintrag im Startskript verändert wird. Der Securelevel kann während des Systemstarts durch das Setzen von `kern_securelevel_enable` auf `YES` und der Wert der Variable `kern_securelevel` auf den gewünschten Securelevel in der `/etc/rc.conf` erhöht werden.

Sobald der Securelevel auf den Wert 1 oder höher gesetzt ist, werden die append-only und die unveränderlichen Dateien geschützt, die Flags können nicht abgeschaltet werden und der Zugriff auf raw Devices ist verboten. Höhe-

re Levels verbieten sogar noch weitere Aktionen. Eine vollständige Beschreibung aller Securelevels finden Sie in security(7) und init(8).

Anmerkung

Das Erhöhen des Securelevels auf 1 oder höher kann einige Probleme mit Xorg, verursachen, da der Zugriff auf `/dev/io` geblockt wird, ebenso die Installation von FreeBSD aus den Quellen, da der `installworld` Teil zeitweilig die append-only und die unveränderlichen Flags einiger Dateien zurücksetzen muss. Manchmal kann es, wie bei Xorg, durch das sehr frühe Starten von xdm(1) im Boot Prozess möglich sein, dies zu umgehen, wenn der Securelevel noch niedrig genug ist. Workarounds wie dieser sind nicht für alle Securelevels und für alle Einschränkungen, die sie schaffen, möglich. Ein bisschen Vorausplanung ist eine gute Idee. Das Verständnis für die Beschränkungen, die durch jedes Securelevel verursacht werden, ist wichtig, da sie die einfache Benutzung des Systems verschlechtern. Es vereinfacht auch die Wahl einer Standardeinstellung und schützt vor Überraschungen.

Wenn das Securelevel des Kernel auf einen Wert von 1 oder höher gesetzt ist, kann es sinnvoll sein das `schg` Flag auf kritische Startdateien, Verzeichnisse und Skripte zu setzen. Ein weniger strenger Kompromiss ist es, das System auf einem höheren Securelevel laufen zu lassen, aber keine `schg` Flags für alle Systemdateien und Verzeichnisse zu setzen. Eine andere Möglichkeit ist es, die Verzeichnisse `/` und `/usr` read-only zu mounten. Es sei darauf hingewiesen, dass Sie nicht vor lauter Überlegen das Wichtigste, nämlich die Entdeckung eines Eindringens, vergessen.

14.3.6. Überprüfen der Integrität von Dateien

Sie können die Systemkonfiguration und die Dateien nur so weit schützen, wie es die Benutzbarkeit des Systems nicht einschränkt. Wenn Sie zum Beispiel mit `chflags` die Option `schg` auf die meisten Dateien in `/` und `/usr` setzen, kann das Ihre Arbeit mehr behindern als nützen. Die Maßnahme schützt zwar die Dateien, schließt aber auch eine Möglichkeit, Veränderungen zu entdecken, aus. Sicherheitsmaßnahmen sind nutzlos, oder schlimmer noch, vermitteln ein falsches Gefühl von Sicherheit, wenn der potentielle Angreifer nicht entdeckt wird. Die Aufgabe besteht nicht darin, den Angreifer aufzuhalten, sondern seine Angriffe zu verzögern, um ihn dann auf frischer Tat zu ertappen.

Der beste Weg, einen Einbruch zu entdecken, ist es, nach veränderten, fehlenden oder unerwarteten Dateien zu suchen. Der wiederum beste Weg, nach veränderten Dateien zu suchen, ist es, die Suche von einem anderen (oft zentralen) besonders geschützten System durchzuführen. Es ist wichtig, dass Ihre Sicherheitsüberprüfungen vor einem Angreifer verborgen bleiben und daher sind sie auf einem besonders geschützten System gut aufgehoben. Um dies optimal auszunutzen, müssen Sie dem besonders geschützten System Zugriffsrechte auf die zu schützenden Systeme geben. Sie können die Dateisysteme der zu schützenden Systeme schreibgeschützt für das besonders geschützte System exportieren, oder Sie können der besonders geschützten Maschine SSH auf die anderen Maschinen erlauben, indem Sie SSH-Schlüsselpaare installieren. Mit Ausnahme des verursachten Netzwerkverkehrs ist die NFS-Methode die am wenigsten sichtbare. Sie erlaubt es Ihnen, nahezu unentdeckt die Dateisysteme der Clients zu beobachten. Wenn Ihr besonders geschütztes System mit den Clients über einen Switch verbunden ist, ist die NFS-Methode oft das Mittel der Wahl. Wenn das besonders geschützte System allerdings mit einem Hub verbunden ist, oder der Zugriff über mehrere Router geschieht, ist die NFS-Methode aus der Netzwerksicht zu unsicher. In einem solchen Fall ist SSH besser geeignet, auch wenn es deutliche Spuren hinterlässt.

Wenn das besonders geschützte System lesenden Zugriff auf die Clients hat, müssen Sie Skripten schreiben, die die Überwachung durchführen. Wenn Sie die NFS-Methode verwenden, können Sie dazu einfache Systemwerkzeuge wie find(1) und md5(1) benutzen. Am besten berechnen Sie einmal am Tag MD5-Prüfsummen der Dateien, Konfigurationsdateien in `/etc` und `/usr/local/etc` sollten öfter überprüft werden. Wenn Unstimmigkeiten zwischen den auf der besonders geschützten Maschine gehaltenen MD5-Prüfsummen und den ermittelten Prüfsummen festgestellt werden, sollte Ihr System einen Systemadministrator benachrichtigen, der den Unstimmigkeiten dann nachgehen sollte. Ein gutes Skript überprüft das System auch auf verdächtige SUID-Programme sowie gelöschte oder neue Dateien in `/` und `/usr`.

Wenn Sie SSH anstelle von NFS benutzen, wird das Erstellen der Skripten schwieriger. Sie müssen die Skripten und die Programme wie `find` mit `scp` auf den Client kopieren. Damit machen Sie die Überprüfung für einen Angreifer sichtbar. Außerdem kann der SSH-Client auf dem Zielsystem schon kompromittiert sein. Zusammenfassend kann der Einsatz von SSH nötig sein, wenn Sie über ungesicherte Verbindungen arbeiten, aber der Umgang mit dieser Methode ist auch sehr viel schwieriger.

Ein gutes Sicherheitsskript wird auch Dateien von Benutzern, die den Zugriff auf ein System ermöglichen, wie `.rhosts`, `.shosts`, `.ssh/authorized_keys` usw., auf Veränderungen untersuchen, die über die Möglichkeiten einer Überprüfung mit `MD5` (die ja nur Veränderungen erkennen kann) hinausgehen.

Wenn Sie über große Partitionen verfügen, kann es zu lange dauern, jede Datei zu überprüfen. In diesem Fall sollten Sie beim Einhängen des Dateisystems Optionen setzen, die das Ausführen von SUID-Programmen verbieten. mount(8) stellt dazu `nosuid` zur Verfügung. Sie sollten diese Dateien aber trotzdem mindestens einmal die Woche überprüfen, da das Ziel dieser Schicht das Aufdecken eines Einbruchs, auch wenn er nicht erfolgreich war, ist.

Die Prozessüberwachung (siehe accton(8)) des Betriebssystems steht ein günstiges Werkzeug zur Verfügung, dass sich bei der Analyse eines Einbruchs als nützlich erweisen kann. Insbesondere können Sie damit herausfinden, wie der Einbrecher in das System eingedrungen ist, vorausgesetzt die Dateien der Prozessüberwachung sind noch alle intakt.

Schließlich sollten die Sicherheitsskripten die Logdateien analysieren. Dies sollte so sicher wie möglich durchgeführt werden, nützlich ist das Schreiben von Logdateien auf entfernte Systeme mit `syslog`. Ein Einbrecher wird versuchen, seine Spuren zu verwischen. Die Logdateien sind wichtig für den Systemadministrator, da er aus ihnen den Zeitpunkt und die Art des Einbruchs bestimmen kann. Eine Möglichkeit, die Logdateien unverändert aufzuheben, ist es, die Systemkonsole auf einen seriellen Port zu legen und die Informationen dort von einer gesicherten Maschine auszulesen.

14.3.7. Paranoia

Es schadet nicht, ein bisschen paranoid zu sein. Grundsätzlich darf ein Systemadministrator jede Sicherheitsmaßnahme treffen, die die Bedienbarkeit des Systems nicht einschränkt. Er kann auch Maßnahmen treffen, die die Bedienbarkeit einschränken, wenn er diese vorher genau durchdacht hat. Was noch wichtiger ist: Halten Sie sich nicht sklavisch an dieses Dokument, sondern führen Sie eigene Maßnahmen ein, um nicht einem künftigen Angreifer, der auch Zugriff auf dieses Dokument hat, alle Ihre Methoden zu verraten.

14.3.8. Denial-of-Service Angriffe

Dieser Abschnitt behandelt Denial-of-Service Angriffe (DoS). Ein DoS-Angriff findet typischerweise auf der Paketebene statt. Während Sie nicht viel gegen moderne Angriffe mit falschen Paketen, die das Netzwerk sättigen, ausrichten können, können Sie sehr wohl den Schaden begrenzen, den solche Angriffe verursachen können und insbesondere einen kompletten Serverausfall verhindern, indem Sie beispielsweise folgende Vorkehrungen treffen:

1. Begrenzen von `fork()` Aufrufen.
2. Begrenzen von Sprungbrett-Angriffen (ICMP response Angriffen, ping zu Broadcast-Adressen usw.).
3. Kernel-Cache für Routen.

Ein häufiger DoS-Angriff gegen forkende Server versucht den Server dazu zu bringen, solange neue Prozesse zu starten, bis das System den ganzen Speicher und alle Dateideskriptoren verbraucht hat, was dann zu einem Ausfall des Servers führt. inetd(8) besitzt einige Optionen, um diese Art von Angriffen zu begrenzen. Beachten Sie bitte, dass es möglich ist, einen Ausfall einer Maschine zu verhindern, doch ist es generell nicht möglich, den Ausfall eines Dienstes bei dieser Art von Angriffen zu verhindern. Lesen Sie sich bitte die Manualpages von inetd gut durch und achten Sie speziell auf die Optionen `-c`, `-C` und `-R`. Angriffe mit gefälschten IP-Adressen umgehen `-C`, so dass normalerweise eine Kombination der Optionen benutzt werden muss. Manche Server, die nicht von inetd gestartet werden, besitzen Optionen, um den Start über `fork()` einzuschränken.

Sendmail besitzt die Option `-OMaxDaemonChildren`, die besser als die eingebauten Optionen zur Begrenzung der Systemauslastung funktioniert. Sie sollten beim Start von Sendmail `MaxDaemonChildren` so hoch setzen, dass Sie die

erwartete Auslastung gut abfangen können. Allerdings sollten Sie den Wert nicht so hoch setzen, dass der Rechner über seine eigenen Füße fällt. Es ist auch klug, Sendmail im Queue-Modus (`-ODeliveryMode=queued`) laufen zu lassen. Der Daemon (`sendmail -bd`) sollte getrennt von den Queue-Läufen (`sendmail -q15m`) laufen. Wenn Sie trotzdem eine sofortige Auslieferung der Post wünschen, können Sie die Queue in einem geringeren Intervall, etwa `-q1m`, abarbeiten. Geben Sie für *dieses* Sendmail aber einen vernünftigen Wert für `MaxDaemonChildren` an, um Fehler zu verhindern.

Syslogd kann direkt angegriffen werden. Daher empfehlen wir Ihnen unbedingt die Option `-s` zu benutzen. Sollte das nicht möglich sein, benutzen Sie bitte `-a`.

Vorsicht ist auch mit Diensten geboten, die automatisch eine Rückverbindung eröffnen, wie der reverse-identd der TCP-Wrapper. Diese Funktion der TCP-Wrapper sollten Sie normalerweise nicht benutzen.

Es empfiehlt sich sehr, interne Dienste vor externen Zugriffen durch eine Firewall an der Grenze Ihres Netzwerks zu schützen. Dahinter steckt mehr die Idee, das Netzwerk vor Überlastung durch Angriffe von außen zu schützen, als interne Dienste vor einem `root`-Zugriff aus dem Netz zu schützen. Konfigurieren Sie immer eine Firewall, die alle Zugriffe blockiert, das heißt blockieren Sie *alles* außer den Ports A, B, C, D und M-Z. Damit können Sie Zugriffe auf alle niedrigen Ports blockieren und Zugriffe auf spezielle Dienste wie named, wenn Sie den primären Namensdienst für eine Zone anbieten, ntalkd oder Sendmail erlauben. Wenn Sie die Firewall so konfigurieren, das sie in der Voreinstellung alle Zugriffe erlaubt, ist es sehr wahrscheinlich, dass Sie vergessen, eine Reihe von Diensten zu blockieren bzw. einen internen Dienst einführen und dann vergessen die Firewall zu aktualisieren. Sie können immer die höheren Portnummern öffnen, ohne die niedrigen Portnummern, die nur von `root` benutzt werden dürfen, zu kompromittieren. Beachten Sie bitte auch, dass es FreeBSD erlaubt, die Portnummern, die für dynamische Verbindungen zur Verfügung stehen, zu konfigurieren. Mit `sysctl` lassen sich verschiedene Bereiche der `net.inet.ip.portrange` Variablen setzen (eine Liste erhalten Sie mit `sysctl -a | fgrep portrange`). So können Sie zum Beispiel die Portnummern 4000 bis 5000 für den normalen Bereich und die Nummern 49152 bis 65535 für den hohen Bereich vorsehen. Dies erleichtert Ihnen die Konfiguration der Firewall, da Sie nun Zugriffe auf Ports unterhalb von 4000, mit Ausnahme der Dienste, die von außen erreichbar sein sollen, blockieren können.

Eine andere Form eines DoS-Angriffs nutzt einen Server als Sprungbrett, der Server wird dabei so angegriffen, dass seine Antworten ihn selber, das lokale Netzwerk oder einen anderen Server überlasten. Der am häufigsten verwendete Angriff dieser Art ist der *ICMP ping broadcast Angriff*. Der Angreifer fälscht dazu ping-Pakete, die zu der Broadcast-Adresse Ihres LANs gesendet werden, indem er darin als Quelladresse die Adresse des Opfers einsetzt. Wenn die Router an der Grenze Ihres Netzwerks ping-Pakete auf Broadcast-Adressen nicht abwehren, wird Ihr LAN genügend Netzwerkverkehr generieren, um das Ziel des Angriffs zu überlasten. Dies kann besonders effektiv sein, wenn der Angreifer diese Methode mit mehreren Dutzend Broadcast-Adressen über mehrere Netzwerke einsetzt. Es wurden schon Broadcast-Angriffe mit über 120 Megabit pro Sekunde gemessen. Ein zweiter Sprungbrett-Angriff wird gegen das Fehlerbehandlungssystem von ICMP eingesetzt. Indem ein Angreifer Pakete konstruiert, die eine ICMP-Fehlermeldung hervorrufen, kann er das einkommende Netzwerk des Servers sättigen und diesen wiederum veranlassen sein ausgehendes Netzwerk mit ICMP-Antworten zu sättigen. Diese Art des Angriffs kann den kompletten Speicher des Servers aufbrauchen und damit den Server stilllegen, insbesondere wenn der Server nicht in der Lage ist, die generierten ICMP-Antworten schnell genug abzuführen. Verwenden Sie die sysctl-Variable `net.inet.icmp.icmplim`, um die Auswirkungen solcher Angriffe zu begrenzen. Die letzte weit verbreitete Form von Sprungbrett-Angriffen verwendet interne inetd-Dienste wie den UDP echo-Dienst. Der Angreifer fälscht dazu einfach ein UDP-Paket, indem er als Quellport den echo-Port von Server A und als Zielport den echo-Port von Server B angibt, wobei beide Server in Ihrem LAN stehen. Die beiden Server werden nun dieses Paket zwischen sich hin und her schicken. Der Angreifer kann die beiden Server und das LAN einfach damit überlasten, dass er mehrere Pakete dieser Art generiert. Ähnliche Probleme gibt es mit dem internen chargen-Port, daher sollten Sie die internen inetd-Testdienste abstellen.

Gefälschte IP-Pakete können dazu benutzt werden, den Kernel-Cache für Routen zu überlasten. Schauen Sie sich bitte die `sysctl`-Parameter `net.inet.ip.rtexpire`, `rtminexpire` und `rtmaxcache` an. Ein Angriff der gefälschte Pakete mit zufälligen Quelladressen einsetzt, bewirkt, dass der Kernel eine Route im Route-Cache anlegt, die Sie sich mit `netstat -rna | fgrep W3` ansehen können. Diese Routen verfallen für gewöhnlich nach 1600 Sekunden. Wenn der Kernel feststellt, dass die Routingtabelle im Cache zu groß geworden ist, wird er dynamisch den Wert von `rtexpire` verringern. Dieser Wert wird aber nie kleiner werden als `rtminexpire`. Daraus ergeben sich zwei Probleme:

1. Der Kernel reagiert nicht schnell genug, wenn ein Server mit einer niedrigen Grundlast plötzlich angegriffen wird.

2. `rtminexpire` ist nicht klein genug, um einen anhaltenden Angriff zu überstehen.

Wenn Ihre Server über eine T3 oder eine noch schnellere Leitung mit dem Internet verbunden sind, ist es klug, mit sysctl(8) die Werte für `rtexpire` und `rtminexpire` händisch zu setzen. Setzen Sie bitte keinen der Werte auf Null, außer Sie wollen die Maschine zum Erliegen bringen. Ein Wert von 2 Sekunden für beide Parameter sollte ausreichen, um die Routingtabelle vor einem Angriff zu schützen.

14.3.9. Anmerkungen zum Zugriff mit Kerberos und SSH

Es gibt ein paar Punkte, die Sie beachten sollten, wenn Sie Kerberos oder SSH einsetzen wollen. Kerberos 5 ist ein ausgezeichnetes Authentifizierungsprotokoll. Leider gibt es Fehler in den für Kerberos angepassten Versionen von telnet und rlogin, die sie ungeeignet für den Umgang mit binären Datenströmen machen. Weiterhin verschlüsselt Kerberos Ihre Sitzung nicht, wenn Sie nicht die `-x` Option verwenden, mit SSH wird dagegen alles verschlüsselt.

Ein Problem mit SSH sind Weiterleitungen von Verbindungen. Wenn Sie von einer sicheren Maschine, auf der sich Ihre Schlüssel befinden, eine Verbindung zu einer ungesicherten Maschine aufmachen, wird für die Dauer der Sitzung ein Port für Weiterleitungen geöffnet. Ein Angreifer, der auf der unsicheren Maschine Zugang zu `root` hat, kann diesen Port benutzen, um Zugriff auf andere Maschinen zu erlangen, die mit Ihren Schlüsseln zugänglich sind.

Wir empfehlen Ihnen, für die Logins Ihrer Mitarbeiter immer SSH zusammen mit Kerberos einzusetzen. Damit reduzieren Sie die Abhängigkeit von potentiell gefährdeten Schlüsseln und schützen gleichzeitig die Passwörter mit Kerberos. SSH-Schlüsselpaare sollten nur für automatisierte Aufgaben von einem besonders gesicherten Server eingesetzt werden (Kerberos kann für diese Art von Aufgaben nicht eingesetzt werden). Weiterhin empfehlen wir Ihnen, das Weiterreichen von Schlüsseln in der SSH-Konfiguration abzustellen bzw. die `from=IP/DOMAIN` Option in `authorized_keys` zu verwenden, die den Schlüssel nur von bestimmten Maschinen aus nutzbar macht.

14.4. DES, Blowfish, MD5, SHA256, SHA512 und Crypt

Teile umgeschrieben und aktualisiert von Bill Swingle.

Jedem Benutzer eines UNIX® Systems ist ein Passwort zugeordnet. Es scheint offensichtlich, dass das Passwort nur dem Benutzer und dem System bekannt sein muss. Um die Passwörter geheim zu halten, werden sie mit einer nicht umkehrbaren Hash-Funktion verschlüsselt, das heißt sie können leicht verschlüsselt aber nicht entschlüsselt werden. Was wir gerade als offensichtlich dargestellt haben, ist also nicht wahr: Das Betriebssystem kennt das Passwort *wirklich* nicht, es kennt nur das *verschlüsselte* Passwort. Die einzige Möglichkeit, das originale Passwort herauszufinden, besteht darin, alle möglichen Passwörter auszuprobieren (*brute force* Suche).

Zu der Zeit als UNIX® entstanden ist, war die einzig sichere Möglichkeit Passwörter zu verschlüsseln, leider DES (Data Encryption Standard). Für die Einwohner der USA stellte das kein Problem dar, aber da der Quellcode von DES nicht aus den USA exportiert werden durfte, musste ein Weg gefunden werden, der die Gesetze der USA nicht verletzte und gleichzeitig die Kompatibilität mit anderen UNIX® Systemen, die immer noch DES benutzten, wahrte.

Die Lösung bestand darin, die Verschlüsselungsbibliotheken aufzuspalten. Benutzer in den USA konnten die DES-Bibliotheken installieren und nutzen. In der Grundeinstellung benutzt FreeBSD MD5 als Verschlüsselungsmethode, das exportiert werden durfte und damit von jedem genutzt werden konnte. Es wird davon ausgegangen, dass MD5 sicherer als DES ist, so dass DES nur aus Kompatibilitätsgründen installiert werden sollte.

14.4.1. Erkennen der Verschlüsselungsmethode

Die derzeit unterstützten Hash-Funktionen sind DES, MD5, Blowfish, SHA256 und SHA512. In der Voreinstellung benutzten FreeBSD 9.1 und neuere Versionen SHA512 zur Verschlüsselung von Passwörtern. Ältere Versionen benutzen standardmäßig MD5.

Sie können leicht herausfinden, welche Verschlüsselungsmethode von FreeBSD verwendet wird. Ein Weg besteht darin, die verschlüsselten Passwörter in `/etc/master.passwd` zu untersuchen. Passwörter, die mit MD5 verschlüsselt wurden, sind länger als die mit DES verschlüsselten und beginnen mit den Zeichen `$1$`. Passwörter, die mit `$2a$` anfangen, wurden mit der Blowfish-Funktion verschlüsselt. DES Passwörter besitzen keine offensichtlichen Merkmale, an denen sie identifiziert werden könnten. Sie sind aber kürzer als MD5-Passwörter und sind in einem 64 Zeichen umfassenden Alphabet kodiert, das das `$`-Zeichen nicht enthält. Ein relativ kurzes Passwort, das nicht mit einem `$`-Zeichen anfängt, ist wahrscheinlich ein DES-Passwort. SHA256 und SHA512 beginnen mit dem Zeichen `$6$`.

Die Verschlüsselungsmethode für neue Passwörter wird durch `passwd_format` in `/etc/login.conf` bestimmt. Der Wert dieser Variablen kann entweder `des`, `md5`, `blf`, `sha256` oder `sha512` sein. Näheres schlagen Sie bitte in login.conf(5) nach.

14.5. Einmalpasswörter

In der Voreinstellung unterstützt FreeBSD *One-time Passwords in Everything* (OPIE), das in der Voreinstellung MD5-Hash-Funktionen einsetzt.

Es gibt drei verschiedene Arten von Passwörtern. Das erste ist das normales UNIX®- oder Kerberos-Passwort. Das zweite ist das Einmalpasswort, das von `opiekey` generiert und von `opiepasswd` und dem Anmeldeprompt akzeptiert wird. Das dritte Passwort ist das „geheime Passwort“, das mit `opiekey` (manchmal auch mit `opiepasswd`) zum Erstellen der Einmalpasswörter verwendet wird.

Das geheime Passwort steht in keiner Beziehung zum UNIX®-Passwort. Beide können gleich sein, obwohl das nicht empfohlen wird. Die geheimen Passwörter von OPIE sind nicht auf eine Länge von 8 Zeichen, wie alte UNIX® Passwörter[1], beschränkt. Gebräuchlich sind Passwörter, die sich aus sechs bis sieben Wörtern zusammensetzen. Das OPIE-System arbeitet vollständig unabhängig von den auf UNIX®-Systemen verwendeten Passwort-Mechanismen.

Neben dem Passwort gibt es noch zwei Werte, die für OPIE wichtig sind. Der erste ist der „Initialwert“ (engl. *seed* oder *key*), der aus zwei Buchstaben und fünf Ziffern besteht. Der zweite Wert ist der „Iterationszähler“, eine Zahl zwischen 1 und 100. OPIE generiert das Einmalpasswort, indem es den Initialwert und das geheime Passwort aneinander hängt und dann die MD5-Hash-Funktion so oft, wie durch den Iterationszähler gegeben, anwendet. Das Ergebnis wird in sechs englische Wörter umgewandelt, die das Einmalpasswort ergeben. Das Authentifizierungssystem (meistens PAM) merkt sich das zuletzt benutzte Einmalpasswort und der Benutzer ist authentifiziert, wenn die Hash-Funktion des Passworts dem vorigen Passwort entspricht. Da nicht umkehrbare Hash-Funktionen benutzt werden, ist es unmöglich, aus einem bekannten Passwort weitere gültige Einmalpasswörter zu berechnen. Der Iterationszähler wird nach jeder erfolgreichen Anmeldung um eins verringert und stellt so die Synchronisation zwischen Benutzer und Login-Programm sicher. Wenn der Iterationszähler den Wert 1 erreicht, muss OPIE neu initialisiert werden.

Es gibt ein paar Programme, die in diesen Prozess einbezogen werden. opiekey(1) akzeptiert einen Iterationszähler, einen Initialwert und ein geheimes Passwort. Daraus generiert es ein Einmalpasswort oder eine Liste von Einmalpasswörtern. opiepasswd(1) wird benutzt, um Passwörter, Iterationszähler oder Initialwerte zu ändern. Als Parameter verlangt es entweder ein geheimes Passwort oder einen Iterationszähler oder einen Initialwert und ein Einmalpasswort. opieinfo(1) hingegen gibt den momentanen Iterationszähler und Initialwert eines Benutzers aus, den es aus `/etc/opiekeys` ermittelt.

Es gibt vier verschiedene Arten von Tätigkeiten. Zuerst wird erläutert, wie opiepasswd(1) über eine gesicherte Verbindung eingesetzt werden, um Einmalpasswörter das erste Mal zu konfigurieren oder das Passwort oder den Initialwert zu ändern. Als nächstes wird erklärt, wie opiepasswd(1) über eine nicht gesicherte Verbindung, oder zusammen mit opiekey(1) über eine gesicherte Verbindung eingesetzt werden, um dasselbe zu erreichen. Als drittes wird beschrieben, wie opiekey(1) genutzt wird, um sich über eine nicht gesicherte Verbindung anzumelden. Die vierte Tätigkeit beschreibt, wie mit opiekey(1) eine Reihe von Schlüsseln generiert wird, die Sie sich aufschreiben oder ausdrucken können, um sich von Orten anzumelden, die über keine gesicherten Verbindungen verfügen.

[1] Unter FreeBSD darf das System-Passwort maximal 128 Zeichen lang sein.

14.5.1. Einrichten über eine gesicherte Verbindung

Um OPIE erstmals zu initialisieren, rufen Sie opiepasswd(1) auf:

```
% opiepasswd -c
[grimreaper] ~ $ opiepasswd -f -c
Adding unfurl:
Only use this method from the console; NEVER from remote. If you are using
telnet, xterm, or a dial-in, type ^C now or exit with no password.
Then run opiepasswd without the -c parameter.
Using MD5 to compute responses.
Enter new secret pass phrase:
Again new secret pass phrase:

ID unfurl OTP key is 499 to4268
MOS MALL GOAT ARM AVID COED
```

Nach der Aufforderung `Enter new secret pass phrase:` oder `Enter secret password:` geben Sie bitte Ihr Passwort ein. Dies ist nicht das Passwort, mit dem Sie sich anmelden, sondern es wird genutzt, um das Einmalpasswort zu generieren. Die Zeile, die mit „ID“ anfängt, enthält Ihren Login-Namen, den Iterationszähler und den Initialwert. Diese Werte müssen Sie sich nicht merken, da das System sie zeigen wird, wenn Sie sich anmelden. In der letzten Zeile steht das Einmalpasswort, das aus diesen Parametern und Ihrem geheimen Passwort ermittelt wurde. Bei der nächsten Anmeldung müssen Sie dann dieses Einmalpasswort benutzen.

14.5.2. Einrichten über eine nicht gesicherte Verbindung

Um Einmalpasswörter über eine nicht gesicherte Verbindung einzurichten, oder das geheime Passwort zu ändern, müssen Sie über eine gesicherte Verbindung zu einer Stelle verfügen, an der Sie opiekey(1) ausführen können. Dies kann etwa die Eingabeaufforderung auf einer Maschine sein, der Sie vertrauen. Zudem müssen Sie einen Iterationszähler vorgeben (100 ist ein guter Wert) und einen Initialwert wählen, wobei Sie auch einen zufällig generierten benutzen können. Benutzen Sie opiepasswd(1) über die ungesicherte Verbindung zu der Maschine, die Sie einrichten wollen:

```
% opiepasswd

Updating unfurl:
You need the response from an OTP generator.
Old secret pass phrase:
        otp-md5 498 to4268 ext
        Response: GAME GAG WELT OUT DOWN CHAT
New secret pass phrase:
        otp-md5 499 to4269
        Response: LINE PAP MILK NELL BUOY TROY

ID mark OTP key is 499 gr4269
LINE PAP MILK NELL BUOY TROY
```

Drücken Sie Return, um die Vorgabe für den Initialwert zu akzeptieren. Bevor Sie nun das Zugriffspasswort (engl. *access password*) eingeben, rufen Sie über die gesicherte Verbindung `opikey` mit denselben Parametern auf:

```
% opiekey 498 to4268
Using the MD5 algorithm to compute response.
Reminder: Don not use opiekey from telnet or dial-in sessions.
Enter secret pass phrase:
GAME GAG WELT OUT DOWN CHAT
```

Gehen Sie zurück zu der nicht gesicherten Verbindung und geben dort das eben generierte Einmalpasswort ein.

14.5.3. Erzeugen eines einzelnen Einmalpasswortes

Nachdem Sie OPIE eingerichtet haben, werden Sie beim nächsten Anmelden wie folgt begrüßt:

```
% telnet example.com
```

```
Trying 10.0.0.1...
Connected to example.com
Escape character is '^]'.

FreeBSD/i386 (example.com) (ttypa)

login: <username>
otp-md5 498 gr4269 ext
Password:
```

OPIE besitzt eine nützliche Eigenschaft. Wenn Sie an der Eingabeaufforderung Return drücken, wird die echo-Funktion eingeschaltet, das heißt Sie sehen, was Sie tippen. Dies ist besonders nützlich, wenn Sie ein generiertes Passwort von einem Ausdruck abtippen müssen.

Jetzt müssen Sie das Einmalpasswort generieren, um der Anmeldeaufforderung nachzukommen. Dies muss auf einem gesicherten System geschehen, auf dem Sie opiekey(1) ausführen können. Dieses Programm gibt es auch für Windows®, Mac OS® und FreeBSD. Es benötigt den Iterationszähler sowie den Initialwert als Parameter, die Sie mittels „cut-and-paste“ direkt von der Login-Aufforderung nehmen können.

Auf dem sicheren System:

```
% opiekey 498 to4268
Using the MD5 algorithm to compute response.
Reminder: Do not use opiekey from telnet or dial-in sessions.
Enter secret pass phrase:
GAME GAG WELT OUT DOWN CHAT
```

Sobald das Einmalpasswort generiert wurde, können Sie die Anmeldeprozedur fortsetzen.

14.5.4. Erzeugen von mehreren Einmalpasswörtern

Manchmal haben Sie keinen Zugriff auf eine sichere Maschine oder eine sichere Verbindung. In diesem Fall können Sie vorher mit opiekey(1) einige Einmalpasswörter generieren. Zum Beispiel:

```
% opiekey -n 5 30 zz99999
Using the MD5 algorithm to compute response.
Reminder: Do not use opiekey from telnet or dial-in sessions.
Enter secret pass phrase: <secret password>
26: JOAN BORE FOSS DES NAY QUIT
27: LATE BIAS SLAY FOLK MUCH TRIG
28: SALT TIN ANTI LOON NEAL USE
29: RIO ODIN GO BYE FURY TIC
30: GREW JIVE SAN GIRD BOIL PHI
```

Mit `-n 5` fordern Sie fünf Passwörter der Reihe nach an. Der letzte Iterationszähler wird durch `30` gegeben. Beachten Sie bitte, dass die Passwörter in der *umgekehrten* Reihenfolge, in der sie zu benutzen sind, ausgeben werden. Wirklich paranoide Benutzer können sich jetzt die Passwörter aufschreiben oder ausdrucken. Sie sollten die Passwörter nach Gebrauch durchstreichen.

14.5.5. Einschränken der Benutzung von System-Passwörtern

OPIE kann die Verwendung von UNIX®-Passwörtern abhängig von der IP-Adresse einschränken. Die dazu nötigen Einstellungen werden in `/etc/opieaccess` vorgenommen, die bei der Installation des Systems automatisch erzeugt wird. Weitere Informationen über diese Datei und Sicherheitshinweise zu ihrer Verwendung finden Sie in opieaccess(5).

`opieaccess` könnte beispielsweise die folgende Zeile enthalten:

```
permit 192.168.0.0 255.255.0.0
```

Diese Zeile erlaubt es Benutzern, die sich von einer der angegebenen IP-Adressen anmelden, ihr UNIX®-Passwort zu verwenden. Beachten Sie bitte, dass eine IP-Adresse leicht gefälscht werden kann.

Findet sich in `opieaccess` kein passender Eintrag, muss die Anmeldung mit OPIE erfolgen.

14.6. TCP-Wrapper

Beigetragen von Tom Rhodes.

TCP-Wrapper erweitern die Fähigkeiten von Abschnitt 29.2, „Der inetd „Super-Server““. Beispielsweise können Verbindungen protokolliert, Nachrichten zurückgesandt oder nur interne Verbindungen angenommen werden. Einige dieser Fähigkeiten können auch über eine Firewall implementiert werden, TCP-Wrapper fügen jedoch noch eine weitere Sicherheitsschicht und Kontrollmöglichkeiten hinzu, die eine Firewall nicht bieten kann.

TCP-Wrapper sollten nicht als Ersatz für eine ordentlich konfigurierte Firewall angesehen werden, sondern stattdessen in Verbindung mit einer Firewall und anderen Sicherheitsmechanismen eingesetzt werden.

14.6.1. TCP-Wrapper einrichten

Um TCP-Wrapper unter FreeBSD zu benutzen, muss der inetd(8)-Server aus `rc.conf` mit den Optionen `-Ww` gestartet werden. Anschließend muss `/etc/hosts.allow` richtig konfiguriert werden.

Anmerkung

Im Gegensatz zu anderen Implementierungen der TCP-Wrapper wird vom Gebrauch der Datei `hosts.deny` abgeraten. Die Konfiguration sollte sich vollständig in der Datei `/etc/hosts.allow` befinden.

In der einfachsten Konfiguration werden Dienste abhängig vom Inhalt der Datei `/etc/hosts.allow` erlaubt oder gesperrt. Unter FreeBSD wird in der Voreinstellung jeder von inetd(8) gestartete Dienst erlaubt.

Eine Konfigurationszeile ist wie folgt aufgebaut: `Dienst : Adresse : Aktion`. `Dienst` ist der von inetd(8) gestartete Dienst (auch Daemon genannt). Die `Adresse` ist ein gültiger Rechnername, eine IP-Adresse oder eine IPv6-Adresse in Klammern (`[ ]`). Der Wert `allow` im Feld `Aktion` erlaubt Zugriffe, der Wert `deny` verbietet Zugriffe. Die Zeilen in `hosts.allow` werden für jede Verbindung der Reihe nach abgearbeitet. Trifft eine Zeile auf eine Verbindung zu, wird die entsprechende Aktion ausgeführt und die Abarbeitung ist beendet.

Um beispielsweise einkommende POP3-Verbindungen für den Dienst mail/qpopper zu erlauben, sollte `hosts.allow` um die nachstehende Zeile erweitert werden:

```
# This line is required for POP3 connections:
qpopper : ALL : allow
```

Nachdem Sie die Zeile hinzugefügt haben, muss inetd(8) neu gestartet werden:

```
# service inetd restart
```

14.6.2. Erweiterte Konfiguration

TCP-Wrapper besitzen weitere Optionen, die bestimmen, wie Verbindungen behandelt werden. In einigen Fällen ist es gut, wenn bestimmten Rechnern oder Diensten eine Nachricht geschickt wird. In anderen Fällen soll vielleicht der Verbindungsaufbau protokolliert oder eine E-Mail an einen Administrator versandt werden. Oder ein Dienst soll nur für das lokale Netz bereitstehen. Dies alles ist mit so genannten Wildcards, Metazeichen und der Ausführung externer Programme möglich.

14.6.2.1. Externe Kommandos

Stellen Sie sich vor, eine Verbindung soll verhindert werden und gleichzeitig soll demjenigen, der die Verbindung aufgebaut hat, eine Nachricht geschickt werden. Solch eine Aktion ist mit `twist` möglich. `twist` führt beim Verbindungsaufbau ein Kommando oder ein Skript aus. Ein Beispiel ist in `hosts.allow` enthalten:

```
# Alle anderen Dienste sind geschützt
ALL : ALL \
```

```
        : severity auth.info \
        : twist /bin/echo "You are not welcome to use %d from %h."
```

Für jeden Dienst, der nicht vorher in `hosts.allow` konfiguriert wurde, wird die Meldung „You are not allowed to use `daemon` from `hostname`.“ zurückgegeben. Dies ist nützlich, wenn die Gegenstelle sofort benachrichtigt werden soll, nachdem die Verbindung getrennt wurde. Der Text der Meldung *muss* in Anführungszeichen (`"`) stehen.

Warnung

Ein so konfigurierter Server ist anfällig für Denial-of-Service-Angriffe. Ein Angreifer kann die gesperrten Dienste mit Verbindungsanfragen überfluten.

Eine weitere Möglichkeit bietet `spawn`. Wie `twist` verbietet `spawn` die Verbindung und führt externe Kommandos aus. Allerdings sendet `spawn` der Gegenstelle keine Rückmeldung. Sehen Sie sich die nachstehende Konfigurationsdatei an:

```
# Verbindungen von example.com sind gesperrt:
ALL : .example.com \
 : spawn (/bin/echo %a from %h attempted to access %d >> \
   /var/log/connections.log) \
 : deny
```

Damit sind Verbindungen von der Domain `*.example.com` gesperrt. Jeder Verbindungsaufbau wird zudem in `/var/log/connections.log` protokolliert. Das Protokoll enthält den Rechnernamen, die IP-Adresse und den Dienst, der angesprochen wurde.

In diesem Beispiel wurden die Metazeichen `%a` und `%h` verwendet. Eine vollständige Liste der Metazeichen finden Sie in hosts_access(5).

14.6.2.2. Wildcards

Die Wildcard `ALL` passt auf jeden Dienst, jede Domain oder jede IP-Adresse. Eine andere Wildcard ist `PARANOID`. Sie passt auf jeden Rechner, dessen IP-Adresse möglicherweise gefälscht ist. Dies ist beispielsweise der Fall, wenn der Verbindungsaufbau von einer IP-Adresse erfolgt, die nicht zu dem übermittelten Rechnernamen passt. In diesem Beispiel werden alle Verbindungsanfragen zu sendmail(8) abgelehnt, wenn die IP-Adresse nicht zum Rechnernamen passt:

```
# Block possibly spoofed requests to sendmail:
sendmail : PARANOID : deny
```

Achtung

Die Wildcard `PARANOID` kann einen Dienst unbrauchbar machen, wenn der Client oder der Server eine fehlerhafte DNS-Konfiguration besitzt. Seien Sie daher besonders vorsichtig, wenn Sie diese Wildcard in Ihre Konfiguration aufnehmen wollen.

Weitere Informationen über Wildcards und deren Funktion finden Sie in hosts_access(5).

Damit die gezeigten Beispiele funktionieren, muss die erste Konfigurationszeile in `hosts.allow` auskommentiert werden.

14.7. Kerberos5

Beigetragen von Tillman Hodgson.

Beruht auf einem Beitrag von Mark Murray.

Kerberos ist ein Netzwerk-Protokoll, das Benutzer mithilfe eines sicheren Servers authentifiziert. Die Daten einer Kommunikation können verschlüsselt werden, nachdem die Kommunikationspartner mit Kerberos ihre Identität geprüft haben.

Kerberos hat nur eine Aufgabe: Die sichere Prüfung der Identität eines Benutzers (Authentifizierung) über das Netzwerk. Das System überprüft weder die Berechtigungen der Benutzer (Autorisierung), noch verfolgt es die durchgeführten Aktionen (Audit). Daher sollte Kerberos zusammen mit anderen Sicherheits-Systemen eingesetzt werden, die diese Funktionen bereitstellen.

Die folgenden Anweisungen beschreiben, wie Sie das mit FreeBSD gelieferte Kerberos einrichten. Eine vollständige Beschreibung des Systems entnehmen Sie den entsprechenden Hilfeseiten.

Die Beschreibung der Kerberos-Installation benutzt folgende Namensräume:

- Die DNS-Domain („Zone“) heißt `example.org`.
- Das Kerberos-Realm heißt `EXAMPLE.ORG`.

Anmerkung

Benutzen Sie echte Domain-Namen, wenn Sie Kerberos einrichten. Damit vermeiden Sie DNS-Probleme und stellen die Zusammenarbeit mit anderen Kerberos-Realms sicher.

14.7.1. Geschichte

Das MIT hat Kerberos entwickelt, um Sicherheitsprobleme auf dem Netzwerk zu lösen. Das Kerberos-Protokoll verwendet starke Kryptographie, sodass ein Server die Identität eines Clients (der umgekehrte Vorgang ist auch möglich) über ein unsicheres Netzwerk feststellen kann.

Der Begriff Kerberos wird sowohl für das Protokoll als auch für Programme verwendet, die Kerberos benutzen, wie Kerberos-Telnet. Die aktuelle Protokollversion ist 5 und wird in RFC 1510 beschrieben.

Mehrere Implementierungen des Protokolls stehen frei zur Verfügung und decken viele Betriebssysteme ab. Das Massachusetts Institute of Technology (MIT), an dem Kerberos ursprünglich entwickelt wurde, entwickelt seine Kerberos-Version weiter. In den USA wird diese Version häufig eingesetzt, unterlag aber Export-Beschränkungen, da sie in den USA entwickelt wurde. Die MIT-Version von Kerberos ist als Port oder Paket security/krb5 verfügbar. Heimdal ist eine weitere Implementierung der Protokollversion 5. Sie wurde außerhalb der USA entwickelt und unterliegt daher keinen Export-Beschränkungen. Heimdal-Kerberos befindet sich im Port oder Paket security/heimdal und das Basissystem von FreeBSD enthält eine minimale Installation von Heimdal.

Die folgenden Beispiele verwenden die in FreeBSD enthaltene Heimdal-Distribution.

14.7.2. Das Heimdal KDC einrichten

Kerberos authentifiziert Benutzer an einer zentralen Stelle: dem Key Distribution Center (KDC). Das KDC verteilt *Tickets*, mit denen ein Dienst die Identität eines Benutzers feststellen kann. Alle Mitglieder eines Kerberos-Realms vertrauen dem KDC, daher gelten für das KDC erhöhte Sicherheitsanforderungen.

Obwohl der Kerberos-Server wenig Ressourcen benötigt, sollte das KDC wegen der Sicherheitsanforderungen auf einem separaten Rechner installiert werden.

Das KDC wird in `/etc/rc.conf` wie folgt aktiviert:

```
kerberos5_server_enable="YES"
kadmind5_server_enable="YES"
```

Danach wird `/etc/krb5.conf` wie folgt bearbeitet:

```
[libdefaults]
    default_realm = EXAMPLE.ORG
[realms]
    EXAMPLE.ORG = {
        kdc = kerberos.example.org
 admin_server = kerberos.example.org
    }
[domain_realm]
    .example.org = EXAMPLE.ORG
```

Diese Einstellungen setzen voraus, dass der voll qualifizierte Name des KDCs `kerberos.example.org` ist. Wenn das KDC einen anderen Namen hat, muss in der DNS-Zone ein Alias-Eintrag (CNAME-Record) für das KDC hinzugefügt werden.

Anmerkung

In großen Netzwerken mit einem ordentlich konfigurierten DNS-Server kann die Datei aus dem obigen Beispiel verkürzt werden:

```
[libdefaults]
      default_realm = EXAMPLE.ORG
```

Die Zonendatei von `example.org` muss dann die folgenden Zeilen enthalten:

```
_kerberos._udp      IN  SRV     01 00 88 kerberos.example.org.
_kerberos._tcp      IN  SRV     01 00 88 kerberos.example.org.
_kpasswd._udp       IN  SRV     01 00 464 kerberos.example.org.
_kerberos-adm._tcp  IN  SRV     01 00 749 kerberos.example.org.
_kerberos           IN  TXT     EXAMPLE.ORG
```

Anmerkung

Damit die Clients die Kerberos-Dienste benutzen können, muss `/etc/krb5.conf` entweder die vollständige Konfiguration enthalten oder eine minimale Konfiguration enthalten *und* zusätzlich ein DNS-Server richtig eingerichtet sein.

Im nächsten Schritt wird die Kerberos-Datenbank eingerichtet. Die Datenbank enthält die Schlüssel aller Prinzipale und ist mit einem Passwort geschützt. Dieses Passwort brauchen Sie sich nicht merken, da ein davon abgeleiteter Schlüssel in `/var/heimdal/m-key` gespeichert wird. Um den Schlüssel zu erstellen, rufen Sie kstash(8) auf und geben Sie ein Passwort ein.

Nachdem der Schlüssel erstellt wurde, sollte die Datenbank initialisiert werden. Das Kerberos-Werkzeug kadmin(8) kann mit `kadmin -l` im lokalen Modus benutzt werden, ohne den Netzwerkdienst, welcher zu diesem Zeitpunkt noch nicht läuft, zu verwenden. An der Eingabeaufforderung von kadmin(8) kann mit `init` die Datenbank des Realms initialisiert werden.

Zuletzt wird mit `add` das erste Prinzipal erstellt. Benutzen Sie die voreingestellten Optionen. Die Einstellungen können später `modify` verändert werden. An der Eingabeaufforderung von kadmin(8) zeigt `?` Hilfetexte an.

Zusammengefasst wird die Datenbank wie folgt eingerichtet:

```
# kstash
Master key: xxxxxxxx
Verifying password - Master key: xxxxxxxx
```

```
# kadmin -l
kadmin> init EXAMPLE.ORG
Realm max ticket life [unlimited]:
kadmin> add tillman
Max ticket life [unlimited]:
Max renewable life [unlimited]:
Attributes []:
Password: xxxxxxxx
Verifying password - Password: xxxxxxxx
```

Jetzt kann das KDC gestartet werden. Führen Sie zum Start der Dienste `service kerberos start` und `service kadmind start` aus. Obwohl zu diesem Zeitpunkt noch keine kerberisierten Dienste laufen, kann die Funktion des KDCs schon überprüft werden. Für den eben angelegten Benutzer können Sie sich vom KDC Tickets holen und anzeigen lassen:

```
% kinit tillman
tillman@EXAMPLE.ORG's Password:

% klist
Credentials cache: FILE: /tmp/krb5cc_500
        Principal: tillman@EXAMPLE.ORG

  Issued           Expires          Principal
Aug 27 15:37:58  Aug 28 01:37:58  krbtgt/EXAMPLE.ORG@EXAMPLE.ORG
```

Nachdem der Test abgeschlossen ist, kann das temporäre Ticket zurückgezogen werden:

```
% kdestroy
```

14.7.3. Heimdal Kerberos-Dienste einrichten

Bei der Konfiguration eines Servers für die Kerberos-Authentifizierung muss zuerst sichergestellt werden, dass `/etc/krb5.conf` richtig konfiguriert ist. Die Datei kann entweder vom KDC kopiert, oder auf dem neuen System regeneriert werden.

Als nächstes muss auf dem Server die `/etc/krb5.keytab` erzeugt werden. Dies ist der Hauptbestandteil um Dienste zu „kerberisieren“ und entspricht der Erzeugung eines geheimen Schlüssels zwischen dem Dienst und dem KDC. Das Geheimnis ist ein kryptographischer Schlüssel, der in einem `keytab>` abgelegt wird. Diese Datei enthält den Schlüssel des Servers, mit dem sich der Server und das KDC gegenseitig authentifizieren können. Sie muss in einer sicheren Art und Weise an den Server übertragen werden, da ansonsten die Sicherheit des Servers gefährdet ist, wenn z.B. die Schlüssel öffentlich werden. In der Regel wird die `keytab` auf einem vertrauenswürdigen Rechner mit `kadmin` erzeugt und anschließend sicher auf den Server übertragen, beispielsweise mit scp(1). Wenn die Sicherheitsrichtlinien es erlauben, kann die Datei auch direkt auf dem Server erzeugt werden. Es ist sehr wichtig, dass die `keytab` auf sichere Weise auf den Server übertragen wird. Wenn der Schlüssel einer anderen Partei bekannt wird, kann sich diese Partei den Benutzern als Server ausgeben! Da der Eintrag für das Host-Prinzipal für die KDC-Datenbank auch mit `kadmin` erstellt wird, ist es praktisch, `kadmin` direkt auf dem Server zu benutzen.

Natürlich ist auch `kadmin` ein kerberisierter Dienst: ein Kerberos-Ticket ist erforderlich, um sich gegenüber dem Netzwerkdienst zu authentifizieren und um sicherzustellen, dass der Benutzer, der `kadmin` ausführt, tatsächlich vorhanden ist. `kadmin` wird nach dem Passwort fragen, um ein neues Ticket zu generieren. Das Prinzipal, das sich mit dem kadmin-Dienst authentifiziert, muss über die Zugriffskontrollliste `kadmin.acl` dazu berechtigt sein. Weitere Informationen über Zugriffskontrolllisten finden Sie in den Heimdal-Info-Seiten (`info heimdal`) im Abschnitt „Remote administration“. Wenn der Zugriff auf `kadmin` von entfernten Rechnern verboten ist, kann sich der Administrator entweder über die lokale Konsole oder über ssh(1) mit dem KDC verbinden, um die lokale Administration mit `kadmin -l` durchzuführen.

Nach der Installation von `/etc/krb5.conf` , können Sie das Kommando `add --random-key` in `kadmin` ausführen, um das Host-Prinzipal in die Datenbank zu schreiben. Das Kommando `ext` extrahiert den Schlüssel des Prinzipals in eine eigene keytab:

```
# kadmin
kadmin> add --random-key host/myserver.example.org
Max ticket life [unlimited]:
Max renewable life [unlimited]:
Attributes []:
kadmin> ext host/myserver.example.org
kadmin> exit
```

Beachten Sie, dass `ext` den extrahierten Schlüssel standardmäßig in `/etc/krb5.keytab` speichert. Das ist gut, wenn das Kommando auf dem kerberisierten Server ausgeführt wird, ansonsten sollte das Argument `--keytab` *`pfad/zur/datei`* benutzt werden, wenn die keytab an einen anderen Ort extrahiert wird:

```
# kadmin
kadmin> ext --keytab=/tmp/example.keytab   host/myserver.example.org
kadmin> exit
```

Anschließend kann die erzeugte keytab sicher mit `scp` auf Server oder auf einen Wechseldatenträger kopiert werden. Geben Sie auf jeden Fall einen anderen Namen für die keytab an, weil sonst die keytab des KDCs überschrieben würde.

Wegen der Datei `krb5.conf` kann der Server nun mit dem KDC kommunizieren und seine Identität mithilfe der Datei `krb5.keytab` nachweisen. Jetzt können die kerberisierten Dienste aktiviert werden. Einer der gebräuchlichsten Dienste ist sshd(8), der Kerberos über GSS-API unterstützt. Fügen Sie folgende Zeile in `/etc/ssh/sshd_config` ein:

```
GSSAPIAuthentication yes
```

Nach dieser Änderung muss sshd(8) mit `service sshd restart` neu gestartet werden, damit die neue Konfiguration wirksam wird.

14.7.4. Heimdal Kerberos-Clients einrichten

Genau wie der Server, benötigt auch der Client eine Konfiguration in `/etc/krb5.conf` . Kopien Sie die Datei (sicher) vom KDC auf den Client, oder schreiben Sie die Datei bei Bedarf einfach neu. Testen Sie den Client, indem Sie mit `kinit` Tickets anfordern, mit `klist` Tickets anzeigen und mit `kdestroy` Tickets löschen. Kerberos-Anwendungen sollten auch kerberisierte Server ansprechen können. Wenn das nicht funktioniert, Sie aber Tickets anfordern können, hat wahrscheinlich der kerberisierte Server ein Problem und nicht der Client oder das KDC. Im Falle eines kerberisierten ssh(1) ist GSS-API in der Voreinstellung deaktiviert. Testen Sie daher mit `ssh -o GSSAPIAuthentication=yes` *`hostname`*.

Wenn Sie die kerberisierten Anwendungen testen, können Sie einen Paket-Sniffer wie `tcpdump` benutzen, um sicherzustellen, dass keine sensiblen Informationen im Klartext übertragen werden.

Es stehen verschiedene Kerberos-Anwendungen zur Verfügung. Die Anwendungen, die SASL benutzen, können dann auch GSS-API benutzen. Viele Arten von Anwendungen können Kerberos zur Authentifizierung verwenden, vom Jabber-Client bis zum IMAP-Client.

Normalerweise wird ein Kerberos-Prinzipal auf ein lokales Benutzerkonto abgebildet. Manchmal wird aber Zugriff auf ein lokales Benutzerkonto benötigt, zu dem es keinen passenden Kerberos-Prinzipal gibt. Der Prinzipal `tillman@EXAMPLE.ORG` bräuchte beispielsweise Zugriff auf das Konto `webdevelopers`. Ebenso könnten andere Prinzipale auf dieses Konto zugreifen wollen.

Die Dateien `.k5login` und `.k5users` im Heimatverzeichnis eines Benutzers können verwendet werden, um dieses Problem zu lösen. Mit der folgenden `.k5login` im Heimatverzeichnis des Benutzers `webdevelopers` haben beide Prinzipale auch ohne das gemeinsame Passwort Zugriff auf das Konto:

```
tillmann@example.org
jdoe@example.org
```

Weitere Informationen zu `.k5users` finden Sie in ksu(1).

14.7.5. Tipps und Fehlersuche

- Wenn Sie den Heimdal-Port oder den MIT-Port benutzen, muss in der Umgebungsvariable `PATH` der Pfad zu den Kerberos-Programmen vor dem Pfad zu den Programmen des Systems stehen.
- Wenn die Clients im Realm ihre Uhrzeit nicht synchronisieren, schlägt vielleicht die Authentifizierung fehl. Abschnitt 29.11, „Die Uhrzeit mit NTP synchronisieren“ beschreibt, wie Sie mithilfe von NTP die Uhrzeiten synchronisieren.
- Die MIT- und Heimdal-Systeme arbeiten bis auf `kadmin`, welches nicht standardisiert ist, gut zusammen.
- Wenn Sie den Namen eines Rechners ändern, müssen Sie auch den `host/`-Prinzipal ändern und die `keytab` aktualisieren. Dies betrifft auch spezielle Einträge wie den Prinzipal für Apaches www/mod_auth_kerb.
- Alle Rechner in einem Realm müssen vor- und rückwärts aufgelöst werden können. Entweder über DNS, zumindest aber über `/etc/hosts` . CNAME-Einträge im DNS funktionieren, aber die entsprechenden A- und PTR-Einträge müssen vorhanden und richtig sein. Wenn sich Namen nicht auflösen lassen, ist die Fehlermeldung nicht gerade selbstsprechend: Kerberos5 refuses authentication because Read req failed: Key table entry not found.
- Einige Betriebssysteme installieren `ksu` mit falschen Zugriffsrechten; es fehlt das Set-UID-Bit für `root`. Das hat zur Folge, dass `ksu` nicht funktioniert. Dies ist ein Fehler in den Zugriffsrechten und kein Fehler des KDCs.
- Wenn Sie für einen Prinzipal unter MIT-Kerberos Tickets mit einer längeren Gültigkeit als der vorgegebenen zehn Stunden einrichten wollen, müssen Sie zwei Sachen ändern. Benutzen Sie das `modify_principal` von `kadmin`, um die maximale Gültigkeitsdauer für den Prinzipal selbst und den Prinzipal `krbtgt` zu erhöhen. Das Prinzipal kann dann mit `kinit -l` ein Ticket mit einer längeren Gültigkeit beantragen.
- Mit einem Packet-Sniffer können Sie feststellen, dass Sie sofort nach dem Aufruf von `kinit` eine Antwort vom KDC bekommen – noch bevor Sie überhaupt ein Passwort eingegeben haben! Das ist in Ordnung: Das KDC händigt ein Ticket-Granting-Ticket (TGT) auf Anfrage aus, da es durch einen vom Passwort des Benutzers abgeleiteten Schlüssel geschützt ist. Wenn das Passwort eingegeben wird, wird es nicht zum KDC gesendet, sondern zum Entschlüsseln der Antwort des KDCs benutzt, die `kinit` schon erhalten hat. Wird die Antwort erfolgreich entschlüsselt, erhält der Benutzer einen Sitzungs-Schlüssel für die künftige verschlüsselte Kommunikation mit dem KDC und das TGT. Das TGT wiederum ist mit dem Schlüssel des KDCs verschlüsselt. Diese Verschlüsselung ist für den Benutzer völlig transparent und erlaubt dem KDC, die Echtheit jedes einzelnen TGT zu prüfen.
- Wenn Sie OpenSSH verwenden und Tickets mir einer langen Gültigkeit (beispielsweise einer Woche) benutzen, setzen Sie `TicketCleanup` in `sshd_config` auf `no`. Ansonsten werden die Tickets gelöscht, wenn Sie sich abmelden.
- Host-Prinzipale können Tickets mit längerer Gültigkeit besitzen. Wenn der Prinzipal eines Benutzers über ein Ticket verfügt, das eine Woche gültig ist, das Ticket des Host-Prinzipals aber nur neun Stunden gültig ist, funktioniert der Ticket-Cache nicht wie erwartet. Im Cache befindet sich dann ein abgelaufenes Ticket des Host-Prinzipals.
- Wenn Sie mit `krb5.dict` die Verwendung schlechter Passwörter verhindern wollen, wie in kadmin(8) beschrieben, geht das nur mit Prinzipalen, denen eine Passwort-Policy zugewiesen wurde. Das Format von `krb5.dict` enthält pro Zeile ein Wort. Sie können daher einen symbolischen Link auf `/usr/share/dict/words` erstellen.

14.7.6. Unterschiede zum MIT-Port

Der Hauptunterschied zwischen MIT-Kerberos und Heimdal-Kerberos ist das Kommando `kadmin`. Die Befehlssätze des Kommandos (obwohl funktional gleichwertig) und das verwendete Protokoll unterscheiden sich in beiden Varianten. Das KDC lässt sich nur mit dem `kadmin` Kommando der passenden Kerberos-Variante verwalten.

Für dieselbe Funktion können auch die Client-Anwendungen leicht geänderte Kommandozeilenoptionen besitzen. Folgen Sie bitte der Anleitung auf der Kerberos-Seite `http://web.mit.edu/Kerberos/www/` des MITs. Achten Sie besonders auf den Suchpfad für Anwendungen. Der MIT-Port wird standardmäßig in `/usr/local/` installiert. Wenn die Umgebungsvariable `PATH` zuerst die Systemverzeichnisse enthält, werden die Systemprogramme anstelle der MIT-Programme ausgeführt.

Anmerkung

Wenn Sie den MIT-Port security/krb5 verwenden, erscheint bei der Anmeldung mit `telnetd` und `klogind` die Fehlermeldung incorrect permissions on cache file. Lesen Sie dazu die im Port enthaltene Datei `/usr/local/share/doc/krb5/README.FreeBSD` . Wichtig ist, dass zur Authentifizierung die Binärdatei `login.krb5` verwendet wird, die für durchgereichte Berechtigungen die Eigentümer korrekt ändert.

Wird MIT-Kerberos auf FreeBSD eingesetzt, sollten in `rc.conf` folgende Zeilen aufgenommen werden:

```
kerberos5_server="/usr/local/sbin/krb5kdc"
kadmind5_server="/usr/local/sbin/kadmind"
kerberos5_server_enable="YES"
kadmind5_server_enable="YES"
```

Diese Zeilen sind notwendig, weil die Anwendungen von MIT-Kerberos die Binärdateien unterhalb von `/usr/local` installieren.

14.7.7. Beschränkungen von Kerberos

14.7.7.1. Kerberos muss ganzheitlich verwendet werden

Jeder über das Netzwerk angebotene Dienst muss mit Kerberos zusammenarbeiten oder auf anderen Wegen gegen Angriffe aus dem Netzwerk geschützt sein. Andernfalls können Berechtigungen gestohlen und wiederverwendet werden. Es ist beispielsweise nicht sinnvoll, für Remote-ShellsKerberos zu benutzen, dagegen aber POP3-Zugriff auf einen Mail-Server zu erlauben, da POP3-Passwörter im Klartext versendet.

14.7.7.2. Kerberos ist für Einbenutzer-Systeme gedacht

In Mehrbenutzer-Umgebungen ist Kerberos unsicherer als in Einbenutzer-Umgebungen, da die Tickets im für alle lesbaren Verzeichnis `/tmp` gespeichert werden. Wenn ein Rechner von mehreren Benutzern verwendet wird, ist es möglich, dass Tickets von einem anderen Benutzer gestohlen oder kopiert werden.

Dieses Problem können Sie lösen, indem Sie mit der Kommandozeilenoption `-c` oder besser mit der Umgebungsvariablen `KRB5CCNAME` einen Ort für die Tickets vorgeben. Es reicht, die Tickets im Heimatverzeichnis eines Benutzers zu speichern und mit Zugriffsrechten zu schützen.

14.7.7.3. Das KDC ist verwundbar

Das KDC muss genauso abgesichert werden wie die auf ihm befindliche Passwort-Datenbank. Auf dem KDC sollten absolut keine anderen Dienste laufen und der Rechner sollte physikalisch gesichert sein. Die Gefahr ist groß, da Kerberos alle Passwörter mit einem Schlüssel, dem Haupt-Schlüssel, verschlüsselt. Der Haupt-Schlüssel wiederum wird in einer Datei auf dem KDC gespeichert.

Ein kompromittierter Haupt-Schlüssel ist nicht ganz so schlimm wie allgemein angenommen. Der Haupt-Schlüssel wird nur zum Verschlüsseln der Passwort-Datenbank und zum Initialisieren des Zufallsgenerators verwendet. Solange der Zugriff auf das KDC abgesichert ist, kann ein Angreifer wenig mit dem Haupt-Schlüssel anfangen.

Wenn das KDC nicht zur Verfügung steht, sind auch die Netzwerkdienste nicht benutzbar, da eine Authentifizierung nicht durchgeführt werden kann. Das KDC ist also ein optimales Ziel für einen Denial-of-Service Angriff. Sie

können diesem Angriff entgegenwirken, indem Sie einen KDC-Master und einen oder mehrere Slaves verwenden. Der Rückfall auf ein sekundäres KDC mittels PAM-Authentifizierung muss sorgfältig eingerichtet werden.

14.7.7.4. Mängel von Kerberos

Mit Kerberos können sich Benutzer, Rechner und Dienste gegenseitig authentifizieren. Allerdings existiert kein Mechanismus, der das KDC gegenüber Benutzern, Rechnern oder Diensten authentifiziert. Ein verändertes kinit(1) könnte beispielsweise alle Benutzernamen und Passwörter abfangen. Die von veränderten Programmen ausgehende Gefahr können Sie lindern, indem Sie die Integrität von Dateien mit Werkzeugen wie security/tripwire prüfen.

14.7.8. Weiterführende Dokumentation

- The Kerberos FAQ
- Designing an Authentication System: a Dialogue in Four Scenes
- RFC 1510, The Kerberos Network Authentication Service (V5)
- MIT Kerberos-Seite
- Heimdal Kerberos-Seite

14.8. OpenSSL

Beigetragen von Tom Rhodes.

OpenSSL ist eine freie Implementierung der SSL und TLS-Protokolle. Es bietet eine verschlüsselte Transportschicht oberhalb der normalen Kommunikationsschicht und kann daher zusammen mit vielen Netzdiensten benutzt werden.

Anwendungsbeispiele für OpenSSL sind die verschlüsselte Authentifizierung von E-Mail-Clients oder Web-Transaktionen wie das Bezahlen mit einer Kreditkarte. OpenSSL kann während des Baus in viele Ports, wie www/apache22 und mail/claws-mail, integriert werden.

Anmerkung

Ist beim Aufruf von `make` die Variable `WITH_OPENSSL_BASE` nicht explizit auf `yes` gesetzt, baut die Ports-Sammlung meist den Port security/openssl.

Das in FreeBSD integrierte OpenSSL stellt die Protokolle Secure Sockets Layer v2/v3 (SSLv2/SSLv3) und Transport Layer Security v1 (TLSv1) zur Verfügung. Die OpenSSL-Bibliotheken stellen kryptographische Funktionen bereit.

Anmerkung

Mit OpenSSL kann der IDEA-Algorithmus verwendet werden, wegen Patenten in den USA ist der Algorithmus in der Voreinstellung allerdings deaktiviert. Wenn Sie die IDEA-Lizenz akzeptieren, können Sie den IDEA-Algorithmus aktivieren, indem Sie die Variable `MAKE_IDEA` in `/etc/make.conf` setzen.

Meist wird OpenSSL eingesetzt, um Zertifikate für Anwendungen bereitzustellen. Die Zertifikate stellen die Identität einer Firma oder eines Einzelnen sicher. Wenn ein Zertifikat nicht von einer Zertifizierungsstelle (*Certifica-*

te Authority, CA) gegengezeichnet wurde, erhalten Sie normalerweise eine Warnung. Eine Zertifizierungsstelle ist eine Firma wie VeriSign, die Zertifikate von Personen oder Firmen gegenzeichnet und damit die Korrektheit der Zertifikate bestätigt. Diese Prozedur kostet Geld, ist aber keine Voraussetzung für den Einsatz von Zertifikaten, beruhigt aber sicherheitsbewusste Benutzer.

14.8.1. Zertifikate erzeugen

Ein Zertifikat erzeugen Sie mit dem nachstehenden Kommando:

```
# openssl req -new -nodes -out req.pem -keyout cert.pem
Generating a 1024 bit RSA private key
..............++++++
.....................................++++++
writing new private key to 'cert.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:PA
Locality Name (eg, city) []:Pittsburgh
Organization Name (eg, company) [Internet Widgits Pty Ltd]:My Company
Organizational Unit Name (eg, section) []:Systems Administrator
Common Name (eg, YOUR name) []:localhost.example.org
Email Address []:trhodes@FreeBSD.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:Another Name
```

Beachten Sie, dass die Eingabe bei „Common Name“ ein gültiger Domain-Name sein muss. Eine andere Eingabe erzeugt ein unbrauchbares Zertifikat. Das Zertifikat kann mit einer Gültigkeitsdauer und anderen Verschlüsselungsalgorithmen erzeugt werden. openssl(1) beschreibt die zur Verfügung stehenden Optionen.

Das Verzeichnis, in dem Sie den letzten Befehl ausgeführt haben, enthält nun zwei Dateien: Die Anforderung für ein neues Zertifikat wurde in `req.pem` gespeichert. Diese Datei können Sie an eine CA senden, wo die Angaben geprüft werden. Nach erfolgreicher Prüfung wird das Zertifikat unterschrieben und an Sie zurückgesandt. Die zweite Datei, `cert.pem`, enthält den privaten Schlüssel für das Zertifikat und darf auch keine Fall in fremde Hände geraten, da ein Angreifer sonst in der Lage ist, anderen Personen oder Rechnern vorzugaukeln, dass es sich bei ihm um Sie handelt.

Wenn Sie keine Signatur einer Zertifizierungsstelle benötigen, können Sie ein selbst-signiertes Zertifikat erstellen. Erzeugen Sie dazu zuerst einen RSA-Schlüssel:

```
# openssl dsaparam -rand -genkey -out myRSA.key 1024
```

Erzeugen Sie dann den CA-Schlüssel:

```
# openssl gendsa -des3 -out myca.key myRSA.key
```

Erstellen Sie mit diesem Schlüssel das Zertifikat:

```
# openssl req -new -x509 -days 365 -key myca.key -out new.crt
```

Zwei neue Dateien befinden sich nun im Verzeichnis: Der Schlüssel der Zertifizierungsstelle `myca.key` und das Zertifikat selbst, `new.crt`. Sie sollten in einem Verzeichnis, vorzugsweise unterhalb von `/etc/ssl` abgelegt werden, das nur von `root` lesbar ist. Die Zugriffsrechte der Dateien können mit chmod(1) auf `0700` gesetzt werden.

14.8.2. Zertifikate benutzen

Mit einem Zertifikat können beispielsweise die Verbindungen zu Sendmail verschlüsselt werden, um eine Klartext-Authentifizierung zu verhindern.

Anmerkung

Einige E-Mail-Programme geben Warnungen aus, wenn ein Zertifikat nicht lokal installiert ist. Weitere Informationen zur Installation von Zertifikaten finden Sie in der Dokumentation der entsprechenden Software.

Ergänzen Sie die Konfigurationsdatei von Sendmail (`.mc`) um die nachstehenden Zeilen:

```
dnl SSL Options
define(`confCACERT_PATH',`/etc/certs')dnl
define(`confCACERT',`/etc/certs/new.crt')dnl
define(`confSERVER_CERT',`/etc/certs/new.crt')dnl
define(`confSERVER_KEY',`/etc/certs/myca.key')dnl
define(`confTLS_SRV_OPTIONS', `V')dnl
```

Im Verzeichnis `/etc/certs` befindet sich der Schlüssel und das Zertifikat. Bauen Sie danach im Verzeichnis `/etc/mail` mit dem Kommando `make install` die `.cf`-Datei. Starten Sie anschließend Sendmail mit `make restart` neu.

Wenn alles gut ging, erscheinen keine Fehlermeldungen in `/var/log/maillog` und Sie sehen Sendmail in der Prozessliste.

Testen Sie nun den Mailserver mit telnet(1):

```
# telnet example.com 25
Trying 192.0.34.166...
Connected to example.com .
Escape character is '^]'.
220 example.com  ESMTP Sendmail 8.12.10/8.12.10; Tue, 31 Aug 2004 03:41:22 -0400 (EDT)
ehlo example.com
250-example.com Hello example.com [192.0.34.166], pleased to meet you
250-ENHANCEDSTATUSCODES
250-PIPELINING
250-8BITMIME
250-SIZE
250-DSN
250-ETRN
250-AUTH LOGIN PLAIN
250-STARTTLS
250-DELIVERBY
250 HELP
quit
221 2.0.0 example.com  closing connection
Connection closed by foreign host.
```

Wenn die Zeile `STARTTLS` erscheint, hat alles funktioniert.

14.9. VPN mit IPsec

Geschrieben von Nik Clayton.

14.9.1. IPsec Grundlagen

Geschrieben von Hiten M. Pandya.

Dieser Abschnitt beschreibt die Einrichtung von IPsec. Um IPsec einzurichten, sollten Sie einen neuen Kernel bauen können (siehe Kapitel 9, *Konfiguration des FreeBSD-Kernels*).

IPsec ist ein Protokoll, das auf dem Internet-Protokoll (IP) aufbaut. Mit IPsec können mehrere Systeme geschützt miteinander kommunizieren. Das in FreeBSD realisierte IPsec-Protokoll baut auf der KAME-Implementierung auf und unterstützt sowohl IPv4 als auch IPv6.

IPsec besteht wiederum aus zwei Protokollen:

- *Encapsulated Security Payload (ESP)* verschlüsselt IP-Pakete mit einem symmetrischen Verfahren wie Blowfish oder 3DES. Damit werden die Pakete vor Manipulationen Dritter geschützt.
- Der *Authentication Header (AH)* enthält eine kryptographische Prüfsumme, die sicher stellt, dass ein IP-Paket nicht verändert wurde. Der Authentication-Header folgt nach dem normalen IP-Header und erlaubt dem Empfänger eines IP-Paketes, dessen Integrität zu prüfen.

ESP und AH können, je nach Situation, zusammen oder einzeln verwendet werden.

IPsec kann in zwei Modi betrieben werden: Der *Transport-Modus* verschlüsselt die Daten zwischen zwei Systemen. Der *Tunnel-Modus* verbindet zwei Subnetze miteinander. Durch einen Tunnel können dann beispielsweise verschlüsselte Daten übertragen werden. Ein Tunnel wird auch als *Virtual-Private-Network* (VPN) bezeichnet. Detaillierte Informationen über das IPsec-Subsystem von FreeBSD finden Sie in ipsec(4).

Die folgenden Optionen in der Kernelkonfiguration aktivieren IPsec:

```
options   IPSEC        #IP security
device    crypto
```

Wenn Sie zur Fehlersuche im IPsec-Subsystem Unterstützung wünschen, sollten Sie die folgende Option ebenfalls aktivieren:

```
options   IPSEC_DEBUG  #debug for IP security
```

14.9.2. VPN zwischen einem Heim- und Firmennetzwerk einrichten

Es gibt keinen Standard, der festlegt, was ein Virtual-Private-Network ist. VPNs können mit verschiedenen Techniken, die jeweils eigene Vor- und Nachteile besitzen, implementiert werden. Dieser Abschnitt stellt Möglichkeiten vor, um ein VPN für das folgende Szenario aufzubauen:

- Es müssen mindestens zwei Netzwerke vorhanden sein, welche intern IP benutzen.
- Beide Netzwerke sind über ein FreeBSD-Gateway mit dem Internet verbunden.
- Der Gateway jedes Netzwerks besitzt mindestens eine öffentliche IP-Adresse.
- Die intern verwendeten IP-Adressen können private oder öffentliche Adressen sein. Sie dürfen sich jedoch nicht überlappen. Zum Beispiel sollten nicht beide Netze `192.168.1.x` benutzen.

14.9.2.1. Konfiguration von IPsec in FreeBSD

Geschrieben von Tom Rhodes.

Als erstes muss security/ipsec-tools aus der Ports-Sammlung installiert werden. Diese Software enthält einige Anwendungen, die bei der Konfiguration von IPsec hilfreich sind.

Als nächstes müssen zwei gif(4)-Pseudogeräte angelegt werden, um die Pakete zu tunneln und dafür zu sorgen, dass beide Netzwerke richtig miteinander kommunizieren können. Geben Sie als `root` die folgenden Befehle ein, wobei Sie *`intern`* und *`extern`* durch die realen internen und externen IP-Adressen der Gateways ersetzen müssen:

```
# ifconfig gif0 create
```

```
# ifconfig gif0 intern1 intern2
```

```
# ifconfig gif0 tunnel extern1 extern2
```

In diesem Beispiel ist die externe IP-Adresse des Firmennetzwerkes (LAN) `172.16.5.4` und die interne IP-Adresse ist `10.246.38.1`. Das Heimnetzwerk (LAN) hat die externe IP-Adresse `192.168.1.12` mit der internen privaten IP-Adresse `10.0.0.5`.

Wenn dies verwirrend erscheint, schauen Sie sich die folgende Ausgabe von ifconfig(8)an:

```
Gateway 1:

gif0: flags=8051 mtu 1280
tunnel inet 172.16.5.4 --> 192.168.1.12
inet6 fe80::2e0:81ff:fe02:5881%gif0 prefixlen 64 scopeid 0x6
inet 10.246.38.1 --> 10.0.0.5 netmask 0xffffff00

Gateway 2:

gif0: flags=8051 mtu 1280
tunnel inet 192.168.1.12 --> 172.16.5.4
inet 10.0.0.5 --> 10.246.38.1 netmask 0xffffff00
inet6 fe80::250:bfff:fe3a:c1f%gif0 prefixlen 64 scopeid 0x4
```

Wenn Sie fertig sind, sollten beide internen Adressen über ping(8) erreichbar sein:

```
priv-net# ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5): 56 data bytes
64 bytes from 10.0.0.5: icmp_seq=0 ttl=64 time=42.786 ms
64 bytes from 10.0.0.5: icmp_seq=1 ttl=64 time=19.255 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=64 time=20.440 ms
64 bytes from 10.0.0.5: icmp_seq=3 ttl=64 time=21.036 ms
--- 10.0.0.5 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 19.255/25.879/42.786/9.782 ms

corp-net# ping 10.246.38.1
PING 10.246.38.1 (10.246.38.1): 56 data bytes
64 bytes from 10.246.38.1: icmp_seq=0 ttl=64 time=28.106 ms
64 bytes from 10.246.38.1: icmp_seq=1 ttl=64 time=42.917 ms
64 bytes from 10.246.38.1: icmp_seq=2 ttl=64 time=127.525 ms
64 bytes from 10.246.38.1: icmp_seq=3 ttl=64 time=119.896 ms
64 bytes from 10.246.38.1: icmp_seq=4 ttl=64 time=154.524 ms
--- 10.246.38.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 28.106/94.594/154.524/49.814 ms
```

Wie erwartet, können nun beiden Seiten ICMP-Pakete von ihren privaten Adressen senden und empfangen. Als nächstes müssen beide Gateways so konfiguriert werden, dass sie die Pakete des anderen Netzwerkes richtig routen. Mit dem folgenden Befehl erreicht man das Ziel:

```
# corp-net# route add 10.0.0.0 10.0.0.5 255.255.255.0
```

```
# corp-net# route add net 10.0.0.0: gateway 10.0.0.5
```

```
# priv-net# route add 10.246.38.0 10.246.38.1 255.255.255.0
```

```
# priv-net# route add host 10.246.38.0: gateway 10.246.38.1
```

Ab jetzt sollten die Rechner von den Gateways sowie von den Rechnern hinter den Gateways erreichbar sein. Dies können Sie wieder mit ping(8) überprüfen:

```
corp-net# ping 10.0.0.8
```

```
PING 10.0.0.8 (10.0.0.8): 56 data bytes
64 bytes from 10.0.0.8: icmp_seq=0 ttl=63 time=92.391 ms
64 bytes from 10.0.0.8: icmp_seq=1 ttl=63 time=21.870 ms
64 bytes from 10.0.0.8: icmp_seq=2 ttl=63 time=198.022 ms
64 bytes from 10.0.0.8: icmp_seq=3 ttl=63 time=22.241 ms
64 bytes from 10.0.0.8: icmp_seq=4 ttl=63 time=174.705 ms
--- 10.0.0.8 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 21.870/101.846/198.022/74.001 ms

priv-net# ping 10.246.38.107
PING 10.246.38.1 (10.246.38.107): 56 data bytes
64 bytes from 10.246.38.107: icmp_seq=0 ttl=64 time=53.491 ms
64 bytes from 10.246.38.107: icmp_seq=1 ttl=64 time=23.395 ms
64 bytes from 10.246.38.107: icmp_seq=2 ttl=64 time=23.865 ms
64 bytes from 10.246.38.107: icmp_seq=3 ttl=64 time=21.145 ms
64 bytes from 10.246.38.107: icmp_seq=4 ttl=64 time=36.708 ms
--- 10.246.38.107 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 21.145/31.721/53.491/12.179 ms
```

Das Konfigurieren der Tunnel ist der einfache Teil. Die Konfiguration einer sicheren Verbindung geht viel mehr in die Tiefe. Die folgende Konfiguration benutzt pre-shared (PSK) RSA-Schlüssel. Abgesehen von den IP-Adressen, sind beide `/usr/local/etc/racoon/racoon.conf` identisch und sehen ähnlich aus:

```
path    pre_shared_key  "/usr/local/etc/racoon/psk.txt"; #location of pre-shared key file
log     debug;  #log verbosity setting: set to 'notify' when testing and debugging is ↺
complete

padding # options are not to be changed
{
        maximum_length  20;
        randomize       off;
        strict_check    off;
        exclusive_tail  off;
}

timer   # timing options. change as needed
{
        counter         5;
        interval        20 sec;
        persend         1;
#       natt_keepalive  15 sec;
        phase1          30 sec;
        phase2          15 sec;
}

listen  # address [port] that racoon will listening on
{
        isakmp          172.16.5.4 [500];
        isakmp_natt     172.16.5.4 [4500];
}

remote  192.168.1.12 [500]
{
        exchange_mode   main,aggressive;
        doi             ipsec_doi;
        situation       identity_only;
        my_identifier   address 172.16.5.4;
        peers_identifier        address 192.168.1.12;
        lifetime        time 8 hour;
        passive         off;
        proposal_check  obey;
#       nat_traversal   off;
        generate_policy off;
```

```
                        proposal {
                                encryption_algorithm    blowfish;
                                hash_algorithm          md5;
                                authentication_method   pre_shared_key;
                                lifetime time           30 sec;
                                dh_group                1;
                        }
}

sainfo  (address 10.246.38.0/24 any address 10.0.0.0/24 any)    # address $network/
$netmask $type address $network/$netmask $type ( $type being any or esp)
{                                                               # $network must be the ↺
two internal networks you are joining.
        pfs_group       1;
        lifetime        time    36000 sec;
        encryption_algorithm    blowfish,3des,des;
        authentication_algorithm        hmac_md5,hmac_sha1;
        compression_algorithm   deflate;
}
```

Eine Beschreibung der verfügbaren Optionen finden Sie in der Manualpage von `racoon.conf` .

Die *Security Policy Database* (SPD) muss noch konfiguriert werden, so dass FreeBSD und racoon in der Lage sind den Netzwerkverkehr zwischen den Hosts zu ver- und entschlüsseln.

Dies wird durch ein Shellskript ähnlich wie das folgende, das auf dem Firmennetzwerk-Gateway liegt, ausgeführt. Diese Datei wird während der Systeminitialisierung ausgeführt und sollte unter `/usr/local/etc/racoon/setkey.conf` gespeichert werden.

```
flush;
spdflush;

# To the home network
spdadd 10.246.38.0/24 10.0.0.0/24 any -P out ipsec esp/tunnel/172.16.5.4-192.168.1.12/
use;
spdadd 10.0.0.0/24 10.246.38.0/24 any -P in ipsec esp/tunnel/192.168.1.12-172.16.5.4/use;
```

Einmal abgespeichert, kann racoon durch das folgende Kommando auf beiden Gateways gestartet werden:

```
# /usr/local/sbin/racoon -F -f /usr/local/etc/racoon/racoon.conf -l /var/log/racoon.log
```

Die Ausgabe sollte so ähnlich aussehen:

```
corp-net# /usr/local/sbin/racoon -F -f /usr/local/etc/racoon/racoon.conf
Foreground mode.
2006-01-30 01:35:47: INFO: begin Identity Protection mode.
2006-01-30 01:35:48: INFO: received Vendor ID: KAME/racoon
2006-01-30 01:35:55: INFO: received Vendor ID: KAME/racoon
2006-01-30 01:36:04: INFO: ISAKMP-SA established 172.16.5.4[500]-192.168.1.12[500] ↺
spi:623b9b3bd2492452:7deab82d54ff704a
2006-01-30 01:36:05: INFO: initiate new phase 2 negotiation: 172.16.5.4[0]192.168.1.12[0]
2006-01-30 01:36:09: INFO: IPsec-SA established: ESP/Tunnel 192.168.1.12[0]-
>172.16.5.4[0] spi=28496098(0x1b2d0e2)
2006-01-30 01:36:09: INFO: IPsec-SA established: ESP/Tunnel 172.16.5.4[0]-
>192.168.1.12[0] spi=47784998(0x2d92426)
2006-01-30 01:36:13: INFO: respond new phase 2 negotiation: 172.16.5.4[0]192.168.1.12[0]
2006-01-30 01:36:18: INFO: IPsec-SA established: ESP/Tunnel 192.168.1.12[0]-
>172.16.5.4[0] spi=124397467(0x76a279b)
2006-01-30 01:36:18: INFO: IPsec-SA established: ESP/Tunnel 172.16.5.4[0]-
>192.168.1.12[0] spi=175852902(0xa7b4d66)
```

Um sicherzustellen, dass der Tunnel richtig funktioniert, wechseln Sie auf eine andere Konsole und benutzen Sie tcpdump(1) mit dem folgenden Befehl, um sich den Netzwerkverkehr anzusehen. Tauschen Sie `em0` durch die richtige Netzwerkkarte aus:

```
# tcpdump -i em0 host  172.16.5.4 and dst 192.168.1.12
```

Die Ausgabe der Konsole sollte dem hier ähneln. Wenn nicht, gibt es ein Problem und ein Debuggen der ausgegebenen Daten ist notwendig.

```
01:47:32.021683 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com: ESP↺
(spi=0x02acbf9f,seq=0xa)
01:47:33.022442 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com: ESP↺
(spi=0x02acbf9f,seq=0xb)
01:47:34.024218 IP corporatenetwork.com > 192.168.1.12.privatenetwork.com: ESP↺
(spi=0x02acbf9f,seq=0xc)
```

An diesem Punkt sollten beide Netzwerke verfügbar sein und den Anschein haben, dass sie zum selben Netzwerk gehören. Meistens sind beide Netzwerke durch eine Firewall geschützt. Um den Netzwerkverkehr zwischen den beiden Netzwerken zu erlauben, ist es notwendig Regeln zu erstellen. Für die ipfw(8) Firewall fügen Sie folgende Zeilen in die Firewall-Konfigurationsdatei ein:

```
ipfw add 00201 allow log esp from any to any
ipfw add 00202 allow log ah from any to any
ipfw add 00203 allow log ipencap from any to any
ipfw add 00204 allow log udp from any 500 to any
```

Anmerkung

Die Regelnummern müssen eventuell, je nach ihrer Hostkonfiguration, angepasst werden.

Für Benutzer der pf(4)- oder ipf(8)-Firewall sollte folgendes funktionieren:

```
pass in quick proto esp from any to any
pass in quick proto ah from any to any
pass in quick proto ipencap from any to any
pass in quick proto udp from any port = 500 to any port = 500
pass in quick on gif0 from any to any
pass out quick proto esp from any to any
pass out quick proto ah from any to any
pass out quick proto ipencap from any to any
pass out quick proto udp from any port = 500 to any port = 500
pass out quick on gif0 from any to any
```

Zum Ende, um dem Computer den Start vom VPN während der Systeminitialisierung zu erlauben, fügen Sie folgende Zeilen in ihre `/etc/rc.conf` : ein

```
ipsec_enable="YES"
ipsec_program="/usr/local/sbin/setkey"
ipsec_file="/usr/local/etc/racoon/setkey.conf" # allows setting up spd policies on boot
racoon_enable="yes"
```

14.10. OpenSSH

Beigetragen von Chern Lee.

OpenSSH stellt Werkzeuge bereit, um sicher auf entfernte Maschinen zuzugreifen. Zusätzlich können TCP/IP-Verbindungen sicher durch SSH weitergeleitet (getunnelt) werden. Mit SSH werden alle Verbindungen verschlüsselt, dadurch wird verhindert, dass die Verbindung zum Beispiel abgehört oder übernommen (*Hijacking*) werden kann.

OpenSSH wird vom OpenBSD-Projekt gepflegt und wird in der Voreinstellung von FreeBSD installiert. OpenSSH ist mit den SSH-Protokollen der Versionen 1 und 2 kompatibel.

14.10.1. Vorteile von OpenSSH

Wenn Daten unverschlüsselt über das Netzwerk gesendet werden, besteht die Gefahr, das Benutzer/Passwort Kombinationen oder alle Daten an beliebiger Stelle zwischen dem Client und dem Server abgehört werden. Mit OpenSSH stehen eine Reihe von Authentifizierungs- und Verschlüsselungsmethoden zur Verfügung, um das zu verhindern.

14.10.2. Den SSH-Server aktivieren

Um zu überprüfen, ob sshd(8) auf dem System aktiviert ist, suchen Sie in `rc.conf` nach der folgenden Zeile:

```
sshd_enable="YES"
```

Ist diese Zeile vorhanden, wird sshd(8), der OpenSSH-Daemon, beim Systemstart automatisch aktiviert. Alternativ kann OpenSSH auch über service(8) gestartet werden:

```
# service sshd start
```

14.10.3. SSH Client

Benutzen Sie ssh(1) um sich mit einem System zu verbinden, auf dem sshd(8) läuft. Verwenden Sie dazu den Benutzernamen und den Namen des Rechners, mit dem Sie sich verbinden möchten:

```
# ssh user@example.com
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Host 'example.com' added to the list of known hosts.
user@example.com's password: *******
```

SSH speichert einen Fingerabdruck des Serverschlüssels. Die Aufforderung, `yes` einzugeben, erscheint nur bei der ersten Verbindung zu einem Server. Weitere Verbindungen zu dem Server werden gegen den gespeicherten Fingerabdruck des Schlüssels geprüft und der Client gibt eine Warnung aus, wenn sich der empfangene Fingerabdruck von dem gespeicherten unterscheidet. Die Fingerabdrücke werden in `~/.ssh/known_hosts` gespeichert.

In der Voreinstellung akzeptieren aktuelle Versionen von sshd(8) nur SSH v2 Verbindungen. Wenn möglich, wird der Client versuchen Version 2 zu verwenden, ist dies nicht möglich, fällt er auf Version 1 zurück. Der Client kann gezwungen werden, nur eine der beiden Versionen zu verwenden, indem die Option `-1` oder `-2` übergeben wird. Die Unterstützung für Version 1 ist nur noch aus Kompatibilitätsgründen zu älteren Versionen enthalten.

14.10.4. Secure Copy

Mit scp(1) lassen sich Dateien in einer sicheren Weise auf entfernte Maschinen übertragen.

```
#  scp user@example.com:/COPYRIGHT COPYRIGHT
user@example.com's password:
COPYRIGHT            100% |*****************************|  4735
00:00
#
```

Da der Fingerabdruck schon im vorigen Beispiel abgespeichert wurde, wird er bei der Verwendung von `scp` in diesem Beispiel überprüft. Da die Fingerabdrücke übereinstimmen, wird keine Warnung ausgegeben.

Die Argumente, die scp(1) übergeben werden, gleichen denen von cp(1) in der Beziehung, dass die ersten Argumente die zu kopierenden Dateien sind und das letzte Argument den Bestimmungsort angibt. Da die Dateien über das Netzwerk kopiert werden, können ein oder mehrere Argumente die Form `user@host:<path_to_remote_file>` besitzen.

14.10.5. Konfiguration

Die für das ganze System gültigen Konfigurationsdateien des OpenSSH-Daemons und des Clients befinden sich in `/etc/ssh`.

Die Client-Konfiguration befindet sich in `ssh_config`, die des Servers befindet sich in `sshd_config`. Für beide Dateien existieren Manualpages, welche die einzelnen Konfigurationsoptionen beschreiben.

14.10.6. ssh-keygen(1)

Mit ssh-keygen(1) können DSA- oder RSA-Schlüssel für einen Benutzer erzeugt werden, die anstelle von Passwörtern verwendet werden können:

```
% ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_dsa):
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_dsa.
Your public key has been saved in /home/user/.ssh/id_dsa.pub.
The key fingerprint is:
bb:48:db:f2:93:57:80:b6:aa:bc:f5:d5:ba:8f:79:17 user@host.example.com
```

ssh-keygen(1) erzeugt einen öffentlichen und einen privaten Schlüssel für die Authentifizierung. Der private Schlüssel wird in `~/.ssh/id_dsa` oder `~/.ssh/id_rsa` gespeichert, während sich der öffentliche Schlüssel in `~/.ssh/id_dsa.pub` oder `~/.ssh/id_rsa.pub` befindet, je nachdem, ob es sich um einen DSA- oder einen RSA-Schlüssel handelt. Der öffentliche Schlüssel muss sowohl für RSA- als auch für DSA-Schlüssel in `~/.ssh/authorized_keys` auf dem entfernten Rechner aufgenommen werden, damit der Schlüssel funktioniert.

Damit werden Verbindungen zu der entfernten Maschine über SSH-Schlüsseln anstelle von Passwörtern authentifiziert.

Warnung

Viele Benutzer denken, dass die Verwendung von Schlüsseln generell sicher ist. Sie verwenden dann einen Schlüssel ohne eine Passphrase. Dies ist jedoch sehr *gefährlich*. Ein Administrator kann überprüfen, ob ein Schlüsselpaar mit einer Passphrase geschützt ist. Wenn die Datei mit dem privaten Schlüssel den Text `ENCRYPTED` enthält, dann hat der Benutzer eine Passphrase verwendet. Um die Benutzer zusätzlich zu schützen, kann ein `from`-Feld in der Datei des öffentlichen Schlüssels hinzugefügt werden. Zum Beispiel würde das Hinzufügen von `from="192.168.10.5"` vor dem `ssh-rsa` - oder `ssh-dsa` -Präfix dafür sorgen, dass sich ein bestimmter Benutzer nur noch von dieser IP-Adresse anmelden darf.

Wenn bei der Erstellung der Schlüssel mit ssh-keygen(1) eine Passphrase angegeben wurde, wird der Benutzer bei jeder Anmeldung zur Eingabe des Passworts aufgefordert. Um den Umgang mit SSH-Schlüsseln zu erleichtern, kann ssh-agent(1) die Verwaltung dieser Schlüssel für Sie übernehmen. Lesen Sie dazu den Abschnitt 14.10.7, „Verwendung von SSH-Agent“.

Warnung

Die Optionen und Dateinamen sind abhängig von der OpenSSH-Version. Die für das System gültigen Optionen finden Sie in ssh-keygen(1).

14.10.7. Verwendung von SSH-Agent

Mit ssh-agent(1) und ssh-add(1) ist es möglich, SSH-Schlüssel in den Speicher zu laden, damit die Passphrase nicht jedes Mal eingegeben werden muss.

ssh-agent(1) übernimmt die Authentifizierung von ihm geladener privater Schlüssel. ssh-agent(1) sollte nur dazu verwendet werden, ein anderes Programm zu starten, beispielsweise eine Shell oder einen Window-Manager.

Um ssh-agent(1) in einer Shell zu verwenden, muss es mit einer Shell als Argument aufgerufen werden. Zudem muss die zu verwaltende Identität mit ssh-add(1) sowie deren Passphrase für den privaten Schlüssel übergeben werden. Nachdem dies erledigt ist, kann sich ein Benutzer über ssh(1) auf jedem Rechner anmelden, der einen entsprechenden öffentlichen Schlüssel besitzt. Dazu ein Beispiel:

```
% ssh-agent csh
% ssh-add
Enter passphrase for /home/user/.ssh/id_dsa:
Identity added: /home/user/.ssh/id_dsa (/home/user/.ssh/id_dsa)
%
```

Um ssh-agent(1) unter Xorg zu verwenden, muss ssh-agent(1) in `~/.xinitrc` aufgenommen werden. Dadurch können alle unter Xorg gestarteten Programme die Dienste von ssh-agent(1) nutzen. `~/.xinitrc` könnte etwa so aussehen:

```
exec ssh-agent startxfce4
```

Dadurch wird bei jedem Start von Xorg zuerst ssh-agent(1) aufgerufen, das wiederum XFCE startet. Nachdem diese Änderung durchgeführt wurde, muss Xorg neu gestartet werden. Danach können Sie mit ssh-add(1) die SSH-Schlüssel laden.

14.10.8. SSH-Tunnel

Mit OpenSSH ist es möglich, einen Tunnel zu erstellen, in dem ein anderes Protokoll verschlüsselt übertragen wird.

Das folgende Kommando erzeugt einen Tunnel für telnet(1):

```
% ssh -2 -N -f -L 5023:localhost:23 user@foo.example.com
%
```

Dieses Beispiel verwendet die folgenden Optionen:

`-2`
: Zwingt ssh(1) dazu, die Version 2 des Protokolls zu verwenden, um sich mit dem Server zu verbinden.

`-N`
: Zeigt an, dass ein Tunnel erstellt werden soll. Ohne diese Option würde ssh(1) eine normale Sitzung öffnen.

`-f`
: Zwingt ssh(1) im Hintergrund zu laufen.

`-L`
: Ein lokaler Tunnel wird in der Form *`localport:remotehost:remoteport`* angegeben. Die Verbindung wird dabei von dem lokalen Port *`localport`* auf einen entfernten Rechner weitergeleitet.

`user@foo.example.com`
: Gibt den Anmeldenamen auf dem entfernten SSH-Server an.

Ein SSH-Tunnel erzeugt einen Socket auf `localhost` und dem angegebenen Port. Jede Verbindung, die auf dem angegebenen Socket aufgemacht wird, wird dann auf den spezifizierten entfernten Rechner und Port weitergeleitet.

Im Beispiel wird der Port *5023* auf die entfernte Maschine und dort auf `localhost` Port *23* weitergeleitet. Da der Port *23* für telnet(1) reserviert ist, erzeugt das eine sichere telnet(1)-Verbindung durch einen SSH-Tunnel.

Diese Vorgehensweise kann genutzt werden, um jedes unsichere TCP-Protokoll wie SMTP, POP3 und FTP weiterzuleiten.

Beispiel 14.1. Mit ssh(1) einen sicheren Tunnel für SMTP erstellen

```
% ssh -2 -N -f -L 5025:localhost:25 user@mailserver.example.com
user@mailserver.example.com's password: *****
% telnet localhost 5025
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 mailserver.example.com ESMTP
```

Zusammen mit ssh-keygen(1) und zusätzlichen Benutzer-Accounts können leicht benutzbare SSH-Tunnel aufgebaut werden. Anstelle von Passwörtern können Schlüssel benutzt werden und jeder Tunnel kann unter einem eigenen Benutzer laufen.

14.10.8.1. Praktische Beispiele für SSH-Tunnel

14.10.8.1.1. Sicherer Zugriff auf einen POP3-Server

In diesem Beispiel gibt es einen SSH-Server, der Verbindungen von außen akzeptiert. Im selben Netzwerk befindet sich zudem noch ein Mail-Server, der POP3 spricht. Um E-Mails auf sichere Weise abzurufen, bauen Sie eine SSH-Verbindung zu dem SSH-Server im Netzwerk auf und tunneln von dort zum Mail-Server weiter.

```
% ssh -2 -N -f -L 2110:mail.example.com:110 user@ssh-server.example.com
user@ssh-server.example.com's password: ******
```

Wenn Sie den Tunnel eingerichtet haben, konfigurieren Sie den Mail-Client so, dass er POP3 Anfragen zu `localhost` auf Port 2110 sendet. Diese Verbindung wird dann über den gesicherten Tunnel zu `mail.example.com` weitergeleitet.

14.10.8.1.2. Umgehen einer strengen Firewall

Einige Netzwerkadministratoren stellen sehr drakonische Firewall-Regeln auf, die nicht nur einkommende Verbindungen filtern, sondern auch ausgehende. Es kann sein, dass Sie externe Maschinen nur über die Ports 22 und 80 (SSH und Web) erreichen.

Die Lösung hier ist es, eine SSH-Verbindung zu einer Maschine außerhalb der Firewall aufzumachen und durch diese zum gewünschten Dienst zu tunneln.

```
% ssh -2 -N -f -L 8888:music.example.com:8000 user@unfirewalled-system.example.org
user@unfirewalled-system.example.org's password: *******
```

In diesem Beispiel benutzt ein Ogg Vorbis Client `localhost` und Port 8888. Die Verbindung wird dann zu `music.example.com` Port 8000 weitergeleitet. Die Firewall wurde somit erfolgreich umgangen.

14.10.9. Die Option `AllowUsers`

Es ist in der Regel ein gute Idee, festzulegen, welche Benutzer sich von welchem Rechner aus anmelden können. Dies lässt sich beispielsweise über die Option `AllowUsers` festlegen. Soll sich etwa nur `root` vom Rechner mit der IP-Adresse `192.168.1.32` aus einwählen dürfen, würden Sie folgenden Eintrag in `/etc/ssh/sshd_config` aufnehmen:

```
AllowUsers root@192.168.1.32
```

Damit sich `admin` von jedem Rechner aus anmelden kann, geben Sie nur den Benutzernamen an:

```
AllowUsers admin
```

Sie können auch mehrere Benutzer in einer Zeile aufführen:

```
AllowUsers root@192.168.1.32 admin
```

Anmerkung

Nur Benutzer, die in dieser Liste aufgeführt ist, dürfen sich auf diesem Rechner anmelden.

Nachdem Sie `/etc/ssh/sshd_config` angepasst haben, muss sshd(8) seine Konfigurationsdateien neu einlesen. Dazu geben Sie Folgendes ein:

```
# /etc/rc.d/sshd reload
```

14.10.10. Weiterführende Informationen

OpenSSH

ssh(1) scp(1) ssh-keygen(1) ssh-agent(1) ssh-add(1) ssh_config(5) für Client Optionen.

sshd(8) sftp-server(8) sshd_config(5) für Server Optionen.

14.11. Zugriffskontrolllisten für Dateisysteme

Beigetragen von Tom Rhodes.

Zugriffskontrolllisten (*Access Control Lists*, ACL) erweitern die normalen Zugriffsrechte von UNIX® Systemen auf eine kompatible (POSIX®.1e) Weise und bieten feiner granulierte Sicherheitsmechanismen.

Der `GENERIC`-Kernel von FreeBSD bietet ACL-Unterstützung für UFS-Dateisysteme. Benutzer, die es vorziehen einen eigenen Kernel zu übersetzen, müssen die folgende Option in die Kernelkonfigurationsdatei aufnehmen:

```
options UFS_ACL
```

Das System gibt eine Warnung aus, wenn ein Dateisystem mit ACLs eingehangen werden soll und die Unterstützung für ACLs nicht im Kernel aktiviert ist. Das Dateisystem muss weiterhin erweiterte Attribute zur Verfügung stellen, damit ACLs verwendet werden können. UFS2 stellt diese Attribute standardmäßig zur Verfügung.

Anmerkung

Die Konfiguration erweiterter Attribute auf UFS1 ist mit einem höheren Aufwand als die Konfiguration erweiterter Attribute auf UFS2 verbunden. Zugriffskontrolllisten sollten daher mit UFS2 verwendet werden.

Die Angabe der Option `acl` in `/etc/fstab` aktiviert Zugriffskontrolllisten für ein Dateisystem. Die bevorzugte Möglichkeit ist die Verwendung von Zugriffskontrolllisten mit tunefs(8) (Option `-a`), im Superblock des Dateisystems festzuschreiben. Diese Möglichkeit hat mehrere Vorteile:

- Nochmaliges Einhängen eines Dateisystems (Option `-u` von mount(8)) verändert den Status der Zugriffskontrolllisten nicht. Die Verwendung von Zugriffskontrolllisten kann nur durch Abhängen und erneutes Einhängen eines Dateisystems verändert werden. Das heißt auch, dass Zugriffskontrolllisten nicht nachträglich auf dem Root-Dateisystem aktiviert werden können.

- Die Zugriffskontrolllisten auf den Dateisystemen sind, unabhängig von den Optionen in `/etc/fstab` oder Namensänderungen der Geräte, immer aktiv. Dies verhindert auch, dass Zugriffskontrolllisten aus Versehen auf Dateisystemen ohne Zugriffskontrolllisten aktiviert werden und durch falsche Zugriffsrechte Sicherheitsprobleme entstehen.

Anmerkung

Es kann sein, dass sich der Status von Zugriffskontrolllisten später durch nochmaliges Einhängen des Dateisystems (Option `-u` von mount(8)) ändern lässt. Die momentane Variante ist aber sicherer, da der Status der Zugriffskontrolllisten nicht versehentlich geändert werden kann. Allgemein sollten Zugriffskontrolllisten auf einem Dateisystem, auf dem sie einmal verwendet wurden, nicht deaktiviert werden, da danach die Zugriffsrechte falsch sein können. Werden Zugriffskontrolllisten auf einem solchen Dateisystem wieder aktiviert, werden die Zugriffsrechte von Dateien, die sich zwischenzeitlich geändert haben, überschrieben, was zu erneuten Problemen führt.

Die Zugriffsrechte einer Datei werden durch ein + (Plus) gekennzeichnet, wenn die Datei durch Zugriffskontrolllisten geschützt ist:

```
drwx------  2 robert  robert  512 Dec 27 11:54 private
drwxrwx---+ 2 robert  robert  512 Dec 23 10:57 directory1
drwxrwx---+ 2 robert  robert  512 Dec 22 10:20 directory2
drwxrwx---+ 2 robert  robert  512 Dec 27 11:57 directory3
drwxr-xr-x  2 robert  robert  512 Nov 10 11:54 public_html
```

In diesem Beispiel sind die Verzeichnisse `directory1`, `directory2` und `directory3` durch Zugriffskontrolllisten geschützt, wohingegen das Verzeichnis `public_html` nicht geschützt ist.

14.11.1. Zugriffskontrolllisten benutzen

Das Werkzeug getfacl(1) zeigt Zugriffskontrolllisten an. Das folgende Kommando zeigt die ACLs auf der Datei `test`:

```
% getfacl test
 #file:test
 #owner:1001
 #group:1001
 user::rw-
 group::r--
 other::r--
```

Das Werkzeug setfacl(1) ändert oder entfernt ACLs auf Dateien. Zum Beispiel:

```
% setfacl -k test
```

Die Option `-k` entfernt alle ACLs einer Datei oder eines Dateisystems. Besser wäre es, die Option `-b` zu verwenden, da sie die erforderlichen Felder beibehält.

```
% setfacl -m u:trhodes:rwx,g:web:r--,o::--- test
```

Mit dem vorstehenden Kommando werden die eben entfernten Zugriffskontrolllisten wiederhergestellt. Der Befehl gibt die Fehlermeldung Invalid argument aus, wenn Sie nicht existierende Benutzer oder Gruppen als Parameter angeben.

14.12. Sicherheitsprobleme in Software Dritter überwachen

Beigetragen von Tom Rhodes.

In den letzten Jahren wurden zahlreiche Verbesserungen in der Einschätzung und dem Umgang mit Sicherheitsproblemen erzielt. Die Gefahr von Einbrüchen in ein System wird aber immer größer, da Softwarepakete von Dritten auf nahezu jedem Betriebssystem installiert und konfiguriert werden.

Die Einschätzung der Verletzlichkeit eines Systems ist ein Schlüsselfaktor für dessen Sicherheit. FreeBSD veröffentlicht zwar Sicherheitshinweise (*security advisories*) für das Basissystem, das Projekt ist allerdings nicht dazu in der Lage, dies auch für die zahlreichen Softwarepakete von Dritten zu tun. Dennoch gibt es einen Weg, auch diese Programmpakete zu überwachen. Das in der Ports-Sammlung enthaltene Programm portaudit wurde gezielt dafür entwickelt.

Der Port ports-mgmt/portaudit fragt dazu eine Datenbank, die vom FreeBSD Security Team sowie den Ports-Entwicklern aktualisiert und gewartet wird, auf bekannte Sicherheitsprobleme ab.

Bevor Sie portaudit verwenden können, müssen Sie es über die Ports-Sammlung installieren:

```
# cd /usr/ports/security/portaudit && make install clean
```

Während der Installation werden die Konfigurationsdateien für periodic(8) aktualisiert, was es portaudit erlaubt, seine Ausgabe in den täglichen Sicherheitsbericht einzufügen. Stellen Sie auf jeden Fall sicher, dass diese (an das E-Mail-Konto von `root` gesendeten) Sicherheitsberichte auch gelesen werden. An dieser Stelle ist keine weitere Konfiguration nötig.

Nach der Installation kann ein Administrator die unter `/var/db/portaudit` lokal gespeicherte Datenbank aktualisieren und sich danach durch folgenden Befehl über mögliche Sicherheitslücken der von ihm installierten Softwarepakete informieren:

```
# portaudit -Fda
```

Anmerkung

Die Datenbank wird automatisch aktualisiert, wenn periodic(8) ausgeführt wird. Der eben genannte Befehl ist daher optional, er wird aber für das folgende Beispiel benötigt.

Nach erfolgter Installation der Datenbank kann ein Administrator über die Ports-Sammlung installierte Softwarepakete Dritter jederzeit überprüfen. Dazu muss er lediglich folgenden Befehl eingeben:

```
# portaudit -a
```

Existiert in Ihren installierten Softwarepaketen eine Sicherheitslücke, wird portaudit eine Ausgabe ähnlich der folgenden produzieren:

```
Affected package: cups-base-1.1.22.0_1
Type of problem: cups-base -- HPGL buffer overflow vulnerability.
Reference: <http://www.FreeBSD.org/ports/portaudit/40a3bca2-6809-11d9-a9e7-0001020eed82.↺
html>

1 problem(s) in your installed packages found.

You are advised to update or deinstall the affected package(s) immediately.
```

Wenn Sie die angegebene URL über einen Internetbrowser aufrufen, erhalten Sie weitere Informationen über die bestehende Sicherheitslücke, wie die betroffenen Versionen, die Version des FreeBSD-Ports sowie Hinweise auf weitere Seiten, die ebenfalls Sicherheitshinweise zu diesem Problem bieten.

Portaudit ist ein mächtiges Werkzeug und insbesondere in Zusammenarbeit mit dem Port Portupgrade äußerst hilfreich.

14.13. FreeBSD Sicherheitshinweise

Beigesteuert von Tom Rhodes.

Wie für andere hochwertige Betriebssysteme auch werden für FreeBSD Sicherheitshinweise herausgegeben. Die Hinweise werden gewöhnlich auf den Sicherheits-Mailinglisten und in den Errata veröffentlicht, nachdem das Sicherheitsproblem behoben ist. Dieser Abschnitt beschreibt den Umgang mit den Sicherheitshinweisen.

14.13.1. Wie sieht ein Sicherheitshinweis aus?

FreeBSD Sicherheitshinweise haben das folgende Format:

```
=============================================================================
FreeBSD-SA-XX:XX.UTIL                                       Security Advisory
                                                            The FreeBSD Project

Topic:          denial of service due to some problem❶

Category:       core❷
Module:         sys❸
Announced:      2003-09-23❹
Credits:        Person❺
Affects:        All releases of FreeBSD❻
                FreeBSD 4-STABLE prior to the correction date
Corrected:      2003-09-23 16:42:59 UTC (RELENG_4, 4.9-PRERELEASE)
                2003-09-23 20:08:42 UTC (RELENG_5_1, 5.1-RELEASE-p6)
                2003-09-23 20:07:06 UTC (RELENG_5_0, 5.0-RELEASE-p15)
                2003-09-23 16:44:58 UTC (RELENG_4_8, 4.8-RELEASE-p8)
                2003-09-23 16:47:34 UTC (RELENG_4_7, 4.7-RELEASE-p18)
                2003-09-23 16:49:46 UTC (RELENG_4_6, 4.6-RELEASE-p21)
                2003-09-23 16:51:24 UTC (RELENG_4_5, 4.5-RELEASE-p33)
                2003-09-23 16:52:45 UTC (RELENG_4_4, 4.4-RELEASE-p43)
                2003-09-23 16:54:39 UTC (RELENG_4_3, 4.3-RELEASE-p39)❼
CVE Name:       CVE-XXXX-XXXX❽

For general information regarding FreeBSD Security Advisories,
including descriptions of the fields above, security branches, and the
following sections, please visit
http://www.FreeBSD.org/security/.

I.   Background❾


II.  Problem Description❿


III. Impact⓫


IV.  Workaround⓬


V.   Solution⓭


VI.  Correction details⓮


VII. References⓯
```

❶ Das Feld `Topic` enthält eine Beschreibung des Sicherheitsproblems und benennt das betroffene Programm.

❷ Das Feld `Category` beschreibt den betroffenen Systemteil. Mögliche Werte für dieses Feld sind `core`, `contrib` oder `ports`. Die Kategorie `core` gilt für Kernkomponenten des FreeBSD-Betriebssystems, die Kategorie `cont-`

`rib` beschreibt zum Basissystem gehörende Software Dritter beispielsweise Sendmail. Die Kategorie `ports` beschreibt Software, die Teil der Ports-Sammlung ist.

❸ Das Feld `Module` beschreibt die betroffene Komponente. Im Beispiel ist `sys` angegeben, das heißt dieses Problem betrifft eine Komponente, die vom Kernel benutzt wird.

❹ Das Feld `Announced` gibt den Zeitpunkt der Bekanntgabe des Sicherheitshinweises an. Damit existiert das Sicherheitsproblem, ist vom Sicherheits-Team bestätigt worden und eine entsprechende Korrektur wurde in das Quellcode-Repository von FreeBSD gestellt.

❺ Das Feld `Credits` gibt die Person oder Organisation an, die das Sicherheitsproblem bemerkte und gemeldet hat.

❻ Welche FreeBSD-Releases betroffen sind, ist im Feld `Affects` angegeben. Die Version einer Datei, die zum Kernel gehört, können Sie schnell mit ident(1) ermitteln. Bei Ports ist die Versionsnummer angegeben, die Sie im Verzeichnis `/var/db/pkg` finden. Wenn Sie Ihr System nicht täglich aktualisieren, ist Ihr System wahrscheinlich betroffen.

❼ Wann das Problem in welchem Release behoben wurde, steht im Feld `Corrected`.

❽ Reserviert für Informationen, über die in der Common Vulnerabilities Database nach Sicherheitslücken gesucht werden kann.

❾ Im Feld `Background` wird das betroffene Werkzeug beschrieben. Meist finden Sie hier warum das Werkzeug Bestandteil von FreeBSD ist, wofür es benutzt wird und eine kurze Darstellung der Herkunft des Werkzeugs.

❿ Im Feld `Problem Description` befindet sich eine genaue Darstellung des Sicherheitsproblems. Hier wird fehlerhafter Code beschrieben oder geschildert, wie ein Werkzeug ausgenutzt wird.

⓫ Das Feld `Impact` beschreibt die Auswirkungen des Sicherheitsproblems auf ein System, beispielsweise erweiterte Rechte oder gar Superuser-Rechte für normale Benutzer.

⓬ Im Feld `Workaround` wird eine Umgehung des Sicherheitsproblems beschrieben. Die Umgehung ist für Administratoren gedacht, die ihr System aus Zeitnot, Netzwerk-technischen oder anderen Gründen nicht aktualisieren können. Nehmen Sie Sicherheitsprobleme ernst: Auf einem betroffenen System sollte das Problem entweder behoben oder, wie hier beschrieben, umgangen werden.

⓭ Im Feld `Solution` enthält eine getestete Schritt-für-Schritt Anleitung, die das Sicherheitsproblem behebt.

⓮ Das Feld `Correction Details` enthält die Subversion-Tags der betroffenen Dateien zusammen mit zugehörigen Revisionsnummern.

⓯ Im Feld `References` finden sich Verweise auf weitere Informationsquellen. Dies können URLs zu Webseiten, Bücher, Mailinglisten und Newsgroups sein.

14.14. Prozess-Überwachung

Beigetragen von Tom Rhodes.

Prozess-Überwachung (*Process accounting*) ist ein Sicherheitsverfahren, bei dem ein Administrator verfolgt, welche Systemressourcen verwendet werden und wie sich diese auf die einzelnen Anwender verteilen. Dadurch kann das System überwacht werden und es ist sogar möglich, zu kontrollieren, welche Befehle ein Anwender eingibt.

Diese Fähigkeiten haben sowohl Vor- als auch Nachteile. Positiv ist, dass man einen Einbruchsversuch bis an den Anfang zurückverfolgen kann. Von Nachteil ist allerdings, dass durch diesen Prozess Unmengen an Protokolldateien erzeugt werden, die auch dementsprechenden Plattenplatz benötigen. Dieser Abschnitt beschreibt die Grundlagen der Prozess-Überwachung.

14.14.1. Die Prozess-Überwachung aktivieren und konfigurieren

Bevor Sie die Prozess-Überwachung verwenden können, müssen Sie diese über die folgenden Befehle aktivieren:

```
# touch /var/account/acct
# chmod 600 /var/account/acct
# accton /var/account/acct
# echo 'accounting_enable="YES"' >> /etc/rc.conf
```

Einmal aktiviert, wird sofort mit der Überwachung von CPU-Statistiken, Befehlen und anderen Vorgängen begonnen. Protokolldateien werden in einem nur von Maschinen lesbaren Format gespeichert und können über sa(8)

aufgerufen werden. Ohne Optionen gibt sa(8) Informationen wie die Anzahl der Aufrufe pro Anwender, die abgelaufene Zeit in Minuten, die gesamte CPU- und Anwenderzeit in Minuten und die durchschnittliche Anzahl der Ein- und Ausgabeoperationen aus.

Um Informationen über ausgeführte Befehle zu erhalten, verwenden Sie lastcomm(1). So können Sie etwa ermitteln, welche Befehle von wem auf welchem ttys(5) ausgeführt wurden. Dieses Beispiel zeigt die Nutzung von ls(1) durch `trhodes` auf dem Terminal `ttyp1`:

```
# lastcomm ls trhodes ttyp1
```

Zahlreiche weitere nützliche Optionen finden Sie lastcomm(1), acct(5) sowie sa(8).

14.15. Einschränkung von Ressourcen

Beigetragen von Tom Rhodes.

Seit Jahren benutzt FreeBSD die Datenbank `/etc/login.conf` um Ressourcen zu beschränken. Obwohl dies immer noch unterstützt wird, ist es nicht die optimale Methode um die Beschränkung von Ressourcen zu steuern, da Benutzer in verschiedene Gruppen (Login-Klassen) aufgeteilt werden müssen und bei Änderungen immer die Datei und die Passwortdatenbank bearbeitet werden muss. Möglicherweise benötigt ein eingeschränkter Benutzer eine zusätzliche Klasse, dann müsste die Datenbank mit `cap_mkdb` neu gebaut werden und `/etc/master.passwd` müsste ebenfalls bearbeitet werden. Zusätzlich müsste die Passwortdatenbank mit `pwd_mkdb` neu gebaut werden. Dieser Prozess kann sehr zeitaufwendig sein, abhängig davon, wie viele Benutzer bearbeitet werden müssen.

Mit rctl(8) können Ressourcen für Benutzer sehr detailliert gesteuert werden. Die Befehl unterstützt nicht nur die Kontrolle der Ressourcen für Benutzer, sondern auch die Beschränkung auf Prozesse, Jails und den ursprünglichen Login-Klassen. Diese erweiterten Funktionen bieten Administratoren und Benutzern die Möglichkeit, Ressourcen über die Kommandozeile oder über eine Konfigurationsdatei zu steuern.

Um diese Eigenschaft zu aktivieren, fügen Sie folgende Zeile in die Kernelkonfigurationsdatei:

```
options         RACCT
options        RCTL
```

Das System muss nun neu übersetzt werden. Dieser Vorgang wird in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben. Anschließend kann `rctl` benutzt werden, um die Regeln für das System festzulegen.

Die Syntax der Regeln ist einfach und wird durch *subject*, *subject-id*, *resource* und *action* gesteuert. Hier ein Beispiel für eine Regel:

```
user:trhodes:maxproc :deny=10/user
```

Diese Regel zeigt den grundlegenden Aufbau, hier mit dem Subjekt `user` und der Subjekt-ID `trhodes`. `maxproc` definiert die Anzahl der Prozesse. Die „Aktion“ `deny` verhindert, dass neue Prozesse erstellt werden. Im vorherigen Beispiel wurde für den Benutzer `trhodes` eine Beschränkung von `10` (zehn) Prozessen konfiguriert. Es sind noch weitere Aktionen verfügbar, beispielsweise die Protokollierung auf der Konsole, Benachrichtigungen an devd(8) oder das Senden eines `SIGTERM` an einen Prozess.

Beim hinzufügen von Regeln müssen einige Dinge beachtet werden. Das obige Beispiel würde den Benutzer sogar daran hindern, einfachste Dinge zu tun, nachdem er sich anmeldet und eine `screen` Sitzung gestartet hat. Sobald die Begrenzung für eine Ressource erreicht ist, wird folgende Meldung ausgegeben:

```
# man test
/usr/bin/man: Cannot fork: Resource temporarily unavailable
eval: Cannot fork: Resource temporarily unavailable
```

rctl(8) kann auch benutzt werden, um einer Jail eine Speichergrenze zuzuweisen. Eine solche Regel könnte wie folgt festgelegt werden:

```
# rctl -a jail:httpd:memoryuse:deny=2G/jail
```

Damit die Regeln auch nach einem Neustart erhalten bleiben, müssen sie in `/etc/rctl.conf` hinzugefügt werden. Dazu schreiben Sie einfach die Regel, ohne das vorhergehende Kommando. Zum Beispiel:

```
# Block jail from using more than 2G memory:
jail:httpd:memoryuse:deny=2G/jail
```

Mit `rctl` können auch Regeln entfernt werden:

```
# rctl -r user:trhodes:maxproc:deny=10/user
```

Die Manualpage zeigt auch eine Möglichkeit, alle Regeln zu entfernen. Falls es erforderlich ist alle Regeln für einen einzelnen Benutzer zu entfernen, kann dieser Befehl verwendet werden:

```
# rctl -r user:trhodes
```

Es gibt noch viele weitere Ressourcen, die verwendet werden können, um zusätzliche `subjects` zu kontrollieren. Weitere Informationen zu diesem Thema finden Sie in rctl(8).

Kapitel 15. Jails

Beigetragen von Matteo Riondato.
Übersetzt von Oliver Peter, Dirk Arlt und Johann Kois.

15.1. Übersicht

Da die Systemadministration eine schwierige Aufgabe ist, wurden viele Werkzeuge entwickelt, die Administratoren bei der Installation, Konfiguration und Wartung ihrer Systeme unterstützen sollen. Eines dieser Werkzeuge, die verwendet werden können um die Sicherheit eines FreeBSD-Systems zu erhöhen, sind *Jails*. Jails sind seit FreeBSD 4.X verfügbar und werden ständig in ihrer Nützlichkeit, Leistung, Zuverlässigkeit und Sicherheit verbessert.

Jails setzen auf dem chroot(2)-Konzept auf, das dazu verwendet wird das root-Verzeichnis einer Reihe von Prozessen zu ändern, um so eine separate, sichere Umgebung zu schaffen. Prozesse, die in einer chroot-Umgebung erstellt wurden, können nicht auf Dateien oder Ressourcen zugreifen, die sich außerhalb dieser Umgebung befinden. Dadurch ist es einem kompromittierten Dienst nicht möglich, das gesamte System zu kompromittieren. Im Laufe der Zeit wurden viele Wege gefunden, um aus einer chroot-Umgebung auszubrechen, so dass es für die Sicherung von Diensten nicht die ideale Lösung ist.

Jails verbessern das traditionelle chroot-Konzept auf unterschiedlichste Art und Weise. In einer traditionellen chroot-Umgebung sind Prozesse auf den Bereich des Dateisystems beschränkt, auf den sie zugreifen können. Der Rest der Systemressourcen (wie zum Beispiel eine Reihe von Systembenutzern, die laufenden Prozesse oder das Netzwerk-Subsystem) teilen sich die chroot-Prozesse mit dem Host-System. Jails erweitern dieses Modell nicht nur auf die Virtualisierung des Zugriffs auf das Dateisystem, sondern auch auf eine Reihe von Benutzern und das Netzwerk-Subsystem. Zudem stehen weitere Möglichkeiten zur Verfügung, den Zugriff auf eine Jail-Umgebung zu kontrollieren.

Eine Jail zeichnet sich durch folgende Merkmale aus:

- Ein Unterverzeichnisbaum: dies ist der Ausgangspunkt der Jail. Einem Prozess, der innerhalb der Jail läuft, ist es nicht mehr möglich, aus diesem Unterverzeichnis auszubrechen.
- Ein Hostname: dieser Name wird für die Jail verwendet.
- Eine IP Adresse: diese Adresse wird der Jail zugewiesen. Die IP-Adresse einer Jails ist üblicherweise ein Adress-Alias auf eine existierende Netzwerkschnittstelle.
- Ein Kommando: der Pfad einer ausführbaren Datei, die innerhalb der Jail ausgeführt werden soll. Dieser Pfad wird relativ zum root-Verzeichnis der Jail-Umgebung angegeben.

Jails haben einen eigenen Satz von Benutzern und ihren eigenen `root`-Konto. Die Rechte dieser Benutzer sind nur auf die Jail-Umgebung beschränkt. Der Benutzer `root` der Jail-Umgebung ist nicht dazu berechtigt, kritische Operationen am System außerhalb der angebundenen Jail-Umgebung durchzuführen.

Dieses Kapitel erklärt, was FreeBSD-Jails sind und wie sie eingesetzt werden. Jails sind ein sehr mächtiges Werkzeug für Administratoren, jedoch kann deren grundlegende Verwendung auch für fortgeschrittene Anwender nützlich sein.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie

- Wissen, was eine Jail ist und welche Verwendungszwecke es dafür unter FreeBSD gibt.
- Wissen, wie man eine Jail erstellt, startet und und anhält.
- Die Grundlagen der Jail-Administration (sowohl innerhalb als auch ausserhalb des Jails) kennen.

Wichtig

Jails sind ein mächtiges Werkzeug, aber sie sind kein Sicherheits-"Allheilmittel". Es ist wichtig zu beachten, dass es für einen Prozess in der Jail nicht möglich ist, von selbst auszubrechen. Es gibt jedoch Möglichkeiten, in denen ein unprivilegierter Benutzer außerhalb der Jail, mit einem privilegierten Benutzer innerhalb der Jail kooperiert, und somit erhöhte Rechte in der Host-Umgebung erlangt.

Den meisten dieser Angriffe kann vorgebeugt werden, indem sichergestellt wird, dass das Rootverzeichnis der Jail für unprivilegierte Benutzer der Host-Umgebung nicht zugänglich ist.

15.2. Jails - Definitionen

Um die für den Einsatz von Jails benötigten FreeBSD-Funktionen, deren Interna sowie die Art und Weise, mit der diese mit anderen Teilen des Betriebssystems interagieren, zu erläutern, werden in diesem Kapitel folgende Definitionen verwendet:

chroot(8) (-Befehl)
: Ein Werkzeug, das den FreeBSD-Systemaufruf chroot(2) verwendet, um das Wurzelverzeichnis eines Prozesses und all seiner Nachkömmlinge zu ändern.

chroot(2) (-Umgebung)
: Die Umgebung eines Prozesses, der in einem „chroot“ läuft. Diese beinhaltet Ressourcen, wie zum Beispiel sichtbare Abschnitte des Dateisystems, verfügbare Benutzer- und Gruppenkennungen, Netzwerkschnittstellen und weitere IPC-Mechanismen und so weiter.

jail(8) (-Befehl)
: Das Systemadministrationswerkzeug, welches es erlaubt, Prozesse innerhalb der Jail-Umgebung zu starten.

Host (-Benutzer, -Prozess, -System)
: Das verwaltende System einer Jail-Umgebung. Das Host-System hat Zugriff auf alle verfügbaren Hardwareressourcen und kann sowohl innerhalb als auch ausserhalb der Jail-Umgebung Prozesse steuern. Einer der wichtigsten Unterschiede des Host-System einer Jails ist, dass die Einschränkungen, welche für die Superuser-Prozesse innerhalb eines Jails gelten, nicht für die Prozesse des Host-Systems gelten.

Gast (-Benutzer, -Prozess, -System)
: Ein Prozess, ein Benutzer oder eine andere Instanz, deren Zugriff durch eine FreeBSD-Jail eingeschränkt ist.

15.3. Einrichtung und Verwaltung von Jails

Einige Administratoren unterscheiden zwei verschiedene Jail-Arten: „Komplette“ Jails, die ein echtes FreeBSD darstellen und Jails für einen bestimmten „Dienst“, die nur einer bestimmten Anwendung oder einem Dienst (der möglicherweise mit besonderen Privilegien laufen soll) gewidmet sind. Dies ist aber nur eine konzeptuelle Unterscheidung, die Einrichtung einer Jail bleibt davon gänzlich unberührt.

```
# setenv D /hier/ist/die/jail
# mkdir -p $D       ❶
# cd /usr/src
# make buildworld   ❷
# make installworld DESTDIR=$D    ❸
# make distribution DESTDIR=$D    ❹
```

```
# mount -t devfs devfs $D/dev     ❺
```

❶ Das Festlegen des Installationsorts für das Jail eignet sich am besten als Startpunkt. Hier wird sich die Jail innerhalb des Host-Dateisystems befinden. Eine gute Möglichkeit wäre etwa `/usr/jail/name_der_jail` , wobei *`name_der_jail`* den Hostname darstellt, über den die Jail identifiziert werden soll. Das Dateisystem unterhalb von `/usr/` stellt normalerweise ausreichend Platz für eine Jail zur Verfügung (bedenken Sie, dass eine „komplette“ Jail ein Replikat einer jeden Datei der Standardinstallation des FreeBSD-Basissystems enthält.

❷ Wenn Sie bereits ihre Systemanwendungen mittels `make world` oder `make buildworld` neu erstellt haben, können Sie diesen Schritt überspringen und die Systemanwendungen in die neue Jail installieren.

❸ Dieser Befehl wird den Verzeichnisbaum mit allen notwendigen Binärdateien, Bibliotheken, Manualpages usw. erstellen.

❹ Der `distribution`-Befehl lässt make alle benötigten Konfigurationsdateien installieren, es werden also alle installierbaren Dateien aus `/usr/src/etc/` in das Verzeichnis `/etc` der Jail installiert (also nach `$D/etc/`).

❺ Das Einhängen des devfs(8)-Dateisystems innerhalb der Jail ist nicht unbedingt notwendig. Allerdings benötigt fast jede Anwendung Zugriff auf wenigstens ein Gerät. Es ist daher sehr wichtig, den Zugriff auf Devices aus der Jail heraus zu kontrollieren, da unsaubere Einstellungen es einem Angreifer erlauben könnten, in das System einzudringen. Die Kontrolle über devfs(8) erfolgt durch die in den Manualpages devfs(8) und devfs.conf(5) beschriebenen Regeln.

Ist eine Jail einmal erst erstellt, kann sie durch jail(8) gestartet werden. jail(8) benötigt zwingend mindestens vier Argumente, die im Abschnitt Abschnitt 15.1, „Übersicht“ des Handbuchs beschrieben sind. Weitere Argumente sind möglich, um beispielsweise die Jail mit den Berechtigungen eines bestimmten Benutzers laufen zu lassen. Das Argument *`command`* hängt vom Typ der Jail ab; für ein *virtuelles System* ist `/etc/rc` eine gute Wahl, da dies dem Startvorgang eines echten FreeBSD-Systems entspricht. Bei einer *Service*-Jail hängt dieses von der Art des Dienstes ab, der in der Jail laufen soll.

Jails werden häufig mit dem Betriebssystem gestartet, da der `rc`-Mechanismus von FreeBSD dafür eine einfach zu realisierende Möglichkeit bietet.

1. Eine Liste der Jails, die mit dem Betriebssystem gestartet werden sollen, wird in die Datei rc.conf(5) geschrieben:

   ```
   jail_enable="YES"   # Set to NO to disable starting of any jails
   jail_list="www"     # Space separated list of names of jails
   ```

   ### Anmerkung

   Die Namen der Jails in der `jail_list` sollten nur alphanumerische Zeichen enthalten.

2. Für jede Jail in der `jail_list` sollten in rc.conf(5) einige Einstellungen vorgenommen werden:

   ```
   jail_www_rootdir="/usr/jail/www"     # jail's root directory
   jail_www_hostname="www.example.org"  # jail's hostname
   jail_www_ip="192.168.0.10"           # jail's IP address
   jail_www_devfs_enable="YES"          # mount devfs in the jail
   jail_www_devfs_ruleset="www_ruleset " # devfs ruleset to apply to jail
   ```

   Beim Start einer in rc.conf(5) konfigurierten Jail wird das `/etc/rc` -Skript der Jail (das "annimmt", dass es sich in einem kompletten System befindet) aufgerufen. Für Service-Jails sollten die Startskripte der Jail durch das Setzen der Option `jail_`*`jailname`*`_exec_start` entsprechend angepasst werden.

Anmerkung

Eine vollständige Liste der Optionen findet sich in der Manualpage zu rc.conf(5).

service(8) kann zum manuellen Starten und Stoppen der Jail genutzt werden, wenn ein Eintrag in `rc.conf` angelegt wurde:

```
# service jail start  www
# service jail stop  www
```

Es gibt momentan keinen sauberen Weg, eine jail(8) zu stoppen. Dies liegt daran, dass die Kommandos zum sauberen Herunterfahren eines Systems innerhalb einer Jail nicht ausgeführt werden können. Der beste Weg eine Jail zu beenden ist es daher, innerhalb der Jail den folgenden Befehl auszuführen (alternativ können Sie auch jexec(8) von außerhalb der Jail aufrufen):

```
# sh /etc/rc.shutdown
```

Weitere Informationen zu diesem Thema finden Sie in der Manualpage jail(8).

15.4. Feinabstimmung und Administration

Es gibt verschiedene Optionen, die für jede Jail gesetzt werden können und verschiedene Wege, ein FreeBSD-Host-System mit Jails zu kombinieren. Dieser Abschnitt zeigt Ihnen:

- Einige zur Verfügung stehende Optionen zur Abstimmung des Verhaltens und der Sicherheitseinstellungen, die mit einer Jail-Installation ausgeführt werden können.
- Einige der Anwendungsprogramme für das Jail-Management, die über die FreeBSD Ports-Sammlung verfügbar sind und genutzt werden können, um Jail-basierte Lösungen allumfassend umzusetzen.

15.4.1. Systemwerkzeuge zur Feinabstimmung von Jails in FreeBSD

Die Feinabstimmung einer Jail-Konfiguration erfolgt zum Großteil durch das Setzen von sysctl(8)-Variablen. Es gibt einen speziellen sysctl-Zweig, der als Basis für die Organisation aller relevanten Optionen dient: Die `security.jail.*`-Hierarchie der FreeBSD-Kerneloptionen. Die folgende Liste enthält alle jail-bezogenen sysctls (inklusiver ihrer Voreinstellungen). Die Namen sollten selbsterklärend sein, für weitergehende Informationen lesen Sie bitte die Manualpages jail(8) und sysctl(8).

- `security.jail.set_hostname_allowed: 1`
- `security.jail.socket_unixiproute_only: 1`
- `security.jail.sysvipc_allowed: 0`
- `security.jail.enforce_statfs: 2`
- `security.jail.allow_raw_sockets: 0`
- `security.jail.chflags_allowed: 0`
- `security.jail.jailed: 0`

Diese Variablen können vom Administrator des *Host-Systems* genutzt werden, um Beschränkungen hinzuzufügen oder aufzuheben, die dem Benutzer `root` als Vorgabe auferlegt sind. Beachten Sie, dass es einige Beschränkungen

gibt, die nicht verändert werden können. Der Benutzer `root` darf innheralb der jail(8) keine Dateisysteme mounten und unmounten. Ebenso ist es ihm untersagt, das devfs(8)-Regelwerk zu laden oder zu entladen. Er darf weder Firewallregeln setzen, noch administrative Aufgaben erledigen, die Modifikationen am Kernel selbst erfordern (wie bespielsweise das Setzen des `Securelevels` des Kernel.

Das FreeBSD-Basissystem enthält einen Basissatz an Werkzeugen, um Informationen über aktive Jails zu erlangen und einer Jail administrative Befehle zuzuordnen. Die Befehle jls(8) und jexec(8) sind Teil des FreeBSD-Basissystems und können für folgende Aufgaben verwendet werden:

- Das Anzeigen einer Liste der aktiven Jails und ihrer zugehörigen Jail Identifier (JID), ihrer IP-Addresse, ihres Hostnames und ihres Pfades.
- Das Herstellen einer Verbindung mit einer laufenden Jail, das Starten eines Befehls aus dem gastgebenen System heraus oder das Ausführen einer administrativen Aufgabe innerhalb der Jail selbst. Dies ist insbesondere dann nützlich, wenn der Benutzer `root` die Jail sauber herunterfahren möchte. jexec(8) kann auch zum Starten einer Shell innerhalb der Jail genutzt werden, um adminstrative Aufgaben durchzuführen:

```
# jexec 1 tcsh
```

15.4.2. High-Level-Werkzeuge zur Jail-Administration in der FreeBSD Ports-Sammlung

Unter den zahlreichen Fremdwerkzeugen für die Administration von Jails sind die sysutils/jailutils die vollständigsten und brauchbarsten. Dabei handelt es sich um eine Sammlung kleiner Anwendungen, die das jail(8)-Management vereinfachen. Weitere Informationen zu diesen Werkzeugen finden Sie auf den entsprechenden Internetseiten.

15.5. Mehrere Jails aktualisieren

Beigetragen von Daniel Gerzo.
Basierend auf einer Idee von Simon L. B. Nielsen.
Artikel geschrieben von Ken Tom.

Die Verwaltung von mehreren Jails kann problematisch sein, da jede Jail bei jedem Upgrade komplett neu gebaut werden muss. Dieser Prozess kann sehr zeitaufwändig sein, wenn eine große Anzahl von Jails erstellt oder manuell aktualisiert werden müssen.

Dieser Abschnitt beschreibt eine Methode zur Lösung dieses Problems, indem so viel wie möglich zwischen Jails, auf sichere Art und Weise, durch den Einsatz von mount_nullfs(8)-Mounts geteilt wird. Dadurch werden Aktualisierungen erleichtert und das Verteilen von verschiedenen Diensten, wie HTTP, DNS und SMTP, auf verschiedene Jails wird attraktiver. Außerdem bietet dieses Verfahren einen einfachen Weg, Jails zu erstellen, zu entfernen und zu aktualisieren.

Anmerkung

Es existieren auch einfachere Lösungen, wie zum Beispiel sysutils/ezjail, das einfachere Methoden zur Administration von Jails verwendet und daher nicht so anspruchsvoll ist, wie der hier beschriebene Aufbau.

Die Ziele des in diesem Abschnitt beschriebenen Aufbaus sind:

- Das Erstellen einer einfachen und gut verständlichen Jail Struktur, die es nicht erfordert für jede Jail ein vollständiges installworld laufen lassen zu müssen.
- Es einfach zu machen, neue Jails zu erstellen oder alte zu entfernen.

- Es einfach zu machen, bestehende Jails zu aktualisieren.
- Es einfach zu machen, einen angepassten FreeBSD-Zweig zu nutzen.
- Paranoid bezüglich Sicherheit zu sein und Angriffsmöglichkeiten weitgehend zu reduzieren.
- Soviel Platz und Inodes wie möglich einzusparen.

Dieses Design ist darauf angewiesen, dass eine read-only-Hauptvorlage in jede Jail hinein gemountet wird und dass jede Jail über wenigstens ein beschreibbares Gerät verfügt. Das Gerät kann hierbei eine separate physikalische Platte oder ein vnode unterstütztes Speichergerät sein. Im folgenden Beispiel wird ein read/write nullfs-Mount genutzt.

Das Layout des Dateisystems ist wie folgt:

- Die Jails befinden sich unterhalb der `/home` Partition.
- Jede Jail wird unterhalb des `/home/j` -Verzeichnisses gemountet.
- `/home/j/mroot` ist die Vorlage für jede Jail und die nur lesbare Partition für alle Jails.
- Unterhalb von `/home/j` wird für jede Jail ein leeres Verzeichnis angelegt.
- Jede Jail bekommt ein `/s`-Verzeichnis, das zum read/write-Teilbereich des Systems verlinkt wird.
- Jede Jail bekommt ihr eigenes read/write-System, das auf `/home/j/skel` basiert.
- Der read/write-Teilbereich jeder Jail wird in `/home/js` erstellt.

15.5.1. Erstellen der Vorlage

Dieser Abschnitt beschreibt die Schritte, die zum Erstellen der Hauptvorlage notwendig sind.

Es wird empfohlen, zunächst das FreeBSD Host-System nach den Anweisungen in Abschnitt 24.7, „Das komplette Basissystem neu bauen“ auf den aktuellen -RELEASE-Zweig zu aktualisieren. Darüber hinaus verwendet diese Vorlage sysutils/cpdup, sowie portsnap zum herunterladen der FreeBSD Ports-Sammlung.

1. Zuerst erstellen wir eine Verzeichnissstruktur für das read-only-Dateisystem, das die FreeBSD-Binärdateien für die Jails enthalten wird. Anschließend wechseln wir in den FreeBSD-Quellcodebaum und installieren das read-only-Dateisystem in die (Vorlage-)Jail.

```
# mkdir /home/j /home/j/mroot
# cd /usr/src
# make installworld DESTDIR=/home/j/mroot
```

2. Als nächstes bereiten wir die Ports-Sammlung fü die Jails vor und kopieren den FreeBSD Quellcodebaum in die Jail, da dieser für mergemaster benötigt wird:

```
# cd /home/j/mroot
# mkdir usr/ports
# portsnap -p /home/j/mroot/usr/ports fetch extract
# cpdup /usr/src /home/j/mroot/usr/src
```

3. Danach wird die Struktur für den read/write-Bereich des Systems erstellt:

```
# mkdir /home/j/skel /home/j/skel/home /home/j/skel/usr-X11R6 /home/j/skel/distfiles
# mv etc /home/j/skel
# mv usr/local /home/j/skel/usr-local
# mv tmp /home/j/skel
# mv var /home/j/skel
# mv root /home/j/skel
```

4. Nutzen Sie mergemaster, um fehlende Konfigurationsdateien zu installieren. Anschließend werden die von mergemaster erstellten Extra-Verzeichnisse entfernt:

```
# mergemaster -t /home/j/skel/var/tmp/temproot -D /home/j/skel -i
# cd /home/j/skel
# rm -R bin boot lib libexec mnt proc rescue sbin sys usr dev
```

5. Nun wird das read/write-Dateisystem mit dem read-only-Dateisystem verlinkt. Vergewissern Sie sich, dass die symbolischen Links an den korrekten `s/` Positionen erstellt werden, weil echte Verzeichnisse oder an falschen Positionen erstellte Verzeichnisse die Installation fehlschlagen lassen.

```
# cd /home/j/mroot
# mkdir s
# ln -s s/etc etc
# ln -s s/home home
# ln -s s/root root
# ln -s s/usr-local usr/local
# ln -s s/usr-X11R6 usr/X11R6
# ln -s s/distfiles usr/ports/distfiles
# ln -s s/tmp tmp
# ln -s s/var var
```

6. Zuletzt erstellen Sie eine allgemeine `/home/j/skel/etc/make.conf` mit folgendem Inhalt:

```
WRKDIRPREFIX?=  /s/portbuild
```

Dies erlaubt es, die FreeBSD-Ports innerhalb jeder Jail zu kompilieren. Das Ports-Verzeichnis ist Teil des read-only System. Der angepasste Pfad des `WRKDIRPREFIX` macht es möglich, innerhalb des read/write-Bereichs der Jail Ports zu bauen.

15.5.2. Jails erstellen

Die Jailvorlage kann nun verwendet werden, um die Jails einzurichten und in `/etc/rc.conf` zu konfigurieren. In diesem Beispiel werden drei Jails erstellt: `NS`, `MAIL` und `WWW`.

1. Fügen Sie die folgenden Zeilen in `/etc/fstab` ein, damit die read-only-Vorlage und der read/write-Bereich für alle Jails verfügbar sind:

```
/home/j/mroot   /home/j/ns     nullfs  ro  0   0
/home/j/mroot   /home/j/mail   nullfs  ro  0   0
/home/j/mroot   /home/j/www    nullfs  ro  0   0
/home/js/ns     /home/j/ns/s   nullfs  rw  0   0
/home/js/mail   /home/j/mail/s nullfs  rw  0   0
/home/js/www    /home/j/www/s  nullfs  rw  0   0
```

Um zu verhindern, dass fsck die nullfs-Mounts während des Bootens überprüft oder dass dump die Mounts sichert, müssen die letzten beiden Spalten auf `0` gesetzt werden.

2. Konfigurieren Sie die Jails in `/etc/rc.conf` :

```
jail_enable="YES"
jail_set_hostname_allow="NO"
jail_list="ns mail www"
jail_ns_hostname="ns.example.org"
jail_ns_ip="192.168.3.17"
jail_ns_rootdir="/usr/home/j/ns"
jail_ns_devfs_enable="YES"
jail_mail_hostname="mail.example.org"
jail_mail_ip="192.168.3.18"
jail_mail_rootdir="/usr/home/j/mail"
jail_mail_devfs_enable="YES"
jail_www_hostname="www.example.org"
jail_www_ip="62.123.43.14"
jail_www_rootdir="/usr/home/j/www"
```

```
jail_www_devfs_enable="YES"
```

Die Variable `jail_name_rootdir` zeigt nach `/usr/home` statt nach `/home`, da der physikalische Pfad von `/home` unter FreeBSD `/usr/home` lautet. Die Variable `jail_name_rootdir` darf im Pfad aber *keinen symbolischen Link* enthalten, weil das Jail ansonsten nicht gestartet werden kann.

3. Erstellen Sie die notwendigen Mountpunkte für die nur lesbaren Bereiche jeder Jail:

```
# mkdir /home/j/ns /home/j/mail /home/j/www
```

4. Installieren Sie mit sysutils/cpdup die read/write-Vorlage in jede Jail:

```
# mkdir /home/js
# cpdup /home/j/skel /home/js/ns
# cpdup /home/j/skel /home/js/mail
# cpdup /home/j/skel /home/js/www
```

5. An dieser Stelle werden die Jails erstellt und fü den Betrieb vorbereitet. Mounten Sie zuerst die notwendigen Dateisysteme für jede Jail. Danach starten Sie die Jails:

```
# mount -a
# service jail start
```

Die Jails sollten nun laufen. Um zu prüfen, ob sie korrekt gestartet wurden, verwenden Sie `jls`. Die Ausgabe sollte ähnlich der folgenden sein:

```
# jls
   JID  IP Address      Hostname                      Path
     3  192.168.3.17    ns.example.org                /home/j/ns
     2  192.168.3.18    mail.example.org              /home/j/mail
     1  62.123.43.14    www.example.org               /home/j/www
```

An diesem Punkt sollte es möglich sein, sich an jeder Jail anzumelden, Benutzer anzulegen und Dienste zu konfigurieren. Die Spalte `JID` gibt die Jail-Identifikationsnummer jeder laufenden Jail an. Nutzen Sie den folgenden Befehl, um administrative Aufgaben in der Jail mit der `JID 3` durchzuführen:

```
# jexec 3 tcsh
```

15.5.3. Jails aktualisieren

Das Design dieses Aufbaus bietet einen einfachen Weg, bestehende Jails zu aktualisieren, während die Ausfallzeiten minimiert werden. Außerdem bietet es die Möglichkeit, zu älteren Versionen zurückzukehren, falls irgendwelche Probleme auftreten.

1. Im ersten Schritt wird das Host-System aktualisiert. Anschließend wird eine temporäre neue read-only Vorlage `/home/j/mroot2` erstellt.

```
# mkdir /home/j/mroot2
# cd /usr/src
# make installworld DESTDIR=/home/j/mroot2
# cd /home/j/mroot2
# cpdup /usr/src usr/src
# mkdir s
```

`installworld` erzeugt einige unnötige Verzeichnisse, die nun entfernt werden sollten:

```
# chflags -R 0 var
# rm -R etc var root usr/local tmp
```

2. Erzeugen Sie neue symbolische Links für das Hauptdateisystem:

```
# ln -s s/etc etc
# ln -s s/root root
```

```
# ln -s s/home home
# ln -s ../s/usr-local usr/local
# ln -s ../s/usr-X11R6 usr/X11R6
# ln -s s/tmp tmp
# ln -s s/var var
```

3. Nun können die Jails gestoppt werden:

```
# service jail stop
```

4. Hängen Sie die originalen Dateisysteme aus, da die read/write-Systeme an das read-only System (`/s`) angeschlossen sind:

```
# umount /home/j/ns/s
# umount /home/j/ns
# umount /home/j/mail/s
# umount /home/j/mail
# umount /home/j/www/s
# umount /home/j/www
```

5. Verschieben Sie das alte read-only-Dateisystem und ersetzen Sie es durch das neue Dateisystem. Das alte Dateisystem kann so als Backup dienen, falls etwas schief geht. Die Namensgebung entspricht hier derjenigen bei der Erstellung eines neuen read-only-Dateisystems. Verschieben Sie die originale FreeBSD Ports-Sammlung in das neue Dateisystem, um Platz und Inodes zu sparen:

```
# cd /home/j
# mv mroot mroot.20060601
# mv mroot2 mroot
# mv mroot.20060601/usr/ports mroot/usr
```

6. Nun ist die neue read-only-Vorlage fertig. Sie müssen daher nur noch die Dateisysteme erneut mounten und die Jails starten:

```
# mount -a
# service jail start
```

Nutzen Sie `jls` um zu prüfen, ob die Jails korrekt gestartet wurden. Führen Sie innerhalb jeder Jail `mergemaster` aus, damit die Konfigurationsdateien aktualisiert werden.

Kapitel 16. Verbindliche Zugriffskontrolle

Written by Tom Rhodes.
Übersetzt von Benjamin Lukas.

16.1. Übersicht

In FreeBSD 5.X wurden neue Sicherheits-Erweiterungen verfügbar, die aus dem TrustedBSD-Projekt übernommen wurden und auf dem Entwurf POSIX®.1e basieren. Die beiden bedeutendsten neuen Sicherheits-Mechanismen sind Berechtigungslisten (Access Control Lists, ACL) und die verbindliche Zugriffskontrolle (Mandatory Access Control, MAC). Durch die MAC können Module geladen werden, die neue Sicherheitsrichtlinien bereitstellen. Mit Hilfe einiger Module kann beispielsweise ein eng umgrenzter Bereich des Betriebssystems gesichert werden, indem die Sicherheitsfunktionen spezieller Dienste unterstützt bzw. verstärkt werden. Andere Module wiederum betreffen in ihrer Funktion das gesamte System - alle vorhandenen Subjekte und Objekte. Das "Verbindliche" in der Namensgebung erwächst aus dem Fakt, dass die Kontrolle allein Administratoren und dem System obliegt und nicht dem Ermessen der Nutzer, wie es mit Hilfe der benutzerbestimmbaren Zugriffskontrolle (Discrectionary Access Control / DAC), dem Zugriffstandard für Dateien, gar der System V IPC in FreeBSD, normalerweise umgesetzt wird.

Dieses Kapitel wird sich auf die Grundstruktur der Verbindlichen Zugriffskontrolle und eine Auswahl der Module, die verschiedenste Sicherheitsfunktionen zur Verfügung stellen, konzentrieren.

Beim Durcharbeiten dieses Kapitels erfahren Sie:

- Welche MAC Module für Sicherheitsrichtlinien derzeit in FreeBSD eingebettet sind und wie die entsprechenden Mechanismen funktionieren.
- Was die einzelnen MAC Module an Funktionen realisieren und auch, was der Unterschied zwischen einer Richtlinie, die *mit* Labels arbeitet, und einer, die *ohne* Labels arbeitet, ist.
- Wie Sie die MAC in ein System einbetten und effizient einrichten.
- Wie die verschiedenen Richtlinienmodule einer MAC konfiguriert werden.
- Wie mit einer MAC und den gezeigten Beispielen eine sicherere Umgebung erstellt werden kann.
- Wie die Konfiguration einer MAC auf korrekte Einrichtung getestet wird.

Vor dem Lesen dieses Kapitels sollten Sie bereits:

- Grundzüge von UNIX® und FreeBSD verstanden haben. (Kapitel 4, *Grundlagen des UNIX Betriebssystems*).
- Mit den Grundzügen der Kernelkonfiguration und -kompilierung vertraut sein (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).
- Einige Vorkenntnisse über Sicherheitskonzepte im Allgemeinen und deren Umsetzung in FreeBSD im Besonderen mitbringen (Kapitel 14, *Sicherheit*).

Warnung

Der unsachgemäße Gebrauch der in diesem Kapitel enthaltenen Informationen kann den Verlust des Systemzugriffs, Ärger mit Nutzern oder die Unfähigkeit, grundlegende Funk-

tionen des X-Windows-Systems zu nutzen, verursachen. Wichtiger noch ist, dass man sich nicht allein auf die MAC verlassen sollte, um ein System zu sichern. Die MAC verbessert und ergänzt lediglich die schon existierenden Sicherheits-Richtlinien - ohne eine gründliche und fundierte Sicherheitspraxis und regelmäßige Sicherheitsprüfungen wird Ihr System nie vollständig sicher sein.

Außerdem sollte angemerkt werden, dass die Beispiele in diesem Kapitel auch genau dasselbe sein sollen, nämlich Beispiele. Es wird nicht empfohlen, diese bestimmten Beispiele auf einem Arbeitssystem umzusetzen. Das Einarbeiten der verschiedenen Sicherheitsmodule erfordert eine Menge Denkarbeit und viele Tests. Jemand, der nicht versteht, wie diese Module funktionieren, kann sich schnell darin wiederfinden, dass er (oder sie) das ganze System durchforsten und viele Dateien und Verzeichnisse neu konfigurieren muß.

16.1.1. Was in diesem Kapitel nicht behandelt wird

Dieses Kapitel behandelt einen großen Teil sicherheitsrelevanter Themen, bezogen auf die Verbindliche Zugriffskontrolle (MAC). Die gegenwärtige Entwicklung neuer MAC Module ist nicht abgedeckt. Einige weitere Module, die im MAC Framework enthalten sind, haben besondere Charakteristika, die zum Testen und Entwickeln neuer Module gedacht sind. Dies sind unter anderem mac_test(4), mac_stub(4) und mac_none(4). Für weitere Informationen zu diesen Modulen und den entsprechend angebotenen Funktionen lesen Sie bitte die Manpages.

16.2. Schlüsselbegriffe

Bevor Sie weiterlesen, müssen noch einige Schlüsselbegriffe geklärt werden. Dadurch soll jegliche auftretende Verwirrung von vornherein beseitigt und die plötzliche Einführung neuer Begriffe und Informationen vermieden werden.

- *Verbund*: Ein Verbund ist ist ein Satz von Programmen und Daten, die speziell und zusammen abgeschottet wurden, um Nutzern Zugriff auf diese ausgewiesenen Systembereiche zu gewähren. Man kann sagen, ein solcher Verbund ist eine Gruppierung, ähnlich einer Arbeitsgruppe, einer Abteilung, einem Projekt oder einem Thema. Durch die Nutzung von Verbünden (*compartments*) kann man Sicherheitsrichtlinien erstellen, die alles notwendige Wissen und alle Werkzeuge zusammenfassen.

- *Hochwassermarkierung*: Eine solche Richtlinie erlaubt die Erhöhung der Sicherheitsstufe in Abhängigkeit der Klassifikation der gesuchten bzw. bereitzustellenden Information. Normalerweise wird nach Abschluss des Prozesses die ursprüngliche Sicherheitsstufe wieder hergestellt. Derzeit enthält die MAC Grundstruktur keine Möglichkeit, eine solche Richtlinie umzusetzen, der Vollständigkeit halber ist die Definition hier jedoch aufgeführt.

- *Integrität*: Das Schlüsselkonzept zur Klassifizierung der Vertraulichkeit von Daten nennt man Integrität. Je weiter die Integrität erhöht wird, umso mehr kann man den entsprechenden Daten vertrauen.

- *Label*: Ein Label ist ein Sicherheitsmerkmal, welches mit Dateien, Verzeichnissen oder anderen Elementen im System verbunden wird. Man sollte es wie einen Vertraulichkeitsstempel auffassen, der Dateien angehört wie beispielsweise die Zugriffszeit, das Erstellungsdatum oder auch der Name; sobald Dateien derart gekennzeichnet werden, bezeichnen diese Label die sicherheitsrelevanten Eigenschaften. Zugriff ist nur noch dann möglich, wenn das zugreifende Subjekt eine korrespondierende Kennzeichnung trägt. Die Bedeutung und Verarbeitung der Label-Werte ist von der Einrichtung der Richtlinie abhängig: Während einige Richtlinien das Label zum Kennzeichnen der Vertraulichkeit oder Geheimhaltungsstufe eines Objekts nutzen, können andere Richtlinien an derselben Stelle Zugriffsregeln festschreiben.

- *Level*: Eine erhöhte oder verminderte Einstellung eines Sicherheitsmerkmals. Wenn das Level erhöht wird, wird auch die ensprechende Sicherheitsstufe angehoben.

- *Niedrigwassermarkierung*: Eine solche Richtlinie erlaubt das Herabstufen des Sicherheitslevels, um weniger sensible Daten verfügbar zu machen. In die meisten Fällen wird das ursprüngliche Sicherheitslevel des Nutzers wie-

derhergestellt, sobald der Vorgang abgeschlossen ist. Das einzige Modul in FreeBSD, welches von dieser Richtlinie Gebrauch macht, ist mac_lomac(4).

- *Multilabel*: Die Eigenschaft `multilabel` ist eine Dateisystemoption, die entweder im Einzelbenutzermodus mit Hilfe des Werkzeugs tunefs(8), während des Bootvorgangs in der Datei fstab(5) oder aber beim Erstellen einen neues Dateisystems aktiviert werden kann. Diese Option erlaubt einem Administrator, verschiedenen Objekten unterschiedliche Labels zuzuordnen - kann jedoch nur zusammen mit Modulen angewendet werden, die auch tatsächlich mit Labels arbeiten.

- *Objekt*: Ein Objekt oder auch Systemobjekt ist theoretisch eine Einheit, durch welche Information fließt, und zwar unter der Lenkung eines *Subjektes*. Praktisch schliesst diese Definition Verzeichnisse, Dateien, Felder, Bildschirme, Tastaturen, Speicher, Bandlaufwerke, Drucker und jegliche anderen Datenspeicher- oder -verarbeitungsgeräte ein. Im Prinzip ist ein Objekt ein Datenkontainer oder eine Systemressource - Zugriff auf ein *Objekt* bedeutet, auf Daten zuzugreifen.

- *Richtlinie*: Eine Sammlung von Regeln, die definiert, wie Zielvorgaben umgesetzt werden, nennt man Richtlinie. Eine *Richtlinie* dokumentiert normalerweise, wie mit bestimmten Elementen umgegangen wird. Dieses Kapitel faßt den Begriff in diesem Kontext als *Sicherheitsrichtlinie* auf; als eine Sammlung von Regeln, die den Fluß von Daten und Informationen kontrolliert und die gleichzeitig definiert, wer auf diese Daten und Informationen zugreifen darf.

- *Anfälligkeit*: Dieser Begriff wird normalerweise verwendet, wenn man über MLS (Multi Level Security) spricht. Das Anfälligkeits-Level beschreibt, wie wichtig oder geheim die Daten sein sollen. Um so höher das Anfälligkeits-Level, um so wichtiger die Geheimhaltung bzw. Vertraulichkeit der Daten.

- *Einzel-Label*: Von einem Einzel-Label spricht man, wenn für ein ganzes Dateisystem lediglich ein einziges Label verwendet wird, um Zugriffskontrolle über den gesamten Datenfluss zu erzwingen. Sobald diese Option verwendet wird - und das ist zu jeder Zeit, wenn die Option `multilabel` nicht explizit gesetzt wurde - sind alle Dateien und Verzeichnisse mit dem gleichen Label gekennzeichnet.

- *Subjekt*: Ein Subjekt ist jedwede Einheit, die Information in Fluss zwischen Objekten bringt: Zum Beispiel ein Nutzer, ein Nutzerprozessor, ein Systemprozeß usw. In FreeBSD handelt es sich meistens um einen Thread, der als Prozeß im Namen eines Nutzers arbeitet.

16.3. Erläuterung

Mit all diesen neuen Begriffen im Kopf können wir nun überlegen, wie die Möglichkeiten der verbindlichen Zugriffskontrolle (MAC) die Sicherheit eines Betriebssystems als Ganzes erweitern. Die verschiedenen Module, die durch die MAC bereitgestellt werden, können verwendet werden, um das Netzwerk oder Dateisysteme zu schützen, Nutzern den Zugang zu bestimmten Ports oder Sockets zu verbieten und vieles mehr. Die vielleicht beste Weise, die Module zu verwenden, ist, sie miteinander zu kombinieren, indem mehrere Sicherheitsrichtlinienmodule gleichzeitig eine mehrschichtige Sicherheitsumgebung schaffen. Das ist etwas anderes als singuläre Richtlinien wie zum Beispiel die Firewall, die typischerweise Elemente eines Systems stabilisiert, das nur für einen speziellen Zweck verwendet wird. Der Verwaltungsmehraufwand ist jedoch von Nachteil, zum Beispiel durch die Verwendung von mehreren Labels oder dem eigenhändigen Erlauben von Netzwerkzugriffen für jeden einzelnen Nutzer.

Solche Nachteile sind allerdings gering im Vergleich zum bleibenden Effekt der erstellten Struktur. Die Möglichkeit zum Beispiel, für konkrete Anwendungen genau die passenden Richtlinien auszuwählen und einzurichten, senkt gleichzeitig die Arbeitskosten. Wenn man unnötige Richtlinien aussortiert, kann man die Gesamtleistung des Systems genauso steigern wie auch eine höhere Anpassungsfähigkeit gewährleisten. Eine gute Umsetzung der MAC beinhaltet eine Prüfung der gesamten Sicherheitsanforderungen und einen wirksamen Einsatz der verschiedenen Module.

Ein System, auf dem eine MAC verwendet wird, muß zumindest garantieren, dass einem Nutzer nicht gestattet wird, Sicherheitsmerkmale nach eigenem Ermessen zu verändern; dass Arbeitswerkzeuge, Programme und Skripte, innerhalb der Beschränkungen arbeiten können, welche die Zugriffsregeln der ausgewählten Module dem System auferlegen; und dass die volle Kontrolle über die Regeln der MAC beim Administrator ist und bleibt.

Es ist die einsame Pflicht des zuständigen Administrators, die richtigen Module sorgfältig auszuwählen. Einige Umgebungen könnten eine Beschränkung der Zugriffe über die Netzwerkschnittstellen benötigen - hier wären die Module mac_portacl(4), mac_ifoff(4) und sogar mac_biba(4) ein guter Anfang. In anderen Fällen muß man sehr strenge Vertraulichkeit von Dateisystemobjekten gewährleisten - dafür könnte man mac_bsdextended(4) oder mac_mls(4) einsetzen.

Die Entscheidung, welche Richtlinien angewandt werden, kann auch anhand der Netzwerk-Konfiguration getroffen werden. Nur bestimmten Benutzern soll erlaubt werden, via ssh(1) auf das Netzwerk oder Internet zuzugreifen - mac_portacl(4) wäre eine gute Wahl. Aber für was entscheidet man sich im Falle eines Dateisystems? Soll der Zugriff auf bestimmte Verzeichnisse von spezifischen Nutzern oder Nutzergruppen separiert werden? Oder wollen wir den Zugriff durch Nutzer oder Programme auf spezielle Dateien einschränken, indem wir gewisse Objekte als geheim einstufen?

Der Zugriff auf Objekte kann einigen vertraulichen Nutzern gestattet werden, anderen wiederum verwehrt. Als Beispiel sei hierzu ein großes Entwicklerteam angeführt, das in kleine Gruppen von Mitarbeitern aufgeteilt wurde. Die Entwickler von Projekt A dürfen nicht auf Objekte zugreifen, die von den Entwicklern von Projekt B geschrieben wurden. Sie müssen aber trotzdem auf Objekte zugreifen können, die von einem dritten Entwicklerteam geschaffen wurden - alles in allem eine verzwickte Situation. Wenn man die verschiedenen Module der MAC richtig verwendet, können Anwender in solche Gruppen getrennt und ihnen der Zugriff zu den gewünschten Systemobjekten gestattet werden - ohne Angst haben zu müssen, dass Informationen in die falschen Hände geraten.

So hat jedes Modul, das eine Sicherheitsrichtlinie verfügbar macht, einen eigenen Weg, die Sicherheit des Systems zu verstärken. Die Auswahl der Module sollte auf einem gut durchdachten Sicherheitskonzept gründen. In vielen Fällen muß das gesamte Konzept eines Systems überarbeitet und neu eingepflegt werden. Ein guter Überblick über die Möglichkeiten der verschiedenen von der MAC angebotenen Module hilft einem Administrator, die besten Richtlinien für seine spezielle Situation auszuwählen.

Im FreeBSD-Standardkernel ist die Option zur Verwendung der MAC nicht enthalten. Daher muß die Zeile

```
options     MAC
```

der Kernelkonfiguration hinzugefügt und der Kernel neu übersetzt und installiert werden.

Achtung

Verschiedenen Anleitungen für die MAC empfehlen, die einzelnen Module direkt in den Kernel einzuarbeiten. Dabei ist es jedoch möglich, das System aus dem Netzwerk auszusperren oder gar schlimmeres. Die Arbeit mit der MAC ist ähnlich der Arbeit mit einer Firewall - man muß, wenn man sich nicht selbst aus dem System aussperren will, genau aufpassen. Man sollte sich eine Möglichkeit zurechtlegen, wie man eine Implementation einer MAC rückgängig machen kann - genauso wie eine Ferninstallation über das Netzwerk nur mit äußerster Vorsicht vorgenommen werden sollte. Es wird daher empfohlen, die Module nicht in den Kernel einzubinden, sondern sie beim Systemstart via `/boot/loader.conf` zu laden.

16.4. MAC Labels verstehen

MAC Label sind Sicherheitsmerkmale, die, wenn sie zum Einsatz kommen, allen Subjekten und Objekten im System zugeordnet werden.

Wenn ein Administrator ein solches Merkmal bzw. Attribut setzen will, muß er/sie verstehen können, was da genau passiert. Die Attribute, die im speziellen Fall zu vergeben sind, hängen vom geladenen Modul und den darin jeweils implementierten Richtlinien ab. Jedes dieser Richtlinienmodule setzt die Arbeit mit seinen entsprechenden Attributen in individueller Weise um. Falls der Nutzer nicht versteht, was er da konfiguriert, oder auch, was seine Kon-

figuration für Begleiterscheinungen mit sich bringt, ergibt sich meist als Resultat ein unerwartetes, ja sogar unerwünschtes Verhalten des gesamten Systems.

Ein Label, einem Objekt verliehen, wird verwendet, um anhand einer Richtlinie eine sicherheitsrelevante Entscheidung über Zugriffsrechte zu fällen. In einigen Richtlinien enthält bereits das Label selbst alle dafür nötigen Informationen. Andere Richtlinien verwenden diese Informationen, um zunächst ein komplexes Regelwerk abzuarbeiten.

Wenn man zum Beispiel einer Datei das Attribut `biba/low` zuordnet, wird dieses durch das Biba Sicherheitsrichtlinienmodul, und zwar mit dem Wert „low“, verarbeitet.

Einige der Richtlinienmodule, die die Möglichkeit zum Vergeben von Labels unter FreeBSD unterstützen, bieten drei vordefinierte Labels an. Dieses nennen sich „high“, „low“ und „equal“. Obwohl die verschiedenen Module die Zugriffskontrolle auf verschiedene Weisen regeln, kann man sich sicher sein, das das „low“-Label der untersten, unsichersten Einstellung entspricht, das „equal“-Label die Verwendung des Moduls für das jeweilige Objekt oder Subjekt deaktiviert - und das „high“-Label die höchstmögliche Einstellung erzwingt. Im Speziellen gilt diese Aussage für die Richtlinien(-module) MLS und Biba.

In den meisten Umgebungen, sogenannten Single Label Environments, wird Objekten nur ein einzelnes Label zugewiesen. Dadurch wird nur ein Regelsatz für die Zugriffskontrolle auf das gesamte System verwendet - und das ist meistens auch tatsächlich ausreichend. Es gibt wenige Fälle, in denen mehrere Labels auf Dateisystemobjekte oder -subjekte verwendet werden. In einem solchen Fall muß das Dateisystem mit der tunefs(8)-Option `multilabel` angepaßt werden, da `single label` die Standardeinstellung ist.

Bei der Verwendung von Biba oder MLS kann man numerische Labels vergeben, die genau das Level angeben, an welcher Stelle in der Hierarchie das Subjekt oder Objekt einzuordnen ist. Dieses numerische Level wird verwendet, um Informationen in verschiedene Gruppen aufzuteilen oder zu sortieren - damit zum Beispiel nur Subjekte, die zu einer gewissen Vertraulichkeitsstufe gehören, Zugang zu einer Gruppe von Objekten erhalten.

In den meisten Fällen wird ein Administrator nur ein einzelnes Label für das gesamte Dateisystem verwenden.

Moment mal, dass ist doch dasselbe wie DAC! Ich dachte, MAC würde die Kontrolle strengstens an den Administrator binden! Diese Aussage hält immer noch stand - `root` ist derjenige, der die Kontrolle ausübt und die Richtlinie konfiguriert, so dass Nutzer in die entsprechenden, angemessenen Kategorien / Zugriffsklassen eingeordnet werden. Nunja, einige Module schränken `root` selbst ein. Die Kontrolle über Objekte wird dann einer Gruppe zugewiesen, jedoch hat `root` die Möglichkeit, die Einstellungen jederzeit zu widerrufen oder zu ändern. Dies ist das Hierarchie/Freigabe-Modell, das durch Richtlinien wie MLS oder Biba bereitgestellt wird.

16.4.1. Konfigurieren der Labels

Gewissermaßen alle Aspekte der Labelkonfiguration werden durch Werkzeuge das Basissystems umgesetzt. Die entsprechenden Kommandos bieten eine einfache Schnittstelle zum Konfigurieren, Manipulieren und auch Verifizieren der gekennzeichneten Objekte.

Mit den beiden Kommandos setfmac(8) und setpmac(8) kann man eigentlich schon alles machen. Das Kommando `setfmac` wird verwendet, um ein MAC-Label auf einem Systemobjekt zu setzen, `setpmac` hingegen zum Setzen von Labels auf Systemsubjekte. Als Beispiel soll hier dienen:

```
# setfmac biba/high test
```

Wenn bei der Ausführung dieses Kommandos keine Fehler aufgetreten sind, gelangt man zur Eingabeaufforderung zurück. Nur wenn ein Fehler auftritt, verhalten sich diese Kommandos nicht still, ganz wie auch die Kommandos chmod(1) und chown(8). In einigen Fällen wird dieser Fehler Permission denied lauten und gewöhnlich dann auftreten, wenn ein Label an einem Objekt angebracht oder verändert werden soll, das bereits (Zugriffs-)Beschränkungen unterliegt.[1] Der Systemadministrator kann so eine Situation mit Hilfe der folgenden Kommandos überwinden:

[1] Andere Vorbedingungen führen natürlich zu anderen Fehlern. Zum Beispiel wenn das Objekt nicht dem Nutzer gehört, der das Label ändern möchte, das Objekt vielleicht gar nicht existiert oder es sich um ein nur lesbares Objekt handelt. Oder eine verbindliche Richtlinie erlaubt dem

```
# setfmac biba/high test
Permission denied
# setpmac biba/low setfmac biba/high test
# getfmac test
test: biba/high
```

Wie wir hier sehen, kann `setpmac` verwendet werden, um die vorhandene Einstellungen zu umgehen, indem dem gestarteten Prozeß ein anderes, valides Label zugeordnet wird. Das Werkzeug `getpmac` wird normalerweise auf gerade laufende Prozesse angewendet. Ähnlich sendmail: Als Argument wird statt eines Kommandos eine eine Prozeß-ID übergeben, es verbirgt sich doch dieselbe Logik dahinter. Wenn ein Nutzer versucht, eine Datei zu verändern, auf die er keinen Zugriff hat, entsprechend der Regeln eines geladenen Richtlinienmoduls, wird der Fehler Operation not permitted durch die Funktion `mac_set_link` angezeigt.

16.4.1.1. Übliche Typen von Labeln

Wenn man die Module mac_biba(4), mac_mls(4) und mac_lomac(4) verwendet, hat man die Möglichkeit, einfache Label zu vergeben. Diese nennen sich `high`, `low` und `equal`. Es folgt eine kurze Beschreibung, was diese Labels bedeuten:

- Das Label `low` ist definitionsgemäß das niedrigeste Label, das einem Objekt oder Subjekt verliehen werden kann. Wird es gesetzt, kann die entsprechende Entität nicht mehr auf Entitäten zugreifen, die das Label `high` tragen.
- Das Label `equal` wird Entitäten verliehen, die von der Richtlinie ausgenommen sein sollen.
- Das Label `high` verleiht einer Entität die höchstmögliche Einstellung.

Unter Beachtung jedes einzelnen Richtlinienmoduls moduliert und beschränkt jede dieser Einstellungen den Informationsfluß unterschiedlich. Genaue Erklärungen zu den Charakteristika der einfachen Labels in den verschiedenen Modulen finden sich im entsprechenden Unterabschnitt dieses Kapitels oder in den Manpages.

16.4.1.1.1. Fortgeschrittene Label-Konfiguration

Numerische klassifizierte Labels werden verwendet in der Form `Klasse:Verbund+Verbund`. Demnach ist das Label

```
biba/10:2+3+6(5:2+3-15:2+3+4+5+6)
```

folgendermaßen zu lesen:

„Biba Policy Label“/„effektive Klasse 10“ :„Verbund 2,3 und 6“: („Low-Klasse 5:...“- „High-Klasse 15:...“)

In diesem Beispiel ist die erstgenannte Klasse als „effektive Klasse“ zu bezeichnen. Ihr werden die „effektiven Verbünde“ zugeordnet. Die zweite Klasse ist die „Low“-Klasse und die letzte die „high“-Klasse. Die allermeisten Konfigurationen kommen ohne die Verwendungen von solchen Klassen aus, nichtsdestotrotz kann man sie für erweiterte Konfigurationen verwenden.

Sobald sie auf *Systemsubjekte* angewendet werden, haben diese eine gegenwärtige Klasse/Verbund- Konfiguration und diese muß im definierten Rahmen gegebenenfalls angepaßt (erhöht oder gesenkt) werden. Im Gegensatz dazu haben *Systemobjekte* alle eingestellten (effektive, High- und Low-Klasse) gleichzeitig. Dies ist notwendig, damit auf Sie von den *Systemsubjekten* in den verschiedenen Klassen gleichzeitig zugegriffen werden kann.

Die Klasse und und die Verbünde in einem Subjekt-Objekt-Paar werden zum Erstellen einer sogenannten Dominanz-Relation verwendet, in welcher entweder das Subjekt das Objekt, das Objekt das Subjekt, keines das andere dominiert oder sich beide gegenseitig dominieren. Der Fall, dass sich beide dominieren, tritt dann ein, wenn die beiden Labels gleich sind. Wegen der Natur des Informationsflusses in Biba kann man einem Nutzer Rechte für einen Reihe von Abteilungen zuordnen, die zum Beispiel mit entsprechenden Projekten korrespondieren. Genauso

Prozeß die Veränderung des Labels nicht, weil die Eigenschaften der Datei, die Eigenschaften des Prozesses oder der Inhalt des neuen Labels nicht akzeptiert werden. Beispiel: Ein Anwender mit geringer Vertraulichkeit versucht, das Label einer Datei mit hoher Vertraulichkeit zu ändern. Oder er versucht, eine Datei mit geringer Vertraulichkeit zu einer Datei mit hoher Vertraulichkeit zu machen.

können aber auch Objekten mehrere Abteilungen zugeordnet sein. Die Nutzer müssen eventuell ihre gegenwärtigen Rechte mithilfe von `su` or `setpmac` anpassen um auf Objekte in einer Abteilung zuzugreifen, zu der sie laut ihrer effektiven Klasse nicht berechtigt sind.

16.4.1.2. Nutzer- und Label-Einstellungen

Nutzer selbst brauchen Labels damit ihre Dateien und Prozesse korrekt mit der Sicherheitsrichtlinie zusammenarbeitet, die für das System definiert wurde. Diese werden in der Datei `login.conf` durch die Verwendung von Login- Klassen zugeordnet. Jedes Richtlinienmodul, das Label verwendet, arbeitet mit diesen Login-Klassen.

Beispielhaft wird der folgende Eintrag, der für jede Richtlinie eine Einstellung enthält, gezeigt:

```
default:\
:copyright=/etc/COPYRIGHT:\
:welcome=/etc/motd:\
:setenv=MAIL=/var/mail/$,BLOCKSIZE=K:\
:path=~/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/usr/local/bin:\
:manpath=/usr/share/man /usr/local/man:\
:nologin=/usr/sbin/nologin:\
:cputime=1h30m:\
:datasize=8M:\
:vmemoryuse=100M:\
:stacksize=2M:\
:memorylocked=4M:\
:memoryuse=8M:\
:filesize=8M:\
:coredumpsize=8M:\
:openfiles=24:\
:maxproc=32:\
:priority=0:\
:requirehome:\
:passwordtime=91d:\
:umask=022:\
:ignoretime@:\
:label=partition/13,mls/5,biba/10(5-15),lomac/10[2]:
```

Die Label-Option in der letzten Zeile legt fest, welches Standard-Label für einen Nutzer erzwungen wird. Nutzern darf niemals gestattet werden, diese Werte selbst zu verändern, demnach haben Nutzer in dieser Beziehung auch keine Wahlfreiheit. In einer richtigen Konfiguration jedoch wird kein Administrator alle Richtlinienmodule aktivieren wollen. Es wird an dieser Stelle ausdrücklich empfohlen, dieses Kapitel zu Ende zu lesen, bevor irgendein Teil dieser Konfiguration ausprobiert wird.

Anmerkung

Nutzer können ihr eigenes Label nach dem Loginvorgang durchaus ändern. Jedoch kann diese Änderung nur unter den Auflagen der gerade gültigen Richtlinie geschehen. Im Beispiel oben wird für die Biba-Richtlinie eine minimale Prozeßintegrität von 5, eine maximale von 15 angegeben, aber die Voreinstellung des tatsächlichen Labels ist 10. Der Nutzerprozeß läuft also mit einer Integrität von 10 bis das Label verändert wird, zum Beispiel durch eine Anwendung des Kommandos `setpmac`, welches jedoch auf den Bereich eingeschränkt wird, der zum Zeitpunkt des Logins angegeben wurde, in diesem Fall von 5 bis 15.

Nach einer Änderung der Datei `login.conf` muß in jedem Fall die Befähigungsdatenbank mit dem Kommando `cap_mkdb` neu erstellt werden - und das gilt für alle im weiteren Verlauf gezeigten Beispiele und Diskussionspunkte.

Es ist nützlich anzumerken, dass viele Einsatzorte eine große Anzahl von Nutzern haben, die wiederum viele verschiedenen Nutzerklassen angehören sollen. Hier ist eine Menge Planungsarbeit notwendig, da die Verwaltung sehr unübersichtlich und schwierig ist.

16.4.1.3. Netzwerkschnittstellen und die zugehörigen Label

Labels können auch, wenn man sie an Netzwerkschittstellen vergibt, helfen, den Datenfluß durch das Netzwerk zu kontrollieren. Das funktioniert in allen Fällen genau so wie mit Objekten. Nutzer, die in der Biba-Richtlinie das Label `high` tragen, dürfen nicht auf Schnittstellen zugreifen, die `low` markiert sind usw.

Die Option `maclabel` wird via `ifconfig` übergeben. Zum Beispiel

```
# ifconfig bge0 maclabel biba/equal
```

belegt die Schnittstelle bge(4) mit dem MAC Label `biba/equal` . Wenn eine komplexe Einstellung wie `biba/high(low-high)` verwendet wird, muß das gesamte Label in Anführungszeichen geschrieben werden, da sonst eine Fehlermeldung zurückgegeben wird.

Jedes Richtlinienmodul, das die Vergabe von Labels unterstützt, stellt einen Parameter bereit, mit dem das MAC Label für Netzwerkschnittstellen deaktiviert werden kann. Das Label der Netzwerkschnittstelle auf `equal` zu setzen, führt zum selben Ergebnis. Beachten Sie die Ausgabe von `sysctl` , die Manpages der verschiedenen Richtlinien oder eben die Informationen, die im weiteren Verlauf dieses Kapitels angeboten werden, um mehr zu diesen Parametern zu erfahren.

16.4.2. Single- oder Multilabel?

Als Standardeinstellung verwendet das System die Option `single label`. Was bedeutet das für den Administrator? Es gibt einige Unterschiede zwischen `single label` und `multilabel`. In ihrer ureigenen Weise bieten beide Vor- und Nachteile bezogen auf die Flexibilität bei der Modellierung der Systemsicherheit.

Die Option `single label` gibt jedem Subjekt oder Objekt genau ein einziges Label, zum Beispiel `biba/high` . Mit dieser Option hat man einen geringeren Verwaltungsaufwand, aber die Flexibilität beim Einsatzes von Richtlinien ist ebenso gering. Viele Administratoren wählen daher auch die Option `multilabel` im Sicherheitsmodell, wenn die Umstände es erfordern.

Die Option `multilabel` gestattet, jedem einzelnen Subjekt oder Objekt seine eigenen unabhängigen Label zu zuzuordnen. Die Optionen `multilabel` und `singlelabel` betreffen jedoch nur die Richtlinien, die Labels als Leistungsmerkmal verwenden, einschließlich der Richtlinien Biba, Lomac, MLS und SEBSD.

Wenn Richtlinien benutzt werden sollen, die ohne Labels auskommen, wird die Option `multilabel` nicht benötigt. Dies betrifft die Richtlinien `seeotheruids`, `portacl` und `partition`.

Man sollte sich dessen bewußt sein, dass die Verwendung der Option `multilabel` auf einer Partition und die Erstellung eines Sicherheitsmodells auf der Basis der FreeBSD `multilevel` Funktionalität einen hohen Verwaltungsaufwand bedeutet, da alles im Dateisystem ein Label bekommt. Jedes Verzeichnis, jede Datei und genauso jede Schnittstelle.

Das folgende Kommando aktiviert `multilabel` für ein Dateisystem. Dies funktioniert nur im Einzelbenutzermodus:

```
# tunefs -l enable /
```

In einer Swap-Partition wird dies nicht benötigt.

Anmerkung

Falls Sie Probleme beim Setzen der Option `multilabel` auf der Root-Partition bemerken, lesen Sie bitte Abschnitt 16.17, „Fehler im MAC beheben“ dieses Kapitels.

16.5. Planung eines Sicherheitsmodells

Wann immer eine neue Technologie eingepflegt werden soll, ist es wichtig, vorher einen Plan zu erstellen. In den verschiedenen Etappen der Planung sollte der Administrator nie das „Große Ganze“ aus den Augen verlieren und mindestens die folgenden Punkte beachten:

- Die Anforderungen
- Die Ziele

Wenn Sie MAC verwenden möchten, sind das im Besonderen folgende Punkte:

- Wie werden Informationen und Ressourcen auf den Zielsystemen klassifiziert?
- Welche Arten von Informationen bzw. Ressourcen sollen im Zugang beschränkt sein und welche Art Einschränkung soll verwendet werden?
- Welche(s) MAC Modul(e) wählt man, um sein Ziel zu erreichen?

Es ist immer möglich, die Einstellungen des Systems und der Systemressourcen im Nachhinein zu „optimieren“. Es ist aber wirklich lästig, das gesamte Dateisystem zu durchsuchen, um Dateien oder Benutzerkonten zu reparieren. Eine gute Planung hilft dem Administrator, sich einer sorgenfreien und effizienten Umsetzung eines Sicherheitsmodells zu versichern. Testlauf des Sicherheitsmodells *vor* dem Einsatz in seiner richtigen Arbeitsumgebung ist auf jeden Fall empfehlenswert. Die Idee, ein System mit einer MAC einfach loslaufen zu lassen, ist wie direkt auf einen Fehlschlag hinzuarbeiten.

Jede Umgebung hat ihre eigenen Anforderungen. Ein tiefgreifendes und vollständiges Sicherheitsprofil zu erstellen spart weitere Änderungen, nachdem das System in Betrieb genommen wurde. Also werden die folgenden Abschnitte die verschiedenen Module vorstellen, die den Administratoren zur Verfügung gestellt werden, die Nutzung und Konfiguration der einzelnen Module beschreiben; und in einigen Fällen Einblicke gewähren, für welche Situationen welche Module besonders geeignet sind. Zum Beispiel ein Webserver kann von der Verwendung der mac_biba(4) oder der mac_bsdextended(4) Richtlinie profitieren. In anderen Fällen, an einem Rechner mit nur wenigen lokalen Benutzern, ist die mac_partition(4) die Richtlinie der Wahl.

16.6. Modulkonfiguration

Jedes Modul, das in der MAC enthalten ist, kann entweder direkt in den Kernel eingefügt werden oder als Kernelmodul in der Laufzeit des Systems geladen werden. Empfohlen wird, den Modulnamen in der Datei `/boot/loader.conf` anzufügen, so dass das Modul am Anfang des Bootvorgangs eingebunden wird.

Die folgenden Abschnitte werden verschiedene MAC Module und ihre jeweiligen Vor- und Nachteile vorstellen. Außerdem wird erklärt, wie sie in bestimmte Umgebungen eingearbeitet werden können. Einige Module unterstützen die Verwendung von `Labels`, das heißt Zugriffskontrolle durch hinzufügen einer Kennzeichnung in der Art von „dieses ist erlaubt, jenes aber nicht“. Eine Label-Konfigurationdatei kontrolliert unter anderem, wie auf Dateien zugegriffen oder wie über das Netzwerk kommuniziert werden darf. Im vorangehenden Abschnitt wurde bereits erläutert, wie die Option `multilabel` auf Dateisysteme angewendet wird, um eine Zugriffskontrolle auf einzelne Dateien oder ganze Dateisysteme zu konfigurieren.

Eine `single label` Konfiguration erzwingt ein einzelnes Label für das gesamte System. Daher wird die `tunefs`-Option `multilabel` genannt.

16.7. Das MAC Modul seeotheruids

Modulename: `mac_seeotheruids.ko`

Parameter in der Kernelkonfiguration: `options MAC_SEEOTHERUIDS`

Bootparameter: `mac_seeotheruids_load="YES"`

Das Modul mac_seeotheruids(4) erweitert die `sysctl`-Variablen `security.bsd.see_other_uids` und `security.bsd.see_other_gids`. Diese Optionen benötigen keine im Vorhinein zu setzenden Labels und können leicht durchschaubar mit den anderen MAC-Modulen zusammenarbeiten.

Nachdem das Modul geladen wurde, können die folgenden `sysctl` Variablen verwendet werden.

- `security.mac.seeotheruids.enabled` dient zur Aktivierung des Moduls, zunächst mit den Standardeinstellungen. Diese verhindern, dass Nutzer Prozesse und Sockets sehen können, die ihnen nicht selbst gehöen.
- `security.mac.seeotheruids.specificgid_enabled` kann eine spezifizierte Nutzergruppe von dieser Richtlinie ausnehmen. Die entsprechende Gruppe muß an den Parameter `security.mac.seeotheruids.specificgid=XXX` übergeben werden, wobei *XXX* die ID der Gruppe ist, die von der Richtlinie ausgenommen werden soll.
- `security.mac.seeotheruids.primarygroup_enabled` kann verwendet werden, um eine spezifische, *primäre* Nutzergruppe von der Richtlinie auszuschliessen. Dieser Parameter und `security.mac.seeotheruids.specificgid_enabled` schließen einander aus.

16.8. Das MAC Modul bsdextended

Modulname: `mac_bsdextended.ko`

Parameter in der Kernelkonfiguration: `options MAC_BSDEXTENDED`

Bootparameter: `mac_bsdextended_load="YES"`

Das Modul mac_bsdextended(4) erstellt eine Firewall für das Dateisystem und ist eine Erweiterung des sonst üblichen Rechtemodells. Es erlaubt einem Administrator einen Regelsatz zum Schutz von Dateien, Werkzeugen und Verzeichnissen in der Dateisystemhierarchie zu erstellen, der einer Firewall ähnelt. Sobald auf ein Objekt im Dateisystem zugegriffen werden soll, wird eine Liste von Regel abgearbeitet, bis eine passende Regel gefunden wird oder die Liste zu Ende ist. Das Verhalten kann durch die Änderung des sysctl(8) Parameters `security.mac.bsdextended.firstmatch_enabled` eingestellt werden. Ähnlich wie bei den anderen Firewallmodulen in FreeBSD wird eine Datei erstellt, welche die Zugriffsregeln enthält. Diese wird beim Systemstart durch eine Variable in rc.conf(5) eingebunden.

Der Regelsatz kann mit dem Programm ugidfw(8) eingepflegt werden, welches eine Syntax bereitstellt, die der von ipfw(8) gleicht. Weitere Werkzeuge können auch selbst erstellt werden, indem die Funktionen der Bibliothek libugidfw(3) verwendet werden.

Bei der Arbeit mit diesem Modul ist äußerste Vorsicht geboten - falscher Gebrauch kann den Zugriff auf Teile des Dateisystems komplett unterbinden.

16.8.1. Beispiele

Nachdem das Modul mac_bsdextended(4) erfolgreich geladen wurde, zeigt das folgende Kommando die gegenwärtig aktiven Regeln an:

```
# ugidfw list  0 slots, 0 rules
```

Wie erwartet, sind keine Regeln definiert. Das bedeutet, das auf alle Teile des Dateisystems zugegriffen werden kann. Um eine Regel zu definieren, die jeden Zugriff durch Nutzer blockiert und nur die Rechte von `root` unangetastet läßt, muß lediglich dieses Kommando ausgeführt werden:

```
# ugidfw add subject not uid root new object not uid root mode n
```

Das ist allerdings keine gute Idee, da nun allen Nutzern der Zugriff auf selbst die einfachsten Programme wie `ls` untersagt wird. Angemessener wäre etwas wie:

```
# ugidfw set 2 subject uid user1 object uid user2 mode n
# ugidfw set 3 subject uid user1 object gid user2 mode n
```

Diese Befehle bewirken, dass `user1` keinen Zugriff mehr auf Dateien und Programme hat, die *`user2`* gehören. Dies schließt das Auslesen von Verzeichniseinträgen ein.

Anstelle `uid user1` könnte auch `not uid` *`user2`* als Parameter übergeben werden. Dies würde diesselben Einschränkungen für alle Nutzer bewirken anstatt nur einen einzigen.

Anmerkung

`root` ist von diesen Einstellungen nicht betroffen.

Dies sollte als Überblick ausreichen, um zu verstehen, wie das Modul mac_bsdextended(4) helfen kann, das Dateisystem abzuschotten. Weitere Informationen bieten die Manpages mac_bsdextended(4) und ugidfw(8).

16.9. Das MAC Modul ifoff

Modulname: `mac_ifoff.ko`

Parameter für die Kernelkonfiguration: `options MAC_IFOFF`

Bootparameter: `mac_ifoff_load="YES"`

Das Modul mac_ifoff(4) ist einzig dazu da, Netzwerkschnittstellen im laufenden Betrieb zu deaktivieren oder zu verhindern, das Netzwerkschnittstellen während der Bootphase gestartet werden. Dieses Modul benötigt für seinen Betrieb weder Labels, die auf dem System eingerichtet werden müssen, noch hat es Abhängigkeiten zu anderen MAC Modulen.

Der größte Teil der Kontrolle geschieht über die im folgenden aufgelisteten `sysctl` -Parameter:

- `security.mac.ifoff.lo_enabled` schaltet den gesamten Netzwerkverkehr auf der Loopback-Schnittstelle lo(4) an bzw. aus.
- `security.mac.ifoff.bpfrecv_enabled` macht das Gleiche für den Berkeley Paket Filter bpf(4).
- `security.mac.ifoff.other_enabled` schaltet den Verkehr für alle anderen Netzwerkschnittstellen.

Die wahrscheinlich häufigste Nutzung von mac_ifoff(4) ist die Überwachung des Netzwerks in einer Umgebung, in der kein Netzwerkverkehr während des Bootvorgangs erlaubt werden soll. Eine andere mögliche Anwendung wäre ein Script, das mit Hilfe von security/aide automatisch alle Schnittstellen blockiert, sobald Dateien in geschützten Verzeichnissen angelegt oder verändert werden.

16.10. Das MAC Modul portacl

Modulname: `mac_portacl.ko`

Parameter für die Kernelkonfiguration: `options MAC_PORTACL`

Bootparameter: `mac_portacl_load="YES"`

Mit Hilfe des Moduls mac_portacl(4) können die Anbindungen an die lokalen TCP und UDP Ports durch eine Vielzahl von `sysctl` Variablen beschränkt werden. Genauer gesagt ermöglicht mac_portacl(4) Nutzern ohne `root`-Rechten den Zugriff auf zu bestimmende privilegierte Ports, also denen innerhalb der ersten 1024.

Sobald das Modul geladen wurde, ist die Richtlinie für alle Sockets verfügbar. Die folgenden Variablen können für die Konfiguration verwendet werden:

- `security.mac.portacl.enabled` schaltet die Anwendung der Richtlinie ein oder aus.
- `security.mac.portacl.port_high` gibt den höchsten Port an, der von der Richtlinie mac_portacl(4) betroffen sein soll.
- `security.mac.portacl.suser_exempt` nimmt, wenn es einen Wert ungleich Null zugewiesen bekommt, `root` von der Richtlinie aus.
- `security.mac.portacl.rules` enthält als Wert die eigentliche `mac_portacl` Richtlinie.

Die eigentliche Konfiguration der `mac_portacl` Richtlinie wird der `sysctl`-Variablen `security.mac.portacl.rules` als Zeichenkette der Form `rule[,rule,...]` übergeben. Jede einzelne Regel hat die Form `idtype:id:protocol:port`. Der Parameter *`idtype`* ist entweder `uid` oder `gid` und wird verwendet, um den Parameter *`id`* als Nutzer-ID oder Gruppen-ID zu kennzeichnen. Der Parameter *`protocol`* gibt an, ob die Regel ür TCP oder UDP gelten soll (indem man den Wert auf `tcp` oder `udp` setzt). Und der letzte Parameter, *`port`*, enthält die Nummer des Ports, auf den der angegebene Nutzer bzw. die angegebene Gruppe Zugriff erhalten soll.

Anmerkung

Da der Regelsatz direkt vom Kernel ausgewertet wird, können nur Zahlenwerte übergeben werden. Das heißt, Namen von Nutzern, Gruppen oder Dienstnamen aus der Datei `/etc/services` funktionieren nicht.

Auf UNIX®-artigen Betriebssystemen sind die Ports kleiner 1024 privilegierten Prozessen vorbehalten, müssen also mit als/von `root` gestartet werden und weiterhin laufen. Damit mac_portacl(4) die Vergabe von Ports kleiner als 1024 an nicht privilegierte Prozesse übernehmen kann, muß die UNIX® Standardeinstellung deaktiviert werden. Dazu ändert man die sysctl(8) Variablen `net.inet.ip.portrange.reservedlow` und `net.inet.ip.portrange.reservedhigh` auf den Wert „0“.

Weiterführende Informationen entnehmen Sie bitte den unten aufgeführten Beispielen oder der Man-Page mac_portacl(4)!

16.10.1. Beispiele

Die folgenden Beispiele sollten ein wenig Licht in die obige Diskussion bringen:

```
# sysctl security.mac.portacl.port_high=1023
# sysctl net.inet.ip.portrange.reservedlow=0 net.inet.ip.portrange.reservedhigh=0
```

Zunächst bestimmen wir, dass mac_portacl(4) für alle privilegierten Ports gelten soll und deaktivieren die normale UNIX®-Beschränkung.

```
# sysctl security.mac.portacl.suser_exempt=1
```

Da `root` von dieser Richtlinie nicht beeinträchtigt werden soll, setzen wir hier `security.mac.portacl.suser_exempt` auf einen Wert ungleich Null. Das Modul mac_portacl(4) ist nun so eingerichtet, wie es UNIX®-artige Betriebssysteme normal ebenfalls tun.

```
# sysctl security.mac.portacl.rules=uid:80:tcp:80
```

Nun erlauben wir dem Nutzer mit der UID 80, normalerweise dem Nutzer `www`, den Port 80 zu verwenden. Dadurch kann der Nutzer `www` einen Webserver betreiben, ohne dafür mit `root`-Privilegien ausgestattet zu sein.

```
# sysctl security.mac.portacl.rules=uid:1001:tcp:110,uid:1001:tcp:995
```

Hier wird dem Nutzer mit der UID 1001 erlaubt, die TCP Ports 110 („pop3“) und 995 („pop3s“) zu verwenden. Dadurch kann dieser Nutzer einen Server starten, der Verbindungen an diesen beiden Ports annehmen kann.

16.11. Das MAC Modul partition

Modulname: `mac_partition.ko`

Parameter für die Kernelkonfiguration: `options MAC_PARTITION`

Bootparameter `mac_partition_load="YES"`

Die Richtlinie mac_partition(4) setzt Prozesse in spezielle „Partitionen“, entsprechend dem zugewiesenen MAC Label. Man kann sich das vorstellen wie eine spezielle Art jail(8), auch wenn das noch kein wirklich guter Vergleich ist.

Es wird empfohlen, dieses Modul durch einen Eintrag in loader.conf(5) zu aktivieren, so dass die Richtlinie während des Bootvorganges eingebunden wird.

Der Großteil der Konfiguration geschieht mit dem Kommando setpmac(8), wie gleich erklärt wird. Außerdem gibt es folgenden `sysctl` Parameter für diese Richtlinie.

- `security.mac.partition.enabled` erzwingt die Verwendung von MAC Prozeß-Partitionen.

Sobald diese Richtlinie aktiv ist, sehen Nutzer nur noch ihre eigenen Prozesse, und alle anderen Prozesse, die ebenfalls derselben Prozeß-Partition zugeordnet sind. Sie können jedoch nicht auf Prozesse oder Werkzeuge außerhalb des Anwendungsbereich dieser Partition zugreifen. Das bedeutet unter anderem, das ein Nutzer, der einer Klasse `insecure` zugeordnet ist, nicht auf das Kommando `top` zugreifen kann - wie auch auf viele anderen Befehle, die einen eigenen Prozeß erzeugen.

Um einen Befehl einer Prozeß-Partition zuzuordnen, muß dieser durch das Kommando `setpmac` mit einem Label versehen werden:

```
# setpmac partition/13 top
```

Diese Zeile fügt das Kommando `top` dem Labelsatz für Nutzer der Klasse `insecure` hinzu, sofern die Partition 13 mit der Klasse `insecure` übereinstimmt. Beachten Sie, dass alle Prozesse, die von Nutzern dieser Klasse erzeugt werden, das Label `partition/13` erhalten, und dieses auch nicht durch den Nutzer geändert werden kann.

16.11.1. Beispiele

Der folgende Befehl listet die vergebenen Label für Prozeß-Partitionen und die laufenden Prozesse auf.

```
# ps Zax
```

Das nächste Kommando liefert das Label der Prozeß-Partition eines anderen Nutzers `trhodes` und dessen gegenwärtig laufenden Prozesse zurück.

```
# ps -ZU trhodes
```

Anmerkung

Jeder Nutzer kann die Prozesse in der Prozeß-Partition von `root` betrachten, solange nicht die Richtlinie mac_seeotheruids(4) geladen wurde.

Eine ausgefeilte Umsetzung dieser Richtlinie deaktiviert alle Dienste in `/etc/rc.conf` und startet diese dann später durch ein Skript, das jedem Dienst das passende Label zuordnet.

Anmerkung

Die folgenden Richtlinien verwenden Zahlenwerte anstatt der drei Standardlabels. Diese Optionen, und ihre Grenzen, werden in den zugehörigen Manpages genauer erklärt.

16.12. Das MAC Modul Multi-Level Security

Modulname: `mac_mls.ko`

Parameter für die Kernelkonfiguration: `options MAC_MLS`

Bootparameter: `mac_mls_load="YES"`

Die Richtlinie mac_mls(4) kontrolliert die Zugriffe zwischen Subjekten und Objekten, indem sie den Informationsfluß strengen Regeln unterwirft.

In MLS Umgebungen wird jedem Subjekt oder Objekt ein „Freigabe“-Level zugeordnet, und diese werden wiederum zu einzelnen Verbünden zusammengefaßt. Da diese Freigabe- oder Anfälligkeits-Level Zahlen größer 6000 erreichen können, ist es für jeden Systemadministrator eine undankbare Aufgabe, jede Entität von Grund auf zu konfigurieren. Zum Glück gibt es 3 „instant“ Labels, die in der Richtlinie zur Anwendung bereit stehen.

Diese drei Labels heißen `mls/low`, `mls/equal` und `mls/high`. Da sie in den Manpages mac_mls(4) ausführlich beschrieben werden, gibt es hier nur einen kurzen Abriß:

- Das Label `mls/low` ist eine niedrige Einstellung, die von allen anderen dominiert werden darf. Alles, was mit `mls/low` versehen wird, hat ein niedriges Freigabe-Level und darf auf keine Informationen zugreifen, denen ein höheres Freigabe-Level zugeordnet wurde. Einem Objekt mit diesem Label kann außerdem keine Information durch ein Objekt höherer Freigabe übergeben werden, es kann also auch nicht durch solche Objekte editiert oder überschrieben werden.
- Das Label `mls/equal` wird an Objekte vergeben, die von dieser Richtlinie ausgenommen werden sollen.
- Das Label `mls/high` verkörpert das höchstmögliche Freigabe-Level. Objekte, denen dieses Label zugeordnet wird, dominieren alle anderen Objekte des Systems. Trotzdem können sie Objekten mit einem niedrigeren Freigabe-Level keine Informationen zuspielen.

MLS bietet:

- Eine hierarchische Sicherheitsschicht und Zuordnung nichthierarchischer Kategorien;
- Feste Regeln: kein „Read-Up“, kein „Write-Down“ (ein Subjekt kann nur Objekte gleicher oder *niedrigerer* Stufe lesen, und es kann nur Objekte gleicher oder *höherer* Stufe schreiben);
- Geheimhaltung (indem unangemessene Offenlegung von Daten verhindert wird);
- Eine Basis zum Entwerfen von Systemen, die Daten verschiedener Vertraulichkeitsebenen gleichzeitig handhaben sollen (ohne das geheime und vertrauliche Informationen untereinander ausgetauscht werden können).

Nachfolgend werden die `sysctl`-Variablen vorgestellt, die für die Einrichtung spezieller Dienste und Schnittstellen vorhanden sind.

- `security.mac.mls.enabled` schaltet die Richtlinie MLS ein (oder aus).

- `security.mac.mls.ptys_equal` sorgt dafür, dass während der Initialisierung alle pty(4)-Geräte als `mls/equal` gekennzeichnet werden.
- `security.mac.mls.revocation_enabled` sorgt dafür, dass die Zugriffsrechte von Objekten wieder zurückgesetzt werden, nachdem deren Label vorübergehend auf ein niedrigeres Freigabe-Level geändert wurde.
- `security.mac.mls.max_compartments` gibt die maximale Anzahl von Verbünden an. Im Prinzip ist es die höchste Nummer eines Verbundes auf dem System.

Um die Labels der MLS Richtlinie zu bearbeiten verwendet man setfmac(8). Um ein Objekt zu kennzeichnen, benutzen Sie folgendes Kommando:

```
# setfmac mls/5 test
```

Um das MLS-Label der Datei `test` auszulesen, verwenden Sie dieses Kommando:

```
# getfmac test
```

Dies ist eine Zusammenstellung der Merkmale von `test`. Ein anderer Ansatz ist, für diese Richtlinie eine Konfigurationsdatei in `/etc` abzulegen, die alle Informationen enthält und mit der dann das Kommando `setfmac` gefüttert wird. Diese Vorgehensweise wird erklärt, nachdem alle Richtlinien vorgestellt wurden.

16.12.1. Verbindlicher Vertraulichkeit in der Planungsphase

Mit dem Richtlinienmodul `Multi-Level Security` bereitet sich ein Administrator darauf vor, den Fluß vertraulicher Informationen zu kontrollieren. Beim Starten der Richtlinie ist immer `mls/low` voreingestellt - alles kann auf alles zugreifen. Der Administrator ändert dies während der eigentlichen Konfiguration, indem er die Vertraulichkeit bestimmter Objekte erhöht.

Jenseits der drei Grundeinstellungen des Labels kann der Administrator einzelne Nutzer oder Nutzergruppen nach Bedarf zusammenschließen und den Informationsaustausch zwischen diesen gestatten oder unterbinden. Es ist sicher eine Vereinfachung, die Freigabe-Level mit Begriffen wie `vertraulich`, `geheim` oder `streng geheim` zu bezeichnen. Einige Administratoren erstellen einfach verschiedene Gruppen auf der Ebene von gegenwärtigen Projekten. Ungeachtet der Herangehensweise bei der Klassifizierung muß ein gut durchdachter Plan existieren, bevor eine derart einengende Richtlinie umgesetzt wird.

Exemplarisch für die Anwendung dieses Moduls bzw. dieser Richtlinie seien angeführt:

- Ein E-Commerce Webserver
- Ein Dateiserver, der vertrauliche Informationen einer Firma oder eines Konzerns speichert
- Umgebungen in Finanzeinrichtungen

Der unsinnigste Einsatzort für diese Richtlinie wäre ein Arbeitsplatzrechner mit nur zwei oder drei Benutzern.

16.13. Das MAC Modul Biba

Modulname: `mac_biba.ko`

Parameter für die Kernelkonfiguration: `options MAC_BIBA`

Bootparameter: `mac_biba_load="YES"`

Das Modul mac_biba(4) lädt die MAC Biba Richtlinie. Diese ähnelt stark der MLS Richtlinie, nur das die Regeln für den Informationsfluß ein wenig vertauscht sind. Es wird in diesem Fall der absteigende Fluß sicherheitskritischer Information geregelt, während die MLS Richtlinie den aufsteigenden Fluß regelt. In gewissen Sinne treffen dieses und das vorangegangene Unterkapitel also auf beide Richtlinien zu.

In einer Biba-Umgebung wird jedem Subjekt und jedem Objekt ein „Integritäts“-Label zugeordnet. Diese Labels sind in hierarchischen Klassen und nicht-hierarchischen Komponenten geordnet. Je höher die Klasse, um so höher die Integrität.

Die unterstützten Labels heißen `biba/low` , `biba/equal` und `biba/high` . Sie werden im Folgenden erklärt:

- `biba/low` ist die niedrigste Stufe der Integrität, die einem Objekt verliehen werden kann. Wenn sie einem Objekt oder Subjekt zugeordnet wird, kann dieses auf Objekte oder Subjekte, die biba/high markiert wurden, zwar lesend zugreifen, nicht jedoch schreibend.
- Das Label `biba/equal` ist, wie der aufmerksame Leser sicherlich schon ahnt, für die Ausnahmen dieser Richtlinie gedacht und sollte nur diesen Ausnahmen entsprechenden Objekten verliehen werden.
- `biba/high` markierte Subjekte und Objekte können Objekte niedrigerer Stufe schreiben , nicht jedoch lesen. Es wird empfohlen, dass dieses Label an Objekte vergeben wird, die sich auf Integrität des gesamten Systems auswirken.

Biba stellt bereit:

- Hierarchische Integritätsstufen mit einem Satz nichthierarchischer Integritätskategorien;
- Festgeschriebene Regeln: kein „Write-Up“, kein „Read-Down“ (der Gegensatz zu MLS - ein Subjekt erhält schreibenden Zugriff auf Objekte gleicher oder geringerer Stufe, aber nicht bei höherer, und lesenden Zugriff bei gleicher Stufe oder höerer, aber nicht bei niedrigerer);
- Integrität (es wird die Echtheit der Daten gewährleistet, indem unangemessene Veränderungen verhindert werden);
- Eine Abstufung der Gewährleistung (im Gegensatz zu MLS, bei der eine Abstufung der Vertraulichkeit vorgenommen wird).

Folgende `sysctl` Parameter werden zur Nutzung der Biba-Richtlinie angeboten:

- `security.mac.biba.enabled` zum Aktivieren/Deaktivieren der Richtlinie auf dem Zielsystem.
- `security.mac.biba.ptys_equal` wird verwendet, um die Biba-Richtlinie auf der pty(4)-Schnittstelle zu deaktivieren.
- `security.mac.biba.revocation_enabled` erzwingt das Zurücksetzen des Labels, falls dieses zeitweise geändert wurde um ein Subjekt zu dominieren.

Um Einstellungen der Biba Richtlinie für Systemobjekte zu verändern werden die Befehle `setfmac` und `getfmac` verwendet:

```
# setfmac biba/low test
# getfmac test
test: biba/low
```

16.13.1. Verbindliche Integrität in der Planungsphase

Integrität garantiert, im Unterschied zu Sensitivität, dass Informationen nur durch vertraute Parteien verändert werden können. Dies schließt Informationen ein, die zwischen Subjekten ausgetauscht werden, zwischen Objekt, oder auch zwischen den beiden. Durch Integrität wird gesichert, das Nutzer nur Informationen verändern, oder gar nur lesen können, die sie explizit benötigen.

Das Modul mac_biba(4) eröffnet einem Administrator die Möglichkeit zu bestimmen, welche Dateien oder Programme ein Nutzer oder eine Nutzergruppe sehen bzw. aufrufen darf. Gleichzeitig kann er zusichern, dass dieselben Programme und Dateien frei von Bedrohungen sind und das System die Echtheit gewährleistet - für diesen Nutzer oder die Nutzergruppe.

Während der anfänglichen Phase der Planung muß der Administrator vorbereitet sein, Nutzer in Klassen, Stufen und Bereiche einzuteilen. Der Zugriff auf Dateien und insbesondere auch Programme wird verhindert sowohl vor als auch nachdem sie gestartet wurden. Das System selbst erhält als Voreinstellung das Label `biba/high` sobald das Modul aktiviert wird - und es liegt allein am Administrator, die verschiedenen Klassen und Stufen für die einzelnen Nutzer zu konfigurieren. Anstatt mit Freigaben zu arbeiten, wie weiter oben gezeigt wurde, könnte man auch Überbegriffe für Projekte oder Systemkomponenten entwerfen. Zum Beispiel, ausschließlich Entwicklern den Vollzugriff auf Quellcode, Compiler und Entwicklungswerkzeuge gewähren, während man andere Nutzer in Kategorien wie Tester, Designer oder einfach nur „allgemeiner Nutzer“ zusammenfaßt, die für diese Bereiche lediglich lesenden Zugriff erhalten sollen.

Mit seinem ursprünglichen Sicherheits-Standpunkt ist ein Subjekt niedrigerer Integrität unfähig, ein Subjekt höherer Integrität zu verändern. Ein Subjekt höherer Integrität kann ein Subjekt niedrigerer Integrität weder beobachten noch lesen. Wenn man ein Label für die niedrigstmögliche Klasse erstellt, kann man diese allen Subjekten verwehren. Einige weitsichtig eingerichtete Umgebungen, die diese Richtlinie verwenden, sind eingeschränkte Webserver, Entwicklungs- oder Test-Rechner oder Quellcode-Sammlungen. Wenig sinnvoll ist diese Richtlinie auf einer Arbeitsstation, oder auf Rechnern die als Router oder Firewall verwendet werden.

16.14. Das MAC Modul LOMAC

Modulname: `mac_lomac.ko`

Parameter für die Kernelkonfiguration: `options MAC_LOMAC`

Bootparameter: `mac_lomac_load="YES"`

Anders als die Biba Richtlinie erlaubt die mac_lomac(4) Richtlinie den Zugriff auf Objekte niedrigerer Integrität nur, nachdem das Integritätslevel gesenkt wurde. Dadurch wird eine Störung derIntegritätsregeln verhindert.

Die MAC Version der „Low-Watermark“ Richtlinie, die nicht mit der älteren lomac(4)-Implementierung verwechselt werden darf, arbeitet fast genauso wie Biba. Anders ist, dass hier „schwebende“ Label verwendet werden, die ein Herunterstufen von Subjekten durch Hilfsverbünde ermöglichen. Dieser zweite Verbund wird in der Form `[auxgrade]` angegeben und sollte in etwa aussehen wie `lomac/10[2]` , wobei die Ziffer zwei (2) hier den Hilfsverbund abbildet.

Die MAC Richtlinie `LOMAC` beruht auf einer durchgängigen Etikettierung aller Systemobjekte mit Integritätslabeln, die Subjekten das Lesen von Objekten niedriger Integrität gestatten und dann das Label des Subjektes herunterstufen - um zukünftige Schreibvorgänge auf Objekte hoher Integrität zu unterbinden. Dies ist die Funktion der Option `[auxgrade]`, die eben vorgestellt wurde. Durch sie erhält diese Richtlinie eine bessere Kompatibilität und die Initialisierung ist weniger aufwändig als bei der Richtlinie Biba.

16.14.1. Beispiele

Wie schon bei den Richtlinien Biba und MLS werden die Befehle `setfmac` und `setpmac` verwendet, um die Labels an den Systemobjekten zu setzen:

```
# setfmac /usr/home/trhodes lomac/high[low]
# getfmac /usr/home/trhodes  lomac/high[low]
```

Beachten Sie, dass hier der Hilfswert auf `low` gesetzt wurde - dieses Leistungsmerkmal ist nur in der MAC `LOMAC` Richtlinie enthalten.

16.15. Beispiel 1: Nagios in einer MAC Jail

Die folgende Demonstration setzt eine sichere Umgebung mithilfe verschiedener MAC Module und sorgfältig konfigurierter Richtlinien um. Es handelt sich jedoch nur um einen Test und sollte nicht als Antwort auf jedes Problem

in Fragen Sicherheit gesehen werden. Eine Richtlinie nur umzusetzen und dann einfach laufen zu lassen, funktioniert nie und kann eine echte Arbeitsumgebung in eine Katastrophe stürzen.

Bevor es losgeht, muß jedes Dateisystem mit der Option `multilabel`, wie weiter oben beschrieben, markiert werden. Dies nicht zu tun, führt zu Fehlern. Außerdem müssen die Ports net-mngt/nagios-plugins, net-mngt/nagios und www/apache22 installiert und konfiguriert sein, so dass sie ordentlich laufen.

16.15.1. Erstellen einer Nutzerklasse `insecure`

Beginnen wir die Prozedur mit dem Hinzufügen einer Nutzerklasse in der Datei `/etc/login.conf`:

```
insecure:\
:copyright=/etc/COPYRIGHT:\
:welcome=/etc/motd:\
:setenv=MAIL=/var/mail/$,BLOCKSIZE=K:\
:path=~/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/usr/local/bin--
:manpath=/usr/share/man /usr/local/man:\
:nologin=/usr/sbin/nologin:\
:cputime=1h30m:\
:datasize=8M:\
:vmemoryuse=100M:\
:stacksize=2M:\
:memorylocked=4M:\
:memoryuse=8M:\
:filesize=8M:\
:coredumpsize=8M:\
:openfiles=24:\
:maxproc=32:\
:priority=0:\
:requirehome:\
:passwordtime=91d:\
:umask=022:\
:ignoretime@:\
:label=biba/10(10-10):
```

Zusätzlich fügen wir beim Standardnutzer folgende Zeile hinzu:

```
:label=biba/high:
```

Anschließend muß die Datenbank neu erstellt werden:

```
# cap_mkdb /etc/login.conf
```

16.15.2. Boot-Konfiguration

Starten Sie den Rechner noch nicht neu. Fügen Sie zunächst noch die folgenden Zeilen in die Datei `/boot/loader.conf` ein, damit die benötigten Module während des Systemstarts geladen werden:

```
mac_biba_load="YES"
mac_seeotheruids_load="YES"
```

16.15.3. Nutzer einrichten

Ordnen Sie den Superuser `root` der Klasse `default` zu:

```
# pw usermod root -L default
```

Alle Nutzerkonten, die weder `root` noch Systemkonten sind, brauchen nun eine Loginklasse, da sie sonst keinen Zugriff auf sonst übliche Befehle erhalten, wie bspw. vi(1). Das folgende `sh` Skript wird diese Aufgabe erledigen:

```
# for x in `awk -F: '($3 >= 1001) && ($3 != 65534) { print $1 }' \
      /etc/passwd`; do pw usermod $x -L default; done;
```

Verschieben Sie die Nutzer `nagios` und `www` in die `insecure` Klasse:

```
# pw usermod nagios -L insecure
```

```
# pw usermod www -L insecure
```

16.15.4. Die Kontextdatei erstellen

Nun muß eine Kontextdatei erstellt werden. Die folgende Beispieldatei soll dazu in `/etc/policy.contexts` gespeichert werden:

```
# This is the default BIBA policy for this system.

# System:
/var/run                        biba/equal
/var/run/*                      biba/equal

/dev                            biba/equal
/dev/*                          biba/equal

/var         biba/equal
/var/spool                      biba/equal
/var/spool/*                    biba/equal

/var/log                        biba/equal
/var/log/*                      biba/equal

/tmp         biba/equal
/tmp/*         biba/equal
/var/tmp          biba/equal
/var/tmp/*        biba/equal

/var/spool/mqueue               biba/equal
/var/spool/clientmqueue      biba/equal

# For Nagios:
/usr/local/etc/nagios
/usr/local/etc/nagios/*         biba/10

/var/spool/nagios               biba/10
/var/spool/nagios/*             biba/10

# For apache
/usr/local/etc/apache           biba/10
/usr/local/etc/apache/*         biba/10
```

Die Richtlinie erzwingt Sicherheit, indem der Informationsfluß Einschränkungen unterworfen wird. In der vorliegenden Konfiguration kann kein Nutzer, weder `root` noch andere, auf Nagios zugreifen. Konfigurationsdateien und die Prozesse, die Teil von Nagios sind, werden durch unsere MAC vollständig abgegrenzt.

Die Kontextdatei kann nun vom System eingelesen werden, indem folgender Befehl ausgeführt wird:

```
# setfmac -ef /etc/policy.contexts /
# setfmac -ef /etc/policy.contexts /
```

Anmerkung

Das obenstehende Dateisystem-Layout kann, je nach Umgebung, sehr unterschiedlich aussehen. Außerdem muß es auf jedem einzelnen Dateisystem ausgeführt werden.

In die Datei `/etc/mac.conf` müssen nun noch diese Änderungen eingetragen werden:

```
default_labels file ?biba
```

```
default_labels ifnet ?biba
default_labels process ?biba
default_labels socket ?biba
```

16.15.5. Netzwerke einbinden

Tragen Sie die folgende Zeile in die Datei `/boot/loader.conf` ein:

```
security.mac.biba.trust_all_interfaces=1
```

Und das Folgende gehört in Datei `rc.conf` zu den Optionen für die Netzwerkkarte. Falls die Netzwerkverbindung(-en) via DHCP konfiguriert werden, muß man dies nach jedem Systemstart eigenhändig nachtragen:

```
maclabel biba/equal
```

16.15.6. Testen der Konfiguration

Versichern Sie sich, dass der Webserver und Nagios nicht automatisch geladen werden und starten Sie den Rechner neu. Prüfen Sie nun, ob `root` wirklich keinen Zugriff auf die Dateien im Konfigurationsverzeichnis von Nagios hat. Wenn `root` den Befehl ls(1) auf `/var/spool/nagios` ausführen kann, ist irgendwas schief gelaufen. Es sollte ein permission denied Fehler ausgegeben werden.

Wenn alles gut aussieht, können Nagios, Apache und Sendmail gestartet werden - allerdings auf eine Weise, die unserer Richtlinie gerecht wird. Zum Beispiel durch die folgenden Kommandos:

```
# cd /etc/mail && make stop && \
setpmac biba/equal make start && setpmac biba/10\(10-10\) apachectl start && \
setpmac biba/10\(10-10\) /usr/local/etc/rc.d/nagios.sh forcestart
```

Versichern Sie sich lieber doppelt, dass alles ordentlich läuft. Wenn nicht, prüfen Sie die Logs und Fehlermeldungen. Verwenden Sie das sysctl(8) Werkzeug um die Sicherheitsrichtlinie sysctl(8) zu deaktivieren und versuchen Sie dann alles noch einmal zu starten.

Anmerkung

Der Superuser kann den Vollzug der Richtlinie schalten und die Konfiguration ohne Furcht verändern. Folgender Befehl stuft eine neu gestartete Shell herunter:

```
# setpmac biba/10 csh
```

Um dies zu vermeiden, werden die Nutzer durch login.conf(5) eingeschränkt. Wenn setpmac(8) einen Befehl außerhalb der definierten Schranken ausführen soll, wird ein Fehler zurückgeliefert. In so einem Fall muß `root` auf `biba/high(high-high)` gesetzt werden.

16.16. Beispiel 2: User Lock Down

Grundlage dieses Beispiels ist ein relativ kleines System zur Datenspeicherung mit weniger als 50 Benutzern. Diese haben die Möglichkeit, sich einzuloggen und dürfen nicht nur Daten speichern, sondern auch auf andere Ressourcen zugreifen.

Die Richtlinien mac_bsdextended(4) und mac_seeotheruids(4) können gleichzeitig eingesetzt werden. Zusammen kann man mit ihnen nicht nur den Zugriff auf Systemobjekte einschränken, sondern auch Nutzerprozesse verstecken.

Beginnen Sie, indem Sie die folgende Zeile in die Datei `/boot/loader.conf` eintragen:

```
mac_seeotheruids_load="YES"
```

Die Richtlinie mac_bsdextended(4) wird durch den anschließenden Eintrag in `/etc/rc.conf` hinzugefügt:

```
ugidfw_enable="YES"
```

Die Standardregeln, welche in `/etc/rc.bsdextended` gespeichert sind, werden zum Systemstart geladen. Sie müssen aber noch angepaßt werden. Da dieser Computer nur Nutzern dienen soll und weitere Dienste gestartet werden, kann alles bis auf die beiden letzten Zeilen auskommentiert werden. Das sorgt dafür dass jeder Nutzer seine eigenen Systemobjekte erhält.

Nun fügen wir alle benötigten Nutzer auf der Maschine hinzu und starten neu. Zum Testen der Einstellungen loggen Sie sich parallel zwei mal mit unterschiedlichen Nutzernamen ein und starten Sie das Kommando `ps aux`. Dort sehen Sie, dass Sie die Prozesse des anderen Nutzers nicht sehen können. Versuchen Sie, ls(1) auf das Heimatverzeichnis eines anderen Nutzers auszuführen. Auch dieser Versuch wird fehlschlagen.

Solange nicht die speziellen `sysctl`-Variablen geändert wurden, hat der Superuser noch vollen Zugriff. Sobald auch diese Einstellungen angepaßt wurden, führen Sie ruhig auch den obigen Test als `root` aus.

Anmerkung

Wenn ein neuer Benutzer hinzugefügt wird, ist für diesen zunächst keine mac_bsdextended(4) Regel im Regelsatz vorhanden. Schnelle Abhilfe schafft hier, einfach das Kernelmodul mit kldunload(8) zu entladen und mit kldload(8) erneut einzubinden.

16.17. Fehler im MAC beheben

Während der Entwicklung des Frameworks haben einige Nutzer auf Probleme hingewiesen. Einige davon werden hier aufgeführt:

16.17.1. Die Option `multilabel` greift nicht auf der /-Partition

Es scheint, dass etwa jedem fünfzigsten Nutzer dieses Problem widerfährt. Und in der Tat - auch wir kennen es aus der Entwicklung. Genauere Untersuchungen dieses „Bugs“ machten uns glauben, dass es sich entweder um einen Fehler in oder eine fehlerhafte Interpretation der Dokumentation handelt. Warum auch immer dieser Fehler auftritt - er kann mit folgender Prozedur behoben werden:

1. Öffnen Sie die Datei `/etc/fstab` und setzen Sie die Rootpartition auf `ro` wie „read-only“.
2. Starten Sie in den Einzelnutzermodus.
3. Rufen Sie `tunefs -l enable` für `/` auf.
4. Starten Sie in den Mehrbenutzermodus.
5. Führen Sie `mount -urw /` aus und ändern Sie anschließend in der Datei `/etc/fstab` die Option `ro` zurück in `rw`. Starten Sie das System noch einmal neu.
6. Achten Sie besonders auf die Ausgabe von `mount` um sich zu versichern, dass die `multilabel` korrekt für das root-Dateisystem gesetzt wurde.

16.17.2. Mit der aktivierten MAC kann ich keinen X11 Server starten

Dies kann durch die Richtlinie `partition` oder einer fehlerhaften Verwendung einer Richtlinie, die mit Labels arbeitet, auftreten. Zum debuggen versuchen Sie folgendes:

1. Schauen Sie sich die Fehlermeldungen genau an. Wenn der Nutzer einer `insecure` Klasse angehört, ist wahrscheinlich die Richtlinie `partition` die Ursache. Versuchen Sie, die Nutzerklasse auf `default` zu stellen und danach die Datenbank mit `cap_mkdb` zu erneuern. Wenn das Problem dadurch nicht gelöst wird, gehen Sie weiter zu Schritt 2.

2. Gehen Sie die Label-Richtlinien Schritt für Schritt nocheinmal durch. Achten Sie darauf, dass für den Nutzer, bei dem das Problem auftritt, für X11 und das Verzeichnis `/dev` alle Einstellungen korrekt sind.

3. Falls all dies nicht helfen sollte, senden Sie die Fehlermeldung und eine Beschreibung ihrer Arbeitsumgebung an die (englisch-sprachige) TrustedBSD Diskussionsliste auf der TrustedBSD Webseite oder an die FreeBSD general questions Mailingliste.

16.17.3. Error: _secure_path(3) cannot stat `.login_conf`

Wenn ich versuche, von `root` zu einem anderen Nutzer des Systems zu wechseln, erhalte ich die Fehlermeldung _secure_path: unable to state .login_conf.

Diese Meldung wird gewöhnlich ausgegeben, wenn der Nutzer ein höhere Label-Einstellung hat als der, dessen Identität man annehmen möchte. Ausführlich: Wenn ein Nutzer `joe` als `biba/low` gelabelt wurde, kann `root`, der `biba/high` als Voreinstellung trägt, das Heimatverzeichnis von `joe` nicht einsehen. Das passiert unabhänig davon, ob `root` vorher mit `su` die Identität von `joe` angenommen hat oder nicht, da das Label sich nicht ändert. Hier haben wir also einen Fall, in dem das Gewährleistungsmodell von Biba verhindert, das der Superuser Objekte einer niedrigeren Integrität betrachten kann.

16.17.4. Der Nutzer `root` ist kaputt!

Im normalen oder sogar im Einzelbenutzermodus wird `root` nicht anerkannt. Das Kommando `whoami` liefert 0 (null) und `su` liefert who are you? zurück. Was geht da vor?

Das kann passieren, wenn eine Label-Richtlinie ausgeschaltet wird - entweder durch sysctl(8) oder wenn das Richtlinienmodul entladen wurde. Wenn eine Richtlinie deaktiviert oder auch nur vorübergehen deaktiviert wird, muß die Befähigungsdatenbank neu konfiguriert werden, d.h. die `label` Option muß entfernt werden. Überprüfen Sie, ob alle `label` Einträge aus der Datei `/etc/login.conf` entfernt wurden und bauen Sie die Datenbank mit `cap_mkdb` neu.

Dieser Fehler kann auch auftreten, wenn eine Richtlinie den Zugriff auf die Datei `master.passwd` einschränkt. Normalerweise passiert das nur, wenn ein Administrator ein Label an diese Datei vergibt, das mit der allgemeingültigen Richtlinie, die das System verwendet, in Konflikt steht. In solchen Fällen werden die Nutzerinformationen vom System ausgelesen und jeder weitere Zugriff wird blockiert, sobald das neue Label greift. Wenn man die Richtlinie via sysctl(8) ausschaltet, sollte es erstmal wieder gehen.

Kapitel 17. Security Event Auditing

Geschrieben von Tom Rhodes und Robert Watson.
Übersetzt von Daniel Seuffert.

17.1. Einleitung

FreeBSD bietet Unterstützung für Sicherheits-Auditing. Ereignis-Auditing bietet zuverlässige, feingranulierte und konfigurierbare Aufzeichnung einer Vielzahl von sicherheitsrelevanten Systemereignissen einschliesslich Benutzereingaben, Konfigurationsänderungen sowie Datei- und Netzwerkzugriffen. Diese Log-Datensätze können unschätzbar wertvoll sein für direkte Systemüberwachung, Einbruchserkennung und Post-Mortem-Analyse. FreeBSD implementiert Sun™s öffentlich zugängliches Basic Security Module (BSM) Application Programming Interface (API) und Dateiformat, und kann mit den Audit-Implementierungen von Sun™ Solaris™ und Apple® Mac OS® X zusammenarbeiten.

Dieses Kapitel konzentriert sich auf die Installation und Konfiguration des Ereignis-Auditings. Es erklärt Audit-Richtlinien und stellt ein Beispiel einer Audit-Konfiguration vor.

Nach dem Lesen dieses Kapitels werden Sie Folgendes wissen:

- Was Ereignis-Auditing ist und wie es funktioniert.
- Wie man Ereignis-Auditing in FreeBSD für Benutzer und Prozesse konfiguriert.
- Wie man den Audit-Pfad mittels Audit-Reduktion und Revisionswerkzeugen überprüft.

Vor dem Lesen dieses Kapitels sollten Sie:

- Sowohl UNIX® als auch FreeBSD-Basismechanismen beherrschen (Kapitel 4, *Grundlagen des UNIX Betriebssystems*).
- Mit den grundlegenden Mechanismen der Kernel-Konfiguration und -Kompilierung vertraut sein (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).
- Mit den Maßnahmen zur Sicherung von FreeBSD vertraut sein (Kapitel 14, *Sicherheit*).

Warnung

Die Audit-Funktionalität in FreeBSD hat einige bekannte Einschränkungen. Nicht alle sicherheitsrelevanten System-Ereignisse sind auditierbar, und einige Anmelde-Mechanismen, wie beispielsweise Xorg-basierte Bildschirm-Manager und Dienste von Drittanbietern, konfigurieren das Auditing für Benutzeranmeldungen nicht korrekt.

Das Sicherheits-Auditing ist in der Lage, sehr detaillierte Log-Dateien von Systemaktivitäten zu erzeugen. Auf einem ausgelasteten System kann die Pfad-Datei sehr groß werden, wenn sie für hohe Auflösung konfiguriert ist, und im Extremfall pro Woche um mehrere Gigabyte anwachsen. Administratoren sollten daher den benötigten Plattenplatz in Verbindung mit umfangreichen Audit-Konfigurationen berücksichtigen. So kann es wünschenswert sein, ein eigenes Dateisystem für `/var/audit` einzusetzen, damit andere Dateisysteme nicht betoffen sind, wenn das Dateisystem des Audit voll läuft.

17.2. Schlüsselbegriffe

Die folgenden Begriffe stehen im Zusammenhang mit Ereignis-Auditing:

- *event*: ein auditierbares Ereignis ist jedes Ereignis, das mit dem Audit-Subsystem aufgezeichnet werden kann. Beispiele für sicherheitsrelevante Systemereignisse sind etwa das Anlegen von Dateien, das Erstellen einer Netzwerkverbindung oder eine Benutzeranmeldung. Ereignisse sind entweder „attributierbar“, können also zu einen authentifizierten Benutzer zurückverfolgt werden, oder sind „nicht-attributierbar“. Nicht-attributierbare Ereignisse erfolgen daher vor der Authentifizierung im Anmeldeprozess (beispielsweise die Eingabe eines falschen Passworts).
- *class*: benannte Zusammenstellungen von zusammengehörenden Ereignissen, die in Auswahl-Ausdrücken benutzt werden. Häufig genutzte Klassen von Ereignissen schließen „file creation“ (fc, Anlegen von Dateien), „exec“ (ex, Ausführung) und „login_logout“ (lo, Anmeldung-Abmeldung) ein.
- *record*: ein Audit-Logeintrag, der ein Sicherheitsereignis enthält. Jeder Datensatz enthält einen Ereignistyp, Informationen über den Gegenstand (Benutzer), welcher die Aktion durchführt, Datums- und Zeitinformationen, Informationen über jedes Objekt oder Argument sowie den Zustand hinsichtlich Erfolg oder Scheitern der Operation.
- *trail*: eine Log-Datei bestehend aus einer Reihe von Audit-Datensätzen, die Sicherheitsereignisse beschreiben. Pfade sind in grober zeitlicher Reihenfolge bezüglich des Zeitpunktes, an welchem ein Ereignis beendet wurde. Nur authorisierte Prozesse dürfen Datensätze zum Audit-Pfad hinzufügen.
- *selection expression*: eine Zeichenkette, welche eine Liste von Präfixen und Audit-Ereignisklassennamen enthält, um Ereignisse abzugleichen.
- *preselection*: der Prozess, durch den das System erkennt, welche Ereignisse von Interesse für den Administrator sind, um die Erzeugung von Datensätze zu verhindern, welche nicht von Belang sind. Die Konfiguration der Vorauswahl benutzt eine Reihe von Auswahl-Ausdrücken, um zu erkennen, welche Klassen von Ereignissen für welche Benutzer aufgezeichnet werden sollen sowie globale Einstellungen, welche sowohl auf authorisierte als auch unauthorisierte Prozesse angewendet werden.
- *reduction*: Die Reduzierung ist der Prozess, durch den Datensätze von bestehenden Audit-Pfaden ausgewählt werden für Speicherung, Ausdruck oder Analyse. Ebenso der Prozess, durch den unerwünschte Datensätze aus dem Audit-Pfad entfernt werden. Mittels Reduzierung können Administratoren Richtlinien für die Speicherung von Audit-Daten vorgeben. Zum Beispiel können ausführliche Audit-Pfade für einen Monat gespeichert werden, um danach den Pfad für archivarische Zwecke auf die Anmeldeinformationen zu reduzieren.

17.3. Audit Konfiguration

Userspace-Untersützung für Ereignis-Auditing ist Bestandteil des FreeBSD-Betriebssystems. Kernel-Unterstützung kann durch Hinzufügen der folgenden Zeile in `/etc/rc.conf` aktiviert werden:

```
auditd_enable="YES"
```

Starten Sie anschließend den Audit-Daemon:

```
# service auditd start
```

Benutzer, die es bevorzugen einen angepassten Kernel zu kompilieren, müssen folgende Zeile in die Kernelkonfigurationsdatei aufnehmen:

```
options   AUDIT
```

17.3.1. Ereignis-Auswahlausdrücke

Auswahlausdrücke werden an einigen Stellen der Audit-Konfiguration benützt, um zu bestimmen, welche Ereignisse auditiert werden sollen. Die Ausdrücke enthalten eine Liste der Ereignisklassen, welche verglichen werden

sollen, jede mit einem Präfix, welches anzeigt, ob verglichene Datensätze akzeptiert oder ignoriert werden sollen und optional, um anzuzeigen, ob der Eintrag beabsichtigt, erfolgreiche oder fehlgeschlagene Operationen zu vergleichen. Auswahlausdrücke werden von links nach rechts ausgewertet und zwei Ausdrücke werden durch Aneinanderhängen miteinander kombiniert.

Tabelle 17.1, „Audit-Ereignisklassen“ fasst die Audit-Ereignisklassen zusammen:

Tabelle 17.1. Audit-Ereignisklassen

Name der Klasse	Beschreibung	Aktion
all	all	Vergleicht alle Ereisnisklassen.
ad	administrative	Administrative Aktionen, ausgeführt auf dem System als Ganzes.
ap	application	Aktionen definiert für Applikationen.
cl	file close	Audit-Aufrufe für den Systemaufruf `close`.
ex	exec	Ausführung des Audit-Programms. Auditierung von Befehlszeilen-Argumenten und Umgebungsvariablen wird gesteuert durch audit_control(5) mittels der `argv` und `envv`-Parameter gemäß der `Richtlinien`-Einstellungen.
fa	file attribute access	Auditierung des Zugriffs auf Objektattribute wie stat(1) und pathconf(2).
fc	file create	Audit-Ereignisse, bei denen eine Datei als Ergebnis angelegt wird.
fd	file delete	Audit-Ereignisse, bei denen Dateilöschungen vorkommen.
fm	file attribute modify	Audit-Ereignisse, bei denen Dateiattribute geändert werden, wie chown(8), chflags(1) und flock(2).
fr	file read	Audit-Ereignisse, bei denen Daten gelesen oder Dateien zum lesen geöffnet werden.
fw	file write	Audit-Ereignisse, bei denen Daten geschrieben oder Dateien geschrieben oder verändert werden.
io	ioctl	Nutzung des Systemaufrufes `ioctl` durch Audit.
ip	ipc	Auditierung verschiedener Formen von Inter-Prozess-Kommunikation einschließlich POSIX-Pipes und System V IPC-Operationen.
lo	login_logout	Audit-Ereignisse von login(1) und logout(1).
na	non attributable	Auditierung nicht-attributierbarer Ereignisse.

Name der Klasse	Beschreibung	Aktion
no	invalid class	Kein Abgleich von Audit-Ereignissen.
nt	network	Audit-Ereignisse in Zusammenhang mit Netzwerkaktivitäten wie connect(2) und accept(2)
ot	other	Auditierung verschiedener Ereignisse.
pc	process	Auditierung von Prozess-Operationen wie exec(3) und exit(3).

Diese Ereignisklassen können angepasst werden durch Modifizierung der Konfigurationsdateien `audit_class` und `audit_event` .

Jede Audit-Klasse ist kombiniert mit einem Präfix, welches anzeigt, ob erfolgreiche/gescheiterte Operationen abgebildet werden, und ob der Eintrag den Abgleich hinzufügt oder entfernt für die Klasse und den Typ. Tabelle 17.2, „Präfixe für Audit-Ereignisklassen“ fasst die verfügbaren Präfixe zusammen.

Tabelle 17.2. Präfixe für Audit-Ereignisklassen

Präfix	Aktion
+	Auditiert erfolgreiche Ereignisse in dieser Klasse.
-	Auditiert fehlgeschlagene Ereignisse in dieser Klasse.
^	Auditiert weder erfolgreiche noch fehlgeschlagene Ereignisse.
^+	Auditiert keine erfolgreichen Ereignisse in dieser Klasse.
^-	Auditiert keine fehlgeschlagenen Ereignisse in dieser Klasse.

Wenn kein Präfix vorhanden ist, werden sowohl erfolgreiche als auch fehlgeschlagene Ereignisse auditiert.

Das folgende Beispiel einer Auswahl-Zeichenkette wählt erfolgreiche und gescheiterte Anmelde/Abmelde-Ereignisse aus, aber nur erfolgreich beendete Ausführungs-Ereignisse:

```
lo,+ex
```

17.3.2. Konfigurationsdateien

Die folgenden Konfigurationsdateien für Sicherheits-Auditing befinden sich in `/etc/security` .

- `audit_class` : enthält die Definitionen der Audit-Klassen.
- `audit_control`: steuert die Eigenschaften des Audit-Subsystems, wie Standard-Audit-Klassen, Mindestfestplattenspeicher auf dem Audit-Log-Volume und die maximale Größe des Audit-Trails.
- `audit_event` : Namen und Beschreibungen der Audit-Ereignisse, und eine Liste von Klassen mit den dazugehörigen Ereignissen.
- `audit_user`: benutzerspezifische Audit-Anforderungen, kombinierbar mit den globalen Standardeinstellungen bei der Anmeldung.
- `audit_warn` : ein anpassbares Skript, das von auditd(8) verwendet wird, um in bestimmten Situationen Warnmeldungen zu generieren, z.B. wenn der Platz für Audit-Protokolle knapp wird, oder wenn die Datei des Audit-Trails rotiert wurde.

Warnung

Konfigurationsdateien von Audit sollten sorgfältig bearbeitet und gepflegt werden, da Fehler in der Konfiguration zu einer fehlerhaften Protokollierung der Ereignisse führen können.

In den meisten Fällen werden Administratoren nur `audit_control` und `audit_user` änpassen müssen. Die erste Datei steuert systemweite Audit-Eigenschaften, sowie Richtlinien. Die zweite Datei kann für die Feinabstimmung bei der Auditierung von Benutzern verwendet werden.

17.3.2.1. Die `audit_control`-Datei

Die `audit_control`-Datei legt eine Anzahl Vorgabewerte fest:

```
dir:/var/audit
dist:off
flags:lo,aa
minfree:5
naflags:lo,aa
policy:cnt,argv
filesz:2M
expire-after:10M
```

Die Option `dir` wird genutzt, um eines oder mehrere Verzeichnisse festzulegen, in welchen Audit-Protokolle gespeichert werden. Gibt es mehrere Verzeichniseinträge, werden diese in der angegebenen Reihenfolge genutzt, bis sie jeweils gefüllt sind. Es ist üblich, Audit so zu konfigurieren, dass die Audit-Logs auf einem dedizierten Dateisystem abgelegt werden, um Wechselwirkungen zwischen dem Audit-Subsystem und anderen Subsystemen zu verhindern, falls das Dateisystem voll läuft.

Das `flags` -Feld legt die systemweite Standard-Vorauswahl-Maske für attributierbare (direkt einem Benutzer zuordenbare) Ereignisse fest. Im obigen Beispiel werden alle gescheiterten und erfolgreichen Anmelde- und Abmelde-Ereignisse für alle Benutzer aufgezeichnet.

Die Option `minfree` definiert den minimalen Prozentsatz an freiem Plattenplatz für das Dateisystem, in welchem der Audit-Pfad abgespeichert wird. Wenn diese Schwelle überschritten ist, wird ein Warnhinweis erzeugt. Das obige Beispiel legt den minimalen freien Platz auf zwanzig Prozent fest.

Die `naflags`-Option bestimmt diejenigen Audit-Klassen, für die nicht-attributierbare Ereignisse aufgezeichnet werden sollen (beispielsweise Anmeldeprozesse und System-Daemonen.

Die Option `policy` legt eine durch Kommata getrennte Liste von policy-Flags fest, welche verschiedene Aspekte des Audit-Verhaltens steuern. Der vorgegebene Flag `cnt` zeigt an, dass das System trotz eines Audit-Fehlers weiterlaufen soll (dieses Flag wird dringend angeraten). Ein anderes, häufig genutztes Flag ist `argv`, welches dazu führt, dass Befehlszeilen-Argumente für den Systemauruf execve(2) als Teil der Befehlsausführung aufgezeichnet werden.

Die `filesz` -Option spezifiziert die maximale Größe in Bytes, welche eine Audit-Pfad-Datei wachsen darf, bevor sie automatisch beendet und rotiert wird. Die Standardvorgabe `0` setzt die automatische Log-Rotation ausser Kraft. Falls die angeforderte Dateigröße größer Null und gleichzeitig unterhalb des Minimums von 512K ist, dann wird die Angabe verworfen und ein Log-Hinweis wird erzeugt.

17.3.2.2. Die Datei `audit_user`

Die `audit_user`-Datei erlaubt es dem Administrator, weitere Audit-Erfordernisse für bestimmte Benutzer festzulegen. Jede Zeile konfiguriert das Auditing für einen Benutzer über zwei Felder: Das erste Feld ist `alwaysaudit`, welches eine Ansammlung von Ereignissen vorgibt, welche immer für diesen Benutzer aufgezeichnet werden. Das zweite Feld `neveraudit` legt eine Menge an Ereignissen fest, die niemals für diesen Benutzer auditiert werden sollen.

Das folgende Beispiel einer `audit_user`-Datei zeichnet Anmelde/Abmelde-Ereignisse, erfolgreiche Befehlsausführungen für den Benutzer `root`, Anlegen von Dateien und erfolgreiche Befehlsausführungen für den Benutzer `www` auf. Falls das Beispiel zusammen mit der vorstehend als Beispiel gezeigten Datei `audit_control` benutzt wird, dann ist der Eintrag `lo` für `root` überflüssig und Anmelde/Abmelde-Ereignisse werden für `www` ebenfalls aufgezeichnet.

```
root:lo,+ex:no
www:fc,+ex:no
```

17.4. Administration des Audit-Subsystems

17.4.1. Audit-Pfade betrachten

Audit-Pfade werden im binären BSM-Format gespeichert, daher benötigen Sie spezielle Werkzeuge, um derartige Dateien zu ändern oder Sie in Textdateien zu konvertieren. Der Befehl praudit(1) wandelt alle Pfad-Dateien in ein einfaches Textformat um. Der Befehl auditreduce(1) kann genutzt werden, um die Pfad-Dateien für Analyse, Ausdruck, Archivierung oder andere Zwecke zu reduzieren. Eine Reihe von Auswahl-Parametern werden von auditreduce(1) unterstützt, einschliesslich Ereignistyp, Ereignisklasse, Benutzer, Datum oder Uhrzeit des Ereignisses und den Dateipfad oder das Objekt, mit dem gearbeitet wurde.

Das Dienstprogramm praudit(1) schreibt zum Beispiel den gesamten Inhalt einer angegebenen Audit-Protokolldatei in eine simple Textdatei:

```
# praudit /var/audit/AUDITFILE
```

`AUDITFILE` ist hier die zu schreibende Protokolldatei.

Audit-Pfade bestehen aus einer Reihe von Datensätzen, die wiederum aus Kürzeln (token) gebildet werden, die von praudit(1) fortlaufend zeilenweise ausgegeben werden. Jedes Kürzel ist von einem bestimmten Typ, z.B. enthält `header` einen audit-Datensatz-Header oder `path` enthält einen Dateipfad von einer Suche. Hier ein Beispiel eines `execve`-Ereignisses:

```
header,133,10,execve(2),0,Mon Sep 25 15:58:03 2006, + 384 msec
exec arg,finger,doug
path,/usr/bin/finger
attribute,555,root,wheel,90,24918,104944
subject,robert,root,wheel,root,wheel,38439,38032,42086,128.232.9.100
return,success,0
trailer,133
```

Dieser Audit stellt einen erfolgreichen `execve`-Aufruf dar, in welchem der Befehl `finger doug` ausgeführt wurde. Das Kürzel des Argumentes enthält die Befehlszeile, welche die Shell an den Kernel weiterleitet. Das Kürzel `path` enthält den Pfad zur ausführbaren Datei (wie vom Kernel wahrgenommen). Das Kürzel `attribute` beschreibt die Binärdatei (insbesondere den Datei-Modus, der genutzt werden kann, um zu bestimmen, ob setuid auf die Applikation angewendet wurde). Das Kürzel `subject` beschreibt den untergeordneten Prozess und speichert daher in Aufeinanderfolge Audit-Benutzer-ID, effektive Benutzer-ID und Gruppen-ID, wirkliche Benutzer-ID und Grppen-ID, Process-ID, Session- ID, Port-ID und Anmelde-Adresse. Beachten Sie, dass Audit-Benutzer-ID und wirkliche Benutzer-ID abweichen: Der Benutzer `robert` wurde zum Benutzer `root`, bevor er diesen Befehl ausführte, aber er wird auditiert mit dem ursprünglich authentifizierten Benutzer. Schließlich zeigt das Kürzel `return` die erfolgreiche Ausführung an und `trailer` schließt den Datensatz ab.

praudit(1) unterstützt auch die Ausgabe im XML-Format (die sie über die Option `-x` auswählen können).

17.4.2. Audit-Pfade reduzieren

Da Audit-Protokolldateien sehr groß sein können, wird ein Administrator höchstwahrscheinlich eine Auswahl an Datensätzen verwenden, wie z.B. alle Datensätze zu einem bestimmten Benutzer:

```
# auditreduce -u trhodes /var/audit/AUDITFILE | praudit
```

Dies wird alle Audit-Datensätze des Benutzers `trhodes` auswählen, die in *`AUDITFILE`* gespeichert sind.

17.4.3. Delegation von Rechten für Audit-Reviews

Mitglieder der Gruppe `audit` haben die Erlaubnis, Audit-Pfade in `/var/audit` zu lesen; standardmässig ist diese Gruppe leer, daher kann nur der Benutzer `root` die Audit-Pfade lesen. Benutzer können der Gruppe `audit` hinzugefügt werden, um Rechte für Audit-Reviews zu gewähren. Da die Fähigkeit, Inhalte von Audit-Protokolldateien zu verfolgen, tiefgreifende Einblicke in das Verhalten von Benutzern und Prozessen erlaubt, wird empfohlen, dass die Gewährung von Rechten für Audit-Reviews mit Bedacht erfolgt.

17.4.4. Aktive Überwachung mittles Audit-Pipes

Audit-Pipes sind nachgebildete (geklonte) Pseudo-Geräte im Dateisystem des Gerätes, welche es Applikationen erlauben, die laufenden Audit-Datensätze anzuzapfen. Dies ist vorrangig für Autoren von Intrusion Detection Software und Systemüberwachungsprogrammen von Bedeutung. Allerdings ist für den Administrator das Audit-Pipe-Gerät ein angenehmer Weg, aktive Überwachung zu gestatten, ohne Gefahr von Problemen durch Besitzerrechte der Audit-Pfad-Datei oder Unterbrechung des Stroms von Ereignissen durch Log-Rotation. Um den laufenden Audit-Ereignisstrom zu verfolgen, geben Sie bitte folgende Befehlszeile ein:

```
# praudit /dev/auditpipe
```

In der Voreinstellung kann nur der Benutzer `root` auf die Audit-Pipe-Geräte-Knotenpunkte zugreifen. Um sie allen Mitgliedern der Gruppe `audit` zugänglich zu machen, fügen Sie eine `devfs` -Regel in `devfs.rules` hinzu:

```
add path 'auditpipe*' mode 0440 group audit
```

Lesen Sie devfs.rules(5) für weitere Informationen, wie das devfs-Dateisystem konfiguriert wird.

Warnung

Es ist sehr leicht, Rückmeldungszyklen von Audit-Ereignissen hervorzurufen, in welcher das Betrachten des Resultates eines Audit-Ereignisses in die Erzeugung von mehr Audit-Ereignissen mündet. Wenn zum Beispiel der gesamte Netzwerk-I/O auditiert wird, während praudit(1) in einer SSH-Sitzung gestartet wurde, dann wird ein kontinuierlicher, mächtiger Strom von Audit-Ereignissen erzeugt, da jedes ausgegebene Ereignis wiederum neue Ereignisse erzeugt. Es ist anzuraten, praudit(1) an einem Audit-Pipe-Gerät nur von Sitzungen anzuwenden (ohne feingranuliertes I/O-Auditing), um dies zu vermeiden.

17.4.5. Rotation von Audit-Pfad-Dateien

Audit-Pfade werden nur vom Kernel geschrieben und nur vom Audit-Daemon auditd(8) verwaltet. Administratoren sollten nicht versuchen, newsyslog.conf(5) oder andere Werkzeuge zu benutzen, um Audit-Protokolldateien direkt zu rotieren. Stattdessen sollte das audit(8) Management-Werkzeug benutzt werden, um die Auditierung zu beenden, das Audit-System neu zu konfigurieren und eine Log-Rotation durchzuführen. Der folgende Befehl veranlasst den Audit-Daemon, eine neue Protokolldatei anzulegen und dem Kernel zu signalisieren, die neue Datei zu nutzen. Die alte Datei wird beendet und umbenannt. Ab diesem Zeitpunkt kann sie vom Administrator bearbeitet werden.

```
# audit -n
```

Warnung

Falls der auditd(8)-Daemon gegenwärtig nicht läuft, wird dieser Befehl scheitern und eine Fehlermeldung wird ausgegeben.

Das Hinzufügen der folgenden Zeile in `/etc/crontab` wird die Log-Rotation alle zwölf Stunden durch cron(8) erzwingen:

```
0     */12       *       *       *       root    /usr/sbin/audit -n
```

Die Änderung wird wirksam, sobald Sie die neue `/etc/crontab` gespeichert haben.

Die automatische Rotation der Audit-Pfad-Datei in Abhängigkeit von der Dateigröße ist möglich durch die Angabe der Option `filesz` in audit_control(5). Dieser Vorgang ist im Abschnitt Konfigurationsdateien dieses Kapitels beschrieben.

17.4.6. Komprimierung von Audit-Pfaden

Da Audit-Pfad-Dateien sehr groß werden können, ist es oft wünschenswert, Pfade zu komprimieren oder anderweitig zu archivieren, sobald sie vom Audit-Daemon geschlossen wurden. Das Skript `audit_warn` kann genutzt werden, um angepasste Aktionen für eine Vielzahl von audit-bezogenen Ereignissen auszuführen, einschliesslich der sauberen Beendigung von Audit-Pfaden, wenn diese geschlossen werden. Zum Beispiel kann man die folgenden Zeilen in das `audit_warn` -Skript aufnehmen, um Audit-Pfade beim Beenden zu komprimieren:

```
#
# Compress audit trail files on close.
#
if [ "$1" = closefile -]; then
        gzip -9 $2
fi
```

Andere Archivierungsaktivitäten können das Kopieren zu einem zentralen Server, die Löschung der alten Pfad-Dateien oder die Reduzierung des alten Audit-Pfades durch Entfernung nicht benötigter Datensätze einschliessen. Das Skript wird nur dann ausgeführt, wenn die Audit-Pfad-Dateien sauber beendet wurden, daher wird es nicht auf Pfaden laufen, welche durch ein unsauberes Herunterfahren des Systems nicht beendet wurden.

Kapitel 18. Speichermedien

Übersetzt von Bernd Warken und Martin Heinen.

18.1. Übersicht

Dieses Kapitel behandelt die Benutzung von Laufwerken unter FreeBSD. Hierzu zählen speichergestützte Laufwerke, Netzlaufwerke, SCSI/IDE-Geräte und Geräte, die die USB-Schnittstelle benutzen.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie Folgendes wissen:

- Die Begriffe, die FreeBSD verwendet, um die Organisation der Daten auf einem physikalischen Laufwerk zu beschreiben.
- Wie Sie zusätzliche Laufwerke zu einem FreeBSD-System hinzufügen.
- Wie Sie FreeBSD zur Verwendung von USB-Speichermedien konfigurieren.
- Wie virtuelle Dateisysteme, zum Beispiel RAM-Disks, eingerichtet werden.
- Wie Sie mit Quotas die Benutzung von Laufwerken einschränken können.
- Wie Sie Partitionen verschlüsseln, um Ihre Daten zu schützen.
- Wie unter FreeBSD CDs und DVDs gebrannt werden.
- Wie Sie die unter FreeBSD erhältlichen Backup-Programme benutzen.
- Was Dateisystem-Schnappschüsse sind und wie sie eingesetzt werden.

Bevor Sie dieses Kapitel lesen,

- sollten Sie wissen, wie Sie einen neuen FreeBSD-Kernel konfigurieren und installieren können.

18.2. Gerätenamen

Die folgende Tabelle zeigt die von FreeBSD unterstützten Speichergeräte und deren Gerätenamen.

Tabelle 18.1. Namenskonventionen von physikalischen Laufwerken

Laufwerkstyp	Gerätename
IDE-Festplatten	`ad` oder `ada`
IDE-CD-ROM-Laufwerke	`acd` oder `cd`
SATA-Festplatten	`ad` oder `ada`
SATA-CD-ROM-Laufwerke	`acd` oder `cd`
SCSI-Festplatten und USB-Speichermedien	`da`
SCSI-CD-ROM-Laufwerke	`cd`
Verschiedene proprietäre CD-ROM-Laufwerke	`mcd` für Mitsumi CD-ROM und `scd` für Sony CD-ROM
Diskettenlaufwerke	`fd`
SCSI-Bandlaufwerke	`sa`
IDE-Bandlaufwerke	`ast`
Flash-Laufwerke	`fla` für DiskOnChip® Flash-Device

Laufwerkstyp	Gerätename
RAID-Laufwerke	`aacd` für Adaptec® AdvancedRAID, `mlxd` und `mlyd` für Mylex®, `amrd` für AMI MegaRAID®, `idad` für Compaq Smart RAID, `twed` für 3ware® RAID.

18.3. Hinzufügen von Laufwerken

Im Original von David O'Brian.

Dieser Abschnitt beschreibt, wie Sie ein neues SATA-Laufwerk zu einer Maschine hinzufügen, die momentan nur ein Laufwerk hat. Dazu schalten Sie zuerst den Rechner aus und installieren das Laufwerk entsprechend der Anleitungen Ihres Rechners, Ihres Controllers und des Laufwerkherstellers. Starten Sie das System neu und melden Sie sich als Benutzer `root` an.

Kontrollieren Sie `/var/run/dmesg.boot` , um sicherzustellen, dass das neue Laufwerk gefunden wurde. In diesem Beispiel erscheint das neu hinzugefügte SATA-Laufwerk als `ada1`.

In diesem Beispiel wird eine einzige große Partition auf der Festplatte erstellt. Verwendet wird das GPT-Partionsschema, welches gegenüber dem älteren und weniger vielseitigen MBR-Schema bevorzug wird.

Anmerkung

Wenn die hinzugefügte Festplatte nicht leer ist, können alte Partitionsinformationen mit `gpart delete` entfernt werden. Details finden Sie in gpart(8).

Zuerst wird das Partitionsschema erstellt und dann eine einzelne Partition angefügt:

```
# gpart create -s GPT ada1
# gpart add -t freebsd-ufs ada1
```

Je nach Anwendung kann es wünschenswert sein, mehrere kleinere Partitionen zu haben. In gpart(8) finden Sie Optionen zum Erstellen von kleineren Partitionen.

Ein Dateisystem wird auf der neuen, leeren Festplatte erstellt:

```
# newfs -U /dev/ada1p1
```

Ein leeres Verzeichnis wird als Mountpunkt erstellt, also ein Speicherort für die Montage der neuen Festplatte im originalen Dateisystem:

```
# mkdir /newdisk
```

Abschließend wird ein Eintrag in `/etc/fstab` hinzugefügt, damit die neue Festplatte automatisch beim Start eingehängt wird:

```
/dev/ada1p1 /newdisk      ufs    rw     2   2
```

Die neue Festplatte kann manuell montiert werden, ohne das System neu zu starten:

```
# mount /newdisk
```

18.4. Partitionen vergrößern

Beigetragen von Allan Jude.
Übersetzt von Björn Heidotting.

Die Kapazität einer Festplatte kann sich ohne Änderungen an bereits vorhandenen Daten erhöhen. Dies geschieht üblicherweise mit virtuellen Maschinen, wenn sich herausstellt, dass die virtuelle Festplatte zu klein ist und vergrößert werden soll. Zuweilen wird auch ein Abbild einer Platte auf einen USB-Stick geschrieben, ohne dabei die volle Kapazität zu nutzen. Dieser Abschnitt beschreibt, wie man Platten vergrößert, bzw. *erweitert*, um die Vorteile der erhöhten Kapazität zu nutzen.

Überprüfen Sie `/var/run/dmesg.boot` , um den Gerätenamen der Festplatte zu bestimmen, die vergrößert werden soll. In diesem Beispiel gibt es nur eine SATA-Festplatte im System, so dass die Platte als `ada0` angezeigt wird.

Um die aktuelle Konfiguration der Partitionen auf der Festplatte anzuzeigen:

```
# gpart show ada0
=>      34  83886013  ada0  GPT  (48G) [CORRUPT]
        34       128     1  freebsd-boot  (64k)
       162  79691648     2  freebsd-ufs  (38G)
  79691810   4194236     3  freebsd-swap  (2G)
  83886046         1        - free -  (512B)
```

Anmerkung

Wenn die Festplatte mit dem GPT-Partitionsschema formatiert wurde kann es vorkommen, dass sie als „corrupted“ angezeigt wird, weil sich die Sicherung der GPT-Partitionstabellen nicht mehr am Ende des Laufwerks befinden. Reparieren Sie in so einem Fall die Partitionstabelle mit `gpart`:

```
# gpart recover ada0
ada0 recovered
```

Nun steht der zusätzliche Speicherplatz zur Verfügung und kann verwendet werden, um eine neue Partition anzulegen oder eine bestehende Partition zu erweitern:

```
# gpart show ada0
=>       34  102399933  ada0  GPT  (48G)
         34        128     1  freebsd-boot  (64k)
        162   79691648     2  freebsd-ufs  (38G)
   79691810    4194236     3  freebsd-swap  (2G)
   83886046   18513921        - free -  (8.8G)
```

Partitionen können nur auf zusammenhängenden, freien Speicherplatz vergrößert werden. In diesem Beispiel wird die letzte Partition der Platte als Swap-Speicher genutzt, aber die zweite Partition ist die, dessen Größe verändert werden soll. Weil der Swap-Speicher nur temporäre Daten enthält, kann er gefahrlos ausgehangen, gelöscht und nachdem die Partition vergrößert wurde, neu erstellt werden.

```
# swapoff /dev/ada0p3
# gpart delete -i 3 ada0
ada0p3 deleted
# gpart show ada0
=>       34  102399933  ada0  GPT  (48G)
         34        128     1  freebsd-boot  (64k)
        162   79691648     2  freebsd-ufs  (38G)
 79691810   22708157        - free -  (10G)
```

Warnung

Es besteht die Gefahr von Datenverlust, wenn die Partitionstabelle eines eingehangenen Dateisystems verändert wird. Es empfiehlt sich daher, die folgenden Schritte auf einem aus-

gehangenen Dateisystem durchzuführen, während die Umsetzung über eine Live-CD-ROM oder von einem USB-Gerät erfolgt. Wenn es jedoch absolut notwendig ist, kann ein eingehangenes Dateisystem auch vergrößert werden, nachdem die Sicherheitsfunktionen von GEOM deaktiviert wurden:

```
# sysctl kern.geom.debugflags=16
```

Vergrößern Sie die Partition und lassen Sie Platz, um die Swap-Partition in der gewünschten Größe neu erstellen zu können. Dies ändert nur die Größe der Partition. Das Dateisystem innerhalb der Partition wird in einem separaten Schritt erweitert.

```
# gpart resize -i 2 -a 4k -s 47G ada0
ada0p2 resized
# gpart show ada0
=>       34  102399933  ada0  GPT  (48G)
         34        128     1  freebsd-boot  (64k)
        162   98566144     2  freebsd-ufs  (47G)
   98566306    3833661        - free -  (1.8G)
```

Erstellen Sie die Swap-Partition neu:

```
# gpart add -t freebsd-swap -a 4k  ada0
ada0p3 added
# gpart show ada0
=>       34  102399933  ada0  GPT  (48G)
         34        128     1  freebsd-boot  (64k)
        162   98566144     2  freebsd-ufs  (47G)
   98566306    3833661     3  freebsd-swap  (1.8G)
# swapon /dev/ada0p3
```

Erweitern Sie das UFS-Dateisystem, um die Kapazität der vergrößerten Partition zu nutzen:

Anmerkung

Ab FreeBSD 10.0-RELEASE ist es möglich, ein eingehangenes Dateisystem zu erweitern. Bei älteren Versionen muss das Dateisystem zuvor ausgehangen werden.

```
# growfs /dev/ada0p2
Device is mounted read-write; resizing will result in temporary write suspension for /.
It's strongly recommended to make a backup before growing the file system.
OK to grow file system on /dev/ada0p2, mounted on /, from 38GB to 47GB? [Yes/No] Yes
super-block backups (for fsck -b #) at:
 80781312, 82063552, 83345792, 84628032, 85910272, 87192512, 88474752,
 89756992, 91039232, 92321472, 93603712, 94885952, 96168192, 97450432
```

Sowohl die Partition als auch das Dateisystem wurden jetzt vergrößert, um den neu zur Verfügung stehenden Speicherplatz zu nutzen.

18.5. USB Speichermedien

Beigetragen von Marc Fonvieille.

Der Universal Serial Bus (USB) wird von vielen externen Speichern benutzt: Festplatten, USB-Thumbdrives sowie von CD- und DVD-Brennern. FreeBSD bietet Unterstützung für Geräte mit USB 1.x, 2.0 und 3.0.

Anmerkung

Die Unterstützung fürUSB 3.0 ist mit einiger Hardware, einschließlich Haswell (Lynx Point) Chipsätzen, nicht kompatibel. Wenn FreeBSD beim Booten mit dem Fehler failed with error 19 abbricht, müssen Sie xHCI/USB3 im BIOS deaktivieren.

Unterstützung für USB-Massenspeicher ist im `GENERIC` -Kernel enthalten. Für einen angepassten Kernel müssen die nachstehenden Zeilen in der Kernelkonfigurationsdatei enthalten sein:

```
device scbus>>>>>>>>># SCSI bus (required for ATA/SCSI)
device da>>>>>>>># Direct Access (disks)
device pass>>>>>># Passthrough device (direct ATA/SCSI access)
device uhci>>>>>># provides USB 1.x support
device ohci>>>>>># provides USB 1.x support
device ehci>>>>>># provides USB 2.0 support
device xhci>>>>>># provides USB 3.0 support
device usb>>>>>># USB Bus (required)
device umass>>>>># Disks/Mass storage - Requires scbus and da
device cd>>>>>>># needed for CD and DVD burners
```

FreeBSD benutzt den umass(4)-Treiber, der das SCSI-Subsystem verwendet um auf USB-Geräte zuzugreifen. Da alle USB-Geräte vom System als SCSI-Geräte erkannt werden, dürfen Sie *nicht* `device atapicam` in die Kernelkonfigurationsdatei aufnehmen, wenn es sich bei dem Gerät um einen CD- oder DVD-Brenner handelt.

Der übrige Abschnitt beschreibt, wie Sie überprüfen können ob ein USB-Gerät von FreeBSD erkannt wird und wie Sie das Gerät so konfigurieren, dass es verwendet werden kann.

18.5.1. Konfiguration von Geräten

Um die USB-Konfiguration zu testen, schließen Sie das USB-Gerät an. Verwenden Sie `dmesg` um zu überprüfen, ob das Gerät in den Systemmeldungen erscheint. Dies sollte in etwa so aussehen:

```
umass0: <STECH Simple Drive, class 0/0, rev 2.00/1.04, addr 3> on usbus0
umass0:  SCSI over Bulk-Only; quirks = 0x0100
umass0:4:0:-1: Attached to scbus4
da0 at umass-sim0 bus 0 scbus4 target 0 lun 0
da0: <STECH Simple Drive 1.04> Fixed Direct Access SCSI-4 device
da0: Serial Number WD-WXE508CAN263
da0: 40.000MB/s transfers
da0: 152627MB (312581808 512 byte sectors: 255H 63S/T 19457C)
da0: quirks=0x2<NO_6_BYTE>
```

Fabrikat, Gerätedatei (`da0`), Geschwindigkeit und Kapazität werden je nach Gerät unterschiedlich sein.

Da ein USB-Gerät als SCSI-Gerät erkannt wird, kann `camcontrol` benutzt werden, um die mit dem System verbundenen USB-Massenspeicher anzuzeigen:

```
# camcontrol devlist
<STECH Simple Drive 1.04>          at scbus4 target 0 lun 0 (pass3,da0)
```

Alternativ kann `usbconfig` benutzt werden, um die Geräte aufzulisten. Weitere Informationen zu diesem Kommando finden Sie in usbconfig(8).

```
# usbconfig
ugen0.3: <Simple Drive STECH> at usbus0, cfg=0 md=HOST spd=HIGH (480Mbps) pwr=ON (2mA)
```

Wenn das Gerät noch nicht formatiert ist, finden Sie in Abschnitt 18.3, „Hinzufügen von Laufwerken“ Informationen, wie Sie USB-Laufwerke formatieren und Partitionen einrichten. Wenn das Laufwerk bereits ein Dateisystem

enthält, kann es von `root` nach den Anweisungen in Abschnitt 4.7, „Anhängen und Abhängen von Dateisystemen“ eingehängt werden.

Warnung

Aus Sicherheitsgründen sollten Sie Benutzern, denen Sie nicht vertrauen, das Einhängen (z.B. durch die unten beschriebene Aktivierung von `vfs.usermount`) beliebiger Medien verbieten. Die meisten Dateisysteme wurden nicht entwickelt, um sich vor böswilligen Geräten zu schützen.

Um auch normalen Anwendern das Einhängen des Laufwerks zu gestatten, könnten Sie beispielsweise mit pw(8) alle potentiellen Benutzer dieser Gerätedateien in die Gruppe `operator` aufnehmen. Außerdem muss sichergestellt werden, dass `operator` Schreib- und Lesezugriff auf diese Gerätedateien haben. Hierfür werden die folgenden Zeilen in `/etc/devfs.rules` hinzugefügt:

```
[localrules=5]
add path 'da*' mode 0660 group operator
```

Anmerkung

Verfügt das System über interne SCSI-Laufwerke, so verändern Sie die zweite Zeile wie folgt:

```
add path 'da[3-9]*' mode 0660 group operator
```

Dies wird die ersten drei SCSI-Laufwerke (`da0` bis `da2`) davon ausschließen, in die Gruppe `operator` aufgenommen zu werden. Ersetzen Sie `3` durch die Anzahl der SCSI-Laufwerke. Weitere Informationen zu dieser Datei finden Sie in devfs.rules(5).

Aktivieren Sie nun die Regeln in `/etc/rc.conf` :

```
devfs_system_ruleset="localrules"
```

Als nächstes müssen Sie das System anweisen, auch normalen Benutzern das mounten von Dateisystemen zu erlauben, indem Sie die folgende Zeile in `/etc/sysctl.conf` hinzufügen:

```
vfs.usermount=1
```

Da diese Einstellung erst nach einem Neustart wirksam wird, können Sie diese Variable mit `sysctl` auch direkt setzen:

```
# sysctl vfs.usermount=1
vfs.usermount: 0 -> 1
```

Zuletzt müssen Sie noch ein Verzeichnis anlegen, in das das USB-Laufwerk eingehängt werden soll. Dieses Verzeichnis muss dem Benutzer gehören, der das USB-Laufwerk in den Verzeichnisbaum einhängen will. Dazu legen Sie als `root` ein Unterverzeichnis `/mnt/`*`username`* an, wobei Sie *`username`* durch den Login des jeweiligen Benutzers sowie *`usergroup`* durch die primäre Gruppe des Benutzers ersetzen:

```
# mkdir /mnt/username
# chown username:usergroup /mnt/username
```

Wenn Sie nun beispielsweise einen USB-Stick anschließen, wird automatisch die Gerätedatei `/dev/da0s1` erzeugt. Ist das Gerät mit einem FAT-Dateisystem formatiert, kann es der Benutzer mit dem folgenden Befehl in den Verzeichnisbaum einhängen:

```
% mount -t msdosfs -o -m=644,-M=755 /dev/da0s1 /mnt/  username
```

Bevor das Gerät entfernt werden kann, *muss* es abgehängt werden:

```
# umount /mnt/username
```

Nach Entfernen des Geräts stehen in den Systemmeldungen Einträge, ähnlich der folgenden:

```
umass0: at uhub3, port 2, addr 3 (disconnected)
da0 at umass-sim0 bus 0 scbus4 target 0 lun 0
da0: <STECH Simple Drive 1.04> s/n WD-WXE508CAN263          detached
(da0:umass-sim0:0:0:0): Periph destroyed
```

18.6. Erstellen und Verwenden von CDs

Beigesteuert von Mike Meyer.

CDs besitzen einige Eigenschaften, die sie von konventionellen Laufwerken unterscheiden. Sie wurden so entworfen, dass sie ununterbrochen, ohne Verzögerungen durch Kopfbewegungen zwischen den Spuren, gelesen werden können. CDs besitzen Spuren, aber damit ist der Teil Daten gemeint, der ununterbrochen gelesen wird, und nicht eine physikalische Eigenschaft der CD. Das ISO 9660-Dateisystem wurde entworfen, um mit diesen Unterschieden umzugehen.

Die FreeBSD Ports-Sammlung bietet einige Werkzeuge zum Brennen und Kopieren von Audio- und Daten-CDs. Dieses Kapitel beschreibt die Verwendung von mehreren Kommandozeilen-Werkzeugen. Wenn Sie eine graphische Oberfläche zum Brennen von CDs benutzen, können Sie sysutils/xcdroast oder sysutils/k3b installieren.

18.6.1. Unterstützte Geräte

Beigetragen von Marc Fonvielle.

Mit diesem Treiber kann auf ATAPI-Geräte, wie CD- oder DVD-Laufwerke, mithilfe des SCSI-Subsystems zugegriffen werden. Dies erlaubt die Benutzung von SCSI-Werkzeugen, wie sysutils/cdrdao oder cdrecord(1).

Um den Treiber zu benutzen, fügen Sie die folgende Zeile in `/boot/loader.conf` ein:

```
atapicam_load="YES"
```

Danach muss das System neu gestartet werden.

Anmerkung

Benutzer, die es vorziehen atapicam(4)-Unterstützung statisch in den Kernel zu komplillieren, sollten diese Zeile in die Kernelkonfigurationsdatei hinzufügen:

```
device atapicam
```

Die folgenden Zeilen werden ebenfalls benötigt, sollten aber schon Teil der Kernelkonfiguration sein:

```
device ata
device scbus
device cd
device pass
```

Übersetzen und installieren Sie den neuen Kernel. Starten Sie anschließend das System neu.

Während des Bootvorgangs sollte der Brenner ungefähr so angezeigt werden:

```
acd0: CD-RW <MATSHITA CD-RW/DVD-ROM UJDA740> at ata1-master PIO4
cd0 at ata1 bus 0 target 0 lun 0
```

```
cd0: <MATSHITA CDRW/DVD UJDA740 1.00> Removable CD-ROM SCSI-0 device
cd0: 16.000MB/s transfers
cd0: Attempt to query device size failed: NOT READY, Medium not present - tray closed
```

Über den Gerätenamen `/dev/cd0` kann nun auf das Laufwerk zugegriffen werden. Um beispielsweise eine CD-ROM in `/mnt` einzuhängen, geben Sie folgendes ein:

```
# mount -t cd9660 /dev/cd0 /mnt
```

Die SCSI-Adresse des Brenners können Sie als `root` wie folgt ermitteln:

```
# camcontrol devlist
<MATSHITA CDRW/DVD UJDA740 1.00>   at scbus1 target 0 lun 0 (pass0,cd0)
```

Die SCSI-Adresse `1,0,0` kann mit den SCSI-Werkzeugen, zum Beispiel cdrecord(1), verwendet werden.

Weitere Informationen über das ATAPI/CAM- und das SCSI-System finden Sie in atapicam(4) und cam(4).

18.6.2. Ein ISO-Dateisystem erstellen

Die Datendateien müssen vorbereitet sein, bevor sie auf eine CD gebrannt werden. In FreeBSD wird `mkisofs` vom Paket oder Port sysutils/cdrtools installiert. Dieses Programm kann aus einem UNIX® Verzeichnisbaum ein ISO 9660-Dateisystem erzeugen. Im einfachsten Fall müssen Sie lediglich den Namen der zu erzeugenden ISO-Datei und den Pfad zu den Dateien angeben, die auf dem ISO 9660-Dateisystem platziert werden:

```
# mkisofs -o imagefile.iso /path/to/tree
```

Dieses Kommando erstellt *`imagefile.iso`*, die ein ISO 9660-Dateisystem enthält, das eine Kopie des Baumes unter */path/to/tree* ist. Dabei werden die Dateinamen auf Namen abgebildet, die den Restriktionen des ISO 9660-Dateisystems entsprechen. Dateien mit Namen, die im ISO 9660-Dateisystem nicht gültig sind, bleiben unberücksichtigt.

Es einige Optionen, um diese Beschränkungen zu überwinden. Die unter UNIX® Systemen üblichen Rock-Ridge-Erweiterungen werden durch `-R` aktiviert, `-J` aktiviert die von Microsoft Systemen benutzten Joliet-Erweiterungen und `-hfs` dient dazu, um das von Mac OS® benutzte HFS zu erstellen.

Für CDs, die nur auf FreeBSD-Systemen verwendet werden sollen, kann `-U` genutzt werden, um alle Beschränkungen für Dateinamen aufzuheben. Zusammen mit `-R` wird ein Abbild des Dateisystems, identisch zu angegebenen FreeBSD-Dateibaum, erstellt, obwohl dies den ISO 9660 Standard verletzen kann.

Die letzte übliche Option ist `-b`. Sie wird benutzt, um den Ort eines Bootimages einer „El Torito“ bootbaren CD anzugeben. Das Argument zu dieser Option ist der Pfad zu einem Bootimage ausgehend von der Wurzel des Baumes, der auf die CD geschrieben werden soll. In der Voreinstellung erzeugt mkisofs(8) ein ISO-Image im „Diskettenemulations“-Modus. Dabei muss das Image genau 1200, 1440 oder 2880 KB groß sein. Einige Bootloader, darunter der auf den FreeBSD-Disks verwendete, kennen keinen Emulationsmodus. Daher sollten Sie in diesen Fällen `-no-emul-boot` verwenden. Wenn `/tmp/myboot` ein bootbares FreeBSD-System enthält, dessen Bootimage sich in `/tmp/myboot/boot/cdboot` befindet, können Sie ein Abbild eines ISO 9660-Dateisystems in `/tmp/bootable.iso` erstellen:

```
# mkisofs -R -no-emul-boot -b boot/cdboot -o /tmp/bootable.iso /tmp/myboot
```

Wenn `md` im Kernel konfiguriert ist, kann das Dateisystem als speicherbasiertes Laufwerk eingehängt werden:

```
# mdconfig -a -t vnode -f /tmp/bootable.iso -u 0
# mount -t cd9660 /dev/md0 /mnt
```

Jetzt können Sie überprüfen, dass `/mnt` und `/tmp/myboot` identisch sind.

Sie können das Verhalten von mkisofs(8) mit einer Vielzahl von Optionen beeinflussen. Details dazu entnehmen Sie bitte mkisofs(8).

18.6.3. Einen ATAPI-Brenner benutzen

Für ATAPI-CD-Brenner kann `burncd` benutzt werden, um ein ISO-Image auf CD zu brennen. `burncd` ist Teil des Basissystems und unter `/usr/sbin/burncd` installiert. Da es nicht viele Optionen hat, ist es leicht zu benutzen:

```
# burncd -f cddevice data imagefile.iso fixate
```

Dieses Kommando brennt eine Kopie von *`imagefile.iso`* auf das Gerät *`cddevice`*. In der Grundeinstellung wird das Gerät `/dev/acd0` benutzt. burncd(8) beschreibt, wie die Schreibgeschwindigkeit gesetzt wird, die CD ausgeworfen wird und Audiodaten geschrieben werden.

18.6.4. Einen USB- oder SCSI-Brenner benutzen

Für Systeme ohne ATAPI-CD-Brenner kann `cdrecord` benutzt werden, um CDs zu brennen. `cdrecord` ist nicht Bestandteil des Basissystems und muss entweder als Paket sysutils/cdrtools oder Port installiert werden. Änderungen im Basissystem können Fehler im binären Programm verursachen und führen möglicherweise dazu, dass Sie einen „Untersetzer“ brennen. Es wird daher empfohlen den Port aktualisieren, wenn das System aktualisiert wird, oder für Benutzer die STABLE verfolgen, den Port zu aktualisieren, wenn es eine neue Version gibt.

Obwohl `cdrecord` viele Optionen besitzt, ist die grundlegende Anwendung einfach. Ein ISO 9660-Image wird wie folgt erstellt:

```
# cdrecord dev= device imagefile.iso
```

Der Knackpunkt in der Benutzung von `cdrecord` besteht darin, das richtige Argument zu `dev` zu finden. Benutzen Sie `-scanbus`, der eine ähnliche Ausgabe wie die folgende produziert:

```
# cdrecord -scanbus
Cdrecord 1.9 (i386-unknown-freebsd7.0) Copyright (C) 1995-2004 Jörg Schilling
Using libscg version 'schily-0.1'
scsibus0:
        0,0,0     0) 'SEAGATE ' 'ST39236LW       ' '0004' Disk
        0,1,0     1) 'SEAGATE ' 'ST39173W        ' '5958' Disk
        0,2,0     2) *
        0,3,0     3) 'iomega  ' 'jaz 1GB         ' 'J.86' Removable Disk
        0,4,0     4) 'NEC     ' 'CD-ROM DRIVE:466' '1.26' Removable CD-ROM
        0,5,0     5) *
        0,6,0     6) *
        0,7,0     7) *
scsibus1:
        1,0,0   100) *
        1,1,0   101) *
        1,2,0   102) *
        1,3,0   103) *
        1,4,0   104) *
        1,5,0   105) 'YAMAHA  ' 'CRW4260         ' '1.0q' Removable CD-ROM
        1,6,0   106) 'ARTEC   ' 'AM12S           ' '1.06' Scanner
        1,7,0   107) *
```

Für die aufgeführten Geräte in der Liste wird das passende Argument zu `dev` gegeben. Benutzen Sie die drei durch Kommas separierten Zahlen, die zum CD-Brenner angegeben sind, als Argument für `dev`. Im Beispiel ist das CDRW-Gerät 1,5,0, so dass die passende Eingabe **`dev=1,5,0`** ist. Einfachere Wege das Argument anzugeben, sowie Informationen über Audiospuren und das Einstellen der Geschwindigkeit, sind in cdrecord(1) beschrieben.

18.6.5. Kopieren von Audio-CDs

Um eine Kopie einer Audio-CD zu erstellen, kopieren Sie die Stücke der CD in einzelne Dateien und brennen diese Dateien dann auf eine leere CD. Das genaue Verfahren hängt davon ab, ob Sie ATAPI- oder SCSI-Laufwerke verwenden.

Prozedur 18.1. SCSI-Laufwerke

1. Kopieren Sie die Audiodaten mit `cdda2wav` :

   ```
   % cdda2wav -vall -D2,0 -B -Owav
   ```

2. Die erzeugten `.wav` Dateien schreiben Sie mit `cdrecord` auf eine leere CD:

```
% cdrecord -v dev=2,0 -dao -useinfo  *.wav
```

Das Argument von `dev` gibt das verwendete Gerät an, das wie in Abschnitt 18.6.4, „Einen USB- oder SCSI-Brenner benutzen“ ermittelt werden kann.

Prozedur 18.2. ATAPI-Laufwerke

Anmerkung

Über das Modul ATAPI/CAM kann `cdda2wav` auch mit ATAPI-Laufwerken verwendet werden. Diese Methode ist für die meisten Anwender besser geeignet als die im folgenden beschriebenen Methoden (Jitter-Korrektur, Big-/Little-Endian-Probleme und anderes mehr spielen hierbei eine Rolle).

1. Der ATAPI-CD-Treiber stellt die einzelnen Stücke der CD über die Dateien `/dev/acd`*`d`*`t`*`nn`*, zur Verfügung. *d* bezeichnet die Laufwerksnummer und *nn* ist die Nummer des Stücks. Die Nummer ist immer zweistellig, das heißt es wird, wenn nötig, eine führende Null ausgegeben. Die Datei `/dev/acd0t01` ist also das erste Stück des ersten CD-Laufwerks. `/dev/acd0t02` ist das zweite Stück und `/dev/acd0t03` das dritte.

   Überprüfen Sie stets, ob die entsprechenden Dateien im Verzeichnis `/dev` auch angelegt werden. Sind die Einträge nicht vorhanden, weisen Sie Ihr System an, das Medium erneut zu testen:

```
# dd if=/dev/acd0 of=/dev/null count=1
```

Anmerkung

Unter FreeBSD 4.X werden diese Einträge nicht mit dem Wert Null vordefiniert. Falls die entsprechenden Einträge unter `/dev` nicht vorhanden sind, müssen Sie diese hier von `MAKEDEV` anlegen lassen:

```
# cd /dev
# sh MAKEDEV acd0t99
```

2. Die einzelnen Stücke werden mit dd(1) kopiert. Achten Sie darauf, eine Blockgröße anzugeben:

```
# dd if=/dev/acd0t01 of=track1.cdr bs=2352
# dd if=/dev/acd0t02 of=track2.cdr bs=2352
...
```

3. Die kopierten Dateien können Sie dann mit `burncd` brennen. Geben Sie an, dass es sich um Audio-Daten handelt und dass `burncd` das Medium am Ende fixieren soll:

```
# burncd -f /dev/acd0 audio track1.cdr track2.cdr  ... fixate
```

18.6.6. Kopieren von Daten-CDs

Es ist möglich eine Daten-CD in eine Datei zu kopieren, die einem Image entspricht, das mit mkisofs(8) erstellt wurde. Mit Hilfe dieses Images können Sie jede Daten-CD kopieren. Das folgende Beispiel verwendet `acd0` für das CD-ROM-Gerät. Wird ein anderes Laufwerk benutzt, muss der Name entsprechend ersetzt werden.

```
# dd if=/dev/acd0 of=file.iso bs=2048
```

Danach existiert ein Image, das wie oben beschrieben, auf CD gebrannt werden kann.

18.6.7. Einhängen von Daten-CDs

Es ist möglich, eine Daten-CD zu mounten und die Daten zu lesen. Standardmäßig erwartet mount(8) ein Dateisystem vom Typ `ufs`. Wenn Sie das folgende Kommando ausführen:

```
# mount /dev/cd0 /mnt
```

erhalten Sie die Fehlermeldung Incorrect super block, und die CD konnte nicht eingehängt werden. Die CD benutzt nicht das `UFS`-Dateisystem, deshalb schlägt der Versuch fehl, sie als solches einzuhängen. Sie müssen mount(8) durch die Angabe von `-t cd9660` sagen, dass es sich um ein Dateisystem vom Typ `ISO9660` handelt. Wenn Sie also die CD-ROM `/dev/cd0` in `/mnt` einhängen wollen, führen Sie folgenden Befehl aus:

```
# mount -t cd9660 /dev/cd0 /mnt
```

Ersetzen Sie `/dev/cd0` durch den Gerätenamen des CD-Gerätes. Die Angabe von `-t cd9660` führt mount_cd9660(8) aus, was äquivalent zu folgendem Befehl ist:

```
# mount_cd9660 /dev/cd0 /mnt
```

Auf diese Weise können Sie Daten-CDs von jedem Hersteller verwenden. Es kann allerdings zu Problemen mit CDs kommen, die verschiedene ISO9660-Erweiterungen benutzen. So speichern Joliet-CDs alle Dateinamen unter Verwendung von zwei Byte langen Unicode-Zeichen. Zwar unterstützt der FreeBSD-Kernel derzeit noch kein Unicode, der CD9660-Treiber erlaubt es aber, zur Laufzeit eine Konvertierungstabelle zu laden. Tauchen also statt bestimmter Zeichen nur Fragezeichen auf, so müssen Sie über die Option `-C` den benötigten Zeichensatz angeben. Weitere Informationen zu diesem Problem finden Sie in der Manualpage mount_cd9660(8).

Anmerkung

Damit der Kernel diese Zeichenkonvertierung (festgelegt durch die Option `-C`) erkennt, müssen Sie das Kernelmodul `cd9660_iconv.ko` laden. Dazu fügen Sie folgende Zeile in `loader.conf` ein:

```
cd9660_iconv_load="YES"
```

Danach müssen Sie allerdings Ihr System neu starten. Alternativ können Sie das Kernelmodul auch direkt über kldload(8) laden.

Manchmal werden Sie die Meldung Device not configured erhalten, wenn Sie versuchen, eine CD-ROM einzuhängen. Für gewöhnlich liegt das daran, dass das Laufwerk meint es sei keine CD eingelegt, oder dass das Laufwerk auf dem Bus nicht erkannt wird. Es kann einige Sekunden dauern, bevor das Laufwerk merkt, dass eine CD eingelegt wurde. Seien Sie also geduldig.

Manchmal wird ein SCSI-CD-ROM nicht erkannt, weil es keine Zeit hatte, auf das Zurücksetzen des Busses zu antworten. Wenn Sie ein SCSI-CD-ROM besitzen, sollten Sie die folgende Zeile in die Kernelkonfiguration aufnehmen und einen neuen Kernel bauen:

```
options SCSI_DELAY=15000
```

Die Zeile bewirkt, dass nach dem Zurücksetzen des SCSI-Busses beim Booten 15 Sekunden gewartet wird, um dem CD-ROM-Laufwerk genügend Zeit zu geben, darauf zu antworten.

18.6.8. Brennen von rohen CDs

Es ist möglich eine Datei auch direkt auf eine CD zu brennen, ohne vorher auf ihr ein ISO 9660-Dateisystem einzurichten. Einige Leute nutzen dies, um Datensicherungen durchzuführen. Diese Vorgehensweise hat den Vorteil, dass sie schneller als das Brennen einer normalen CD ist.

```
# burncd -f /dev/acd1 -s 12 data archive.tar.gz fixate
```

Um die Daten einer solchen CD abzurufen, müssen die Daten direkt von dem rohen Gerät gelesen werden:

```
# tar xzvf /dev/acd1
```

Eine auf diese Weise gefertigte CD kann nicht in das Dateisystem eingehangen werden. Sie kann auch nicht auf einem anderen Betriebssystem gelesen werden. Wenn Sie die erstellten CDs in das Dateisystem einhängen oder mit anderen Betriebssystemen austauschen wollen, müssen Sie mkisofs(8) wie oben beschrieben benutzen.

18.7. DVDs benutzen

Beigetragen von Marc Fonvieille.
Mit Beiträgen von Andy Polyakov.

Nach der CD ist die DVD die nächste Generation optischer Speichermedien. Auf einer DVD können mehr Daten als auf einer CD gespeichert werden. DVDs werden als Standardmedium für Videos verwendet.

Für beschreibbare DVDs existieren fünf Medienformate:

- DVD-R: Dies war das erste verfügbare Format. Das Format wurde vom DVD-Forum festgelegt. Die Medien sind nur einmal beschreibbar.
- DVD-RW: Dies ist die wiederbeschreibbare Version des DVD-R Standards. Eine DVD-RW kann ungefähr 1000 Mal beschrieben werden.
- DVD-RAM: Dies ist ein wiederbeschreibbares Format, das wie ein Wechsellaufwerk betrachtet werden kann. Allerdings sind die Medien nicht kompatibel zu den meisten DVD-ROM-Laufwerken und DVD-Video-Spielern, da das DVD-RAM-Format nur von wenigen Brennern unterstützt wird. Informationen zur Nutzung von DVD-RAM finden Sie in Abschnitt 18.7.8, „DVD-RAM“.
- DVD+RW: Ist ein wiederbeschreibbares Format, das von der DVD+RW Alliance festgelegt wurde. Eine DVD+RW kann ungefähr 1000 Mal beschrieben werden.
- DVD+R: Dieses Format ist die nur einmal beschreibbare Variante des DVD+RW Formats.

Auf einer einfach beschichteten DVD können 4.700.000.000 Bytes gespeichert werden. Das sind 4,38 GB oder 4485 MB (1 Kilobyte sind 1024 Bytes).

Anmerkung

Die physischen Medien sind unabhängig von der Anwendung. Ein DVD-Video ist eine spezielle Anordnung von Dateien, die auf irgendein Medium, beispielsweise DVD-R, DVD+R oder DVD-RW geschrieben werden kann. Bevor Sie ein Medium auswählen, müssen Sie sicherstellen, dass der Brenner und der DVD-Spieler mit dem Medium umgehen können.

18.7.1. Konfiguration

Benutzen Sie growisofs(1), um DVDs zu beschreiben. Das Kommando ist Bestandteil von sysutils/dvd+rw-tools, und kann mit allen DVD-Medien umgehen.

Diese Werkzeuge verwenden das SCSI-Subsystem, um auf die Geräte zuzugreifen. Daher muss ATAPI/CAM-Unterstützung geladen, oder statisch in den Kernel kompiliert werden. Sollte der Brenner jedoch die USB-Schnittstelle nutzen, wird diese Unterstützung nicht benötigt. Weitere Informationen zur Konfiguration von USB-Geräten finden Sie in Abschnitt 18.5, „USB Speichermedien“.

Für ATAPI-Geräte müssen ebenfalls DMA-Zugriffe aktiviert werden. Dazu wird die folgende Zeile in `/boot/loader.conf` eingefügt:

```
hw.ata.atapi_dma="1"
```

Bevor Sie dvd+rw-tools benutzen, lesen Sie bitte die Hardware-Informationen auf der Seite Hardware Compatibility Notes.

Anmerkung

Für eine grafische Oberfläche sollten Sie sich sysutils/k3b ansehen, das eine benutzerfreundliche Schnittstelle zu growisofs(1) und vielen anderen Werkzeugen bietet.

18.7.2. Daten-DVDs brennen

growisofs(1) erstellt mit dem Programm mkisofs das Dateisystem und brennt anschließend die DVD. Vor dem Brennen braucht daher kein Abbild der Daten erstellt zu werden.

Wenn Sie von den Daten im Verzeichnis `/path/to/data` eine DVD+R oder eine DVD-R brennen wollen, benutzen Sie das nachstehende Kommando:

```
# growisofs -dvd-compat -Z /dev/cd0 -J -R /path/to/data
```

In diesem Beispiel wird `-J -R` an mkisofs(8) durchgereicht und dient zum Erstellen des Dateisystems (hier: ein ISO-9660-Dateisystem mit Joliet- und Rock-Ridge-Erweiterungen). Weiteres entnehmen Sie bitte der Hilfeseite mkisofs(8).

Die Option `-Z` wird für die erste Aufnahme einer Single- oder Multisession benötigt. Ersetzen Sie */dev/cd0* mit dem Gerätenamen des DVD-Gerätes. Die Nutzung von `-dvd-compat` schließt das Medium, weitere Daten können danach nicht mehr angehängt werden. Dies sollte auch eine eine bessere Kompatibilität mit anderen DVD-ROM-Laufwerken bieten.

Um ein vorher erstelltes Abbild der Daten zu brennen, beispielsweise *imagefile.iso*, verwenden Sie:

```
# growisofs -dvd-compat -Z /dev/cd0=imagefile.iso
```

Die Schreibgeschwindigkeit hängt von den verwendeten Medium sowie dem verwendeten Gerät ab und sollte automatisch gesetzt werden. Um die Schreibgeschwindigkeit vorzugeben, verwenden Sie `-speed=`. Beispiele finden Sie in growisofs(1).

Anmerkung

Um größere Dateien als 4.38GB zu unterstützen, ist es notwendig ein UDF/ISO-9660 Hybrid-Dateisystem zu erstellen. Dieses Dateisystem muss mit zusätzlichen Parametern `-udf -iso-level 3` bei mkisofs(8) und allen relevanten Programmen, wie beispielsweise growisofs(1)) erzeugt werden. Dies ist nur notwendig, wenn Sie ein ISO-Image erstellen oder direkt auf eine DVD schreiben wollen. DVDs, die in dieser Weise hergestellt worden sind, müssen als UDF-Dateisystem mit mount_udf(8) eingehangen werden. Sie sind nur auf Betriebssystemen, die UDF unterstützen brauchbar, ansonsten sieht es so aus, als ob sie kaputte Dateien enthalten würden.

Um diese Art von ISO-Datei zu erstellen:

```
% mkisofs -R -J -udf -iso-level 3 -o imagefile.iso /path/to/data
```

Um Daten direkt auf eine DVD zu brennen, geben Sie den folgenden Befehl ein:

```
# growisofs -dvd-compat -udf -iso-level 3 -Z   /dev/cd0  -J -R /path/to/data
```

Wenn ein ISO-Abbild bereits große Dateien enthält, sind keine weiteren Optionen für growisofs(1) notwendig, um das Abbild auf die DVD zu brennen.

Achten Sie darauf, eine aktuelle Version von sysutils/cdrtools zu verwenden, welche mkisofs(8) enthält, da ältere Versionen keinen Support für große Dateien enthalten. Falls die neueste Version nicht funktioniert, installieren Sie sysutils/cdrtools-devel und lesen Sie mkisofs(8).

18.7.3. DVD-Videos brennen

Ein DVD-Video ist eine spezielle Anordnung von Dateien, die auf den ISO-9660 und den micro-UDF (M-UDF) Spezifikationen beruht. Da DVD-Video auf eine bestimmte Datei-Hierarchie angewiesen ist, müssen DVDs mit speziellen Programmen wie multimedia/dvdauthor erstellt werden.

Ist bereits ein Abbild des Dateisystems eines DVD-Videos vorhanden, kann es auf die gleiche Weise wie jedes andere Abbild gebrannt werden. Wenn `dvdauthor` verwendet wurde, um die DVD zu erstellen und die Resultate in `/path/to/video` liegen, kann das folgende Kommando verwendet werden, um ein DVD-Video zu brennen:

```
# growisofs -Z /dev/cd0 -dvd-video /path/to/video
```

`-dvd-video` wird an mkisofs(8) weitergereicht, um die Datei-Hierarchie für ein DVD-Video zu erstellen. Weiterhin bewirkt diese Option, dass growisofs(1) mit `-dvd-compat` aufgerufen wird.

18.7.4. DVD+RW-Medien benutzen

Im Gegensatz zu CD-RW-Medien müssen DVD+RW-Medien erst formatiert werden, bevor sie benutzt werden können. Es wird *empfohlen* growisofs(1) einzusetzen, da das Programm Medien automatisch formatiert, wenn es erforderlich ist. Es ist jedoch möglich, auch `dvd+rw-format` zu nutzen, um die DVD+RW zu formatieren:

```
# dvd+rw-format /dev/cd0
```

Dieser Vorgang muss nur einmal durchgeführt werden. Denken Sie daran, dass nur neue DVD+RWs formatiert werden müssen. Anschließend können DVD+RWs, wie gewohnt gebrannt werden.

Wenn Sie auf einer DVD+RW ein neues Dateisystem erstellen wollen, brauchen Sie die DVD+RW vorher nicht zu löschen. Überschreiben Sie einfach das vorige Dateisystem indem Sie eine neue Session anlegen:

```
# growisofs -Z /dev/cd0 -J -R /path/to/newdata
```

Das DVD+RW-Format erlaubt es, Daten an eine vorherige Aufnahme anzuhängen. Dazu wird eine neue Session mit der schon bestehenden zusammengeführt. Es wird keine Multi-Session geschrieben, sondern growisofs(1) *vergrößert* das ISO-9660-Dateisystem auf dem Medium.

Das folgende Kommando fügt weitere Daten zu einer vorher erstellten DVD+RW hinzu:

```
# growisofs -M /dev/cd0 -J -R /path/to/nextdata
```

Wenn Sie eine DVD+RW erweitern, verwenden Sie dieselben mkisofs(8)-Optionen wie beim Erstellen der DVD+RW.

Anmerkung

Verwenden Sie `-dvd-compat`, um bessere Kompatibilität mit DVD-ROM-Laufwerken zu gewährleisten. Zu einem DVD+RW-Medium können Sie mit dieser Option auch weiterhin Daten hinzufügen.

Um das Medium zu löschen, verwenden Sie:

```
# growisofs -Z /dev/cd0=/dev/zero
```

18.7.5. DVD-RW-Medien benutzen

Eine DVD-RW kann mit zwei Methoden beschrieben werden: *Sequential-Recording* oder *Restricted-Overwrite*. Voreingestellt ist Sequential-Recording.

Eine neue DVD-RW kann direkt beschrieben werden; sie muss nicht vorher formatiert werden. Allerdings muss eine DVD-RW, die mit Sequential-Recording aufgenommen wurde, zuerst gelöscht werden, bevor eine neue Session aufgenommen werden kann.

Der folgende Befehl löscht eine DVD-RW im Sequential-Recording-Modus:

```
# dvd+rw-format -blank=full /dev/cd0
```

Anmerkung

Das vollständige Löschen mit `-blank=full` dauert mit einem 1x Medium ungefähr eine Stunde. Wenn die DVD-RW im Disk-At-Once-Modus (DAO) aufgenommen wurde, kann sie mit `-blank` schneller gelöscht werden. Um eine DVD-RW im DAO-Modus zu brennen, benutzen Sie das folgende Kommando:

```
# growisofs -use-the-force-luke=dao -Z /dev/cd0=imagefile.iso
```

Die Option `-use-the-force-luke=dao` sollte nicht erforderlich sein, da growisofs(1) den DAO-Modus automatisch erkennt.

Der Restricted-Overwrite-Modus sollte mit jeder DVD-RW verwendet werden, da er flexibler als der voreingestellte Sequential-Recording-Modus ist.

Um Daten auf eine DVD-RW im Sequential-Recording-Modus zu schreiben, benutzen Sie dasselbe Kommando wie für die anderen DVD-Formate:

```
# growisofs -Z /dev/cd0 -J -R /path/to/data
```

Um weitere Daten zu einer Aufnahme hinzuzufügen, benutzen Sie `-M` mit growisofs(1). Werden die Daten im Sequential-Recording-Modus hinzugefügt, wird eine neue Session erstellt. Das Ergebnis ist ein Multi-Session-Medium.

Eine DVD-RW im Restricted-Overwrite-Modus muss nicht gelöscht werden, um eine neue Session aufzunehmen. Das Medium kann einfach mit `-Z` überschrieben werden. Mit `-M` kann das ISO-9660-Dateisystem, wie mit einer DVD+RW, vergrößert werden. Die DVD enthält danach eine Session.

Benutzen sie das nachstehende Kommando, um den Restricted-Overwrite-Modus einzustellen:

```
# dvd+rw-format /dev/cd0
```

Das folgende Kommando stellt den Modus wieder auf Sequential-Recording zurück:

```
# dvd+rw-format -blank=full  /dev/cd0
```

18.7.6. Multi-Session

Nur wenige DVD-ROM-Laufwerke unterstützen Multi-Session-DVDs und lesen meist nur die erste Session. Mehrere Sessions werden von DVD+R, DVD-R und DVD-RW im Sequential-Recording-Modus unterstützt. Im Modus Restricted-Overwrite gibt nur eine Session.

Wenn das Medium noch nicht geschlossen ist, erstellt das nachstehende Kommando eine neue Session auf einer DVD+R, DVD-R oder DVD-RW im Sequential-Recording-Modus:

```
# growisofs -M /dev/cd0 -J -R /path/to/nextdata
```

Wird dieses Kommando mit DVD+RW- oder DVD-RW-Medien im Restricted-Overwrite-Modus benutzt, werden die neuen Daten mit den Daten der bestehenden Session zusammengeführt. Das Medium enthält danach eine Session. Nutzen Sie diese Methode, um neue Daten zu einer bestehenden Session hinzuzufügen.

Anmerkung

Für den Anfang und das Ende einer Session wird auf dem Medium zusätzlicher Platz verbraucht. Um den Speicherplatz auf dem Medium optimal auszunutzen, sollten Sie daher Sessions mit vielen Daten hinzufügen. Auf ein DVD+R-Medium passen maximal 154 Sessions, 2000 Sessions auf ein DVD-R-Medium und 127 Sessions auf eine DVD+R Double Layer.

18.7.7. Weiterführendes

`dvd+rw-mediainfo /dev/cd0` zeigt Informationen über eine im Laufwerk liegende DVD an.

Weiteres zu dvd+rw-tools finden Sie in growisofs(1), auf der dvd+rw-tools Web-Seite und in den Archiven der cdwrite-Mailingliste.

Anmerkung

Wenn Sie einen Problembericht zur Nutzung der dvd+rw-tools erstellen, fügen Sie immer die Ausgabe von `dvd+rw-mediainfo` hinzu.

18.7.8. DVD-RAM

DVD-RAM-fähige Brenner nutzten die SCSI- oder ATAPI-Schnittstelle. Für ATAPI-Geräte muss der DMA-Modus aktiviert werden, indem die folgende Zeile in `/boot/loader.conf` hinzugefügt wird:

```
hw.ata.atapi_dma="1"
```

Eine DVD-RAM kann mit einer Wechselplatte verglichen werden. Wie diese, muss auch eine DVD-RAM vor dem ersten Einsatz formatiert werden. In diesem Beispiel wird das gesamte Medium mit dem Standard-UFS2-Dateisystem formatiert:

```
# dd if=/dev/zero of= /dev/acd0  bs=2k count=1
# bsdlabel -Bw  acd0
# newfs  /dev/acd0
```

Denken Sie dabei daran, dass Sie gegebenenfalls die Gerätedatei (hier `acd0`) an Ihre Konfiguration anpassen müssen.

Nachdem die DVD-RAM formatiert ist, kann sie wie eine normale Festplatte gemountet werden:

```
# mount /dev/acd0 /mnt
```

Danach kann schreibend und lesend auf das DVD-RAM Medium zugegriffen werden.

18.8. Disketten benutzen

Dieser Abschnitt beschreibt die Formatierung von 3,5 Zoll Disketten in FreeBSD.

Prozedur 18.3. Disketten formatieren

Bevor eine Diskette benutzt werden kann, muss sie (low-level) formatiert werden, was normalerweise der Hersteller schon gemacht hat. Sie können die Diskette allerdings noch einmal formatieren, um das Medium zu überprüfen. Benutzen Sie fdformat(1), um Disketten unter FreeBSD zu formatieren. Achten Sie dabei auf Fehlermeldungen, die schlechte Speichermedien anzeigen.

1. Um eine Diskette zu formatieren, legen Sie eine 3,5 Zoll Diskette in das erste Diskettenlaufwerk ein und führen das folgende Kommando aus:

   ```
   # /usr/sbin/fdformat -f 1440 /dev/fd0
   ```

2. Nach dem Formatieren muss auf der Diskette ein Disklabel erstellt werden, um die Größe und Geometrie der Diskette zu erkennen. Eine Liste der unterstützten Geometrien finden Sie in `/etc/disktab`.

   Erstellen Sie nun das Label mit bsdlabel(8):

   ```
   # /sbin/bsdlabel -B -w /dev/fd0 fd1440
   ```

3. Auf der Diskette kann nun ein Dateisystem erstellt werden (high-level Formatierung). Das Dateisystem der Diskette kann entweder UFS oder FAT sein, wobei FAT für Disketten in der Regel die bessere Wahl ist.

   Um die Diskette mit FAT zu formatieren, geben Sie folgendes Kommando ein:

   ```
   # /sbin/newfs_msdos /dev/fd0
   ```

Die Diskette kann nun benutzt werden. Um die Diskette zu verwenden, kann sie mit mount_msdosfs(8) eingehängt werden. Man kann auch emulators/mtools aus der Ports-Sammlung installieren, um mit der Diskette zu arbeiten.

18.9. Datensicherung

Die Planung und Umsetzung einer Backup-Strategie ist unerlässlich, um Daten in bestimmten Situationen wiederherstellen zu können, zum Beispiel bei Plattendefekten, versehentlichem Löschen von Dateien, willkürlicher Korrumpierung von Dateien oder der vollständigen Zerstörung des Systems und der Backups, die am gleichen Ort aufbewahrt werden.

Die Art und der Zeitplan des Backups kann variieren, abhängig von der Wichtigkeit der Daten, der benötigten Granularität zur Wiederherstellung von Dateien und der Dauer einer akzeptablen Ausfallzeit. Zu den möglichen Backup-Strategien gehören unter anderem:

- Die Archivierung des kompletten Systems auf externen Datenträgern. Dieser Ansatz schützt zwar vor allen oben aufgeführten Problemen, ist aber zeitaufwändig und unbequem bei der Wiederherstellung, insbesondere für nicht privilegierte Benutzer.

- Dateisystem-Snapshots sind nützlich bei der Wiederherstellung von gelöschten Dateien, bzw. früheren Versionen von Dateien.

- Kopien ganzer Dateisysteme oder Festplatten, die mit einem anderen System im Netzwerk mittels net/rsync synchronisiert werden.
- Hardware oder Software RAID, was im Falle von Plattendefekten die Ausfallzeit minimiert oder vermeidet.

Üblicherweise wird eine Mischung aus verschiedenen Strategien verwendet. Es kann zum Beispiel ein Sicherungsplan erstellt und automatisiert werden, um eine wöchentliche, vollständige Systemsicherung, ergänzt mit stündlichen ZFS-Snapshots, zu erstellen. Darüber hinaus könnte man eine manuelle Sicherung einzelner Verzeichnisse oder Dateien machen, bevor diese bearbeitet oder gelöscht werden.

Dieser Abschnitt beschreibt einige Programme, die zur Erstellung und Verwaltung von Sicherungen unter FreeBSD verwendet werden können.

18.9.1. Sicherung von Dateisystemen

Die traditionellen UNIX®-Programme zum Sichern und Wiederherstellen von Dateisystemen sind dump(8) und restore(8). Diese Programme arbeiten auf der Block-Ebene der Festplatte, also unterhalb des Abstraktionslevels von Dateien, Links und Verzeichnissen, die die Grundlage des Dateisystemkonzepts bilden. Im Gegensatz zu anderen Backup-Programmen sichert `dump` ein ganzes Dateisystem und nicht nur einen Teil des Dateisystems, oder einen Verzeichnisbaum, der mehr als ein Dateisystem umfasst. Anstatt Dateien oder Verzeichnisse zu schreiben, schreibt `dump` die Blöcke, aus denen die Dateien und Verzeichnisse bestehen.

Anmerkung

Wird `dump` benutzt, um das Root-Verzeichnis zu sichern, werden `/home`, `/usr` und viele andere Verzeichnisse nicht gesichert, da dies normalerweise Mountpunkte für andere Dateisysteme oder symbolische Links zu diesen Dateisystemen sind.

Wenn `restore` zum Extrahieren von Daten verwendet wird, werden temporäre Dateien standardmäßig in `/tmp` abgelegt. Wenn Sie von einer Platte mit einem kleinen `/tmp`-Verzeichnis zurücksichern, setzen Sie die Umgebungsvariable `TMPDIR` auf ein Verzeichnis mit mehr freiem Speicherplatz, damit die Wiederherstellung gelingt.

Beachten Sie bei der Verwendung von `dump`, dass es einige Eigenarten aus den frühen Tagen der Version 6 von AT&T UNIX® (ca. 1975) beibehalten hat. Die Standardparameter gehen davon aus, dass auf einem 9-Spur-Band gesichert wird, und nicht auf ein anderes Medium oder auf Sicherungsbänder mit hoher Dichte. Diese Standardwerte müssen auf der Kommandozeile überschrieben werden.

Es ist möglich, das Dateisystem über das Netzwerk auf einem anderen Rechner zu sichern, oder auf einem Bandlaufwerk eines anderen Rechners. Obwohl die Programme rdump(8) und rrestore(8) für diese Zwecke benutzt werden können, gelten sie als nicht sicher.

Verwenden Sie stattdessen `dump` und `restore` in einer sichereren Weise über eine SSH-Verbindung. In diesem Beispiel wird eine vollständige, komprimierte Sicherung des `/usr` Dateisystems erstellt, das anschließend an einen bestimmten Host über eine SSH-Verbindung gesendet wird.

Beispiel 18.1. `dump` mit ssh benutzen

```
# /sbin/dump -0uan -f - /usr | gzip -2 | ssh -c blowfish \
   targetuser@targetmachine.example.com dd of=/mybigfiles/dump-usr-l0.gz
```

In diesem Beispiel wird `RSH` gesetzt, um über eine SSH-Verbindung eine Sicherung auf ein Bandlaufwerk eines entfernten Systems zu schreiben:

Beispiel 18.2. `dump` über ssh mit gesetzter `RSH` benutzen

```
# env RSH=/usr/bin/ssh /sbin/dump -0uan -f tatargetuser@targetmachine.example.com:/
dev/sa0 /usr
```

18.9.2. Sicherung von Verzeichnissen

Einige integrierte Werkzeuge stehen zur Sicherung und Wiederherstellung von bestimmten Dateien und Verzeichnissen bei Bedarf zur Verfügung.

Wenn es um die Sicherung von Dateien in einem Verzeichnis geht, ist tar(1) eine gute Wahl. Dieses Werkzeug stammt aus Version 6 von AT&T UNIX® und erwartet standardmäßig eine rekursive Sicherung auf ein lokales Band. Es können jedoch Optionen angegeben werden, um den Namen einer Sicherungsdatei zu bestimmen.

In diesem Beispiel wird eine komprimierte Sicherung des aktuellen Verzeichnisses nach `/tmp/mybackup.tgz` gespeichert. Achten Sie bei der Sicherungsdatei darauf, dass sie nicht in dem Verzeichnis gepeichert wird, welches gesichert werden soll.

Beispiel 18.3. Das aktuelle Verzeichnis mit `tar` sichern

```
# tar czvf /tmp/mybackup.tgz  .
```

Um eine komplette Sicherung wiederherzustellen, wechseln Sie mit `cd` in das Verzeichnis, in dem Sie die Daten wiederherstellen möchten und geben Sie den Namen der Sicherungsdatei an. Beachten Sie, dass dabei alle Dateien in dem Verzeichnis überschrieben werden. Im Zweifel sichern Sie besser in einem temporären Verzeichnis, oder geben Sie den Verzeichnisnamen bei der Wiederherstellung an.

Beispiel 18.4. Wiederherstellung mit `tar` in das aktuelle Verzeichnis

```
# tar xzvf /tmp/mybackup.tgz
```

Es gibt dutzende Optionen, die in tar(1) beschrieben werden. Das Programm unterstützt auch die Verwendung von Ausschlußmustern, um bestimmte Dateien von der Sicherung oder Wiederherstellung von Verzeichnissen auszuschließen.

Um bestimmte, aufgelistete Dateien und Verzeichnisse zu sichern, ist cpio(1) eine gute Wahl. Im Gegensatz zu `tar` weiß `cpio` nicht wie ein Verzeichnisbaum durchlaufen wird. Daher ist es auf eine Liste von zu sichernden Dateien angewiesen.

So kann beispielsweise eine Liste von Dateien mit `ls` oder `find` erzeugt werden. Dieses Beispiel erstellt eine rekursive Liste des aktuellen Verzeichnisses, die dann über eine Pipe an `cpio` übergeben wird, um eine Sicherung namens `/tmp/mybackup.cpio` zu erstellen.

Beispiel 18.5. Rekursive Sicherung des aktuellen Verzeichnisses mit `ls` und `cpio`

```
# ls -R | cpio -ovF /tmp/mybackup.cpio
```

pax(1) ist ein Programm, welches versucht die Funktionen von `tar` und `cpio` zu kombinieren. Über die Jahre hinweg sind die verschiedenen Versionen von `tar` und `cpio` leicht inkompatibel geworden. Daher hat POSIX® `pax` geschaffen, welches versucht viele der unterschiedlichen `cpio`- und `tar`-Formate zu lesen und zu schreiben, außerdem einige neue, eigene Formate.

Für die vorangegangenen Beispiele wäre ein äquivalenter Aufruf von `pax`:

Beispiel 18.6. Das aktuelle Verzeichnis mit `pax` sichern

```
# pax -wf /tmp/mybackup.pax  .
```

18.9.3. Bandmedien benutzen

Obwohl sich Bandmedien mit der Zeit weiterentwickelt haben, verwenden moderne Backup-Systeme in der Regel *Offsite-Backups* in Verbindung mit lokalen Wechseldatenträgern. FreeBSD unterstützt alle SCSI-Bandlaufwerke, wie etwa LTO und DAT. Zusätzlich gibt es begrenzte Unterstützung für SATA- und USB-Bandlaufwerke.

Für SCSI-Bandlaufwerke nutzt FreeBSD den sa(4) Treiber, der die Schnittstellen `/dev/sa0` , `/dev/nsa0` und `/dev/esa0` bereitstellt. Der Name des physikalischen Geräts ist `/dev/sa0` . Wird `/dev/nsa0` benutzt, dann wird die Backup-Anwendung nach dem Schreibvorgang das Band nicht zurückspulen, was es ermöglicht, mehr als eine Datei auf das Band zu schreiben. Die Verwendung von `/dev/esa0` wirft das Band aus, nachdem das Gerät geschlossen wurde.

FreeBSD nutzt `mt` für die Steuerung der Operationen des Bandlaufwerks, wie die Suche nach Dateien auf einem Band, oder um Kontrollmarkierungen auf ein Band zu schreiben. Beispielsweise können die ersten drei Dateien auf einem Band erhalten bleiben, indem sie übersprungen werden, bevor eine neue Datei auf das Band geschrieben wird

```
# mt -f /dev/nsa0 fsf 3
```

Dieses Werkzeug unterstützt viele Operationen. Weitere Einzelheiten finden Sie in mt(1).

Um eine Datei mit `tar` auf ein Band zu schreiben, geben Sie den Namen des Bandlaufwerks und den Dateinamen an:

```
# tar cvf /dev/sa0 file
```

Wiederherstellung von Dateien aus dem `tar`-Archiv von Band in das aktuelle Verzeichnis:

```
# tar xvf /dev/sa0
```

Benutzen Sie `dump`, um ein UFSDateisystem zu sichern. Dieses Beispiel sichert `/usr`, ohne danach das Band zurückzuspulen:

```
# dump -0aL -b64 -f /dev/nsa0 /usr
```

Interaktive Wiederherstellung von Dateien aus einer dump(8)-Datei von Band in das aktuelle Verzeichnis:

```
# restore -i -f /dev/nsa0
```

18.9.4. Backup-Software von Drittanbietern

Die FreeBSD Ports-Sammlung enthält viele Programme von Drittanbietern, die verwendet werden können um die zeitliche Erstellung von Sicherungen zu planen, zu vereinfachen und bequemer zu machen. Viele dieser Programme basieren auf dem Client-Server-Modell und können benutzt werden, um die Sicherung von einzelnen Systemen oder allen Rechnern in einem Netzwerk zu automatisieren.

Zu den bekannten Programmen gehören Amanda, Bacula, rsync und duplicity.

18.9.5. Die Wiederherstellung in einem Notfall

Zusätzlich zu den regelmäßigen Sicherungen empfiehlt es sich, die folgenden Schritte im Rahmen eines Notfallplans durchzuführen.

Erstellen Sie einen Ausdruck der Ausgabe der folgenden Kommandos:

- `gpart show`
- `more /etc/fstab`
- `dmesg`

Bewahren Sie diesen Ausdruck und eine Kopie des Installationsmediums an einem sicheren Ort auf. Im Falle einer Wiederherstellung im Notfall, starten Sie von dem Installationsmedium und wählen Sie `Live CD`, um eine Rettungs-Shell zu starten. Dieser Rettungmodus kann verwendet werden, um den aktuellen Stand des Systems anzuzeigen, und wenn nötig, Festplatten zu formatieren und Daten aus den Sicherungen wiederherzustellen.

Anmerkung

Das Installationsmedium für FreeBSD/i386 9.3-RELEASE enthält keine Rettungs-Shell. Laden Sie für diese Version ein Abbild der Livefs CD von `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/ISO IMAGES/9.3/FreeBSD-9.3-RELEASE-i386-livefs.iso` .

Als nächstes testen Sie die Rettungs-Shell und die Sicherungen. Dokumentieren Sie diesen Ablauf. Bewahren Sie diese Notizen zusammen mit den Medien, den Ausdrucken und den Sicherungen auf. Diese Notizen können Ihnen im Notfall helfen eine versehentliche Zerstörung der Sicherungen zu verhindern, während Sie unter Stress eine Wiederherstellung durchführen.

Als zusätzliche Sicherheitsvorkehrung kann jeweils die letzte Sicherung an einem entfernten Standort aufbewahrt werden. Dieser Standort sollte räumlich von den Computern und Festplatten durch eine erhebliche Entfernung getrennt sein.

18.10. Netzwerk-, speicher- und dateibasierte Dateisysteme

Verbessert und neu strukturiert von Marc Fonvieille.

Neben physikalischen Laufwerken, wie Disketten, CDs und Festplatten, unterstützt FreeBSD auch *virtuelle Laufwerke*.

Dazu zählen Netzwerkdateisysteme wie Network Filesystem, sowie speicher- und dateibasierte Dateisysteme.

Abhängig von der verwendeten FreeBSD Version werden speicher- und dateibasierte Dateisysteme mit unterschiedlichen Werkzeugen angelegt.

Anmerkung

Gerätedateien werden unter FreeBSD automatisch von devfs(5) angelegt.

18.10.1. Dateibasierte Laufwerke unter FreeBSD

Unter FreeBSD werden virtuelle Laufwerke, md(4), mit mdconfig(8) erzeugt. Dazu muss das Modul md(4) geladen sein. Bei der Verwendung einer eigenen Kernelkonfiguration, muss diese Zeile enthalten sein:

```
device md
```

mdconfig(8) unterstützt verschiedene Arten von speicherbasierten virtuellen Laufwerken: speicherbasierte Laufwerke, deren Speicher von malloc(9) zur Verfügung gestellt wird, und dateibasierte Laufwerke, deren Speicher von einer Datei oder dem Swap-Bereich zur Verfügung gestellt wird. Eine mögliche Anwendung ist das Einhängen von CDs.

Das Abbild eines Dateisystems wird wie folgt eingehangen:

Beispiel 18.7. Einhängen eines existierenden Abbildes unter FreeBSD

```
# mdconfig -a -t vnode -f diskimage -u 0
# mount /dev/md0 /mnt
```

Ein neues Dateisystem-Abbild erstellen Sie mit mdconfig(8) wie folgt:

Beispiel 18.8. Erstellen eines dateibasierten Laufwerks mit `mdconfig`

```
# dd if=/dev/zero of=newimage bs=1k count=5k
5120+0 records in
5120+0 records out
# mdconfig -a -t vnode -f newimage -u 0
# bsdlabel -w md0 auto
# newfs md0a
/dev/md0a: 5.0MB (10224 sectors) block size 16384, fragment size 2048
        using 4 cylinder groups of 1.25MB, 80 blks, 192 inodes.
super-block backups (for fsck -b #) at:
 160, 2720, 5280, 7840
# mount /dev/md0a /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md0a       4710    4  4330     0%    /mnt
```

Wenn keine Gerätenummer mit `-u` angegeben ist, wird von md(4) automatisch eine ungenutzte Gerätenummer zugewiesen. Das zugewiesene Gerät wird auf der Standardausgabe ausgegeben (zum Beispiel `md4`). Weitere Informationen finden Sie in mdconfig(8).

Obwohl mdconfig(8) sehr nützlich ist, benötigt es einige Kommandos, um ein dateibasiertes Dateisystem zu erstellen. FreeBSD enthält auch mdmfs(8), das die notwendigen Schritte in einem Befehl zusammenfasst. Es konfiguriert mit mdconfig(8) ein md(4)-Laufwerk, erstellt darauf mit newfs(8) ein UFS-Dateisystem und hängt es anschließend mit mount(8) ein. Das virtuelle Laufwerk aus dem obigen Beispiel kann mit den nachstehenden Befehlen erstellt werden:

Beispiel 18.9. Mit `mdmfs` ein dateibasiertes Dateisystem erstellen

```
# dd if=/dev/zero of= newimage  bs=1k count= 5k
5120+0 records in
5120+0 records out
# mdmfs -F  newimage  -s 5m md0 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md0        4718    4  4338     0%    /mnt
```

Wenn `md` ohne Gerätenummer verwendet wird, wählt md(4) automatisch ein ungenutztes Gerät aus. Weitere Einzelheiten entnehmen Sie bitte der Hilfeseite mdmfs(8).

18.10.2. Speicherbasierte Laufwerke unter FreeBSD

Bei einem speicherbasierten Dateisystem sollte „swap backing“ aktiviert werden. Das heißt allerdings nicht, dass das speicherbasierte Laufwerk automatisch auf die Festplatte ausgelagert wird, vielmehr wird der Speicherplatz danach aus einem Speicherpool angefordert, der bei Bedarf auf die Platte ausgelagert werden kann. Zusätzlich ist es möglich, malloc(9)-gestützte speicherbasierte Laufwerke zu erstellen. Das Anlegen solcher Laufwerke kann allerdings zu einer System-Panic führen, wenn der Kernel danach über zu wenig Speicher verfügt.

Beispiel 18.10. Erstellen eines speicherbasierten Laufwerks mit `mdconfig`

```
# mdconfig -a -t swap -s  5m -u 1
# newfs -U md1
/dev/md1: 5.0MB (10240 sectors) block size 16384, fragment size 2048
        using 4 cylinder groups of 1.27MB, 81 blks, 192 inodes.
        with soft updates
super-block backups (for fsck -b #) at:
 160, 2752, 5344, 7936
# mount /dev/md 1 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md1        4718    4  4338     0%    /mnt
```

Beispiel 18.11. Erstellen eines speicherbasierten Laufwerks mit `mdmfs`

```
# mdmfs -s 5m md2 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
```

```
/dev/md2        4846    2  4458     0%    /mnt
```

18.10.3. Virtuelle Laufwerke freigeben

Wenn ein virtuelles Laufwerk nicht mehr in Gebrauch ist, sollten seine belegten Ressourcen an das System zurückgegeben werden. Zuerst wird das Dateisystem abgehängt und dann die benutzten Ressourcen mit mdconfig(8) freigegeben.

Alle von `/dev/md4` belegten Ressourcen werden mit dem nachstehenden Kommando freigegeben:

```
# mdconfig -d -u 4
```

Informationen zu eingerichteten md(4)-Geräten werden mit `mdconfig -l` angezeigt.

18.11. Schnappschüsse von Dateisystemen

Beigetragen von Tom Rhodes.

Zusammen mit Soft Updates bietet FreeBSD eine weitere Funktion: Schnappschüsse von Dateisystemen.

UFS-Schnappschüsse sind Dateien, die ein Abbild eines Dateisystems enthalten und müssen auf dem jeweiligen Dateisystem erstellt werden. Pro Dateisystem darf es maximal 20 Schnappschüsse, die im Superblock vermerkt werden, geben. Schnappschüsse bleiben erhalten, wenn das Dateisystem abgehangen, neu eingehangen oder das System neu gestartet wird. Wenn ein Schnappschuss nicht mehr benötigt wird, kann er mit rm(1) gelöscht werden. Es ist egal, in welcher Reihenfolge Schnappschüsse gelöscht werden. Es kann allerdings vorkommen, dass nicht der gesamte Speicherplatz wieder freigegeben wird, da ein anderer Schnappschuss einen Teil der entfernten Blöcke für sich beanspruchen kann.

Das unveränderliche `Snapshot`-Dateiflag wird nach der Erstellung des Snaphshots von mksnap_ffs(8) gesetzt. Durch die Verwendung von unlink(1) ist es allerdings möglich, einen Schnappschuss zu löschen.

Schnappschüsse werden mit mount(8) erstellt. Das folgende Kommando legt einen Schnappschuss von `/var` in `/var/snapshot/snap` ab:

```
# mount -u -o snapshot /var/snapshot/snap /var
```

Alternativ kann der Schnappschuss auch mit mksnap_ffs(8) erstellt werden.

```
# mksnap_ffs /var /var/snapshot/snap
```

Um Schnappschüsse auf einem Dateisystem, beispielsweise `/var` zu finden, kann man find(1) verwenden:

```
# find /var -flags snapshot
```

Nachdem ein Schnappschuss erstellt wurde, können Sie ihn für verschiedene Zwecke benutzen:

- Sie können den Schnappschuss für die Datensicherung benutzen und ihn auf eine CD oder ein Band schreiben.
- Die Intigrität des Schnappschusses kann mit fsck(8) geprüft werden. Wenn das Dateisystem zum Zeitpunkt der Erstellung des Schnappschusses in Ordnung war, sollte fsck(8) immer erfolgreich durchlaufen.
- Sie können den Schnappschuss mit dump(8) sichern. Sie erhalten dann eine konsistente Sicherung des Dateisystems zu dem Zeitpunkt, der durch den Zeitstempel des Schnappschusses gegeben ist. Der Schalter `-L` von dump(8) erstellt für die Sicherung einen Schnappschuss und entfernt diesen am Ende der Sicherung wieder.
- Sie können einen Schnappschuss in den Verzeichnisbaum einhängen und sich dann den Zustand des Dateisystems zu dem Zeitpunkt ansehen, an dem der Schnappschuss erstellt wurde. Der folgende Befehl hängt den Schnappschuss `/var/snapshot/snap` ein:

```
# mdconfig -a -t vnode -f /var/snapshot/snap -u 4
# mount -r /dev/md4 /mnt
```

Der eingefrorene Stand des `/var`-Dateisystems ist nun unterhalb von `/mnt` verfügbar. Mit Ausnahme der früheren Schnappschüsse, die als leere Dateien auftauchen, wird zu Beginn alles so aussehen, wie zum Zeitpunkt der Erstellung des Schnappschusses. Der Schnappschuss kann wie folgt abgehängt werden:

```
# umount /mnt
# mdconfig -d -u 4
```

Weitere Informationen über Soft Updates und Schnappschüsse von Dateisystemen sowie technische Artikel finden Sie auf der Webseite von Marshall Kirk McKusick.

18.12. Disk Quotas

Disk Quotas erlauben dem Administrator, den Plattenplatz und/oder die Anzahl der Dateien eines Benutzers oder der Mitglieder einer Gruppe, auf Dateisystemebene zu beschränken. Dadurch wird verhindert, dass ein Benutzer oder eine Gruppe von Benutzern den ganzen verfügbaren Plattenplatz belegt.

Dieser Abschnitt beschreibt die Konfiguration von Disk Quotas für UFS-Dateisysteme. Lesen Sie Abschnitt 20.4.8, „Dataset-, Benutzer- und Gruppenquotas“, wenn Sie Disk Quotas auf einem ZFS-Dateisystem einrichten möchten.

18.12.1. Disk Quotas aktivieren

Prüfen Sie zunächst, ob der FreeBSD-Kernel Disk Quotas unterstützt:

```
% sysctl kern.features.ufs_quota
kern.features.ufs_quota: 1
```

In diesem Beispiel zeigt die `1` an, das Quotas unterstützt werden. Falls `0` ausgegeben wird, fügen Sie folgende Zeile in die Kernelkonfigurationsdatei ein, und folgen Sie den Anweisungen in Kapitel 9, *Konfiguration des FreeBSD-Kernels* um den Kernel zu aktualisieren:

```
options QUOTA
```

Als nächstes aktivieren Sie Disk Quotas in `/etc/rc.conf` :

```
quota_enable="YES"
```

Normalerweise wird beim Booten die Integrität der Quotas auf allen Dateisystemen mit quotacheck(8) überprüft. Dieses Programm stellt sicher, dass die Quota-Datenbank mit den Daten auf einem Dateisystem übereinstimmt. Dies ist allerdings ein zeitraubender Prozess, der die Zeit, die das System zum Booten braucht, signifikant beeinflusst. Eine Variable in `/etc/rc.config` erlaubt es, diesen Schritt zu überspringen:

```
check_quotas="NO"
```

Zuletzt muss noch `/etc/fstab` bearbeitet werden, um die Plattenquotas auf Dateisystemebene zu aktivieren. Um Quotas pro Benutzer für ein Dateisystem zu aktivieren, geben Sie für dieses Dateisystem `userquota` im Feld Optionen von `/etc/fstab` an. Zum Beispiel:

```
/dev/da1s2g   /home    ufs rw,userquota 1 2
```

Um Quotas für Gruppen einzurichten, verwenden Sie `groupquota`. Um Quotas für Benutzer und Gruppen einzurichten, trennen Sie die Optionen durch Kommata:

```
/dev/da1s2g    /home    ufs rw,userquota,groupquota 1 2
```

Quota-Dateien werden standardmäßig im Rootverzeichnis des Dateisystems unter `quota.user` und `quota.group` abgelegt. Weitere Informationen finden Sie in fstab(5). Es wird nicht empfohlen, Quota-Dateien an anderen Stellen zu speichern.

Sobald die Konfiguration abgeschlossen ist, starten Sie das System neu. `/etc/rc` wird dann automatisch die richtigen Kommandos aufrufen, um die Quota-Dateien für alle in `/etc/rc.conf` definierten Quotas anzulegen.

Normalerweise brauchen die Kommandos quotacheck(8), quotaon(8) oder quotaoff(8) nicht händisch aufgerufen werden, obwohl man die entsprechenden Seiten im Manual lesen sollte, um sich mit ihnen vertraut zu machen.

18.12.2. Setzen von Quota-Limits

Stellen Sie sicher, dass Quotas auch tatsächlich aktiviert sind:

```
# quota -v
```

Für jedes Dateisystem, auf dem Quotas aktiviert sind, sollte eine Zeile mit der Plattenauslastung und den aktuellen Quota-Limits zu sehen sein.

Mit `edquota` können nun Quota-Limits zugewiesen werden.

Mehrere Möglichkeiten stehen zur Verfügung, um Limits für den Plattenplatz, den ein Benutzer oder eine Gruppe verbrauchen kann, oder die Anzahl der Dateien, die angelegt werden dürfen, festzulegen. Die Limits können auf dem Plattenplatz (Block-Quotas), der Anzahl der Dateien (Inode-Quotas) oder einer Kombination von beiden basieren. Jedes Limit wird weiterhin in zwei Kategorien geteilt: Hardlimits und Softlimits.

Ein Hardlimit kann nicht überschritten werden. Hat der Benutzer einmal ein Hardlimit erreicht, so kann er auf dem betreffenden Dateisystem keinen weiteren Platz mehr beanspruchen. Hat ein Benutzer beispielsweise ein Hardlimit von 500 Kilobytes auf einem Dateisystem und benutzt davon 490 Kilobyte, so kann er nur noch 10 weitere Kilobytes beanspruchen. Der Versuch, weitere 11 Kilobytes zu beanspruchen, wird fehlschlagen.

Softlimits können für eine befristete Zeit überschritten werden. Diese Frist beträgt in der Grundeinstellung eine Woche. Hat der Benutzer das Softlimit über die Frist hinaus überschritten, so wird das Softlimit in ein Hardlimit umgewandelt und der Benutzer kann keinen weiteren Platz mehr beanspruchen. Wenn er einmal das Softlimit unterschreitet, wird die Frist wieder zurückgesetzt.

Im folgenden Beispiel wird das Quota des Benutzerkonto `test` bearbeitet. Wenn `edquota` aufgerufen wird, wird der in `EDITOR` definierte Editor aufgerufen, um die Quota-Limts zu konfigurieren. Der Standard-Editor ist vi.

```
# edquota -u test

Quotas for user test:

/usr: kbytes in use: 65, limits (soft = 50, hard = 75)
        inodes in use: 7, limits (soft = 50, hard = 60)
/usr/var: kbytes in use: 0, limits (soft = 50, hard = 75)
        inodes in use: 0, limits (soft = 50, hard = 60)
```

Für jedes Dateisystem, auf dem Quotas aktiv sind, sind zwei Zeilen zu sehen. Eine repräsentiert die Block-Quotas und die andere die Inode-Quotas. Um ein Limit zu modifizieren, ändern Sie einfach den angezeigten Wert. Um beispielsweise das Blocklimit von `/usr` auf ein Softlimit von `500` und ein Hardlimit von `600` zu erhöhen, ändern Sie die Zeile wie folgt:

```
/usr: kbytes in use: 65, limits (soft = 500, hard = 600)
```

Die neuen Limits sind wirksam, sobald der Editor verlassen wird.

Manchmal ist es wünschenswert, die Limits für eine Reihe von Benutzern zu setzen. Dazu weisen Sie zunächst einem Benutzer das gewünschte Quota-Limit zu. Anschließend benutzen Sie `-p`, um das Quota auf einen bestimm-

ten Bereich von Benutzer-IDs (UID) zu duplizieren. Der folgende Befehl dupliziert die Quota-Limits auf die UIDs `10000` bis `19999`:

```
# edquota -p test 10000-19999
```

Weitere Informationen finden Sie in edquota(8).

18.12.3. Überprüfen von Quota-Limits und Plattennutzung

Um die Limits oder die Plattennutzung individueller Benutzer und Gruppen zu überprüfen, kann quota(1) benutzt werden. Ein Benutzer kann nur die eigenen Quotas und die Quotas der Gruppe, der er angehört untersuchen. Nur der Superuser darf sich alle Limits ansehen. Mit repquota(8) erhalten Sie eine Zusammenfassung von allen Limits und der Plattenausnutzung für alle Dateisysteme, auf denen Quotas aktiv sind.

In der Ausgabe von quota(1) werden Dateisysteme, auf denen ein Benutzer keinen Platz verbraucht, nicht angezeigt, auch wenn diesem Quotas zugewiesen wurden. Benutzen Sie `-v` um solche Dateisysteme ebenfalls anzuzeigen. Das folgende Beispiel zeigt die Ausgabe von `quota -v` für einen Benutzer, der Quota-Limits auf zwei Dateisystemen besitzt:

```
Disk quotas for user test (uid 1002):
     Filesystem  usage   quota   limit   grace   files   quota   limit   grace
           /usr      65*     50      75   5days       7      50      60
       /usr/var       0      50      75               0      50      60
```

Im Dateisystem `/usr` liegt der Benutzer momentan 15 Kilobytes über dem Softlimit von 50 Kilobytes und hat noch 5 Tage seiner Frist übrig. Der Stern * zeigt an, dass der Benutzer sein Limit überschritten hat.

18.12.4. Quotas über NFS

Quotas werden von dem Quota-Subsystem auf dem NFS-Server erzwungen. Der rpc.rquotad(8) Daemon stellt `quota` die Quota Informationen auf dem NFS-Client zur Verfügung, so dass Benutzer auf diesen Systemen ihre Quotas abfragen können.

Sie aktivieren `rpc.rquotad` auf dem NFS-Server, indem Sie das Zeichen `#` auf folgender Zeile in `/etc/inetd.conf` entfernen:

```
rquotad/1      dgram rpc/udp wait root /usr/libexec/rpc.rquotad rpc.rquotad
```

Anschließend starten Sie `inetd` neu:

```
# service inetd restart
```

18.13. Partitionen verschlüsseln

Beigetragen von Lucky Green.

FreeBSD bietet ausgezeichnete Möglichkeiten, Daten vor unberechtigten Zugriffen zu schützen. Wenn das Betriebssystem läuft, schützen Zugriffsrechte und vorgeschriebene Zugriffskontrollen (MAC) (siehe Kapitel 16, *Verbindliche Zugriffskontrolle*) die Daten. Die Zugriffskontrollen des Betriebssystems schützen allerdings nicht vor einem Angreifer, der Zugriff auf den Rechner hat. Der Angreifer kann eine Festplatte in ein anderes System einbauen und dort die Daten analysieren.

Die für FreeBSD verfügbaren kryptografischen Subsysteme, GEOM Based Disk Encryption (`gbde`) und `geli` sind in der Lage, Daten auf Dateisystemen auch vor hoch motivierten Angreifern zu schützen, die über erhebliche Mittel verfügen. Dieser Schutz ist unabhängig von der Art und Weise, durch die ein Angreifer Zugang zu einer Festplatte oder zu einem Rechner erlangt hat. Im Gegensatz zu schwerfälligen Systemen, die einzelne Dateien verschlüsseln,

verschlüsseln gbde und `geli` transparent ganze Dateisysteme. Auf der Festplatte werden dabei keine Daten im Klartext gespeichert.

18.13.1. Plattenverschlüsselung mit gbde

gbde(8) benutzt 128-Bit AES im CBC-Modus, um die Daten eines Sektors zu verschlüsseln. Jeder Sektor einer Festplatte wird mit einem anderen AES-Schlüssel verschlüsselt. Weitere Informationen zum kryptographischen Design und wie die Schlüssel für einen Sektor aus der gegebenen Passphrase ermittelt werden, finden Sie in gbde(4).

Anmerkung

sysinstall(8) kann nicht mit verschlüsselten gbde-Geräten umgehen. Vor dem Start von sysinstall(8) müssen alle `*.bde`-Geräte deaktiviert werden, da sysinstall(8) sonst bei der Suche nach abstürzt. Das im Beispiel verwendete Gerät wird mit dem folgenden Befehl deaktiviert:

```
# gbde detach /dev/ad4s1c
```

FreeBSD enthält ein Kernelmodul für gbde, das wie folgt geladen werden kann:

```
# kldload geom_bde
```

Wenn Sie einen angepassten Kernel verwenden, stellen Sie sicher, dass folgende Zeile in der Kernelkonfigurationsdatei enthalten ist:

```
options GEOM_BDE
```

Das folgende Beispiel beschreibt, wie eine Partition auf einer neuen Festplatte verschlüsselt wird. Die Partition wird in `/private` eingehangen. Mit gbde könnten auch `/home` und `/var/mail` verschlüsselt werden. Die dazu nötigen Schritte können allerdings in dieser Einführung nicht behandelt werden.

1. Installieren der Festplatte

   Installieren Sie die Festplatte wie in Abschnitt 18.3, „Hinzufügen von Laufwerken“ beschrieben. Im Beispiel verwenden wir die Partition `/dev/ad4s1c` . Die Gerätedateien `/dev/ad0s1` * sind Standard-Partitionen des FreeBSD-Systems.

   ```
   # ls /dev/ad*
   /dev/ad0        /dev/ad0s1b     /dev/ad0s1e     /dev/ad4s1
   /dev/ad0s1      /dev/ad0s1c     /dev/ad0s1f     /dev/ad4s1c
   /dev/ad0s1a     /dev/ad0s1d     /dev/ad4
   ```

2. Verzeichnis für gbde-Lock-Dateien anlegen

   ```
   # mkdir /etc/gbde
   ```

   Die Lock-Dateien sind für den Zugriff von gbde auf verschlüsselte Partitionen notwendig. Ohne die Lock-Dateien können die Daten nur mit erheblichem manuellen Aufwand wieder entschlüsselt werden (dies wird auch von der Software nicht unterstützt). Jede verschlüsselte Partition benötigt eine gesonderte Lock-Datei.

3. Vorbereiten der gbde-Partition

   Eine von gbde benutzte Partition muss einmalig vorbereitet werden:

   ```
   # gbde init /dev/ad4s1c -i -L /etc/gbde/ad4s1c.lock
   ```

   gbde(8) öffnet eine Vorlage im Standard-Editor, um verschiedene Optionen einstellen zu können. Setzen Sie `sector_size` auf `2048`, wenn Sie UFS1 oder UFS2 benutzen.

   ```
   # $FreeBSD: src/sbin/gbde/template.txt,v 1.1.36.1 2009/08/03 08:13:06 kensmith Exp $
   ```

```
#
# Sector size is the smallest unit of data which can be read or written.
# Making it too small decreases performance and decreases available space.
# Making it too large may prevent filesystems from working.  512 is the
# minimum and always safe.  For UFS, use the fragment size
#
sector_size     =       2048
[...-]
```

gbde(8) fragt zweimal die Passphrase zum Schutz der Daten ab. Die Passphrase muss beide Mal gleich eingegeben werden. Die Sicherheit der Daten hängt allein von der Qualität der gewählten Passphrase ab. Die Auswahl einer sicheren und leicht zu merkenden Passphrase wird auf der Webseite Diceware Passphrase beschrieben.

Mit `gbde init` wurde im Beispiel auch die Lock-Datei `/etc/gbde/ad4s1c.lock` angelegt. gbde-Lockdateien müssen die Dateiendung „.lock“ aufweisen, damit sie von `/etc/rc.d/gbde` , dem Startskript von gbde, erkannt werden.

Achtung

Sichern Sie die Lock-Dateien von gbde immer zusammen mit den verschlüsselten Dateisystemen. Ein entschlossener Angreifer kann die Daten vielleicht auch ohne die Lock-Datei entschlüsseln. Ohne die Lock-Datei können Sie allerdings nicht auf die verschlüsselten Daten zugreifen. Dies ist nur noch mit erheblichem manuellen Aufwand möglich, der nicht gbde(8) wird.

4. Einbinden der verschlüsselten Partition in den Kernel

```
# gbde attach /dev/ad4s1c -l /etc/gbde/ad4s1c.lock
```

Dieses Kommando fragt die Passphrase ab, die bei der Initialisierung der verschlüsselten Partition eingegeben wurde. Das neue verschlüsselte Gerät erscheint danach in `/dev` als `/dev/device_name.bde`:

```
# ls /dev/ad*
/dev/ad0        /dev/ad0s1b     /dev/ad0s1e     /dev/ad4s1
/dev/ad0s1      /dev/ad0s1c     /dev/ad0s1f     /dev/ad4s1c
/dev/ad0s1a     /dev/ad0s1d     /dev/ad4        /dev/ad4s1c.bde
```

5. Dateisystem auf dem verschlüsselten Gerät anlegen

Nachdem die verschlüsselte Partition im Kernel eingebunden ist, kann mit newfs(8) ein Dateisystem erstellt werden. Dieses Beispiel erstellt ein UFS2-Dateisystem mit aktivierten Soft Updates.

```
# newfs -U -O2 /dev/ad4s1c.bde
```

Anmerkung

newfs(8) muss auf einer eingebundenen gbde-Partition ausgeführt werden, welche durch das Suffix `*.bde` identifiziert wird.

6. Einhängen der verschlüsselten Partition

Legen Sie einen Mountpunkt für das verschlüsselte Dateisystem an:

```
# mkdir /private
```

Hängen Sie das verschlüsselte Dateisystem ein:

```
# mount /dev/ad4s1c.bde /private
```

7. Überprüfen des verschlüsselten Dateisystems

   Das verschlüsselte Dateisystem sollte jetzt von df(1) erkannt werden und benutzt werden können.

```
% df -H
Filesystem        Size   Used  Avail Capacity  Mounted on
/dev/ad0s1a      1037M    72M   883M     8%    /
/devfs            1.0K   1.0K     0B   100%    /dev
/dev/ad0s1f       8.1G    55K   7.5G     0%    /home
/dev/ad0s1e      1037M   1.1M   953M     0%    /tmp
/dev/ad0s1d       6.1G   1.9G   3.7G    35%    /usr
/dev/ad4s1c.bde   150G   4.1K   138G     0%    /private
```

18.13.1.1. Einhängen eines existierenden verschlüsselten Dateisystems

Nach jedem Neustart müssen verschlüsselte Dateisysteme dem Kernel wieder bekannt gemacht werden, auf Fehler überprüft werden und eingehangen werden. Die dazu nötigen Befehle müssen als `root` durchgeführt werden.

1. gbde-Partition im Kernel bekannt geben

```
# gbde attach /dev/ad4s1c -l /etc/gbde/ad4s1c.lock
```

   Dieses Kommando fragt nach der Passphrase, die während der Initialisierung der verschlüsselten gbde-Partition festgelegt wurde.

2. Prüfen des Dateisystems

   Das verschlüsselte Dateisystem kann noch nicht automatisch über `/etc/fstab` eingehangen werden. Daher muss es vor dem Einhängen mit fsck(8) geprüft werden:

```
# fsck -p -t ffs /dev/ad4s1c.bde
```

3. Einhängen des verschlüsselten Dateisystems

```
# mount /dev/ad4s1c.bde /private
```

   Das verschlüsselte Dateisystem steht danach zur Verfügung.

Mit einem Skript können verschlüsselte Dateisysteme automatisch bekannt gegeben, geprüft und eingehangen werden. Wir raten Ihnen allerdings aus Sicherheitsgründen davon ab. Starten Sie das Skript manuell an der Konsole oder in einer ssh(1)-Sitzung.

Alternativ existiert ein `rc.d`-Skript, an das über Einträge in rc.conf(5) Argumente übergeben werden können:

```
gbde_autoattach_all="YES"
gbde_devices="ad4s1c"
gbde_lockdir="/etc/gbde"
```

Durch diese Argumente muss beim Systemstart die gbde-Passphrase eingegeben werden. Erst nach Eingabe der korrekten Passphrase wird die gbde-verschlüsselte Partition automatisch in den Verzeichnisbaum eingehängt. Dieses Vorgehen ist nützlich, wenn gbde auf einem Laptop eingesetzt wird.

18.13.2. Plattenverschlüsselung mit `geli`

Beigetragen von Daniel Gerzo.

Mit `geli` ist eine alternative kryptografische GEOM-Klasse verfügbar. `geli` unterscheidet sich von `gbde` durch unterschiedliche Fähigkeiten und einen unterschiedlichen Ansatz für die Verschlüsselung.

Die wichtigsten Merkmale von geli(8) sind:

- Die Nutzung des crypto(9)-Frameworks. Wenn das System über kryptografische Hardware verfügt, wird diese von `geli` automatisch verwendet.
- Die Unterstützung verschiedener kryptografischer Algorithmen, wie beispielsweise AES, Blowfish, und 3DES.
- Die Möglichkeit, die root-Partition zu verschlüsseln. Um auf die verschlüsselte root-Partition zugreifen zu können, muss beim Systemstart die Passphrase eingegeben werden.
- `geli` erlaubt den Einsatz von zwei voneinander unabhängigen Schlüsseln, etwa einem privaten „Schlüssel“ und einem „Unternehmens-Schlüssel“.
- `geli` ist durch einfache Sektor-zu-Sektor-Verschlüsselung sehr schnell.
- Die Möglichkeit, Master-Keys zu sichern und wiederherzustellen. Wenn ein Benutzer seinen Schlüssel zerstört, kann er über seinen zuvor gesicherten Schlüssel wieder auf seine Daten zugreifen.
- `geli` erlaubt es, Platten mit einem zufälligen Einmal-Schlüssel einzusetzen, was für Swap-Partitionen und temporäre Dateisysteme interessant ist.

Weitere Merkmale von `geli` finden Sie in geli(8).

Dieser Abschnitt beschreibt, wie `geli` im FreeBSD-Kernel aktiviert wird und wie ein `geli`-Verschlüsselungs-Provider angelegt wird.

Da der Kernel angepasst werden muss, werden `root`-Privilegien benötigt.

1. Aufnahme der **`geli`**-Unterstützung in Ihre Kernelkonfigurationsdatei

   Stellen Sie bei einer angepassten Kernelkonfigurationsdatei sicher, dass diese Zeile enthalten ist:

   ```
   options GEOM_ELI
   device crypto
   ```

   Alternativ kann auch das `geli`-Kernelmodul beim Systemstart geladen werden, indem folgende Zeile in `/boot/loader.conf` eingefügt wird:

   ```
   geom_eli_load="YES"
   ```

   Ab sofort wird geli(8) vom Kernel unterstützt.

2. Erzeugen des Master-Keys

   Das folgende Beispiel beschreibt, wie eine Schlüsseldatei erzeugt wird, die als Teil des Master-Keys für den Verschlüsselungs-Provider verwendet wird, der unter `/private` in den Verzeichnisbaum eingehängt wird. Die Schlüsseldatei liefert zufällige Daten, die für die Verschlüsselung des Master-Keys benutzt werden. Zusätzlich wird der Master-Key durch eine Passphrase geschützt. Die Sektorgröße des Providers beträgt 4 KB. Das Beispiel beschreibt, wie Sie einen `geli`-Provider aktivieren, ein vom ihm verwaltetes Dateisystem erzeugen, es mounten, mit ihm arbeiten und wie Sie es schließlich wieder unmounten und den Provider deaktivieren.

   Um eine bessere Leistung zu erzielen, wird eine größere Sektorgröße, beispielsweise 4 KB, empfohlen.

   Der Master-Key wird durch eine Passphrase sowie, den Daten der Schlüsseldatei aus `/dev/random` geschützt. Die Sektorgröße des Providers `/dev/`*`da2`*`.eli` beträgt 4 KB.

   ```
   # dd if=/dev/random of=/root/da2.key bs=64 count=1
   # geli init -s 4096 -K /root/da2.key /dev/da2
   Enter new passphrase:
   Reenter new passphrase:
   ```

Es ist nicht zwingend nötig, sowohl eine Passphrase als auch eine Schlüsseldatei zu verwenden. Die einzelnen Methoden können auch unabhängig voneinander eingesetzt werden.

Wird für die Schlüsseldatei „-“ angegeben, wird dafür die Standardeingabe verwendet. Das folgende Beispiel zeigt, dass auch mehr als eine Schlüsseldatei verwendet werden kann:

```
# cat keyfile1 keyfile2 keyfile3 | geli init -K - /dev/da2
```

3. Aktivieren des Providers mit dem erzeugten Schlüssel

```
# geli attach -k /root/da2.key /dev/da2
Enter passphrase:
```

Dadurch wird die (Normaltext-)Gerätedatei `/dev/da2.eli` angelegt.

```
# ls /dev/da2*
/dev/da2  /dev/da2.eli
```

4. Das neue Dateisystem erzeugen

```
# dd if=/dev/random of=/dev/da2.eli bs=1m
# newfs /dev/da2.eli
# mount /dev/da2.eli /private
```

Das verschlüsselte Dateisystem wird nun von df(1) angezeigt und kann ab sofort eingesetzt werden.

```
# df -H
Filesystem     Size   Used  Avail Capacity  Mounted on
/dev/ad0s1a    248M    89M   139M    38%    /
/devfs         1.0K   1.0K     0B   100%    /dev
/dev/ad0s1f    7.7G   2.3G   4.9G    32%    /usr
/dev/ad0s1d    989M   1.5M   909M     0%    /tmp
/dev/ad0s1e    3.9G   1.3G   2.3G    35%    /var
/dev/da2.eli   150G   4.1K   138G     0%    /private
```

5. Das Dateisystem unmounten und den Provider deaktivieren

Wenn Sie nicht mehr mit dem verschlüsselten Dateisystem arbeiten und die unter `/private` eingehängte Partition daher nicht mehr benötigen, sollten Sie diese unmounten und den `geli`-Verschlüsselungs-Provider wieder deaktivieren:

```
# umount /private
# geli detach da2.eli
```

Weitere Informationen zum Einsatz von `geli` finden Sie in geli(8).

18.13.2.1. Der Einsatz des `geli`- `rc.d`-Skripts

`geli` verfügt über ein `rc.d`-Skript, das den Einsatz von `geli` deutlich vereinfacht. Es folgt nun ein Beispiel, in dem `geli` über die Datei rc.conf(5) konfiguriert wird:

```
geli_devices="da2"
geli_da2_flags="-k /root/da2.key"
```

Dies konfiguriert `/dev/da2` als `geli`-Provider mit dem Master-Key `/root/da2.key` . Das System wird den Provider automatisch deaktivieren, bevor es heruntergefahren wird. Während des Startvorgangs fordert das Skript die Passphrase an, bevor der Provider aktiviert wird. Vor und nach der Eingabeaufforderung für die Passphrase werden noch weitere Kernelmeldungen angezeigt. Achten Sie sorgfältig auf die Eingabeaufforderung zwischen den anderen Meldungen, falls es zu Problemen beim Startvorgang kommt. Sobald die richtige Passphrase eingegeben wurde, wird der Provider aktiviert. Anschließend werden die Dateisysteme gemäß `/etc/fstab` eingehängt. Lesen Sie Abschnitt 4.7, „Anhängen und Abhängen von Dateisystemen“ wenn Sie wissen möchten, wie Sie ein Dateisystem konfigurieren, sodass es beim booten automatisch gestartet wird.

Weitere Informationen zur Konfiguration der `rc.d`-Skripten finden Sie im Abschnitt rc.d des Handbuchs.

18.14. Den Auslagerungsspeicher verschlüsseln

Geschrieben von Christian Brüffer.

Wie die Verschlüsselung von Partitionen, wird auch der Auslagerungsspeicher verschlüsselt, um sensible Informationen zu schützen. Stellen Sie sich eine Anwendung vor, die mit Passwörtern umgeht. Solange sich diese Passwörter im Arbeitsspeicher befinden, werden sie nicht auf die Festplatte geschrieben und nach einem Neustart gelöscht. Falls FreeBSD jedoch damit beginnt Speicher auszulagern, um Platz für andere Anwendungen zu schaffen, können die Passwörter unverschlüsselt auf die Festplatte geschrieben werden. Die Verschlüsselung des Auslagerungsspeichers kann in solchen Situationen Abhilfe schaffen.

Dieser Abschnitt zeigt die Konfiguration eines verschlüsselten Auslagerungsspeichers mittels gbde(8) oder geli(8). In den Beispielen repräsentiert `/dev/ada0s1b` die Swap-Partition.

18.14.1. Konfiguration eines verschlüsselten Auslagerungsspeichers

Swap-Partitionen werden standardmäßig nicht verschlüsselt. Sie sollten daher alle sensiblen Daten im Auslagerungsspeicher löschen, bevor Sie fortfahren. Führen Sie folgenden Befehl aus, um die Swap-Partition mit Zufallsdaten zu überschreiben:

```
# dd if=/dev/random of=/dev/ ad0s1b bs=1m
```

Um den Auslagerungsspeicher mit gbde(8) zu verschlüsseln, fügen Sie in `/etc/fstab` das Suffix `.bde` an den Gerätenamen der Swap-Partition hinzu:

```
# Device                Mountpoint      FStype  Options         Dump    Pass#
/dev/ad0s1b.bde         none            swap    sw              0       0
```

Wenn Sie geli(8) benutzen, verwenden Sie stattdessen das Suffix `.eli`, um den Auslagerungsspeicher zu verschlüsseln:

```
# Device                Mountpoint      FStype  Options         Dump    Pass#
/dev/ad0s1b.eli         none            swap    sw              0       0
```

In der Voreinstellung verschlüsselt geli(8) mit dem AES-Algorithmus und einer Schlüssellänge von 128 Bit. Diese Voreinstellungen können mittels `geli_swap_flags` in `/etc/rc.conf` angepasst werden. Die folgende Zeile weist das rc.d-Skript `encswap` an, geli(8)-Swap-Partitionen mit dem Blowfish-Algorithmus und einer Schlüssellänge von 128 Bit zu verschlüsseln. Zusätzlich wird die Sektorgröße auf 4 Kilobyte gesetzt und „detach on last close“ aktiviert:

```
geli_swap_flags="-e blowfish -l 128 -s 4096 -d"
```

Eine Auflistung möglicher Optionen für `onetime` finden Sie in der Manualpage von geli(8).

18.14.2. Überprüfung des verschlüsselten Auslagerungsspeichers

Nachdem das System neu gestartet wurde, kann die korrekte Funktion des verschlüsselten Auslagerungsspeichers mit `swapinfo` geprüft werden.

Wenn Sie gbde(8) einsetzen, erhalten Sie eine Meldung ähnlich der folgenden:

```
% swapinfo
Device          1K-blocks     Used    Avail Capacity
/dev/ad0s1b.bde    542720        0   542720     0%
```

Wenn Sie geli(8) einsetzen, erhalten Sie hingegen eine Ausgabe ähnlich der folgenden:

```
% swapinfo
Device          1K-blocks     Used    Avail Capacity
```

```
/dev/ad0s1b.eli    542720        0   542720     0%
```

18.15. Highly Available Storage (HAST)

Beigetragen von Daniel Gerzo.
Mit Beiträgen von Freddie Cash, Pawel Jakub Dawidek, Michael W. Lucas und Viktor Petersson.
Übersetzt von Benedict Reuschling.

Hochverfügbarkeit ist eine der Hauptanforderungen von ernsthaften Geschäftsanwendungen und hochverfügbarer Speicher ist eine Schlüsselkomponente in solchen Umgebungen. Highly Available STorage (HAST) ist ein Framework in FreeBSD, welches die transparente Speicherung der gleichen Daten über mehrere physikalisch getrennte Maschinen ermöglicht, die über ein TCP/IP-Netzwerk verbunden sind. HAST kann als ein netzbasiertes RAID1 (Spiegel) verstanden werden und ist dem DRBD®-Speichersystem der GNU/Linux®-Plattform ähnlich. In Kombination mit anderen Hochverfügbarkeitseigenschaften von FreeBSD wie CARP, ermöglicht es HAST, hochverfügbare Speichercluster zu bauen, die in der Lage sind, Hardwareausfällen zu widerstehen.

Die Hauptmerkmale von HAST sind:

- Es kann zur Maskierung von I/O-Fehlern auf lokalen Festplatten eingesetzt werden.
- Dateisystem-unabhängig, was es erlaubt, jedes von FreeBSD unterstützte Dateisystem zu verwenden.
- Effiziente und schnelle Resynchronisation: es werden nur die Blöcke synchronisiert, die während der Ausfallzeit eines Knotens geändert wurden.
- Es kann in einer bereits bestehenden Umgebung eingesetzt werden, um zusätzliche Redundanz zu erreichen.
- Zusammen mit CARP, Heartbeat, oder anderen Werkzeugen, ist es möglich, ein robustes und dauerhaftes Speichersystem zu bauen.

Nachdem Sie diesen Abschnitt gelesen haben, werden Sie folgendes wissen:

- Was HAST ist, wie es funktioniert und welche Eigenschaften es besitzt.
- Wie man HAST unter FreeBSD aufsetzt und verwendet.
- Wie man CARP und devd(8) kombiniert, um ein robustes Speichersystem zu bauen.

Bevor Sie diesen Abschnitt lesen, sollten Sie:

- die Grundlagen von UNIX® und FreeBSD verstanden haben (Kapitel 4, *Grundlagen des UNIX Betriebssystems*).
- wissen, wie man Netzwerkschnittstellen und andere Kernsysteme von FreeBSD konfiguriert (Kapitel 12, *Konfiguration und Tuning*).
- ein gutes Verständnis der FreeBSD-Netzwerkfunktionalität besitzen (Teil IV, „Netzwerke“).

Das HAST-Projekt wurde von der FreeBSD Foundation mit Unterstützung der OMCnet Internet Service GmbH und TransIP BV gesponsert.

18.15.1. HAST im Einsatz

HAST bietet eine synchrone Replikation auf Blockebene zwischen zwei Maschinen: einem `primary`, auch bekannt als `master` Knoten, sowie dem `secondary`, oder `slave` Knoten. Diese beiden Maschinen zusammen werden als Cluster bezeichnet.

Da HAST in einer primär-sekundär-Konfiguration funktioniert, ist immer nur ein Knoten des Clusters zu jeder Zeit aktiv. Der primäre Knoten, auch *active* genannt, ist derjenige, der alle I/O-Anfragen verarbeitet, die an die HAST-Schnittstelle gesendet werden. Der sekundäre Knoten wird automatisch vom primären Knoten aus synchronisiert.

Die physischen Komponenten des HAST-Systems sind die lokale Platte am Primärknoten und die entfernte Platte am Sekundärknoten.

HAST arbeitet synchron auf Blockebene, was es für Dateisysteme und Anwendungen transparent macht. HAST stellt gewöhnliche GEOM-Provider in `/dev/hast/` für die Verwendung durch andere Werkzeuge oder Anwendungen zur Verfügung. Es gibt keinen Unterschied zwischen dem Einsatz von HAST bereitgestellten Geräten und herkömmlichen Platten oder Partitionen.

Jede Schreib-, Lösch- oder Entleerungsoperation wird an die lokale und über TCP/IP zu der entfernt liegenden Platte gesendet. Jede Leseoperation wird von der lokalen Platte durchgeführt, es sei denn, die lokale Platte ist nicht aktuell oder es tritt ein I/O-Fehler auf. In solchen Fällen wird die Leseoperation an den Sekundärknoten geschickt.

HAST versucht, eine schnelle Fehlerbereinigung zu gewährleisten. Aus diesem Grund ist es wichtig, die Synchronisationszeit nach dem Ausfall eines Knotens zu reduzieren. Um eine schnelle Synchronisation zu ermöglichen, verwaltet HAST eine Bitmap von unsauberen Bereichen auf der Platte und synchronisiert nur diese während einer regulären Synchronisation (mit Ausnahme der initialen Synchronisation).

Es gibt viele Wege, diese Synchronisation zu behandeln. HAST implementiert mehrere Replikationsarten, um unterschiedliche Methoden der Synchronisation zu realisieren:

- *memsync*: Dieser Modus meldet Schreiboperationen als vollständig, wenn die lokale Schreiboperation beendet ist und der entfernt liegende Knoten die Ankunft der Daten bestätigt hat, jedoch bevor die Daten wirklich gespeichert wurden. Die Daten werden auf dem entfernt liegenden Knoten direkt nach dem Senden der Bestätigung gespeichert. Dieser Modus ist dafür gedacht, Latenzen zu verringern und zusätzlich eine gute Verlässlichkeit zu bieten.
- *fullsync*: Dieser Modus meldet Schreiboperationen als vollständig, wenn sowohl die lokale, als auch die entfernte Schreiboperation abgeschlossen wurde. Dies ist der sicherste und zugleich der langsamste Replikationsmodus. Er stellt den momentanen Standardmodus dar.
- *async*: Dieser Modus meldet Schreiboperationen als vollständig, wenn lokale Schreibvorgänge abgeschlossen wurden. Dies ist der schnellste und gefährlichste Replikationsmodus. Er sollte nur verwendet werden, wenn die Latenz zu einem entfernten Knoten bei einer Replikation zu hoch ist für andere Modi.

18.15.2. HAST-Konfiguration

Das HAST-Framework besteht aus mehreren Komponenten:

- Dem hastd(8)-Daemon, welcher für Datensynchronisation verantwortlich ist. Wenn dieser Daemon gestartet wird, wird automatisch `geom_gate.ko` geladen.
- Dem hastctl(8) Management-Werkzeug.
- Der Konfigurationsdatei hast.conf(5). Diese Datei muss vorhanden sein, bevor hastd gestartet wird.

Alternativ lässt sich die `GEOM_GATE` -Unterstützung in den Kernel statisch einbauen, indem folgende Zeile zur Kernelkonfigurationsdatei hinzugefügt wird. Anschließend muss der Kernel, wie in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben, neu gebaut werden:

```
options GEOM_GATE
```

Das folgende Beispiel beschreibt, wie man zwei Knoten als master-slave / primary-secondary mittels HAST konfiguriert, um Daten zwischen diesen beiden auszutauschen. Die Knoten werden als `hasta` mit der IP-Adresse `172.16.0.1` und `hastb` mit der IP-Adresse `172.16.0.2` bezeichnet. Beide Knoten besitzen eine dedizierte Festplatte `/dev/ad6` mit der gleichen Größe für den HAST-Betrieb. Der HAST-Pool, manchmal auch Ressource genannt, oder der GEOM-Provider in `/dev/hast/` wird als `test` bezeichnet.

Die Konfiguration von HAST wird in `/etc/hast.conf` vorgenommen. Diese Datei sollte auf beiden Knoten gleich sein. Die einfachste Konfiguration ist folgende:

```
resource test {
 on hasta {
  local /dev/ad6
  remote 172.16.0.2
 }
 on hastb {
  local /dev/ad6
  remote 172.16.0.1
 }
}
```

Fortgeschrittene Konfigurationsmöglichkeiten finden Sie in hast.conf(5).

Tipp

Es ist ebenfalls möglich, den Hostnamen in den `remote`-Anweisungen zu verwenden, falls die Rechner aufgelöst werden können und in `/etc/hosts` , oder im lokalen DNS definiert sind.

Sobald die Konfiguration auf beiden Rechnern vorhanden ist, kann ein HAST-Pool erstellt werden. Lassen Sie diese Kommandos auf beiden Knoten ablaufen, um die initialen Metadaten auf die lokale Platte zu schreiben und starten Sie anschliessend hastd(8):

```
# hastctl create test
# service hastd onestart
```

Anmerkung

Es ist *nicht* möglich, GEOM-Provider mit einem bereits bestehenden Dateisystem zu verwenden, um beispielsweise einen bestehenden Speicher in einen von HAST verwalteten Pool zu konvertieren. Dieses Verfahren muss einige Metadaten auf den Provider schreiben und dafür würde nicht genug freier Platz zur Verfügung stehen.

Die Rolle eines HAST Knotens, `primary` oder `secondary`, wird vom einem Administrator, oder einer Software wie Heartbeat, mittels hastctl(8) festgelegt. Auf dem primären Knoten `hasta` geben Sie diesen Befehl ein:

```
# hastctl role primary test
```

Geben Sie folgendes Kommando auf dem sekundären Knoten `hastb` ein:

```
# hastctl role secondary test
```

Überprüfen Sie das Ergebnis mit `hastctl` auf beiden Knoten:

```
# hastctl status test
```

Überprüfen Sie die `status`-Zeile. Wird hier `degraded` angezeigt, dann ist etwas mit der Konfigurationsdatei nicht in Ordnung. Auf jedem Konten sollte `complete` angezeit werden, was bedeutet, dass die Synchronisation zwischen den beiden Knoten gestartet wurde. Die Synchronisierung ist abgeschlossen, wenn `hastctl status` meldet, dass die `dirty`-Bereiche 0 Bytes betragen.

Der nächste Schritt ist, ein Dateisystem auf dem GEOM-Provider anzulegen und dieses ins System einzuhängen. Dies muss auf dem `primary`-Knoten durchgeführt werden. Die Erstellung des Dateisystems kann ein paar Minuten dauern, abhängig von der Größe der Festplatte. Dieses Beispiel erstellt ein UFS-Dateisystem auf `/dev/hast/test` :

```
# newfs -U /dev/hast/ test
```

```
# mkdir /hast/ test
# mount /dev/hast/ test /hast/test
```

Sobald das HAST-Framework richtig konfiguriert wurde, besteht der letzte Schritt nun darin, sicherzustellen, dass HAST während des Systemstarts automatisch gestartet wird. Fügen Sie diese Zeile in `/etc/rc.conf` hinzu:

```
hastd_enable="YES"
```

18.15.2.1. Failover-Konfiguration

Das Ziel dieses Beispiels ist, ein robustes Speichersystem zu bauen, welches Fehlern auf einem beliebigen Knoten widerstehen kann. Wenn der `primary`-Knoten ausfällt, ist der `secondary`-Knoten da, um nahtlos einzuspringen, das Dateisystem zu prüfen, einzuhängen und mit der Arbeit fortzufahren, ohne dass auch nur ein einzelnes Bit an Daten verloren geht.

Um diese Aufgabe zu bewerkstelligen, wird das *Common Address Redundancy Protocol* (CARP) benutzt, welches ein automatisches Failover auf der IP-Schicht ermöglicht. CARP erlaubt es mehreren Rechnern im gleichen Netzsegment, die gleiche IP-Adresse zu verwenden. Setzen Sie CARP auf beiden Knoten des Clusters anhand der Dokumentation in Abschnitt 31.10, „Common Address Redundancy Protocol (CARP)“ auf. In diesem Beispiel hat jeder Knoten seine eigene Management IP-Adresse und die geteilte IP-Adresse *`172.16.0.254`* . Der primäre HAST-Knoten des Clusters muss der CARP-Masterknoten sein.

Der HAST-Pool, welcher im vorherigen Abschnitt erstellt wurde, ist nun bereit für den Export über das Netzwerk auf den anderen Rechner. Dies kann durch den Export über NFS oder Samba erreicht werden, indem die geteilte IP-Addresse *`172.16.0.254`* verwendet wird. Das einzige ungelöste Problem ist der automatische Failover, sollte der primäre Knoten einmal ausfallen.

Falls die CARP-Schnittstelle aktiviert oder deaktiviert wird, generiert das FreeBSD-Betriebssystem ein devd(8)-Ereignis, was es ermöglicht, Zustandsänderungen auf den CARP-Schnittstellen zu überwachen. Eine Zustandsänderung auf der CARP-Schnittstelle ist ein Indiz dafür, dass einer der Knoten gerade ausgefallen oder wieder verfügbar ist. Diese Zustandsänderungen machen es möglich, ein Skript zu starten, welches automatisch den HAST-Failover durchführt.

Um Zustandsänderungen auf der CARP-Schnittstelle abzufangen, müssen diese Zeilen in `/etc/devd.conf` auf jedem Knoten hinzugefügt werden:

```
notify 30 {
 match "system" "IFNET";
 match "subsystem" "carp0";
 match "type" "LINK_UP";
 action "/usr/local/sbin/carp-hast-switch master";
};

notify 30 {
 match "system" "IFNET";
 match "subsystem" "carp0";
 match "type" "LINK_DOWN";
 action "/usr/local/sbin/carp-hast-switch slave";
};
```

Anmerkung

Wenn auf dem System FreeBSD 10 oder höher eingesetzt wird, ersetzen Sie `carp0` durch den Namen der konfigurierten Schnittstelle für CARP.

Starten Sie devd(8) auf beiden Knoten neu, um die neue Konfiguration wirksam werden zu lassen:

```
# service devd restart
```

Wenn die Schnittstelle aktiviert oder deaktiviert wird, erzeugt das System eine Meldung, was es dem devd(8)-Subsystem ermöglicht, ein automatisches Failover-Skript zu starten, `/usr/local/sbin/carp-hast-switch` . Weitere Informationen zu dieser Konfiguration finden Sie in devd.conf(5).

Es folgt ein Beispiel für ein automatisches Failover-Skript:

```
#!/bin/sh

# Original script by Freddie Cash <fjwcash@gmail.com>
# Modified by Michael W. Lucas <mwlucas@BlackHelicopters.org>
# and Viktor Petersson <vpetersson@wireload.net>

# The names of the HAST resources, as listed in /etc/hast.conf
resources="test"

# delay in mounting HAST resource after becoming master
# make your best guess
delay=3

# logging
log="local0.debug"
name="carp-hast"

# end of user configurable stuff

case "$1" in
 master)
  logger -p $log -t $name "Switching to primary provider for ${resources}."
  sleep ${delay}

  # Wait for any "hastd secondary" processes to stop
  for disk in ${resources}; do
   while $( pgrep -lf "hastd: ${disk} \(secondary\)" > /dev/null 2>&1 ); do
    sleep 1
   done

   # Switch role for each disk
   hastctl role primary ${disk}
   if [ $? -ne 0 -]; then
    logger -p $log -t $name "Unable to change role to primary for resource ${disk}."
    exit 1
   fi
  done

  # Wait for the /dev/hast/* devices to appear
  for disk in ${resources}; do
   for I in $( jot 60 ); do
    [ -c "/dev/hast/${disk}" -] && break
    sleep 0.5
   done

   if [ ! -c "/dev/hast/${disk}" -]; then
    logger -p $log -t $name "GEOM provider /dev/hast/${disk} did not appear."
    exit 1
   fi
  done

  logger -p $log -t $name "Role for HAST resources ${resources} switched to primary."


  logger -p $log -t $name "Mounting disks."
  for disk in ${resources}; do
   mkdir -p /hast/${disk}
   fsck -p -y -t ufs /dev/hast/${disk}
   mount /dev/hast/${disk} /hast/${disk}
  done
```

```
 ;;

 slave)
  logger -p $log -t $name "Switching to secondary provider for ${resources}."

  # Switch roles for the HAST resources
  for disk in ${resources}; do
   if ! mount | grep -q "^/dev/hast/${disk} on "
   then
   else
    umount -f /hast/${disk}
   fi
   sleep $delay
   hastctl role secondary ${disk} 2>&1
   if [ $? -ne 0 -]; then
    logger -p $log -t $name "Unable to switch role to secondary for resource ${disk}."
    exit 1
   fi
   logger -p $log -t $name "Role switched to secondary for resource ${disk}."
  done
 ;;
esac
```

Im Kern führt das Skript die folgenden Aktionen durch, sobald ein Knoten zum Master wird:

- Es ernennt den HAST-Pool als den primären für einen gegebenen Knoten.
- Es prüft das Dateisystem, dass auf dem HAST-Pool erstellt wurde.
- Es hängt den Pool ins System ein.

Wenn ein Knoten zum Sekundären ernannt wird:

- Hängt es den HAST-Pool aus dem Dateisystem aus.
- Degradiert es den HAST-Pool zum sekundären.

Achtung

Dieses Skript ist nur ein Beispiel für eine mögliche Lösung. Es behandelt nicht alle möglichen Szenarien, die auftreten können und sollte erweitert bzw. abgeändert werden, so dass z.B. benötigte Dienste gestartet oder gestoppt werden.

Tipp

Für dieses Beispiel wurde ein UFS-Dateisystem verwendet. Um die Zeit für die Wiederherstellung zu verringern, kann ein UFS mit Journal oder ein ZFS-Dateisystem benutzt werden.

Weitere detaillierte Informationen mit zusätzlichen Beispielen können unter http://wiki.FreeBSD.org/HAST abgerufen werden.

18.15.3. Fehlerbehebung

HAST sollte generell ohne Probleme funktionieren. Jedoch kann es, wie bei jeder anderen Software auch, zu gewissen Zeiten sein, dass sie sich nicht so verhält wie angegeben. Die Quelle dieser Probleme kann unterschiedlich sein, jedoch sollte als Faustregel gewährleistet werden, dass die Zeit für alle Knoten im Cluster synchron läuft.

Für die Fehlersuche bei HAST sollte die Anzahl an Debugging-Meldungen von hastd(8) erhöht werden. Dies kann durch das Starten von `hastd` mit `-d` erreicht werden. Diese Option kann mehrfach angegeben werden, um die Anzahl an Meldungen weiter zu erhöhen. Sie sollten ebenfalls die Verwendung von `-F` in Erwägung ziehen, was `hastd` im Vordergrund startet.

18.15.3.1. Auflösung des Split-brain-Zustands

`split-brain` bezeichnet eine Situation, in der beide Knoten des Clusters nicht in der Lage sind, miteinander zu kommunizieren und dadurch beide als primäre Knoten fungieren. Dies ist ein gefährlicher Zustand, weil es beiden Knoten erlaubt ist, Änderungen an den Daten vorzunehmen, die miteinander nicht in Einklang gebracht werden können. Diese Situation muss vom Systemadministrator händisch bereinigt werden.

Der Administrator muss entscheiden, welcher Knoten die wichtigeren Änderungen besitzt, oder die Zusammenführung manuell durchführen. Anschließend kann HAST die volle Synchronisation mit dem Knoten durchführen, der die beschädigten Daten enthält. Um dies zu tun, geben Sie folgende Befehle auf dem Knoten ein, der neu synchronisiert werden muss:

```
# hastctl role init test
# hastctl create test
# hastctl role secondary test
```

Kapitel 19. GEOM: Modulares Framework zur Plattentransformation

Geschrieben von Tom Rhodes.
Übersetzt von Daniel Seuffert und Johann Kois.

19.1. Übersicht

Dieses Kapitel behandelt den Einsatz von Laufwerken mit dem GEOM-Framework in FreeBSD. Dies beinhaltet auch die wichtigen RAID-Überwachungswerkzeuge, welche das Framework zur Konfiguration nutzen. Dieses Kapitel enthält keine tiefschürfenden Betrachtungen, wie GEOM I/O nutzt oder steuert, sein zugrundeliegendes Subsystem oder den Quelltext von GEOM. Solche Informationen werden in geom(4) und seinen zahlreichen „SEE ALSO“-Verweisen bereitgestellt. Dieses Kapitel ist auch kein ausführlicher Leitfaden für RAID-Konfigurationen. Nur durch GEOM unterstützte RAID-Klassen werden erörtert.

Nach Lesen dieses Kapitels werden Sie folgendes wissen:

- Welche Art von RAID-Unterstützung durch GEOM verfügbar ist.
- Wie man die Basis-Dienstprogramme nutzt, um verschiedene RAID-Stufen zu konfigurieren, zu manipulieren und zu warten.
- Wie man mittels GEOM spiegelt, striped, verschlüsselt und entfernte Laufwerke verbindet.
- Wie man an Laufwerken, welche an das GEOM-Framework angeschlossen sind, Fehler behebt.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- Verstehen, wie FreeBSD Laufwerke behandelt (Kapitel 18, *Speichermedien*).
- Wissen wie man einen neuen FreeBSD-Kernel installiert und konfiguriert (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).

19.2. Einführung in GEOM

GEOM erlaubt den Zugriff und die Kontrolle von Klassen, wie beispielsweise Master Boot Records und BSD-Label, durch die Nutzung von Datenträgern (Providern) oder den besonderen Dateien in `/dev`. Verschiedene Software RAID-Konfigurationen unterstützend, gewährt GEOM transparenten Zugriff auf das Betriebssystem und die System-Dienstprogramme.

19.3. RAID0 - Striping

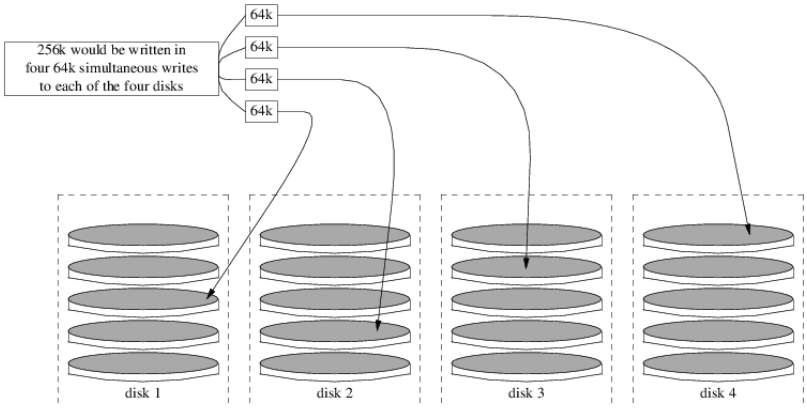
Geschrieben von Tom Rhodes und Murray Stokely.

Striping (stripe = Streifen) fasst verschiedene Laufwerke in einem einzigen Datenträger zusammen. In vielen Fällen wird dies durch die Nutzung von Hardware-Controllern bewerkstelligt. Das GEOM-Subsystem unterstützt Software-RAID0, welches auch als Striping bekannt ist.

In einem RAID0-System werden die Daten in einzelne Blöcke aufgeteilt, welche über alle angeschlossenen Laufwerke in einem Datenfeld (Array) geschrieben werden. Anstatt darauf warten zu müssen, dass 256K auf ein einzel-

nes Laufwerk geschrieben werden, kann ein RAID0-System gleichzeitig 64K auf jedes von 4 Laufwerken schreiben mit entsprechend besserer I/O-Leistung. Dieser Durchsatz kann durch die Verwendung mehrerer Controller noch zusätzlich gesteigert werden.

Jedes Laufwerk in einem RAID0-Stripe muss die gleiche Größe haben, da I/O-Anforderungen für das Lesen und Schreiben abwechselnd auf mehrere Laufwerke parallel erfolgen.



Prozedur 19.1. Erzeugen eines Stripe von unformatierten ATA-Platten

1. Laden Sie das `geom_stripe.ko`-Modul:

   ```
   # kldload geom_stripe
   ```

2. Stellen Sie sicher, dass ein geeigneter Mount-Punkt existiert. Falls dieser Datenträger eine Root-Partition werden soll, dann nutzen Sie zeitweise einen anderen Mount-Punkt, beispielsweise `/mnt`:

   ```
   # mkdir /mnt
   ```

3. Bestimmen Sie die Gerätenamen derjenigen Platten, welche gestriped werden sollen, und erzeugen Sie ein neues Stripe-Gerät. Das folgende Beispiel verwendet zwei unbenutzte und unpartitionierte ATA-Platten, die gestriped werden sollen. Die Gerätenamen lauten `/dev/ad2` und `/dev/ad3`:

   ```
   # gstripe label -v st0 /dev/ad2 /dev/ad3
   Metadata value stored on /dev/ad2.
   Metadata value stored on /dev/ad3.
   Done.
   ```

4. Schreiben Sie einen Standard-Label (auch als Partitions-Tabelle bekannt) auf den neuen Datenträger und installieren Sie den normalen Bootstrap-Code:

   ```
   # bsdlabel -wB /dev/stripe/st0
   ```

5. Dieser Prozess sollte zwei weitere Geräte im Verzeichnis `/dev/stripe` (zusätzlich zum Gerät `st0`) erzeugt haben. Diese schliessen `st0a` und `st0c` ein. Nun kann mit `newfs` ein Dateisystem auf dem Gerät `st0a` erzeugt werden:

   ```
   # newfs -U /dev/stripe/st0a
   ```

   Viele Zahlen rauschen nun über Ihren Bildschirm und nach ein paar Sekunden wird der Prozess abgeschlossen sein. Der Datenträger wurde erzeugt und kann in den Verzeichnisbaum eingehängt werden.

Um das erzeugte Stripe manuell zu mounten:

```
# mount /dev/stripe/st0a /mnt
```

Um das erzeugte Dateisystem automatisch während des Startvorgangs zu mounten, muss die Datenträgerinformation in `/etc/fstab` eingetragen werden. In diesem Beispiel wird ein permanenter Mountpunkt namens `stripe` erstellt:

```
# mkdir /stripe
# echo "/dev/stripe/st0a /stripe ufs rw 2 2" \
    >> /etc/fstab
```

Das `geom_stripe.ko`-Modul muss ebenfalls automatisch beim Systemstart geladen werden (durch die Aufnahme der folgenden Zeile in die Datei `/boot/loader.conf`):

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

19.4. RAID1 - Spiegelung

Spiegelung (RAID1 / Mirroring) ist eine Technik, bei der identische Daten auf mehr als ein Laufwerk geschrieben werden. Spiegel werden in der Regel zum Schutz vor Datenverlust aufgrund von Festplattenausfällen verwendet. Jedes Laufwerk in einem Spiegel enthält eine identische Kopie der Daten. Wenn ein einzelnes Laufwerk ausfällt, funktioniert der Spiegel weiterhin und die Daten werden von den restlichen Festplatten bereit gestellt. Der Rechner läuft einfach weiter und der Administrator hat die Gelegentheit, das defekte Laufwerk auszutauschen.

Zwei häufige Situationen werden in diesem Beispiel erläutert. Im ersten Beispiel wird ein Spiegel aus zwei neuen Laufwerken erstellt, der die existierende Platte ersetzt. Das zweite Beispiel erzeugt ein Spiegel mit einem einzigen Laufwerk, kopiert dann die Daten von der alten Platte und fügt die alte Platte zum Spiegel hinzu. Obwohl dieses Verfahren etwas komplizierter ist, wird nur ein neues Laufwerk benötigt.

Traditionell sind die Laufwerke in einem Spiegel vom gleichen Modell und besitzen die gleiche Kapazität. Dies ist jedoch keine Voraussetzung für gmirror(8). Hier können Spiegel mit unterschiedlichen Kapazitäten verwendet werden. Die Kapazität richtet sich dann nach dem kleinsten Laufwerk im Spiegel. Zusätzlicher Speicherplatz auf größeren Laufwerken bleibt dann ungenutzt. Werden später weitere Laufwerke zum Spiegel hinzugefügt, müssen diese mindestens so viel Kapazität haben wie das kleinste Laufwerk im Spiegel.

Warnung

Die hier gezeigten Verfahren löschen keine Daten. Dennoch sollte, wie bei jeder größeren Operation, zuerst eine vollständige Sicherung erstellt werden.

19.4.1. Probleme mit Metadaten

Viele Plattensysteme speichern Metadaten am Ende der Platte. Alte Metadaten sollten vor der Wiederverwendung in einem Spiegel gelöscht werden, da die meisten Probleme aus zwei Arten von übrig gebliebenen Metadaten resultieren: GPT-Partitionstabellen und alte Metadaten aus einem vorherigen Spiegel.

GPT-Metadaten können mit gpart(8) gelöscht werden. Dieses Beispiel löscht sowohl die primären, als auch die GPT-Partitionstabelle von der Festplatte `ada8`:

```
# gpart destroy -F ada8
```

Mit gmirror(8) kann eine Platte aus einem aktiven Spiegel entfernt und gleichzeitig die Metadaten gelöscht werden. In diesem Beispiel wird die Platte `ada8` aus dem aktiven Spiegel `gm4` entfernt:

```
# gmirror remove gm4 ada8
```

Wenn der Spiegel nicht aktiv ist, sich jedoch noch alte Metadaten auf der Festplatte befinden, benutzen Sie `gmirror clear`, um die Metadaten zu entfernen:

```
# gmirror clear ada8
```

gmirror(8) speichert einen Datenblock an Metadaten am Ende der Festplatte. Da das GPT-Partitionschema die Metadaten auch am Ende der Platte speichert, wird es nicht empfohlen, mit gmirror(8) einen Spiegel aus einem

gesamten GPT-Datenträger zu erstellen. In diesen Fällen sollte eine MBR-Partitionierung benutzt werden, weil hier nur eine Partitionstabelle am Anfang der Platte gespeichert wird und somit nicht mit den Metadaten des Spiegels im Konflikt steht.

19.4.2. Einen Spiegel mit zwei neuen Festplatten erstellen

In diesem Beispiel wurde FreeBSD bereits auf der vorhandenen Festplatte `ada0` installiert. Zwei neue Platten, `ada1` und `ada2`, wurden bereits mit dem System verbunden. Ein neuer Spiegel soll mit diesen beiden Platten erzeugt und verwendet werden, um die alte vorhandene Platte zu ersetzen.

Das Kernelmodul `geom_mirror.ko` muss entweder in den Kernel eingebaut, oder zur Laufzeit geladen werden. Sie können das Modul manuell laden:

```
# gmirror load
```

Erstellen Sie den Spiegel mit den beiden neuen Festplatten:

```
# gmirror label -v gm0 /dev/ada1 /dev/ada2
```

`gm0` ist ein vom Benutzer gewählter Name, der dem neuen Spiegel zugeordnet wird. Nachdem der Spiegel gestartet wurde, erscheint dieser Gerätename in `/dev/mirror/`.

MBR- und bsdlabel-Partitionstabellen können jetzt auf dem neuen Spiegel erzeugt werden. Dieses Beispiel verwendet das herkömmliche Dateisystem-Layout für `/`, swap, `/var`, `/tmp` und `/usr`. Eine einzelne Root- und Swap-Partition würde ebenfalls funktionieren.

Die Partitionen auf dem Spiegel müssen nicht zwingend die gleiche Größe wie die auf der Festplatte haben, aber sie müssen groß genug sein, um alle Daten aufnehmen zu können, die bereits auf `ada0` gespeichert sind.

```
# gpart create -s MBR mirror/gm0
# gpart add -t freebsd -a 4k mirror/gm0
# gpart show mirror/gm0
=>       63  156301423  mirror/gm0  MBR  (74G)
         63         63                    - free -  (31k)
        126  156301299                1  freebsd  (74G)
  156301425         61                    - free -  (30k)
```

```
# gpart create -s BSD mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 2g mirror/gm0s1
# gpart add -t freebsd-swap -a 4k -s 4g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 2g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k -s 1g mirror/gm0s1
# gpart add -t freebsd-ufs  -a 4k mirror/gm0s1
# gpart show mirror/gm0s1
=>        0  156301299  mirror/gm0s1  BSD  (74G)
          0          2                     - free -  (1.0k)
          2    4194304                   1  freebsd-ufs  (2.0G)
    4194306    8388608                   2  freebsd-swap (4.0G)
   12582914    4194304                   4  freebsd-ufs  (2.0G)
   16777218    2097152                   5  freebsd-ufs  (1.0G)
   18874370  137426928                   6  freebsd-ufs  (65G)
  156301298          1                     - free -  (512B)
```

Damit von dem Spiegel gebootet werden kann, muss der Bootcode in den MBR installiert, ein bsdlabel erstellt und die aktive Partition gesetzt werden:

```
# gpart bootcode -b /boot/mbr mirror/gm0
# gpart set -a active -i 1 mirror/gm0
# gpart bootcode -b /boot/boot mirror/gm0s1
```

Erstellen Sie die Dateisysteme auf dem neuen Spiegel und aktivieren Sie Soft-Updates:

```
# newfs -U /dev/mirror/gm0s1a
# newfs -U /dev/mirror/gm0s1d
```

```
# newfs -U /dev/mirror/gm0s1e
# newfs -U /dev/mirror/gm0s1f
```

Die Dateisysteme der vorhandenen Platte `ada0` können jetzt mit dump(8) und restore(8) auf den Spiegel kopiert werden.

```
# mount /dev/mirror/gm0s1a /mnt
# dump -C16 -b64 -0aL -f - / | (cd /mnt && restore -rf -)
# mount /dev/mirror/gm0s1d /mnt/var
# mount /dev/mirror/gm0s1e /mnt/tmp
# mount /dev/mirror/gm0s1f /mnt/usr
# dump -C16 -b64 -0aL -f - /var | (cd /mnt/var && restore -rf -)
# dump -C16 -b64 -0aL -f - /tmp | (cd /mnt/tmp && restore -rf -)
# dump -C16 -b64 -0aL -f - /usr | (cd /mnt/usr && restore -rf -)
```

Fügen Sie die Dateisysteme für den Spiegel in `/etc/rc.conf` hinzu:

```
# Device          Mountpoint      FStype  Options Dump    Pass#
/dev/mirror/gm0s1a      /               ufs     rw      1       1
/dev/mirror/gm0s1b      none            swap    sw      0       0
/dev/mirror/gm0s1d      /var            ufs     rw      2       2
/dev/mirror/gm0s1e      /tmp            ufs     rw      2       2
/dev/mirror/gm0s1f      /usr            ufs     rw      2       2
```

Wenn das Modul `geom_mirror.ko` nicht im Kernel enthalten ist, können Sie `/mnt/boot/loader.conf` bearbeiten, damit das Modul beim Systemstart geladen wird:

```
geom_mirror_load="YES"
```

Starten Sie das System neu und überprüfen Sie, ob alle Daten erfolgreich kopiert wurden. Das BIOS wird den Spiegel vermutlich als zwei einzelne Laufwerke erkennen. Da beide Laufwerke jedoch identisch sind, spielt es keine Rolle, welches Laufwerk zum Booten ausgewählt wird.

Falls es Probleme beim Booten gibt, lesen Sie den Abschnitt 19.4.4, „Fehlerbehebung“. Die alte Festplatte `ada0` kann vom System getrennt und als Offline-Sicherung aufbewahrt werden.

Im laufenden Betrieb verhält sich der Spiegel genau wie ein einzelnes Laufwerk.

19.4.3. Einen Spiegel mit einem vorhandenen Laufwerk erstellen

In diesem Beispiel wurde FreeBSD bereits auf der Festplatte `ada0` installiert und eine weitere Platte, `ada1`, wurde an das System angeschlossen. Zunächst wird ein Spiegel mit einer Festplatte erstellt, dann das vorhandene System auf den Spiegel kopiert. Zuletzt wird die alte Festplatte in den Spiegel eingefügt. Diese etwas komplexere Vorgehensweise ist erforderlich, da `gmirror` 512 Byte an Metadaten am Ende der Festplatte speichert, und die bestehende Platte, `ada0`, in der Regel den Platz bereits belegt hat.

Laden Sie das Kernelmodul `geom_mirror.ko`:

```
# gmirror load
```

Prüfen Sie mit `diskinfo` die Mediengröße der vorhandenen Festplatte:

```
# diskinfo -v ada0 | head -n3
/dev/ada0
        512             # sectorsize
        1000204821504   # mediasize in bytes (931G)
```

Jetzt können Sie den Spiegel auf der neuen Festplatte erzeugen. Um sicherzustellen, dass die Kapazität nicht größer ist, als die Kapazität der vorhandenen Platte `ada0`, benutzen Sie gnop(8) um eine Platte mit der exakt gleichen Größe zu imitieren. Diese Platte speichert keine Daten und wird nur verwendet, um die Größe des Spiegels zu begrenzen. gmirror(8) wird die Kapazität des Spiegels auf die Größe von `gzero.nop` beschränken, auch wenn die neue Festplatte `ada1` mehr Platz zur Verfügung hätte. Beachten Sie, dass *`1000204821504`* in der zweiten Zeile der ermittelten Mediengröße von `diskinfo` entspricht.

```
# geom zero load
# gnop create -s 1000204821504 gzero
# gmirror label -v gm0 gzero.nop ada1
# gmirror forget gm0
```

Da `gzero.nop` keine Daten speichert, sieht der Spiegel sie als nicht verbunden an. Der Spiegel ist so konfiguriert, dass er nicht verbundene Komponenten einfach „vergisst“. Das Ergebnis ist ein Spiegel mit nur einer einzigen Platte, `ada1`.

Sehen Sie sich nach der Erstellung von `gm0` die Partitionstabelle von `ada0` an. Diese Ausgabe stammt von einer 1 TB Festplatte. Falls am Ende der Platte noch freier Speicherplatz ist, kann der Inhalt von `ada0` direkt auf den Spiegel kopiert werden.

Falls jedoch der gesamte Speicherplatz auf der Platte zugeordnet ist, dann gibt es keinen Platz mehr für die 512 Byte Metadaten für den Spiegel am Ende der Platte, wie in dieser Auflistung zu sehen.

```
# gpart show ada0
=>        63  1953525105        ada0  MBR  (931G)
          63  1953525105           1  freebsd  [active]  (931G)
```

In diesem Fall muss die Partitionstabelle bearbeitet werden, um die Kapazität von `mirror/gm0` um einen Sektor zu reduzieren. Dieses Verfahren wird später erläutert.

In beiden Fällen sollte die Partitionstabelle der primären Platte mit `gpart backup` gesichert werden.

```
# gpart backup ada0 > table.ada0
# gpart backup ada0s1 > table.ada0s1
```

Diese Kommandos erstellen zwei Dateien, `table.ada0` und `table.ada0s1`. Das Beispiel verwendet eine 1 TB Festplatte:

```
# cat table.ada0
MBR 4
1 freebsd         63 1953525105   [active]
```

```
# cat table.ada0s1
BSD 8
1  freebsd-ufs          0    4194304
2 freebsd-swap    4194304   33554432
4  freebsd-ufs   37748736   50331648
5  freebsd-ufs   88080384   41943040
6  freebsd-ufs  130023424  838860800
7  freebsd-ufs  968884224  984640881
```

Wenn am Ende der Platte kein Platz vorhanden ist, muss die Größe des Slice und der letzten Partition verringert werden. Bearbeiten Sie die beiden Dateien, und verringern Sie die Größe der Slice und der Partition jeweils um eins. Dies bezieht sich auf die letzten Zahlen in der Liste.

```
# cat table.ada0
MBR 4
1 freebsd         63 1953525104   [active]
```

```
# cat table.ada0s1
BSD 8
1  freebsd-ufs          0    4194304
2 freebsd-swap    4194304   33554432
4  freebsd-ufs   37748736   50331648
5  freebsd-ufs   88080384   41943040
6  freebsd-ufs  130023424  838860800
7  freebsd-ufs  968884224  984640880
```

Wenn mindestens ein Sektor der Platte nicht zugewiesen wurde, kann die Platte ohne Modifikation verwendet werden.

Jetzt kann die Partitionstabelle auf `mirror/gm0` wiederhergestellt werden:

```
# gpart restore mirror/gm0 < table.ada0
# gpart restore mirror/gm0s1 < table.ada0s1
```

Prüfen Sie die Partitionstabellen mit `gpart show`. Dieses Beispiel nutzt `gm0s1a` für `/`, `gm0s1d` für `/var`, `gm0s1e` für `/usr`, `gm0s1f` für `/data1` und `gm0s1g` für `/data2`.

```
# gpart show mirror/gm0
=>        63  1953525104  mirror/gm0  MBR  (931G)
          63  1953525042           1  freebsd  [active]  (931G)
  1953525105          62              - free -  (31k)

# gpart show mirror/gm0s1
=>         0  1953525042  mirror/gm0s1  BSD  (931G)
           0     2097152             1  freebsd-ufs  (1.0G)
     2097152    16777216             2  freebsd-swap  (8.0G)
    18874368    41943040             4  freebsd-ufs  (20G)
    60817408    20971520             5  freebsd-ufs  (10G)
    81788928   629145600             6  freebsd-ufs  (300G)
   710934528  1242590514             7  freebsd-ufs  (592G)
  1953525042          63                - free -  (31k)
```

Sowohl die Slice, als auch die letzte Partition, muss mindestens einen freien Block am Ende der Platte haben.

Erstellen Sie Dateisysteme auf diesen neuen Partitionen:

```
# newfs -U /dev/mirror/gm0s1a
# newfs -U /dev/mirror/gm0s1d
# newfs -U /dev/mirror/gm0s1e
# newfs -U /dev/mirror/gm0s1f
# newfs -U /dev/mirror/gm0s1g
```

Damit Sie von dem Spiegel booten können, müssen Sie den Bootcode in den MBR installieren, ein bsdlabel anlegen und das aktive Slice setzen:

```
# gpart bootcode -b /boot/mbr mirror/gm0
# gpart set -a active -i 1 mirror/gm0
# gpart bootcode -b /boot/boot mirror/gm0s1
```

Bearbeiten Sie `/etc/fstab`, um die neuen Partitionen auf dem Spiegel nutzen zu können. Speichern Sie zunächst eine Kopie der Datei unter `/etc/fstab.orig`:

```
# cp /etc/fstab /etc/fstab.orig
```

Ersetzen Sie in `/etc/fstab` `/dev/ada0` durch `mirror/gm0`.

```
# Device>>>>>>>>>>Mountpoint>>>>>>FStype>>Options>Dump>>>>Pass#
/dev/mirror/gm0s1a>>>>>>/>>>>>>>>>>>>>>>ufs>>>>>rw>>>>>>1>>>>>>>1
/dev/mirror/gm0s1b>>>>>>none>>>>>>>>>>>>swap>>>>sw>>>>>>0>>>>>>>0
/dev/mirror/gm0s1d>>>>>>/var>>>>>>>>>>>>ufs>>>>>rw>>>>>>2>>>>>>>2
/dev/mirror/gm0s1e>>>>>>/usr>>>>>>>>>>>>ufs>>>>>rw>>>>>>2>>>>>>>2
/dev/mirror/gm0s1f>>>>>>/data1>>>>>>>>>>ufs>>>>>rw>>>>>>2>>>>>>>2
/dev/mirror/gm0s1g>>>>>>/data2>>>>>>>>>>ufs>>>>>rw>>>>>>2>>>>>>>2
```

Wenn das Modul `geom_mirror.ko` nicht im Kernel enthalten ist, können Sie `/boot/loader.conf` bearbeiten, damit das Modul beim Systemstart geladen wird:

```
geom_mirror_load="YES"
```

Die Dateisysteme der ursprünglichen Platte können jetzt mit dump(8) und restore(8) auf den Spiegel kopiert werden. Wenn Sie das Dateisystem mit `dump -L` sichern, wird zunächst ein Snapshot des Dateisystems erstellt, was einige Zeit dauern kann.

```
# mount /dev/mirror/gm0s1a /mnt
```

```
# dump -C16 -b64 -0aL -f - / | (cd /mnt && restore -rf -)
# mount /dev/mirror/gm0s1d /mnt/var
# mount /dev/mirror/gm0s1e /mnt/usr
# mount /dev/mirror/gm0s1f /mnt/data1
# mount /dev/mirror/gm0s1g /mnt/data2
# dump -C16 -b64 -0aL -f - /usr | (cd /mnt/usr && restore -rf -)
# dump -C16 -b64 -0aL -f - /var | (cd /mnt/var && restore -rf -)
# dump -C16 -b64 -0aL -f - /data1 | (cd /mnt/data1 && restore -rf -)
# dump -C16 -b64 -0aL -f - /data2 | (cd /mnt/data2 && restore -rf -)
```

Starten Sie das System neu und booten Sie von `ada1`. Wenn alles funktioniert, wird das System von `mirror/gm0` booten, welches jetzt die gleichen Daten enthält wie `ada0`. Lesen Sie Abschnitt 19.4.4, „Fehlerbehebung“, falls es Probleme beim Booten gibt.

An dieser Stelle besteht der Spiegel immer noch aus der einzelnen Platte `ada1`.

Nachdem erfolgreich von `mirror/gm0` gebootet wurde, besteht der letzte Schritt darin, `ada0` in den Spiegel einzufügen.

Wichtig

Wenn Sie `ada0` in den Spiegel einfügen, wird der Inhalt der Platte mit den Daten aus dem Spiegel überschrieben. Sie müssen sicherstellen, das `mirror/gm0` den gleichen Inhalt wie `ada0` hat, bevor Sie `ada0` zum Spiegel hinzufügen. Falls der zuvor mit dump(8) und restore(8) kopierte Inhalt nicht mit dem von `ada0` identisch ist, machen Sie die Änderungen an `/etc/fstab` rückgängig, starten Sie das System neu und beginnen Sie die Prozedur von vorn.

```
# gmirror insert gm0 ada0
GEOM_MIRROR: Device gm0: rebuilding provider ada0
```

Die Synchronisation zwischen den beiden Platten wird direkt gestartet. Verwenden Sie `gmirror status` um den Fortschritt zu beobachten.

```
# gmirror status
      Name    Status  Components
girror/gm0  DEGRADED  ada1 (ACTIVE)
                      ada0 (SYNCHRONIZING, 64%)
```

Nach einer Weile wird die Wiederherstellung abgeschlossen sein.

```
GEOM_MIRROR: Device gm0: rebuilding provider ada0 finished.
# gmirror status
      Name    Status  Components
mirror/gm0  COMPLETE  ada1 (ACTIVE)
                      ada0 (ACTIVE)
```

`mirror/gm0` besteht nun aus den beiden Platten `ada0` und `ada1`. Der Inhalt der beiden Platten wird automatisch miteinander synchronisiert. Im laufenden Betrieb verhält sich `mirror/gm0` wie eine einzelne Festplatte.

19.4.4. Fehlerbehebung

19.4.4.1. Probleme beim Bootvorgang

19.4.4.1.1. BIOS-Einstellungen

Möglicherweise müssen die BIOS-Einstellungen geändert werden, um von dem neuen gespiegelten Laufwerk zu booten. Beide Platten des Spiegels können zum Booten verwendet werden, da sie als Komponenten des Spiegels identische Daten enthalten.

19.4.4.1.2. Probleme beim Booten

Wenn der Bootvorgang mit der folgenden Meldung abbricht, ist irgendwas mit dem Spiegel nicht in Ordnung:

```
Mounting from ufs:/dev/mirror/gm0s1a failed with error 19.

Loader variables:
  vfs.root.mountfrom=ufs:/dev/mirror/gm0s1a
  vfs.root.mountfrom.options=rw

Manual root filesystem specification:
  <fstype>:<device> [options]
      Mount <device> using filesystem <fstype>
      and with the specified (optional) option list.

    eg. ufs:/dev/da0s1a
        zfs:tank
        cd9660:/dev/acd0 ro
          (which is equivalent to: mount -t cd9660 -o ro /dev/acd0 /)

  ?               List valid disk boot devices
  .               Yield 1 second (for background tasks)
  <empty line>    Abort manual input

mountroot>
```

Dieses Problem kann durch ein nicht geladenes Kernelmodul `geom_mirror.ko` in `/boot/loader.conf` verursacht werden. Um das Problem zu beheben, booten Sie von einem FreeBSD-Installationsmedium und wählen Sie `Shell` an der Eingabeaufforderung. Laden Sie dann das Modul und hängen Sie den Spiegel ein:

```
# gmirror load
# mount /dev/mirror/gm0s1a /mnt
```

Bearbeiten Sie dann `/mnt/boot/loader.conf` und fügen Sie eine Zeile für das Kernelmodul hinzu:

```
geom_mirror_load="YES"
```

Speichern Sie die Datei und starten Sie das System neu.

Andere Probleme, die `error 19` verursachen können, sind nur mit mehr Aufwand zu beheben. Obwohl das System von `ada0` booten sollte, wird ein weiterer Prompt erscheinen, wenn `/etc/fstab` fehlerhaft ist. Geben Sie am Loader-Prompt `ufs:/dev/ada0s1a` ein und drücken Sie Enter. Machen Sie die Änderungen an `/etc/fstab` rückgängig und hängen Sie anstelle des Spiegels die originale Festplatte (`ada0`) ein. Starten Sie dann das System neu und versuchen Sie den Vorgang erneut.

```
Enter full pathname of shell or RETURN for /bin/sh:
# cp /etc/fstab.orig /etc/fstab
# reboot
```

19.4.5. Wiederherstellung des Systems nach einem Plattenausfall

Der Vorteil der Plattenspiegelung ist, dass eine Platte ausfallen kann, ohne dass Sie dabei Daten verlieren. Falls `ada0` aus dem obigen Beispiel ausfällt, steht der Spiegel weiterhin zur Verfügung und bietet die Daten von der verbleibenden Platte `ada1` an.

Um das ausgefallene Laufwerk zu ersetzen, muss das System heruntergefahren werden und das ausgefallene Laufwerk durch ein neues Laufwerk von gleicher oder größerer Kapazität ersetzt werden. Hersteller verwenden oft etwas willkürliche Werte für die Kapazität. Der einzige Weg, um wirklich sicher zu sein, ist die Gesamtzahl der Sektoren von `diskinfo -V` zu vergleichen. Ein Laufwerk mit größerer Kapazität wird funktionieren, allerdings wird der zusätzliche Platz ungenutzt bleiben.

Nachdem der Rechner wieder eingeschaltet ist, wird der Spiegel im „degraded“ Modus ausgeführt werden. Der Spiegel wird angewiesen, Laufwerke zu vergessen, die noch nicht verbunden sind:

```
# gmirror forget gm0
```

Alte Metadaten sollten von der Ersatzfestplatte nach den Anweisungen in Abschnitt 19.4.1, „Probleme mit Metadaten“ gelöscht werden. Anschließend kann die Ersatzfestplatte, in diesem Beispiel `ada4`, in den Spiegel eingefügt werden:

```
# gmirror insert gm0 /dev/ada4
```

Die Wiederherstellung beginnt, sobald das neue Laufwerk in den Spiegel eingesetzt wird. Das Kopieren der Daten vom Spiegel auf das neue Laufwerk kann eine Weile dauern. Die Leistung des Spiegels ist während dieser Zeit stark reduziert, deswegen sollten neue Laufwerke idealerweise dann eingefügt werden, wenn der Rechner nicht benötigt wird.

Der Fortschritt der Wiederherstellung kann mit `gmirror status` überwacht werden. Während der Wiederherstellung ist der Status `DEGRADED`. Wenn der Vorgang abgeschlossen ist, wechselt der Status zu `COMPLETE`.

19.5. RAID3 - Byte-Level Striping mit dedizierter Parität

Geschrieben von Mark Gladman und Daniel Gerzo.
Basierend auf Dokumentation von Tom Rhodes und Murray Stokely.
Übersetzt von Björn Heidotting.

RAID3 ist eine Methode, die mehrere Festplatten zu einem einzigen Volume mit einer dedizierten Paritätsfestplatte kombiniert. In einem RAID3-System werden die Daten in einzelne Bytes aufgeteilt und dann über alle Laufwerke, mit Ausnahme der Paritätsfestplatte, geschrieben. Beim Lesen von Daten in einer RAID3 Implementierung werden alle Festplatten im Array parallel genutzt. Die Leistung kann durch den Einsatz von mehreren Controllern weiter erhöht werden. Ein RAID3-Array hat eine Fehlertoleranz von 1 Laufwerk und bietet dabei eine Kapazität von 1 - 1/n der Gesamtkapazität der Laufwerke im Array, wobei n die Anzahl der Festplatten im Array darstellt. So eine Konfiguration ist meistens für die Speicherung von größeren Dateien geeignet, wie beispielsweise Multimediadateien.

Mindestens 3 Festplatten sind erforderlich, um ein RAID3 zu erstellen. Jede Festplatte muss von der gleichen Größe sein, da die I/O-Anfragen für Lesen oder Schreiben auf mehreren Festplatten parallel stattfinden. Aufgrund der Beschaffenheit von RAID3, muss die Anzahl der Laufwerke 3, 5, 9, 17 bzw. 2^n + 1 sein.

Dieser Abschnitt beschreibt, wie ein Software RAID3 auf einem FreeBSD-System erstellt wird.

Anmerkung

Obwohl es theoretisch möglich ist FreeBSD von einem RAID3-Array zu booten, wird von solch einer ungewöhnlichen Konfiguration dringend abgeraten.

19.5.1. Ein dediziertes RAID3-Array erstellen

In FreeBSD wird die Unterstützung für RAID3 über die GEOM-Klasse graid3(8) implementiert. Zum Erstellen eines dedizierten RAID3-Arrays sind folgende Schritte erforderlich.

1. Laden Sie zunächst das Modul `geom_raid3.ko` mit einem der folgenden Befehle:

   ```
   # graid3 load
   ```

   oder:

   ```
   # kldload geom_raid3
   ```

2. Stellen Sie sicher, dass ein geeigneter Mountpoint existiert. Dieser Befehl erstellt ein neues Verzeichnis, welches als Mountpoint verwendet werden kann:

```
# mkdir /multimedia
```

3. Bestimmen Sie die Gerätenamen der Festplatten, die dem Array hinzugefügt werden und erstellen Sie ein neues RAID3 Gerät. Das letzte aufgeführte Gerät wird als dediziertes Paritätslaufwerk verwendet. Dieses Beispiel verwendet drei unpartionierte ATA-Platten: *ada1* und *ada2* für die Daten, sowie *ada3* für die Parität.

```
# graid3 label -v gr0 /dev/ada1 /dev/ada2 /dev/ada3
Metadata value stored on /dev/ada1.
Metadata value stored on /dev/ada2.
Metadata value stored on /dev/ada3.
Done.
```

4. Partitionieren Sie das neu erstelle Gerät `gr0` und erstellen Sie darauf ein UFS-Dateisystem:

```
# gpart create -s GPT /dev/raid3/gr0
# gpart add -t freebsd-ufs /dev/raid3/gr0
# newfs -j /dev/raid3/gr0p1
```

   Viele Zahlen rauschen nun über den Bildschirm und nach einer gewissen Zeit ist der Vorgang abgeschlossen. Das Volume wurde erstellt und kann jetzt in den Verzeichnisbaum eingehangen werden:

```
# mount /dev/raid3/gr0p1 /multimedia/
```

   Das RAID3-Array ist nun einsatzbereit.

Weitere Konfigurationsschritte sind erforderlich, um die Einstellungen nach einem Systemneustart zu erhalten.

1. Das Modul `geom_raid3.ko` muss geladen werden, bevor das Array eingehangen werden kann. Damit das Kernelmodul automatisch beim Systemstart geladen wird, muss die folgende Zeile in `/boot/loader.conf` hinzugefügt werden:

```
geom_raid3_load="YES"
```

2. Die folgenden Informationen über das Volume müssen in `/etc/fstab` hinzugefügt werden, um das Dateisystem des Arrays automatisch beim Systemstart zu aktivieren:

```
/dev/raid3/gr0p1 /multimedia ufs rw 2 2
```

19.6. GEOM Gate Netzwerkgeräte

GEOM unterstützt die Verwendung entfernter Geräte wie Festplatten, CD-ROMs und Dateien mittels Nutzung der Gate-Dienstprogramme. Dies ist vergleichbar mit NFS.

Zu Beginn muss eine Exportdatei erzeugt werden. Diese Datei legt fest, wer Zugriff auf die exportierten Ressourcen hat und welche Zugriffstechniken angeboten werden. Um zum Beispiel den vierten Slice auf der ersten SCSI-Platte zu exportieren, ist die folgende Datei `/etc/gg.exports` mehr als ausreichend:

```
192.168.1.0/24 RW /dev/da0s4d
```

Sie wird allen Hosts innerhalb des privaten Netzwerkes den Zugriff auf das Dateisystem auf der Partition `da0s4d` erlauben.

Um dieses Gerät zu exportieren, stellen Sie bitte sicher, dass es momentan nicht gemounted ist und starten Sie den ggated(8) Server-Daemon:

```
# ggated
```

Um nun `mount` auf der Client-Maschine auszuführen, geben Sie bitte die folgenden Befehle ein:

```
# ggatec create -o rw 192.168.1.1 /dev/da0s4d
ggate0
```

```
# mount /dev/ggate0 /mnt
```

Auf das Gerät kann jetzt über den Mount-Punkt `/mnt` zugegriffen werden.

Anmerkung

Es sollte darauf hingewiesen werden, dass dies scheitern wird, falls das Gerät momentan entweder auf dem Server oder irgendeiner anderen Maschine gemountet ist.

Wenn das Gerät nicht länger gebraucht wird, kann es mit umount(8) ausgehängt werden.

19.7. Das Labeln von Laufwerken

Während der Initialisierung des Systems legt der FreeBSD-Kernel für jedes gefundene Gerät Knotenpunkte an. Diese Methode für die Überprüfung auf vorhandene Geräte wirft einige Fragen auf. Was passiert beispielsweise, wenn ein neues USB-Laufwerk hinzugefügt wird? Es ist sehr wahrscheinlich, dass ein Flash-Speicher-Gerät den Gerätenamen `da0` erhält, während gleichzeitig das bisherige `da0` zu `da1` wird. Dies verursacht Probleme beim Einhängen von Dateisystemen, wenn diese in `/etc/fstab` aufgeführt sind und kann dazu führen, dass das System nicht mehr startet.

Eine Lösung für dieses Problem ist das Aneinanderketten der SCSI-Geräte, damit ein neues Gerät, welches der SCSI-Karte hinzugefügt wird, unbenutzte Gerätenummern erhält. Aber was geschieht, wenn ein USB-Gerät möglicherweise die primäre SCSI-Platte ersetzt? Dies kann passieren, weil USB-Geräte normalerweise vor der SCSI-Karte geprüft werden. Eine Lösung ist das Hinzufügen dieser Geräte, nachdem das System gestartet ist. Eine andere Lösung könnte sein, nur ein einzelnes ATA-Laufwerk zu nutzen und die SCSI-Geräte niemals in der `/etc/fstab` aufzuführen.

Eine bessere Lösung ist die Verwendung von `glabel`, um die Laufwerke zu mit Labeln zu versehen und diese in `/etc/fstab` zu nutzen. Da `glabel` seine Label im letzten Sektor jedes vorhandenen Datenträgers speichert, wird das Label persistent bleiben (auch über Neustarts hinweg). Durch Nutzung dieses Labels als Gerät kann das Dateisystem immer gemountet sein, unabhängig davon, durch welchen Geräte-Knotenpunkt auf ihn zugegriffen wird.

Anmerkung

`glabel` kann permanente (dauerhaft) und vorübergehende Label erstellen. Aber nur dauerhafte Label bleiben konsistent über Neustarts hinweg. Lesen Sie die glabel(8) für weitere Unterschiede zwischen den Label-Typen.

19.7.1. Label-Typen und Beispiele

Permanente Label können generische Label oder Dateisystem-Label sein. Permanente Dateisystem-Label können mit tunefs(8) oder newfs(8) erzeugt werden. Dieser Typ von Label wird in einem Unterverzeichnis von `/dev` angelegt und wird dem Dateisystem entsprechend benannt. UFS2-Dateisystem-Label werden zum Beispiel in `/dev/ufs` angelegt. Permanente Label können außerdem durch den Befehl `glabel label` erzeugt werden. Diese Label sind nicht dateisystemspezisch und werden im Unterverzeichnis `/dev/label` erzeugt.

Temporäre Label werden beim nächsten Systemstart zerstört. Diese Label werden im Verzeichnis `/dev/label` erzeugt und sind ideal für Testzwecke. Ein temporäres Label kann mit `glabel create` erzeugt werden.

Um ein permanentes Label auf einem UFS2-Dateisystem ohne Löschung von Daten zu erzeugen, kann man folgenden Befehl verwenden:

```
# tunefs -L home /dev/da3
```

Warnung

Wenn das Dateisystem voll ist, kann dies zu Datenkorruption führen.

In `/dev/ufs` sollte nun ein Label vorhanden sein, welches zu `/etc/fstab` hinzugefügt werden kann:

```
/dev/ufs/home   /home           ufs     rw              2       2
```

Anmerkung

Das Dateisystem darf nicht gemountet sein beim Versuch, `tunefs` auszuführen.

Nun kann das Dateisystem eingehängt werden:

```
# mount /home
```

Von nun an kann der Geräte-Knotenpunkt sich ohne negative Effekte auf das System ändern, solange das Kernelmodul `geom_label.ko` beim Systemstart mittels `/boot/loader.conf` geladen wird oder die `GEOM_LABEL`-Kernel-Option aktiv ist.

Dateisysteme können auch mit einem Standard-Label erzeugt werden (mittels des Flags `-L` in `newfs`). Lesen Sie newfs(8) für weitere Informationen.

Der folgende Befehl kann genutzt werden, um das Label zu beseitigen:

```
# glabel destroy home
```

Das folgende Beispiel zeigt Ihnen, wie Sie Label für die Partitionen einer Bootplatte erzeugen.

Beispiel 19.1. Die Partitionen einer Bootplatte labeln

Durch das Erstellen von permanenten Labeln für die Partitionen einer Bootplatte sollte das System selbst dann noch normal starten können, wenn Sie die Platte an einen anderen Controller anschließen oder in ein anderes System installieren. In diesem Beispiel nehmen wir an, dass nur eine einzige ATA-Platte verwendet wird, die Ihr System derzeit als `ad0` erkennt. Weiters nehmen wir an, dass Sie das Standard-Partionierungsschema von FreeBSD vewendet haben und Ihre Platte daher die Dateisysteme `/`, `/var`, `/usr` sowie `/tmp` aufweist. Zusätzlich wurde eine Swap-Partition angelegt.

Starten Sie das System neu. Am loader(8)-Prompt drücken Sie die Taste 4, um in den Single-User-Modus zu gelangen. Dort führen Sie die folgenden Befehle aus:

```
# glabel label rootfs /dev/ad0s1a
GEOM_LABEL: Label for provider /dev/ad0s1a is label/rootfs
# glabel label var /dev/ad0s1d
GEOM_LABEL: Label for provider /dev/ad0s1d is label/var
# glabel label usr /dev/ad0s1f
GEOM_LABEL: Label for provider /dev/ad0s1f is label/usr
# glabel label tmp /dev/ad0s1e
GEOM_LABEL: Label for provider /dev/ad0s1e is label/tmp
# glabel label swap /dev/ad0s1b
```

```
GEOM_LABEL: Label for provider /dev/ad0s1b is label/swap
# exit
```

Das System startet daraufhin in den Multi-User-Modus. Nachdem der Startvorgang abgeschlossen ist, editieren Sie `/etc/fstab` und ersetzen die konventionellen Gerätedateien durch die entsprechenden Label. Die modifizierte `/etc/fstab` sollte wie folgt aussehen:

```
# Device                Mountpoint      FStype  Options         Dump    Pass#
/dev/label/swap         none            swap    sw              0       0
/dev/label/rootfs       /               ufs     rw              1       1
/dev/label/tmp          /tmp            ufs     rw              2       2
/dev/label/usr          /usr            ufs     rw              2       2
/dev/label/var          /var            ufs     rw              2       2
```

Starten Sie Ihr System neu. Traten keine Probleme auf, wird das System normal hochfahren und Sie erhalten die folgende Ausgabe, wenn Sie den Befehl `mount` ausführen:

```
# mount
/dev/label/rootfs on / (ufs, local)
devfs on /dev (devfs, local)
/dev/label/tmp on /tmp (ufs, local, soft-updates)
/dev/label/usr on /usr (ufs, local, soft-updates)
/dev/label/var on /var (ufs, local, soft-updates)
```

Beginnend mit FreeBSD 7.2, unterstützt glabel(8) einen neuen Labeltyp für UFS-Dateisysteme. Dieser basiert auf der eindeutigen Dateisystem-ID `ufsid`. Derartige Label finden sich in `/dev/ufsid` und werden während des Systemstarts automatisch erzeugt. Es ist möglich, diese `ufsid`-Label zum automatischen Einhängen von Partitionen in `/etc/fstab` einzusetzen. Verwenden Sie `glabel status`, um eine Liste aller Dateisysteme und ihrer `ufsid`-Label zu erhalten:

```
% glabel status
                  Name  Status  Components
ufsid/486b6fc38d330916     N/A  ad4s1d
ufsid/486b6fc16926168e     N/A  ad4s1f
```

In diesem Beispiel repräsentiert `ad4s1d` das `/var`-Dateisystem, während `ad4s1f` dem `/usr`-Dateisystem entspricht. Wenn Sie die angegebenen `ufsid`-Werte verwenden, können diese Dateisysteme durch die folgenden Einträge in der Datei `/etc/fstab` gemountet werden:

```
/dev/ufsid/486b6fc38d330916        /var        ufs        rw        2       2
/dev/ufsid/486b6fc16926168e        /usr        ufs        rw        2       2
```

Jede Partition, die ein `ufsid`-Label aufweist, kann auf diese Art gemountet werden. Dies hat den Vorteil, dass Sie die permanenten Label nicht manuell anlegen müssen, wobei sich die Platten nach wie vor über geräteunabhängige Namen ansprechen und einhängen lassen.

19.8. UFS Journaling in GEOM

FreeBSD unterstützt Journaling für UFS-Dateisysteme. Diese Funktion wird über das GEOM-Subsystem realisiert und kann über das Werkzeug gjournal(8) eingerichtet werden.

Bei Journaling wird ein Protokoll über alle Dateisystemtransaktionen angelegt, inklusive aller Veränderungen, aus denen ein kompletter Schreibvorgang besteht, bevor diese Änderungen (Metadaten sowie tatsächliche Schreibvorgänge) physisch auf der Festplatte ausgeführt werden. Dieses Protokoll kann später erneut aufgerufen werden, um diese Vorgänge zu wiederholen, damit Systeminkonsistenzen vermieden werden.

Diese Technik bietet eine weitere Möglichkeit, sich vor Datenverlust und Dateisystem-Inkonsistenzen zu schützen. Im Gegensatz zu Soft Updates (die Metadaten-Aktualisierungen verfolgen und erzwingen) und Snapshots (die ein

Image eines Dateisystems darstellen) wird bei Journaling ein tatsächliches Protokoll in einem speziell dafür bereitgestellten Bereich der Festplatte (oder manchmal sogar auf einer separaten Platte) gespeichert.

Im Gegensatz zu anderen Journaling-Dateisystemen arbeitet die `gjournal`-Methode blockbasiert und wurde nicht als Teil des Dateisystems implementiert, sondern als GEOM-Erweiterung.

Um die `gjournal`-Unterstützung zu aktivieren, muss der FreeBSD-Kernel die folgende Option enthalten:

```
options UFS_GJOURNAL
```

Um ein Volume mit Journalunterstützung beim Systemstart automatisch zu mounten, muss das Kernelmodul `geom_journal.ko` ebenfalls automatisch geladen werden (durch einen entsprechenden Eintrag in der Datei `/boot/loader.conf`):

```
geom_journal_load="YES"
```

Alternativ können Sie auch einen angepassten Kernel bauen, der diese Funktionalität enthält, indem Sie die folgende Zeile in die Kernelkonfigurationsdatei aufnehmen:

```
options     GEOM_JOURNAL
```

Das Anlegen eines neuen Journals auf einem freien Dateisystem erfolgt durch die folgenden Schritte (im Folgenden wird angenommen, dass es sich bei `da4` um eine neue SCSI-Platte handelt):

```
# gjournal load
# gjournal label /dev/da4
```

Danach sollten die Gerätedateien `/dev/da4` sowie `/dev/da4.journal` vorhanden sein. Nun können Sie auf diesem Gerät ein Dateisystem anlegen:

```
# newfs -O 2 -J /dev/da4.journal
```

Dieser Befehl erzeugt ein UFS2-Dateisystem auf dem Gerät, für das im letzten Schritt das Journaling aktiviert wurde.

Danach hängen Sie das neue Dateisystem mit `mount` in den Verzeichnisbaum ein:

```
# mount /dev/da4.journal /mnt
```

Anmerkung

Falls auf Ihrem System mehrere Slices angelegt sind (beispielsweise `ad4s1` sowie `ad4s2`), wird `gjournal` für jedes Slice ein Journal anlegen (also `ad4s1.journal` sowie `ad4s2.journal`).

Um die Leistung zu optimieren, kann das Journal auf eine externe Platte ausgelagert werden. In einem solchen Fall geben Sie die Gerätedatei der Platte nach dem Gerät an, für das Sie Journaling aktivieren wollen. Theoretisch ist es auch möglich, Journaling auf bereits existierenden Dateisystemen durch das Werkzeug `tunefs` zu aktivieren. Machen Sie aber *immer* eine Sicherung der Daten, bevor Sie versuchen, ein existierendes Dateisystem zu ändern. `gjournal` wird zwar den Vorgang abbrechen, wenn es das Journal nicht erzeugen kann, allerdings schützt dies nicht vor Datenverlust durch einen fehlerhaften Einsatz von `tunefs`.

Es ist möglich, Journale auch für die Bootplatte eines FreeBSD-Systems zu verwenden. Der Artikel Implementing UFS Journaling on a Desktop PC enthält eine ausführliche Anleitung zu diesem Thema.

Kapitel 20. Das Z-Dateisystem (ZFS)

Geschrieben von Tom Rhodes, Allan Jude, Benedict Reuschling und Warren Block.
Übersetzt von Benedict Reuschling.

Das *Z-Dateisystem*, oder kurz ZFS, ist ein fortgeschrittenes Dateisystem, das entwickelt wurde, um viele der großen Probleme in vorherigen Entwicklungen zu überwinden.

Ursprünglich von Sun™ entworfen, wird die weitere Entwicklung von ZFS heutzutage als Open Source vom OpenZFS Projekt vorangetrieben.

ZFS hat drei große Entwurfsziele:

- Datenintegrität: Alle Daten enthalten eine Prüfsumme (checksum) der Daten. Wenn Daten geschrieben werden, wird die Prüfsumme berechnet und zusammen mit den Daten gespeichert. Wenn diese Daten später wieder eingelesen werden, wird diese Prüfsumme erneut berechnet. Falls die Prüfsummen nicht übereinstimmen, wurde ein Datenfehler festgestellt. ZFS wird versuchen, diesen Fehler automatisch zu korrigieren, falls genug Datenredundanz vorhanden ist.
- Gepoolter Speicher: physikalische Speichermedien werden zu einem Pool zusammengefasst und der Speicherplatz wird von diesem gemeinsam genutzten Pool allokiert. Der Speicherplatz steht allen Dateisystemen zur Verfügung und kann durch das Hinzufügen von neuen Speichermedien vergrößert werden.
- Geschwindigkeit: mehrere Zwischenspeichermechanismen sorgen für erhöhte Geschwindigkeit. Der ARC ist ein weiterentwickelter, hauptspeicherbasierter Zwischenspeicher für Leseanfragen. Auf einer zweiten Stufe kann ein plattenbasierter L2ARC-Lesezwischenspeicher hinzugefügt werden. Zusätzlich ist auch noch ein plattenbasierter, synchroner Schreibzwischenspeicher verfügbar, der sog. ZIL.

Eine vollständige Liste aller Eigenschaften und der dazugehörigen Terminologie ist in Abschnitt 20.8, „ZFS-Eigenschaften und Terminologie“ zu sehen.

20.1. Was ZFS anders macht

ZFS ist signifikant unterschiedlich zu allen bisherigen Dateisystemen, weil es mehr als nur ein Dateisystem ist. Durch die Kombination von traditionell getrennten Rollen von Volumenmanager und Dateisystem ist ZFS mit einzigartigen Vorteilen ausgestattet. Das Dateisystem besitzt jetzt Kenntnis von der zugrundeliegenden Struktur der Speichermedien. Traditionelle Dateisysteme konnten nur auf einer einzigen Platte gleichzeitig angelegt werden. Falls es zwei Festplatten gab, mussten auch zwei getrennte Dateisysteme erstellt werden. In einer traditionellen Hardware-RAID-Konfiguration wurde dieses Problem umgangen, indem dem Betriebssystem nur eine einzige logische Platte angezeigt wurde, die sich aus dem Speicherplatz von der Anzahl an physischen Platten zusammensetzte, auf dem dann das Betriebssystem ein Dateisystem erstellte. Sogar im Fall von Software-RAID-Lösungen, wie die, die von GEOM bereitgestellt werden, war das UFS-Dateisystem der Ansicht, dass es auf nur einem einzigen Gerät angelegt wurde. ZFS's Kombination eines Volumenmanagers und eines Dateisystems löst dies und erlaubt das Erstellen von vielen Dateisystemen, die sich alle den darunterliegenden Pool aus verfügbarem Speicher teilen. Einer der größten Vorteile von ZFS's Kenntnis des physikalischen Layouts der Platten ist, dass existierende Dateisysteme automatisch wachsen können, wenn zusätzliche Platten zum Pool hinzugefügt werden. Dieser neue Speicherplatz wird dann allen Dateisystemen zur Verfügung gestellt. ZFS besitzt ebenfalls eine Menge an unterschiedlichen Eigenschaften, die für jedes Dateisystem angepasst werden können, was viele Vorteile bringt, wenn man unterschiedliche Dateisysteme und Datasets anlegt, anstatt ein einziges, monolithisches Dateisystem zu erzeugen.

20.2. Schnellstartanleitung

Es existiert ein Startmechanismus, der es FreeBSD erlaubt, ZFS-Pools während der Systeminitialisierung einzubinden. Um diesen zu aktivieren, fügen Sie diese Zeile in `/etc/rc.conf` ein:

```
zfs_enable="YES"
```

Starten Sie dann den Dienst:

```
# service zfs start
```

Die Beispiele in diesem Abschnitt gehen von drei SCSI-Platten mit den Gerätenamen *da0*, *da1* und *da2* aus. Nutzer von SATA-Hardware sollten stattdessen die Bezeichnung *ada* als Gerätenamen verwenden.

20.2.1. Pools mit einer Platte

Um einen einfachen, nicht-redundanten Pool mit einem einzigen Gerät anzulegen, geben Sie folgendes ein:

```
# zpool create example /dev/da0
```

Um den neuen Pool anzuzeigen, prüfen Sie die Ausgabe von `df`:

```
# df
Filesystem  1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a   2026030  235230  1628718    13%    /
devfs               1       1        0   100%    /dev
/dev/ad0s1d  54098308 1032846 48737598     2%    /usr
example      17547136       0 17547136     0%    /example
```

Diese Ausgabe zeigt, dass der `example`-Pool erstellt und eingehängt wurde. Er ist nun als Dateisystem verfügbar. Dateien können darauf angelegt werden und Anwender können sich den Inhalt ansehen:

```
# cd /example
# ls
# touch testfile
# ls -al
total 4
drwxr-xr-x   2 root  wheel    3 Aug 29 23:15 .
drwxr-xr-x  21 root  wheel  512 Aug 29 23:12 ..
-rw-r--r--   1 root  wheel    0 Aug 29 23:15 testfile
```

Allerdings nutzt dieser Pool noch keine der Vorteile von ZFS. Um ein Dataset auf diesem Pool mit aktivierter Komprimierung zu erzeugen, geben Sie ein:

```
# zfs create example/compressed
# zfs set compression=gzip example/compressed
```

Das `example/compressed`-Dataset ist nun ein komprimiertes ZFS-Dateisystem. Versuchen Sie, ein paar große Dateien auf `/example/compressed` zu kopieren.

Deaktivieren lässt sich die Komprimierung durch:

```
# zfs set compression=off example/compressed
```

Um ein Dateisystem abzuhängen, verwenden Sie `zfs umount` und überprüfen Sie dies anschließend mit `df`:

```
# zfs umount example/compressed
# df
Filesystem  1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a   2026030  235232  1628716    13%    /
devfs               1       1        0   100%    /dev
/dev/ad0s1d  54098308 1032864 48737580     2%    /usr
example      17547008       0 17547008     0%    /example
```

Um das Dateisystem wieder einzubinden und erneut verfügbar zu machen, verwenden Sie `zfs mount` und prüfen Sie erneut mit `df`:

```
# zfs mount example/compressed
# df
Filesystem         1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a          2026030  235234  1628714    13%    /
```

```
devfs                     1       1        0   100%    /dev
/dev/ad0s1d         54098308 1032864 48737580     2%    /usr
example             17547008       0 17547008     0%    /example
example/compressed  17547008       0 17547008     0%    /example/compressed
```

Den Pool und die Dateisysteme können Sie auch über die Ausgabe von `mount` prüfen:

```
# mount
/dev/ad0s1a on / (ufs, local)
devfs on /dev (devfs, local)
/dev/ad0s1d on /usr (ufs, local, soft-updates)
example on /example (zfs, local)
example/compressed on /example/compressed (zfs, local)
```

Nach der Erstellung können ZFS-Datasets wie jedes andere Dateisystem verwendet werden. Jedoch sind jede Menge andere Besonderheiten verfügbar, die individuell auf Dataset-Basis eingestellt sein können. Im Beispiel unten wird ein neues Dateisystem namens `data` angelegt. Wichtige Dateien werden dort abgespeichert, deshalb wird es so konfiguriert, dass zwei Kopien jedes Datenblocks vorgehalten werden.

```
# zfs create example/data
# zfs set copies=2 example/data
```

Es ist jetzt möglich, den Speicherplatzverbrauch der Daten durch die Eingabe von `df` zu sehen:

```
# df
Filesystem         1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a          2026030  235234  1628714    13%    /
devfs                      1       1        0   100%    /dev
/dev/ad0s1d         54098308 1032864 48737580     2%    /usr
example             17547008       0 17547008     0%    /example
example/compressed  17547008       0 17547008     0%    /example/compressed
example/data        17547008       0 17547008     0%    /example/data
```

Sie haben vermutlich bemerkt, dass jedes Dateisystem auf dem Pool die gleiche Menge an verfügbarem Speicherplatz besitzt. Das ist der Grund dafür, dass in diesen Beispielen `df` verwendet wird, um zu zeigen, dass die Dateisysteme nur die Menge an Speicher verbrauchen, den sie benötigen und alle den gleichen Pool verwenden. ZFS eliminiert Konzepte wie Volumen und Partitionen und erlaubt es mehreren Dateisystemen den gleichen Pool zu belegen.

Um das Dateisystem und anschließend den Pool zu zerstören, wenn dieser nicht mehr benötigt wird, geben Sie ein:

```
# zfs destroy example/compressed
# zfs destroy example/data
# zpool destroy example
```

20.2.2. RAID-Z

Platten fallen aus. Eine Methode, um Datenverlust durch Festplattenausfall zu vermeiden, ist die Verwendung von RAID. ZFS unterstützt dies in seiner Poolgestaltung. Pools mit RAID-Z benötigen drei oder mehr Platten, bieten aber auch mehr nutzbaren Speicher als gespiegelte Pools.

Dieses Beispiel erstellt einen RAID-Z-Pool, indem es die Platten angibt, die dem Pool hinzugefügt werden sollen:

```
# zpool create storage raidz da0 da1 da2
```

Anmerkung

Sun™ empfiehlt, dass die Anzahl der Geräte in einer RAID-Z Konfiguration zwischen drei und neun beträgt. Für Umgebungen, die einen einzelnen Pool benötigen, der aus 10 oder mehr Platten besteht, sollten Sie in Erwägung ziehen, diesen in kleinere RAID-Z-Gruppen aufzuteilen. Falls nur zwei Platten verfügbar sind und Redundanz benötigt wird, ziehen Sie

die Verwendung eines ZFS-Spiegels (mirror) in Betracht. Lesen Sie dazu zpool(8), um weitere Details zu erhalten.

Das vorherige Beispiel erstellte einen ZPool namens `storage`. Dieses Beispiel erzeugt ein neues Dateisystem, genannt `home`, in diesem Pool:

```
# zfs create storage/home
```

Komprimierung und das Vorhalten von mehreren Kopien von Dateien und Verzeichnissen kann aktiviert werden:

```
# zfs set copies=2 storage/home
# zfs set compression=gzip storage/home
```

Um dies als das neue Heimatverzeichnis für Anwender zu setzen, kopieren Sie die Benutzerdaten in dieses Verzeichnis und erstellen passende symbolische Verknüpfungen:

```
# cp -rp /home/* /storage/home
# rm -rf /home /usr/home
# ln -s /storage/home /home
# ln -s /storage/home /usr/home
```

Daten von Anwendern werden nun auf dem frisch erstellten `/storage/home` abgelegt. Überprüfen Sie dies durch das Anlegen eines neuen Benutzers und das anschließende Anmelden als dieser Benutzer.

Versuchen Sie, einen Dateisystemschnappschuss anzulegen, den Sie später wieder zurückrollen können:

```
# zfs snapshot storage/home@08-30-08
```

Schnappschüsse können nur auf einem Dateisystem angelegt werden, nicht auf einem einzelnen Verzeichnis oder einer Datei.

Das Zeichen `@` ist der Trenner zwischen dem Dateisystem- oder dem Volumennamen. Wenn ein wichtiges Verzeichnis aus Versehen gelöscht wurde, kann das Dateisystem gesichert und dann zu einem früheren Schnappschuss zurückgerollt werden, in welchem das Verzeichnis noch existiert:

```
# zfs rollback storage/home@08-30-08
```

Um all verfügbaren Schnappschüsse aufzulisten, geben Sie `ls` im Verzeichnis `.zfs/snapshot` dieses Dateisystems ein. Beispielsweise lässt sich der zuvor angelegte Schnappschuss wie folgt anzeigen:

```
# ls /storage/home/.zfs/snapshot
```

Es ist möglich, ein Skript zu schreiben, um regelmäßig Schnappschüsse von Benutzerdaten anzufertigen. Allerdings verbrauchen Schnappschüsse über lange Zeit eine große Menge an Speicherplatz. Der zuvor angelegte Schnappschuss kann durch folgendes Kommando wieder entfernt werden:

```
# zfs destroy storage/home@08-30-08
```

Nach erfolgreichen Tests kann `/storage/home` zum echten `/home`-Verzeichnis werden, mittels:

```
# zfs set mountpoint=/home storage/home
```

Prüfen Sie mit `df` und `mount`, um zu bestätigen, dass das System das Dateisystem nun als `/home` verwendet:

```
# mount
/dev/ad0s1a on / (ufs, local)
devfs on /dev (devfs, local)
/dev/ad0s1d on /usr (ufs, local, soft-updates)
storage on /storage (zfs, local)
storage/home on /home (zfs, local)
# df
Filesystem   1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a    2026030  235240  1628708    13%    /
```

```
devfs                1       1        0   100%    /dev
/dev/ad0s1d   54098308 1032826 48737618     2%    /usr
storage       26320512       0 26320512     0%    /storage
storage/home  26320512       0 26320512     0%    /home
```

Damit ist die RAID-Z Konfiguration abgeschlossen. Tägliche Informationen über den Status der erstellten Dateisysteme können als Teil des nächtlichen periodic(8)-Berichts generiert werden. Fügen Sie dazu die folgende Zeile in `/etc/periodic.conf` ein:

```
daily_status_zfs_enable="YES"
```

20.2.3. RAID-Z wiederherstellen

Jedes Software-RAID besitzt eine Methode, um den Zustand (`state`) zu überprüfen. Der Status von RAID-Z Geräten wird mit diesem Befehl angezeigt:

```
# zpool status -x
```

Wenn alle Pools Online sind und alles normal ist, zeigt die Meldung folgendes an:

```
all pools are healthy
```

Wenn es ein Problem gibt, womöglich ist eine Platte im Zustand Offline, dann wird der Poolzustand ähnlich wie dieser aussehen:

```
  pool: storage
 state: DEGRADED
status: One or more devices has been taken offline by the administrator.
 Sufficient replicas exist for the pool to continue functioning in a
 degraded state.
action: Online the device using 'zpool online' or replace the device with
 'zpool replace'.
 scrub: none requested
config:

 NAME        STATE     READ WRITE CKSUM
 storage     DEGRADED     0     0     0
   raidz1    DEGRADED     0     0     0
     da0     ONLINE       0     0     0
     da1     OFFLINE      0     0     0
     da2     ONLINE       0     0     0

errors: No known data errors
```

Dies zeigt an, dass das Gerät zuvor vom Administrator mit diesem Befehl abgeschaltet wurde:

```
# zpool offline storage da1
```

Jetzt kann das System heruntergefahren werden, um `da1` zu ersetzen. Wenn das System wieder eingeschaltet wird, kann die fehlerhafte Platte im Pool ersetzt werden:

```
# zpool replace storage da1
```

Von diesem Punkt an kann der Status erneut geprüft werden. Dieses Mal ohne die Option `-x`, damit alle Pools angezeigt werden:

```
# zpool status storage
 pool: storage
 state: ONLINE
 scrub: resilver completed with 0 errors on Sat Aug 30 19:44:11 2008
config:

 NAME        STATE     READ WRITE CKSUM
 storage     ONLINE       0     0     0
   raidz1    ONLINE       0     0     0
```

```
     da0     ONLINE       0     0     0
     da1     ONLINE       0     0     0
     da2     ONLINE       0     0     0

errors: No known data errors
```

In diesem Beispiel ist alles normal.

20.2.4. Daten verifizieren

ZFS verwendet Prüfsummen, um die Integrität der gespeicherten Daten zu gewährleisten. Dies wird automatisch beim Erstellen von Dateisystemen aktiviert.

Warnung

Prüfsummen können deaktiviert werden, dies wird jedoch *nicht* empfohlen! Prüfsummen verbrauchen nur sehr wenig Speicherplatz und sichern die Integrität der Daten. Viele Eigenschaften vom ZFS werden nicht richtig funktionieren, wenn Prüfsummen deaktiviert sind. Es gibt keinen merklichen Geschwindigkeitsunterschied durch das Deaktivieren dieser Prüfsummen.

Prüfsummenverifikation ist unter der Bezeichnung *scrubbing* bekannt. Verifizieren Sie die Integrität der Daten des `storage`-Pools mit diesem Befehl:

```
# zpool scrub storage
```

Die Laufzeit einer Überprüfung hängt ab von der Menge an Daten, die gespeichert sind. Größere Mengen an Daten benötigen proportional mehr Zeit zum überprüfen. Diese Überprüfungen sind sehr I/O-intensiv und es kann auch nur eine Überprüfung zur gleichen Zeit durchgeführt werden. Nachdem eine Prüfung beendet ist, kann der Status mit dem Unterkommando `status` angezeigt werden:

```
# zpool status storage
 pool: storage
 state: ONLINE
 scrub: scrub completed with 0 errors on Sat Jan 26 19:57:37 2013
config:

 NAME        STATE     READ WRITE CKSUM
 storage     ONLINE       0     0     0
   raidz1    ONLINE       0     0     0
     da0     ONLINE       0     0     0
     da1     ONLINE       0     0     0
     da2     ONLINE       0     0     0

errors: No known data errors
```

Das Datum der letzten Prüfoperation wird angezeigt, um zu verfolgen, wann die nächste Prüfung benötigt wird. Routinemässige Überprüfungen helfen dabei, Daten vor stiller Korrumpierung zu schützen und die Integrität des Pools sicher zu stellen.

Lesen Sie zfs(8) und zpool(8), um über weitere ZFS-Optionen zu erfahren.

20.3. `zpool` Administration

Administration von ZFS ist unterteilt zwischen zwei Hauptkommandos. Das `zpool` -Werkzeug steuert die Operationen des Pools und kümmert sich um das Hinzufügen, entfernen, ersetzen und verwalten von Platten. Mit dem `zfs`-Befehl können Datasets erstellt, zerstört und verwaltet werden, sowohl Dateisysteme als auch Volumes.

20.3.1. Pools anlegen und zerstören

Einen ZFS-Pool (*zpool*) anzulegen beinhaltet das Treffen von einer Reihe von Entscheidungen, die relativ dauerhaft sind, weil die Struktur des Pools nachdem er angelegt wurde, nicht mehr geändert werden kann. Die wichtigste Entscheidung ist, welche Arten von vdevs als physische Platten zusammengefasst werden soll. Sehen Sie sich dazu die Liste von vdev-Arten an, um Details zu möglichen Optionen zu bekommen. Nachdem der Pool angelegt wurde, erlauben die meisten vdev-Arten es nicht mehr, weitere Geräte zu diesem vdev hinzuzufügen. Die Ausnahme sind Spiegel, die das Hinzufügen von weiteren Platten zum vdev gestatten, sowie stripes, die zu Spiegeln umgewandelt werden können, indem man zusätzliche Platten zum vdev anhängt. Obwohl weitere vdevs eingefügt werden können, um einen Pool zu vergrößern, kann das Layout des Pools nach dem Anlegen nicht mehr verändert werden. Stattdessen müssen die Daten gesichert, der Pool zerstört und danach neu erstellt werden.

Erstellen eines einfachen gespiegelten Pools:

```
# zpool create mypool mirror /dev/ada1 /dev/ada2
# zpool status
  pool: mypool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada1    ONLINE       0     0     0
            ada2    ONLINE       0     0     0

errors: No known data errors
```

Mehrere vdevs können gleichzeitig angelegt werden. Geben Sie zusätzliche Gruppen von Platten, getrennt durch das vdev-Typ Schlüsselwort, in diesem Beispiel `mirror`, an:

```
# zpool create mypool mirror /dev/ada1 /dev/ada2 mirror /dev/ada3 /dev/ada4
  pool: mypool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada1    ONLINE       0     0     0
            ada2    ONLINE       0     0     0
          mirror-1  ONLINE       0     0     0
            ada3    ONLINE       0     0     0
            ada4    ONLINE       0     0     0

errors: No known data errors
```

Pools lassen sich auch durch die Angabe von Partitionen anstatt von ganzen Platten erzeugen. Durch die Verwendung von ZFS in einer separaten Partition ist es möglich, dass die gleiche Platte andere Partitionen für andere Zwecke besitzen kann. Dies ist besonders von Interesse, wenn Partitionen mit Bootcode und Dateisysteme, die zum starten benötigt werden, hinzugefügt werden können. Das erlaubt es, von Platten zu booten, die auch Teil eines Pools sind. Es gibt keinen Geschwindigkeitsnachteil unter FreeBSD wenn eine Partition anstatt einer ganzen Platte verwendet wird. Durch den Einsatz von Partitionen kann der Administrator die Platten *unter provisionieren*, indem weniger als die volle Kapazität Verwendung findet. Wenn in Zukunft eine Ersatzfestplatte mit der gleichen Größe als die Originalplatte eine kleinere Kapazität aufweist, passt die kleinere Partition immer noch und die Ersatzplatte kann immer noch verwendet werden.

Erstellen eines RAID-Z2 [527]-Pools mit Partitionen:

```
# zpool create mypool raidz2 /dev/ada0p3 /dev/ada1p3 /dev/ada2p3 /dev/ada3p3 /dev/ada4p3 /
dev/ada5p3
```

```
# zpool status
  pool: mypool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          raidz2-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0
            ada3p3  ONLINE       0     0     0
            ada4p3  ONLINE       0     0     0
            ada5p3  ONLINE       0     0     0

errors: No known data errors
```

Ein Pool, der nicht länger benötigt wird, kann zerstört werden, so dass die Platten für einen anderen Einsatzzweck Verwendung finden können. Um einen Pool zu zerstören, müssen zuerst alle Datasets in diesem Pool abgehängt werden. Wenn die Datasets verwendet werden, wird das Abhängen fehlschlagen und der Pool nicht zerstört. Die Zerstörung des Pools kann erzwungen werden durch die Angabe der Option `-f`, jedoch kann dies undefiniertes Verhalten in den Anwendungen auslösen, die noch offene Dateien auf diesen Datasets hatten.

20.3.2. Hinzufügen und Löschen von Geräten

Es gibt zwei Fälle für das Hinzufügen von Platten zu einem Pool: einhängen einer Platte zu einem existierenden vdev mit `zpool attach` oder einbinden von vdevs zum Pool mit `zpool add`. Nur manche vdev-Arten gestatten es, Platten zum vdev hinzuzufügen, nachdem diese angelegt wurden.

Ein Pool mit nur einer einzigen Platte besitzt keine Redundanz. Datenverfälschung kann erkannt, aber nicht repariert werden, weil es keine weiteren Kopien der Daten gibt. Die Eigenschaft copies kann genutzt werden, um einen geringen Fehler wie einen beschädigtem Sektor auszumerzen, enthält aber nicht die gleiche Art von Schutz, die Spiegelung oder RAID-Z bieten. Wenn man mit einem Pool startet, der nur aus einer einzigen vdev-Platte besteht, kann mit dem Kommando `zpool attach` eine zustätzliche Platte dem vdev hinzugefügt werden, um einen Spiegel zu erzeugen. Mit `zpool attach` können auch zusätzliche Platten zu einer Spiegelgruppe eingefügt werden, was die Redundanz und Lesegeschwindigkeit steigert. Wenn die Platten, aus denen der Pool besteht, partitioniert sind, replizieren Sie das Layout der ersten Platte auf die Zweite. Verwenden Sie dazu `gpart backup` und `gpart restore`, um diesen Vorgang einfacher zu gestalten.

Umwandeln eines (stripe) vdevs namens *`ada0p3`* mit einer einzelnen Platte zu einem Spiegel durch das Einhängen von *`ada1p3`*:

```
# zpool status
  pool: mypool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          ada0p3    ONLINE       0     0     0

errors: No known data errors
# zpool attach mypool ada0p3 ada1p3
Make sure to wait until resilver is done before rebooting.

If you boot from pool 'mypool', you may need to update
boot code on newly attached disk 'ada1p3'.

Assuming you use GPT partitioning und 'da0' is your new boot disk
you may use the following command:
```

```
        gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1 da0
# gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1   ada1
bootcode written to ada1
# zpool status
  pool: mypool
 state: ONLINE
status: One or more devices is currently being resilvered.  The pool will
        continue to function, possibly in a degraded state.
action: Wait for the resilver to complete.
  scan: resilver in progress since Fri May 30 08:19:19 2014
        527M scanned out of 781M at 47.9M/s, 0h0m to go
        527M resilvered, 67.53% done
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0  (resilvering)

errors: No known data errors
# zpool status
  pool: mypool
 state: ONLINE
  scan: resilvered 781M in 0h0m with 0 errors on Fri May 30 08:15:58 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0

errors: No known data errors
```

Wenn das Hinzufügen von Platten zu einem vdev keine Option wie für RAID-Z ist, gibt es eine Alternative, nämlich einen anderen vdev zum Pool hinzuzufügen. Zusätzliche vdevs bieten höhere Geschwindigkeit, indem Schreibvorgänge über die vdevs verteilt werden. Jedes vdev ist dafür verantwortlich, seine eigene Redundanz sicherzustellen. Es ist möglich, aber nicht empfehlenswert, vdev-Arten zu mischen, wie zum Beispiel `mirror` und `RAID-Z`. Durch das Einfügen eines nicht-redundanten vdev zu einem gespiegelten Pool oder einem RAID-Z vdev riskiert man die Daten des gesamten Pools. Schreibvorgänge werden verteilt, deshalb ist der Ausfall einer nicht-redundanten Platte mit dem Verlust eines Teils von jedem Block verbunden, der auf den Pool geschrieben wird.

Daten werden über jedes vdev gestriped. Beispielsweise sind zwei Spiegel-vdevs effektiv ein RAID 10, dass über zwei Sets von Spiegeln die Daten schreibt. Speicherplatz wird so allokiert, dass jedes vdev zur gleichen Zeit vollgeschrieben wird. Es gibt einen Geschwindigkeitsnachteil wenn die vdevs unterschiedliche Menge von freiem Speicher aufweisen, wenn eine unproportionale Menge an Daten auf das weniger volle vdev geschrieben wird.

Wenn zusätzliche Geräte zu einem Pool, von dem gebootet wird, hinzugefügt werden, muss der Bootcode aktualisiert werden.

Einbinden einer zweiten Spiegelgruppe (`ada2p3` und `ada3p3`) zu einem bestehenden Spiegel:

```
# zpool status
  pool: mypool
 state: ONLINE
  scan: resilvered 781M in 0h0m with 0 errors on Fri May 30 08:19:35 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0
```

```
errors: No known data errors
# zpool add mypool mirror ada2p3 ada3p3
# gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1   ada2
bootcode written to ada2
# gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1   ada3
bootcode written to ada3
# zpool status
  pool: mypool
 state: ONLINE
  scan: scrub repaired 0 in 0h0m with 0 errors on Fri May 30 08:29:51 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0
          mirror-1  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0
            ada3p3  ONLINE       0     0     0

errors: No known data errors
```

Momentan können vdevs nicht von einem Pool entfernt und Platten nur von einem Spiegel ausgehängt werden, wenn genug Redundanz übrig bleibt. Wenn auch nur eine Platte in einer Spiegelgruppe bestehen bleibt, hört der Spiegel auf zu existieren und wird zu einem stripe, was den gesamten Pool riskiert, falls diese letzte Platte ausfällt.

Entfernen einer Platte aus einem Spiegel mit drei Platten:

```
# zpool status
  pool: mypool
 state: ONLINE
  scan: scrub repaired 0 in 0h0m with 0 errors on Fri May 30 08:29:51 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0

errors: No known data errors
# zpool detach mypool ada2p3
# zpool status
  pool: mypool
 state: ONLINE
  scan: scrub repaired 0 in 0h0m with 0 errors on Fri May 30 08:29:51 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0

errors: No known data errors
```

20.3.3. Den Status eines Pools überprüfen

Der Status eines Pools ist wichtig. Wenn ein Gerät sich abschaltet oder ein Lese-, Schreib- oder Prüfsummenfehler festgestellt wird, wird der dazugehörige Fehlerzähler erhöht. Die `status`-Ausgabe zeigt die Konfiguration und den Status von jedem Gerät im Pool und den Gesamtstatus des Pools. Aktionen, die durchgeführt werden sollten und Details zum letzten `scrub` werden ebenfalls angezeigt.

```
# zpool status
  pool: mypool
 state: ONLINE
  scan: scrub repaired 0 in 2h25m with 0 errors on Sat Sep 14 04:25:50 2013
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          raidz2-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0
            ada3p3  ONLINE       0     0     0
            ada4p3  ONLINE       0     0     0
            ada5p3  ONLINE       0     0     0

errors: No known data errors
```

20.3.4. Fehler beseitigen

Wenn ein Fehler erkannt wurde, werden die Lese-, Schreib- oder Prüfsummenzähler erhöht. Die Fehlermeldung kann beseitigt und der Zähler mit `zpool clear` *`mypool`* zurückgesetzt werden. Den Fehlerzustand zurückzusetzen kann wichtig sein, wenn automatisierte Skripte ablaufen, die den Administrator informieren, sobald der Pool Fehler anzeigt. Weitere Fehler werden nicht gemeldet, wenn der alte Fehlerbericht nicht entfernt wurde.

20.3.5. Ein funktionierendes Gerät ersetzen

Es gibt eine Reihe von Situationen, in denen es nötig ist, eine Platte mit einer anderen auszutauschen. Wenn eine funktionierende Platte ersetzt wird, hält der Prozess die alte Platte während des Ersetzungsvorganges noch aktiv. Der Pool wird nie den Zustand degraded erhalten, was das Risiko eines Datenverlustes minimiert. Alle Daten der alten Platte werden durch das Kommando `zpool replace` auf die Neue übertragen. Nachdem die Operation abgeschlossen ist, wird die alte Platte vom vdev getrennt. Falls die neue Platte grösser ist als die alte Platte , ist es möglich den Pool zu vergrößern, um den neuen Platz zu nutzen. Lesen Sie dazu Einen Pool vergrößern.

Ersetzen eines funktionierenden Geräts in einem Pool:

```
# zpool status
  pool: mypool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0

errors: No known data errors
# zpool replace mypool ada1p3 ada2p3
Make sure to wait until resilver is done before rebooting.

If you boot from pool 'zroot', you may need to update
boot code on newly attached disk 'ada2p3'.

Assuming you use GPT partitioning und 'da0' is your new boot disk
you may use the following command:

        gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1 da0
# gpart bootcode -b /boot/pmbr -p /boot/gptzfsboot -i 1   ada2
# zpool status
  pool: mypool
 state: ONLINE
status: One or more devices is currently being resilvered.  The pool will
```

```
        continue to function, possibly in a degraded state.
action: Wait for the resilver to complete.
  scan: resilver in progress since Mon Jun  2 14:21:35 2014
        604M scanned out of 781M at 46.5M/s, 0h0m to go
        604M resilvered, 77.39% done
config:

        NAME             STATE     READ WRITE CKSUM
        mypool           ONLINE       0     0     0
          mirror-0       ONLINE       0     0     0
            ada0p3       ONLINE       0     0     0
            replacing-1  ONLINE       0     0     0
              ada1p3     ONLINE       0     0     0
              ada2p3     ONLINE       0     0     0  (resilvering)

errors: No known data errors
# zpool status
  pool: mypool
 state: ONLINE
  scan: resilvered 781M in 0h0m with 0 errors on Mon Jun  2 14:21:52 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0

errors: No known data errors
```

20.3.6. Behandlung von fehlerhaften Geräten

Wenn eine Platte in einem Pool ausfällt, wird das vdev zu dem diese Platte gehört, den Zustand degraded erhalten. Alle Daten sind immer noch verfügbar, jedoch wird die Geschwindigkeit möglicherweise reduziert, weil die fehlenden Daten aus der verfügbaren Redundanz heraus berechnet werden müssen. Um das vdev in einen funktionierenden Zustand zurück zu versetzen, muss das physikalische Gerät ersetzt werden. ZFS wird dann angewiesen, den resilver-Vorgang zu beginnen. Daten, die sich auf dem defekten Gerät befanden, werden neu aus der vorhandenen Prüfsumme berechnet und auf das Ersatzgerät geschrieben. Nach Beendigung dieses Prozesses kehrt das vdev zum Status online zurück.

Falls das vdev keine Redundanz besitzt oder wenn mehrere Geräte ausgefallen sind und es nicht genug Redundanz gibt, um dies zu kompensieren, geht der Pool in den Zustand faulted über. Wenn keine ausreichende Anzahl von Geräten wieder an den Pool angeschlossen wird, fällt der Pool aus und die Daten müssen von Sicherungen wieder eingespielt werden.

Wenn eine defekte Platte ausgewechselt wird, wird der Name dieser defekten Platte mit der GUID des Geräts ersetzt. Ein neuer Gerätename als Parameter für `zpool replace` wird nicht benötigt, falls das Ersatzgerät den gleichen Gerätenamen besitzt.

Ersetzen einer defekten Platte durch `zpool replace` :

```
# zpool status
  pool: mypool
 state: DEGRADED
status: One or more devices could not be opened.  Sufficient replicas exist for
        the pool to continue functioning in a degraded state.
action: Attach the missing device und online it using 'zpool online'.
   see: http://illumos.org/msg/ZFS-8000-2Q
  scan: none requested
config:

        NAME                    STATE     READ WRITE CKSUM
        mypool                  DEGRADED     0     0     0
          mirror-0              DEGRADED     0     0     0
```

```
            ada0p3              ONLINE       0     0     0
            316502962686821739  UNAVAIL      0     0     0  was /dev/ada1p3

errors: No known data errors
# zpool replace mypool 316502962686821739 ada2p3
# zpool status
  pool: mypool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function, possibly in a degraded state.
action: Wait for the resilver to complete.
  scan: resilver in progress since Mon Jun  2 14:52:21 2014
        641M scanned out of 781M at 49.3M/s, 0h0m to go
        640M resilvered, 82.04% done
config:

        NAME                        STATE     READ WRITE CKSUM
        mypool                      DEGRADED     0     0     0
          mirror-0                  DEGRADED     0     0     0
            ada0p3                  ONLINE       0     0     0
            replacing-1             UNAVAIL      0     0     0
              15732067398082357289  UNAVAIL      0     0     0  was /dev/ada1p3/old
              ada2p3                ONLINE       0     0     0  (resilvering)

errors: No known data errors
# zpool status
  pool: mypool
 state: ONLINE
  scan: resilvered 781M in 0h0m with 0 errors on Mon Jun  2 14:52:38 2014
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0

errors: No known data errors
```

20.3.7. Einen Pool überprüfen

Es wird empfohlen, dass Pools regelmäßig geprüft (scrubbed) werden, idealerweise mindestens einmal pro Monat. Der `scrub`-Vorgang ist beansprucht die Platte sehr und reduziert die Geschwindigkeit während er läuft. Vermeiden Sie Zeiten, in denen großer Bedarf besteht, wenn Sie `scrub` starten oder benutzen Sie `vfs.zfs.scrub_delay` [524], um die relative Priorität vom `scrub` einzustellen, um zu verhindern, dass es mit anderen Aufgaben kollidiert.

```
# zpool scrub mypool
# zpool status
  pool: mypool
 state: ONLINE
  scan: scrub in progress since Wed Feb 19 20:52:54 2014
        116G scanned out of 8.60T at 649M/s, 3h48m to go
        0 repaired, 1.32% done
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          raidz2-0  ONLINE       0     0     0
            ada0p3  ONLINE       0     0     0
            ada1p3  ONLINE       0     0     0
            ada2p3  ONLINE       0     0     0
            ada3p3  ONLINE       0     0     0
            ada4p3  ONLINE       0     0     0
            ada5p3  ONLINE       0     0     0
```

```
errors: No known data errors
```

Falls eine Überrpüfaktion abgebrochen werden muss, geben Sie `zpool scrub -s` *`mypool`* ein.

20.3.8. Selbstheilung

Die Prüfsummen, welche zusammen mit den Datenblöcken gespeichert werden, ermöglichen dem Dateisystem, sich *selbst zu heilen*. Diese Eigenschaft wird automatisch Daten korrigieren, deren Prüfsumme nicht mit der Gespeicherten übereinstimmt, die auf einem anderen Gerät, das Teil des Pools ist, vorhanden ist. Beispielsweise bei einem Spiegel aus zwei Platten, von denen eine anfängt, Fehler zu produzieren und nicht mehr länger Daten speichern kann. Dieser Fall ist sogar noch schlimmer, wenn auf die Daten seit einiger Zeit nicht mehr zugegriffen wurde, zum Beispiel bei einem Langzeit-Archivspeicher. Traditionelle Dateisysteme müssen dann Algorithmen wie fsck(8) ablaufen lassen, welche die Daten überprüfen und reparieren. Diese Kommandos benötigen einige Zeit und in gravierenden Fällen muss ein Administrator manuelle Entscheidungen treffen, welche Reparaturoperation vorgenommen werden soll. Wenn ZFS einen defekten Datenblock mit einer Prüfsumme erkennt, die nicht übereinstimmt, versucht es die Daten von der gespiegelten Platte zu lesen. Wenn diese Platte die korrekten Daten liefern kann, wird nicht nur dieser Datenblock an die anfordernde Applikation geschickt, sondern auch die falschen Daten auf der Disk reparieren, welche die falsche Prüfsumme erzeugt hat. Dies passiert während des normalen Betriebs des Pools, ohne dass eine Interaktion vom Systemadministrator notwendig wäre.

Das nächste Beispiel demonstriert dieses Verhalten zur Selbstheilung. Ein gespiegelter Pool mit den beiden Platten `/dev/ada0` und `/dev/ada1` wird angelegt.

```
# zpool create healer mirror /dev/ada0 /dev/ada1
# zpool status healer
  pool: healer
 state: ONLINE
  scan: none requested
config:

    NAME        STATE     READ WRITE CKSUM
    healer      ONLINE       0     0     0
      mirror-0  ONLINE       0     0     0
        ada0    ONLINE       0     0     0
        ada1    ONLINE       0     0     0

errors: No known data errors
# zpool list
NAME     SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
healer   960M  92.5K   960M     0%  1.00x  ONLINE  -
```

Ein paar wichtige Daten, die es vor Datenfehlern mittels der Selbstheilungsfunktion zu schützen gilt, werden auf den Pool kopiert. Eine Prüfsumme wird zum späteren Vergleich berechnet.

```
# cp /some/important/data /healer
# zfs list
NAME     SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
healer   960M  67.7M   892M     7%  1.00x  ONLINE  -
# sha1 /healer > checksum.txt
# cat checksum.txt
SHA1 (/healer) = 2753eff56d77d9a536ece6694bf0a82740344d1f
```

Datenfehler werden durch das Schreiben von zufälligen Daten an den Anfang einer Platte des Spiegels simuliert. Um ZFS daran zu hindern, die Daten so schnell zu reparieren, wie es diese entdeckt, wird der Pool vor der Veränderung exportiert und anschließend wieder importiert.

Warnung

Dies ist eine gefährliche Operation, die wichtige Daten zerstören kann. Es wird hier nur zu Demonstrationszwecken gezeigt und sollte nicht während des normalen Betriebs des Pools

versucht werden. Dieses vorsätzliche Korrumpierungsbeispiel sollte auf gar keinen Fall auf einer Platte mit einem anderen Dateisystem durchgeführt werden. Verwenden Sie keine anderen Gerätenamen als diejenigen, die hier gezeigt werden, die Teil des Pools sind. Stellen Sie sicher, dass die passende Sicherungen angefertigt haben, bevor Sie dieses Kommando ausführen!

```
# zpool export healer
# dd if=/dev/random of=/dev/ada1 bs=1m count=200
200+0 records in
200+0 records out
209715200 bytes transferred in 62.992162 secs (3329227 bytes/sec)
# zpool import healer
```

Der Status des Pools zeigt an, dass bei einem Gerät ein Fehler aufgetreten ist. Wichtig zu wissen ist, dass Anwendungen, die Daten vom Pool lesen keine ungültigen Daten erhalten haben. ZFS lieferte Daten vom `ada0`-Gerät mit der korrekten Prüfsumme aus. Das Gerät mit der fehlerhaften Prüfsumme kann sehr einfach gefunden werden, da die Spalte `CKSUM` einen Wert ungleich Null enthält.

```
# zpool status healer
    pool: healer
   state: ONLINE
  status: One or more devices has experienced an unrecoverable error.  An
          attempt was made to correct the error.  Applications are unaffected.
  action: Determine if the device needs to be replaced, und clear the errors
          using 'zpool clear' or replace the device with 'zpool replace'.
     see: http://www.sun.com/msg/ZFS-8000-9P
    scan: none requested
  config:

      NAME        STATE     READ WRITE CKSUM
      healer      ONLINE       0     0     0
        mirror-0  ONLINE       0     0     0
          ada0    ONLINE       0     0     0
          ada1    ONLINE       0     0     1

errors: No known data errors
```

Der Fehler wurde erkannt und korrigiert durch die vorhandene Redundanz, welche aus der nicht betroffenen Platte `ada0` des Spiegels gewonnen wurde. Ein Vergleich der Prüfsumme mit dem Original wird zeigen, ob sich der Pool wieder in einem konsistenten Zustand befindet.

```
# sha1 /healer >> checksum.txt
# cat checksum.txt
SHA1 (/healer) = 2753eff56d77d9a536ece6694bf0a82740344d1f
SHA1 (/healer) = 2753eff56d77d9a536ece6694bf0a82740344d1f
```

Die beiden Prüfsummen, die vor und nach der vorsätzlichen Korrumpierung der Daten des Pools angelegt wurden, stimmen immer noch überein. Dies zeigt wie ZFS in der Lage ist, Fehler automatisch zu erkennen und zu korrigieren, wenn die Prüfsummen nicht übereinstimmen. Beachten Sie, dass dies nur möglich ist, wenn genug Redundanz im Pool vorhanden ist. Ein Pool, der nur aus einer einzigen Platte besteht besitzt keine Selbstheilungsfunktion. Dies ist auch der Grund warum Prüfsummen bei ZFS so wichtig sind und deshalb aus keinem Grund deaktiviert werden sollten. Kein fsck(8) ist nötig, um diese Fehler zu erkennen und zu korrigieren und der Pool war während der gesamten Zeit, in der das Problem bestand, verfügbar. Eine scrub-Aktion ist nun nötig, um die fehlerhaften Daten auf `ada1` zu beheben.

```
# zpool scrub healer
# zpool status healer
  pool: healer
 state: ONLINE
status: One or more devices has experienced an unrecoverable error.  An
            attempt was made to correct the error.  Applications are unaffected.
```

```
action: Determine if the device needs to be replaced, und clear the errors
            using 'zpool clear' or replace the device with 'zpool replace'.
   see: http://www.sun.com/msg/ZFS-8000-9P
  scan: scrub in progress since Mon Dec 10 12:23:30 2012
        10.4M scanned out of 67.0M at 267K/s, 0h3m to go
        9.63M repaired, 15.56% done
config:

    NAME        STATE     READ WRITE CKSUM
    healer      ONLINE       0     0     0
      mirror-0  ONLINE       0     0     0
       ada0     ONLINE       0     0     0
       ada1     ONLINE       0     0   627  (repairing)

errors: No known data errors
```

Durch das scrub werden die Daten von `ada0` gelesen und alle Daten mit einer falschen durch diejenigen mit der richtigen Prüfsumme auf `ada1` ersetzt. Dies wird durch die Ausgabe `(repairing)` des Kommandos `zpool status` angezeigt. Nachdem die Operation abgeschlossen ist, ändert sich der Poolstatus zu:

```
# zpool status healer
  pool: healer
 state: ONLINE
status: One or more devices has experienced an unrecoverable error.  An
        attempt was made to correct the error.  Applications are unaffected.
action: Determine if the device needs to be replaced, und clear the errors
            using 'zpool clear' or replace the device with 'zpool replace'.
   see: http://www.sun.com/msg/ZFS-8000-9P
  scan: scrub repaired 66.5M in 0h2m with 0 errors on Mon Dec 10 12:26:25 2012
config:

    NAME        STATE     READ WRITE CKSUM
    healer      ONLINE       0     0     0
      mirror-0  ONLINE       0     0     0
       ada0     ONLINE       0     0     0
       ada1     ONLINE       0     0 2.72K

errors: No known data errors
```

Nach der scrub-Operation und der anschliessenden Synchronisation der Daten von `ada0` nach `ada1`, kann die Fehlermeldung vom Poolstatus durch die Eingabe von `zpool clear` bereinigt werden.

```
# zpool clear healer
# zpool status healer
  pool: healer
 state: ONLINE
  scan: scrub repaired 66.5M in 0h2m with 0 errors on Mon Dec 10 12:26:25 2012
config:

    NAME        STATE     READ WRITE CKSUM
    healer      ONLINE       0     0     0
      mirror-0  ONLINE       0     0     0
       ada0     ONLINE       0     0     0
       ada1     ONLINE       0     0     0

errors: No known data errors
```

Der Pool ist jetzt wieder in einem voll funktionsfähigen Zustand versetzt worden und alle Fehler wurden beseitigt.

20.3.9. Einen Pool vergrössern

Die verwendbare Größe eines redundant ausgelegten Pools ist durch die Kapazität des kleinsten Geräts in jedem vdev begrenzt. Das kleinste Gerät kann durch ein größeres Gerät ersetzt werden. Nachdem eine replace oder resilver-Operation abgeschlossen wurde, kann der Pool anwachsen, um die Kapazität des neuen Geräts zu nutzen. Nehmen wir als Beispiel einen Spiegel mit einer 1 TB und einer 2 TB Platte. Der verwendbare Plattenplatz beträgt 1 TB.

Wenn die 1 TB Platte mit einer anderen 2 TB Platte ersetzt wird, kopiert der resilver-Prozess die existierenden Daten auf die neue Platte. Da beide Geräte nun 2 TB Kapazität besitzen, kann auch der verfügbare Plattenplatz auf die Größe von 2 TB anwachsen.

Die Erweiterung wird durch das Kommando `zpool online -e` auf jedem Gerät ausgelöst. Nachdem alle Geräte expandiert wurden, wird der Speicher im Pool zur Verfügung gestellt.

20.3.10. Importieren und Exportieren von Pools

Pools werden *exportiert* bevor diese an ein anderes System angeschlossen werden. Alle Datasets werden abgehängt und jedes Gerät wird als exportiert markiert, ist jedoch immer noch gesperrt, so dass es nicht von anderen Festplattensubsystemen verwendet werden kann. Dadurch können Pools auf anderen Maschinen *importiert* werden, die ZFS und sogar andere Hardwarearchitekturen (bis auf ein paar Ausnahmen, siehe zpool(8)) unterstützen. Besitzt ein Dataset offene Dateien, kann `zpool export -f` den Export des Pools erzwingen. Verwenden Sie dies mit Vorsicht. Die Datasets werden dadurch gewaltsam abgehängt, was bei Anwendungen, die noch offene Dateien auf diesem Dataset hatten, möglicherweise zu unerwartetem Verhalten führen kann.

Einen nichtverwendeten Pool exportieren:

```
# zpool export mypool
```

Beim Importieren eines Pool werden auch automatisch alle Datasets eingehängt. Dies ist möglicherweise nicht das bevorzugte Verhalten und wird durch `zpool import -N` verhindert. Durch `zpool import -o` temporäre Eigenschaften nur für diesen Import gesetzt. Mit dem Befehl `zpool import altroot=` ist es möglich, einen Pool mit einem anderen Basiseinhängepunkt anstatt der Wurzel des Dateisystems einzubinden. Wenn der Pool zuletzt auf einem anderen System verwendet und nicht korrekt exportiert wurde, muss unter Umständen ein Import erzwungen werden durch `zpool import -f` . Alle Pools, die momentan nicht durch ein anderes System verwendet werden, lassen sich mit `zpool import -a` importieren.

Alle zum Import verfügbaren Pools auflisten:

```
# zpool import
   pool: mypool
     id: 9930174748043525076
  state: ONLINE
 action: The pool can be imported using its name or numeric identifier.
 config:

        mypool      ONLINE
          ada2p3    ONLINE
```

Den Pool mit einem anderen Wurzelverzeichnis importieren:

```
# zpool import -o altroot= /mnt mypool
# zfs list
zfs list
NAME                 USED  AVAIL  REFER  MOUNTPOINT
mypool               110K  47.0G    31K  /mnt/mypool
```

20.3.11. Einen Pool aktualisieren

Nachdem FreeBSD aktualisiert wurde oder wenn der Pool von einem anderen System, das eine ältere Version von ZFS einsetzt, lässt sich der Pool manuell auf den aktuellen Stand von ZFS bringen, um die neuesten Eigenschaften zu unterstützen. Bedenken Sie, ob der Pool jemals wieder von einem älteren System eingebunden werden muss, bevor Sie die Aktualisierung durchführen. Das aktualisieren eines Pools ist ein nicht umkehrbarer Prozess. ältere Pools lassen sich aktualisieren, jedoch lassen sich Pools mit neueren Eigenschaften nicht wieder auf eine ältere Version bringen.

Aktualisierung eines v28-Pools, um `Feature Flags` zu unterstützen:

```
# zpool status
  pool: mypool
```

```
 state: ONLINE
status: The pool is formatted using a legacy on-disk format.  The pool can
        still be used, but some features are unavailable.
action: Upgrade the pool using 'zpool upgrade'.  Once this is done, the
        pool will no longer be accessible on software that does not support feat
        flags.
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
     ada0    ONLINE       0     0     0
     ada1    ONLINE       0     0     0

errors: No known data errors
# zpool upgrade
This system supports ZFS pool feature flags.

The following pools are formatted with legacy version numbers und can
be upgraded to use feature flags.  After being upgraded, these pools
will no longer be accessible by software that does not support feature
flags.

VER  POOL
---  ------------
28   mypool

Use 'zpool upgrade -v' for a list of available legacy versions.
Every feature flags pool has all supported features enabled.
# zpool upgrade mypool
This system supports ZFS pool feature flags.

Successfully upgraded 'mypool' from version 28 to feature flags.
Enabled the following features on 'mypool':
  async_destroy
  empty_bpobj
  lz4_compress
  multi_vdev_crash_dump
```

Die neueren Eigenschaften von ZFS werden nicht verfügbar sein, bis `zpool upgrade` abgeschlossen ist. `zpool upgrade -v` kann verwendet werden, um zu sehen, welche neuen Eigenschaften durch die Aktualisierung bereitgestellt werden, genauso wie diejenigen, die momentan schon verfügbar sind.

Einen Pool um zusätzliche Feature Flags erweitern:

```
# zpool status
  pool: mypool
 state: ONLINE
status: Some supported features are not enabled on the pool. The pool can
        still be used, but some features are unavailable.
action: Enable all features using 'zpool upgrade'. Once this is done,
        the pool may no longer be accessible by software that does not support
        the features. See zpool-features(7) for details.
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        mypool      ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
     ada0    ONLINE       0     0     0
     ada1    ONLINE       0     0     0

errors: No known data errors
# zpool upgrade
This system supports ZFS pool feature flags.
```

```
All pools are formatted using feature flags.


Some supported features are not enabled on the following pools. Once a
feature is enabled the pool may become incompatible with software
that does not support the feature. See zpool-features(7) for details.

POOL  FEATURE
---------------
zstore
      multi_vdev_crash_dump
      spacemap_histogram
      enabled_txg
      hole_birth
      extensible_dataset
      bookmarks
      filesystem_limits
# zpool upgrade mypool
This system supports ZFS pool feature flags.

Enabled the following features on 'mypool':
  spacemap_histogram
  enabled_txg
  hole_birth
  extensible_dataset
  bookmarks
  filesystem_limits
```

Warnung

Der Bootcode muss auf Systemen, die von dem Pool starten, aktualisiert werden, um diese neue Version zu unterstützen. Verwenden Sie `gpart bootcode` auf der Partition, die den Bootcode enthält. Lesen Sie für weitere Informationen gpart(8).

20.3.12. Aufgezeichnete Historie des Pools anzeigen

Befehle, die den Pool in irgendeiner Form verändern, werden aufgezeichnet. Diese Befehle beinhalten das Erstellen von Datasets, verändern von Eigenschaften oder das Ersetzen einer Platte. Diese Historie ist nützlich um nachzuvollziehen, wie ein Pool aufgebaut ist und welcher Benutzer eine bestimmte Aktion wann und wie getätigt hat. Die aufgezeichnete Historie wird nicht in einer Logdatei festgehalten, sondern ist Teil des Pools selbst. Das Kommando zum darstellen dieser Historie lautet passenderweise `zpool history`:

```
# zpool history
History for 'tank':
2013-02-26.23:02:35 zpool create tank mirror /dev/ada0 /dev/ada1
2013-02-27.18:50:58 zfs set atime=off tank
2013-02-27.18:51:09 zfs set checksum=fletcher4 tank
2013-02-27.18:51:18 zfs create tank/backup
```

Die Ausgabe zeigt `zpool` und `zfs`-Befehle, die ausgeführt wurden zusammen mit einem Zeitstempel. Nur Befehle, die den Pool verändern werden aufgezeichnet. Befehle wie `zfs list` sind dabei nicht enthalten. Wenn kein Name angegeben wird, erscheint die gesamte Historie aller Pools.

Der Befehl `zpool history` kann sogar noch mehr Informationen ausgeben, wenn die Optionen `-i` oder `-l` angegeben werden. Durch `-i` zeigt ZFS vom Benutzer eingegebene, als auch interne Ereignisse an.

```
# zpool history -i
History for 'tank':
2013-02-26.23:02:35 [internal pool create txg:5] pool spa 28; zfs spa 28; zpl 5;uts  9.1-
RELEASE 901000 amd64
```

```
2013-02-27.18:50:53 [internal property set txg:50] atime=0 dataset = 21
2013-02-27.18:50:58 zfs set atime=off tank
2013-02-27.18:51:04 [internal property set txg:53] checksum=7 dataset = 21
2013-02-27.18:51:09 zfs set checksum=fletcher4 tank
2013-02-27.18:51:13 [internal create txg:55] dataset = 39
2013-02-27.18:51:18 zfs create tank/backup
```

Weitere Details lassen sich durch die Angabe von `-l` entlocken. Historische Einträge werden in einem langen Format ausgegeben, einschließlich Informationen wie der Name des Benutzers, welcher das Kommando eingegeben hat und der Hostname, auf dem die Änderung erfolgte.

```
# zpool history -l
History for 'tank':
2013-02-26.23:02:35 zpool create tank mirror /dev/ada0 /dev/ada1 [user 0 (root) ↺
on :global]
2013-02-27.18:50:58 zfs set atime=off tank [user 0 (root) on myzfsbox:global]
2013-02-27.18:51:09 zfs set checksum=fletcher4 tank [user 0 (root) on myzfsbox:global]
2013-02-27.18:51:18 zfs create tank/backup [user 0 (root) on myzfsbox:global]
```

Die Ausgabe zeigt, dass der Benutzer `root` den gespiegelten Pool mit den beiden Platten `/dev/ada0` und `/dev/ada1` angelegt hat. Der Hostname `myzfsbox` wird ebenfalls in den Kommandos angezeigt, nachdem der Pool erzeugt wurde. Die Anzeige des Hostnamens wird wichtig, sobald der Pool von einem System exportiert und auf einem anderen importiert wird. Die Befehle, welche auf dem anderen System verwendet werden, können klar durch den Hostnamen, der bei jedem Kommando mit verzeichnet wird, unterschieden werden.

Beide Optionen für `zpool history` lassen sich auch kombinieren, um die meisten Details zur Historie eines Pools auszugeben. Die Pool Historie liefert wertvolle Informationen, wenn Aktionen nachverfolgt werden müssen oder zur Fehlerbeseitigung mehr Informationen gebraucht werden.

20.3.13. Geschwindigkeitsüberwachung

Ein eingebautes Überwachungssystem kann I/O-Statistiken in Echtzeit liefern. Es zeigt die Menge von freiem und belegtem Speicherplatz auf dem Pool an, wieviele Lese- und Schreiboperationen pro Sekunde durchgeführt werden und die aktuell verwendete I/O-Bandbreite. Standardmäßig werden alle Pools in einem System überwacht und angezeigt. Ein Poolname kann angegeben werden, um die Anzeige auf diesen Pool zu beschränken. Ein einfaches Beispiel:

```
# zpool iostat
               capacity     operations    bundwidth
pool        alloc   free   read  write   read  write
----------  -----  -----  -----  -----  -----  -----
data         288G  1.53T      2     11  11.3K  57.1K
```

Um kontinuierlich die I/O-Aktivität zu überprüfen, kann eine Zahl als letzter Parameter angegeben werden, die ein Intervall in Sekunden angibt, die zwischen den Aktualisierungen vergehen soll. Die nächste Zeile mit Statistikinformationen wird dann nach jedem Intervall ausgegeben. Drücken Sie Ctrl+C, um diese kontinuierliche Überwachung zu stoppen. Alternativ lässt sich auch eine zweite Zahl nach dem Intervall auf der Kommandozeile angeben, welche die Obergrenze von Statistikausgaben darstellt, die angezeigt werden sollen.

Noch mehr Informationen zu I/O-Statistiken können durch Angabe der Option `-v` angezeigt werden. Jedes Gerät im Pool wird dann mit einer eigenen Statistikzeile aufgeführt. Dies ist hilfreich um zu sehen, wieviele Lese- und Schreiboperationen von jedem Gerät durchgeführt werden und kann bei der Diagnose eines langsamen Geräts, das den Pool ausbremst, hilfreich sein. Dieses Beispiel zeigt einen gespiegelten Pool mit zwei Geräten:

```
# zpool iostat -v
                           capacity     operations    bundwidth
pool                     alloc   free   read  write   read  write
-----------------------  -----  -----  -----  -----  -----  -----
data                      288G  1.53T      2     12  9.23K  61.5K
  mirror                  288G  1.53T      2     12  9.23K  61.5K
    ada1                     -      -      0      4  5.61K  61.7K
```

```
    ada2                    -      -      1      4  5.04K  61.7K
----------------------  -----  -----  -----  -----  -----  -----
```

20.3.14. Einen Pool aufteilen

Ein Pool, der aus einem oder mehreren gespiegelten vdevs besteht, kann in zwei Pools aufgespalten werden. Falls nicht anders angegeben, wird das letzte Mitglied eines Spiegels abgehängt und dazu verwendet, einen neuen Pool mit den gleichen Daten zu erstellen. Die Operation sollte zuerst mit der Option `-n` versucht werden. Die Details der vorgeschlagenen Option werden dargestellt, ohne die Aktion in Wirklichkeit durchzuführen. Das hilft dabei zu bestätigen, ob die Aktion das tut, was der Benutzer damit vor hatte.

20.4. `zfs` Administration

Das `zfs`-Werkzeug ist dafür verantwortlich, alle ZFS Datasets innerhalb eines Pools zu erstellen, zerstören und zu verwalten. Der Pool selbst wird durch `zpool` verwaltet.

20.4.1. Datasets erstellen und zerstören

Anders als in traditionellen Festplatten- und Volumenmanagern wird der Plattenplatz in ZFS *nicht* vorher allokiert. Bei traditionellen Dateisystemen gibt es, nachdem der Plattenplatz partitioniert und zugeteilt wurde, keine Möglichkeit, ein zusätzliches Dateisystem hinzuzufügen, ohne eine neue Platte anzuschließen. Mit ZFS lassen sich neue Dateisysteme zu jeder Zeit anlegen. Jedes *Dataset* besitzt Eigenschaften wie Komprimierung, Deduplizierung, Zwischenspeicher (caching), Quotas, genauso wie andere nützliche Einstellungen wie Schreibschutz, Unterscheidung zwischen Groß- und Kleinschreibung, Netzwerkfreigaben und einen Einhängepunkt. Datasets können ineinander verschachtelt werden und Kind-Datasets erben die Eigenschaften ihrer Eltern. Jedes Dataset kann als eine Einheit verwaltet, delegiert, repliziert, mit Schnappschüssen versehen, in Jails gesteckt und zerstört werden. Es gibt viele Vorteile, ein separates Dataset für jede Art von Dateien anzulegen. Der einzige Nachteil einer großen Menge an Datasets ist, dass manche Befehle wie `zfs list` langsamer sind und dass das Einhängen von hunderten oder hunderttausenden von Datasets de FreeBSD-Bootvorgang verzögert.

Erstellen eines neuen Datasets und aktivieren von LZ4 Komprimierung:

```
# zfs list
NAME                  USED  AVAIL  REFER  MOUNTPOINT
mypool                781M  93.2G   144K  none
mypool/ROOT           777M  93.2G   144K  none
mypool/ROOT/default   777M  93.2G   777M  /
mypool/tmp            176K  93.2G   176K  /tmp
mypool/usr            616K  93.2G   144K  /usr
mypool/usr/home       184K  93.2G   184K  /usr/home
mypool/usr/ports      144K  93.2G   144K  /usr/ports
mypool/usr/src        144K  93.2G   144K  /usr/src
mypool/var           1.20M  93.2G   608K  /var
mypool/var/crash      148K  93.2G   148K  /var/crash
mypool/var/log        178K  93.2G   178K  /var/log
mypool/var/mail       144K  93.2G   144K  /var/mail
mypool/var/tmp        152K  93.2G   152K  /var/tmp
# zfs create -o compress=lz4  mypool/usr/mydataset
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 781M  93.2G   144K  none
mypool/ROOT            777M  93.2G   144K  none
mypool/ROOT/default    777M  93.2G   777M  /
mypool/tmp             176K  93.2G   176K  /tmp
mypool/usr             704K  93.2G   144K  /usr
mypool/usr/home        184K  93.2G   184K  /usr/home
mypool/usr/mydataset  87.5K  93.2G  87.5K  /usr/mydataset
mypool/usr/ports       144K  93.2G   144K  /usr/ports
mypool/usr/src         144K  93.2G   144K  /usr/src
mypool/var            1.20M  93.2G   610K  /var
mypool/var/crash       148K  93.2G   148K  /var/crash
```

```
mypool/var/log         178K  93.2G   178K  /var/log
mypool/var/mail        144K  93.2G   144K  /var/mail
mypool/var/tmp         152K  93.2G   152K  /var/tmp
```

Ein Dataset zu zerstören ist viel schneller, als alle Dateien zu löschen, die sich in dem Dataset befindet, da es keinen Scan aller Dateien und aktualisieren der dazugehörigen Metadaten erfordert.

Zerstören des zuvor angelegten Datasets:

```
# zfs list
NAME                   USED  AVAIL  REFER  MOUNTPOINT
mypool                 880M  93.1G   144K  none
mypool/ROOT            777M  93.1G   144K  none
mypool/ROOT/default    777M  93.1G   777M  /
mypool/tmp             176K  93.1G   176K  /tmp
mypool/usr             101M  93.1G   144K  /usr
mypool/usr/home        184K  93.1G   184K  /usr/home
mypool/usr/mydataset   100M  93.1G   100M  /usr/mydataset
mypool/usr/ports       144K  93.1G   144K  /usr/ports
mypool/usr/src         144K  93.1G   144K  /usr/src
mypool/var            1.20M  93.1G   610K  /var
mypool/var/crash       148K  93.1G   148K  /var/crash
mypool/var/log         178K  93.1G   178K  /var/log
mypool/var/mail        144K  93.1G   144K  /var/mail
mypool/var/tmp         152K  93.1G   152K  /var/tmp
# zfs destroy mypool/usr/mydataset
# zfs list
NAME                  USED  AVAIL  REFER  MOUNTPOINT
mypool                781M  93.2G   144K  none
mypool/ROOT           777M  93.2G   144K  none
mypool/ROOT/default   777M  93.2G   777M  /
mypool/tmp            176K  93.2G   176K  /tmp
mypool/usr            616K  93.2G   144K  /usr
mypool/usr/home       184K  93.2G   184K  /usr/home
mypool/usr/ports      144K  93.2G   144K  /usr/ports
mypool/usr/src        144K  93.2G   144K  /usr/src
mypool/var           1.21M  93.2G   612K  /var
mypool/var/crash      148K  93.2G   148K  /var/crash
mypool/var/log        178K  93.2G   178K  /var/log
mypool/var/mail       144K  93.2G   144K  /var/mail
mypool/var/tmp        152K  93.2G   152K  /var/tmp
```

In modernen Versionen von ZFS ist `zfs destroy` asynchron und der freie Speicherplatz kann erst nach ein paar Minuten im Pool auftauchen. Verwenden Sie `zpool get freeing` *`poolname`*, um die Eigenschaft `freeing` aufzulisten, die angibt, bei wievielen Datasets die Blöcke im Hintergrund freigegeben werden. Sollte es Kind-Datasets geben, Schnappschüsse oder andere Datasets, dann lässt sich der Elternknoten nicht zerstören. Um ein Dataset und all seine Kinder zu zerstören, verwenden Sie die Option `-r`, um das Dataset und all seine Kinder rekursiv zu entfernen. Benutzen Sie die Option `-n` und `-v`, um Datasets und Snapshots anzuzeigen, die durch diese Aktion zerstört werden würden, dies jedoch nur zu simulieren und nicht wirklich durchzuführen. Speicherplatz, der dadurch freigegeben würde, wird ebenfalls angezeigt.

20.4.2. Volumes erstellen und zerstören

Ein Volume ist ein spezieller Typ von Dataset. Anstatt dass es als Dateisystem eingehängt wird, stellt es ein Block-Gerät unter `/dev/zvol/` *`poolname/dataset`* dar. Dies erlaubt es, das Volume für andere Dateisysteme zu verwenden, die Festplatten einer virtuellen Maschine bereitzustellen oder über Protokolle wie iSCSI oder HAST exportiert zu werden.

Ein Volume kann mit einem beliebigen Dateisystem formatiert werden oder auch ohne ein Dateisystem als reiner Datenspeicher fungieren. Für den Benutzer erscheint ein Volume als eine gewöhnliche Platte. Indem gewöhnliche Dateisysteme auf diesen *zvols* angelegt werden, ist es möglich, diese mit Eigenschaften auszustatten, welche diese normalerweise nicht besitzen. Beispielsweise wird durch Verwendung der Komprimierungseigenschaft auf einem 250 MB Volume das Erstellen eines komprimierten FAT Dateisystems möglich.

```
# zfs create -V 250m -o compression=on tank/fat32
# zfs list tank
NAME USED AVAIL REFER MOUNTPOINT
tank 258M  670M   31K /tank
# newfs_msdos -F32 /dev/zvol/tank/fat32
# mount -t msdosfs /dev/zvol/tank/fat32 /mnt
# df -h /mnt | grep fat32
Filesystem           Size Used Avail Capacity Mounted on
/dev/zvol/tank/fat32 249M  24k  249M     0%   /mnt
# mount | grep fat32
/dev/zvol/tank/fat32 on /mnt (msdosfs, local)
```

Ein Volume zu zerstören ist sehr ähnlich wie ein herkömmliches Dataset zu entfernen. Die Operation wird beinahe sofort durchgeführt, jedoch kann es mehrere Minuten dauern, bis der freie Speicherplatz im Hintergrund wieder freigegeben ist.

20.4.3. Umbenennen eines Datasets

Der Name eines Datasets lässt sich durch `zfs rename` ändern. Das Eltern-Dataset kann ebenfalls mit diesem Kommando umbenannt werden. Ein Dataset unter einem anderen Elternteil umzubenennen wird den Wert dieser Eigenschaft verändern, die vom Elternteil vererbt wurden. Wird ein Dataset umbenannt, wird es abgehängt und dann erneut unter der neuen Stelle eingehängt (welche vom neuen Elternteil geerbt wird). Dieses Verhalten kann durch die Option `-u` verhindert werden.

Ein Dataset umbenennen und unter einem anderen Elterndataset verschieben:

```
# zfs list
NAME                  USED  AVAIL  REFER  MOUNTPOINT
mypool                780M  93.2G   144K  none
mypool/ROOT           777M  93.2G   144K  none
mypool/ROOT/default   777M  93.2G   777M  /
mypool/tmp            176K  93.2G   176K  /tmp
mypool/usr            704K  93.2G   144K  /usr
mypool/usr/home       184K  93.2G   184K  /usr/home
mypool/usr/mydataset 87.5K  93.2G  87.5K  /usr/mydataset
mypool/usr/ports      144K  93.2G   144K  /usr/ports
mypool/usr/src        144K  93.2G   144K  /usr/src
mypool/var           1.21M  93.2G   614K  /var
mypool/var/crash      148K  93.2G   148K  /var/crash
mypool/var/log        178K  93.2G   178K  /var/log
mypool/var/mail       144K  93.2G   144K  /var/mail
mypool/var/tmp        152K  93.2G   152K  /var/tmp
# zfs rename mypool/usr/mydataset  mypool/var/newname
# zfs list
NAME                  USED  AVAIL  REFER  MOUNTPOINT
mypool                780M  93.2G   144K  none
mypool/ROOT           777M  93.2G   144K  none
mypool/ROOT/default   777M  93.2G   777M  /
mypool/tmp            176K  93.2G   176K  /tmp
mypool/usr            616K  93.2G   144K  /usr
mypool/usr/home       184K  93.2G   184K  /usr/home
mypool/usr/ports      144K  93.2G   144K  /usr/ports
mypool/usr/src        144K  93.2G   144K  /usr/src
mypool/var           1.29M  93.2G   614K  /var
mypool/var/crash      148K  93.2G   148K  /var/crash
mypool/var/log        178K  93.2G   178K  /var/log
mypool/var/mail       144K  93.2G   144K  /var/mail
mypool/var/newname   87.5K  93.2G  87.5K  /var/newname
mypool/var/tmp        152K  93.2G   152K  /var/tmp
```

Schnappschüsse können auf diese Weise ebenfalls umbenannt werden. Aufgrund der Art von Schnappschüssen können diese nicht unter einem anderen Elterndataset eingehängt werden. Um einen rekursiven Schnappschuss umzubenennen, geben Sie die Option `-r` an, um alle Schnappschüsse mit dem gleichen Namen im Kind-Dataset ebenfalls umzubenennen.

```
# zfs list -t snapshot
NAME                                 USED  AVAIL  REFER  MOUNTPOINT
mypool/var/newname@first_snapshot      0      -  87.5K  -
# zfs rename mypool/var/newname@first_snapshot   new_snapshot_name
# zfs list -t snapshot
NAME                                     USED  AVAIL  REFER  MOUNTPOINT
mypool/var/newname@new_snapshot_name       0      -  87.5K  -
```

20.4.4. Festlegen von Dataset-Eigenschaften

Jedes ZFS-Dataset besitzt eine Menge von Eigenschaften, die sein Verhalten beeinflussen. Die meisten Eigenschaften werden automatisch vom Eltern-Dataset vererbt, können jedoch lokal überschrieben werden. Sie legen eine Eigenschaft durch `zfs set property=value dataset` fest. Die meisten Eigenschaften haben eine begrenzte Menge von gültigen Werten. `zfs get` stellt diese dar und zeigt jede mögliche Eigenschaft und gültige Werte an. Die meisten Eigenschaften können über `zfs inherit` wieder auf ihren Ausgangswert zurückgesetzt werden.

Benutzerdefinierte Eigenschaften lassen sich ebenfalls setzen. Diese werden Teil der Konfiguration des Datasets und können dazu verwendet werden, zusätzliche Informationen über das Dataset oder seine Bestandteile zu speichern. Um diese benutzerdefinierten Eigenschaften von den ZFS-eigenen zu unterscheiden, wird ein Doppelpunkt (`:`) verwendet, um einen eigenen Namensraum für diese Eigenschaft zu erstellen.

```
# zfs set custom:costcenter=1234 tank
# zfs get custom:costcenter tank
NAME PROPERTY           VALUE SOURCE
tank custom:costcenter  1234  local
```

Um eine selbstdefinierte Eigenschaft umzubenennen, verwenden Sie `zfs inherit` mit der Option `-r`. Wenn die benutzerdefinierte Eigenschaft nicht in einem der Eltern-Datasets definiert ist, wird diese komplett entfernt (obwohl diese Änderungen natürlich in der Historie des Pools noch aufgezeichnet sind).

```
# zfs inherit -r custom:costcenter tank
# zfs get custom:costcenter tank
NAME    PROPERTY           VALUE              SOURCE
tank    custom:costcenter  -                  -
# zfs get all tank | grep custom:costcenter
#
```

20.4.5. Verwalten von Schnappschüssen

Schnappschüsse sind eine der mächtigen Eigenschaften von ZFS. Ein Schnappschuss bietet einen nur-Lese Zustand eines Datasets zu einem bestimmten Zeitpunkt. Mit Kopieren-beim-Schreiben (Copy-On-Write COW), können Schnappschüsse schnell erstellt werden durch das Aufheben der älteren Version der Daten auf der Platte. Falls kein Snapshot existiert, wird der Speicherplatz wieder für zukünftige Verwendung freigegeben wenn Daten geschrieben oder gelöscht werden. Schnappschüsse sparen Speicherplatz, indem diese nur die Unterschiede zwischen dem momentanen Dataset und der vorherigen Version aufzeichnen. Schnappschüsse sind nur auf ganzen Datasets erlaubt, nicht auf individuellen Dateien oder Verzeichnissen. Wenn ein Schnappschuss eines Datasets erstellt wird, wird alles was darin enthalten ist, dupliziert. Das beinhaltet Dateisystemeigenschaften, Dateien, Verzeichnisse, Rechte und so weiter. Schnappschüsse benötigen keinen zusätzlichen Speicherplatz wenn diese erstmals angelegt werden, nur wenn Blöcke, die diese referenzieren, geändert werden. Rekursive Schnappschüsse, die mit der Option `-r` erstellt, erzeugen einen mit dem gleichen Namen des Datasets und all seinen Kindern, was eine konsistente Momentaufnahme aller Dateisysteme darstellt. Dies kann wichtig sein, wenn eine Anwendung Dateien auf mehreren Datasets ablegt, die miteinander in Verbindung stehen oder voneinander abhängig sind. Ohne Schnappschüsse würde ein Backup Kopien dieser Dateien zu unterschiedlichen Zeitpunkten enthalten.

Schnappschüsse in ZFS bieten eine Vielzahl von Eigenschaften, die selbst in anderen Dateisystemen mit Schnappschussfunktion nicht vorhanden sind. Ein typisches Beispiel zur Verwendung von Schnappschüssen ist, den momentanen Stand des Dateisystems zu sichern, wenn eine riskante Aktion wie das Installieren von Software oder eine Systemaktualisierung durchgeführt wird. Wenn diese Aktion fehlschlägt, so kann der Schnappschuss zurückgerollt werden und das System befindet sich wieder in dem gleichen Zustand, wie zu dem, als der Schnappschuss erstellt wurde. Wenn die Aktualisierung jedoch erfolgreich war, kann der Schnappschuss gelöscht werden, um

Speicherplatz frei zu geben. Ohne Schnappschüsse, wird durch ein fehlgeschlagenes Update eine Wiederherstellung der Sicherung fällig, was oft mühsam und zeitaufwändig ist, außerdem ist währenddessen das System nicht verwendbar. Schnappschüsse lassen sich schnell und mit wenig bis gar keiner Ausfallzeit zurückrollen, selbst wenn das System im normalen Betrieb läuft. Die Zeitersparnis ist enorm, wenn mehrere Terabyte große Speichersysteme eingesetzt werden und viel Zeit für das Kopieren der Daten vom Sicherungssystem benötigt wird. Schnappschüsse sind jedoch keine Ersatz für eine Vollsicherung des Pools, können jedoch als eine schnelle und einfache Sicherungsmethode verwendet werden, um eine Kopie eines Datasets zu einem bestimmten Zeitpunkt zu sichern.

20.4.5.1. Schnappschüsse erstellen

Schnappschüsse werden durch das Kommando `zfs snapshot` *`dataset`*`@`*`snapshotname`* angelegt. Durch Angabe der Option `-r` werden Schnappschüsse rekursive angelegt, mit dem gleichen Namen auf allen Datasets.

Einen rekursiven Schnappschuss des gesamten Pools erzeugen:

```
# zfs list -t all
NAME                                  USED  AVAIL  REFER  MOUNTPOINT
mypool                                780M  93.2G   144K  none
mypool/ROOT                           777M  93.2G   144K  none
mypool/ROOT/default                   777M  93.2G   777M  /
mypool/tmp                            176K  93.2G   176K  /tmp
mypool/usr                            616K  93.2G   144K  /usr
mypool/usr/home                       184K  93.2G   184K  /usr/home
mypool/usr/ports                      144K  93.2G   144K  /usr/ports
mypool/usr/src                        144K  93.2G   144K  /usr/src
mypool/var                           1.29M  93.2G   616K  /var
mypool/var/crash                      148K  93.2G   148K  /var/crash
mypool/var/log                        178K  93.2G   178K  /var/log
mypool/var/mail                       144K  93.2G   144K  /var/mail
mypool/var/newname                   87.5K  93.2G  87.5K  /var/newname
mypool/var/newname@new_snapshot_name     0      -  87.5K  -
mypool/var/tmp                        152K  93.2G   152K  /var/tmp
# zfs snapshot -r mypool@my_recursive_snapshot
# zfs list -t snapshot
NAME                                        USED  AVAIL  REFER  MOUNTPOINT
mypool@my_recursive_snapshot                   0      -   144K  -
mypool/ROOT@my_recursive_snapshot              0      -   144K  -
mypool/ROOT/default@my_recursive_snapshot      0      -   777M  -
mypool/tmp@my_recursive_snapshot               0      -   176K  -
mypool/usr@my_recursive_snapshot               0      -   144K  -
mypool/usr/home@my_recursive_snapshot          0      -   184K  -
mypool/usr/ports@my_recursive_snapshot         0      -   144K  -
mypool/usr/src@my_recursive_snapshot           0      -   144K  -
mypool/var@my_recursive_snapshot               0      -   616K  -
mypool/var/crash@my_recursive_snapshot         0      -   148K  -
mypool/var/log@my_recursive_snapshot           0      -   178K  -
mypool/var/mail@my_recursive_snapshot          0      -   144K  -
mypool/var/newname@new_snapshot_name           0      -  87.5K  -
mypool/var/newname@my_recursive_snapshot       0      -  87.5K  -
mypool/var/tmp@my_recursive_snapshot           0      -   152K  -
```

Schnappschüsse werden nicht durch einen `zfs list`-Befehl angezeigt. Um Schnappschüsse mit aufzulisten, muss `-t snapshot` an das Kommando `zfs list` angehängt werden. Durch `-t all` werden sowohl Dateisysteme als auch Schnappschüsse nebeneinander angezeigt.

Schnappschüsse werden nicht direkt eingehängt, deshalb wird auch kein Pfad in der Spalte `MOUNTPOINT` angezeigt. Ebenso wird kein freier Speicherplatz in der Spalte `AVAIL` aufgelistet, da Schnappschüsse nicht mehr geschrieben werden können, nachdem diese angelegt wurden. Vergleichen Sie den Schnappschuss mit dem ursprünglichen Dataset von dem es abstammt:

```
# zfs list -rt all mypool/usr/home
NAME                                 USED  AVAIL  REFER  MOUNTPOINT
mypool/usr/home                      184K  93.2G   184K  /usr/home
mypool/usr/home@my_recursive_snapshot    0      -   184K  -
```

Durch das Darstellen des Datasets und des Schnappschusses nebeneinander zeigt deutlich, wie Schnappschüsse in COW Manier funktionieren. Sie zeichnen nur die Änderungen (*delta*) auf, die währenddessen entstanden sind und nicht noch einmal den gesamten Inhalt des Dateisystems. Das bedeutet, dass Schnappschüsse nur wenig Speicherplatz benötigen, wenn nur kleine Änderungen vorgenommen werden. Der Speicherverbrauch kann sogar noch deutlicher gemacht werden, wenn eine Datei auf das Dataset kopiert wird und anschließend ein zweiter Schnappschuss angelegt wird:

```
# cp /etc/passwd /var/tmp
# zfs snapshot mypool/var/tmp @after_cp
# zfs list -rt all mypool/var/tmp
NAME                                  USED  AVAIL  REFER  MOUNTPOINT
mypool/var/tmp                        206K  93.2G   118K  /var/tmp
mypool/var/tmp@my_recursive_snapshot   88K      -   152K  -
mypool/var/tmp@after_cp                  0      -   118K  -
```

Der zweite Schnappschuss enthält nur die Änderungen am Dataset, die nach der Kopieraktion gemacht wurden. Dies bedeutet enorme Einsparungen von Speicherplatz. Beachten Sie, dass sich die Größe des Schnappschusses *mypool/var/tmp@my_recursive_snapshot* in der Spalte USED ebenfalls geändert hat, um die Änderungen von sich selbst und dem Schnappschuss, der im Anschluss angelegt wurde, anzuzeigen.

20.4.5.2. Schnappschüsse vergleichen

ZFS enthält ein eingebautes Kommando, um die Unterschiede zwischen zwei Schnappschüssen miteinander zu vergleichen. Das ist hilfreich, wenn viele Schnappschüsse über längere Zeit angelegt wurden und der Benutzer sehen will, wie sich das Dateisystem über diesen Zeitraum verändert hat. Beispielsweise kann `zfs diff` den letzten Schnappschuss finden, der noch eine Datei enthält, die aus Versehen gelöscht wurde. Wenn dies für die letzten beiden Schnappschüsse aus dem vorherigen Abschnitt durchgeführt wird, ergibt sich folgende Ausgabe:

```
# zfs list -rt all mypool/var/tmp
NAME                                  USED  AVAIL  REFER  MOUNTPOINT
mypool/var/tmp                        206K  93.2G   118K  /var/tmp
mypool/var/tmp@my_recursive_snapshot   88K      -   152K  -
mypool/var/tmp@after_cp                  0      -   118K  -
# zfs diff mypool/var/tmp@my_recursive_snapshot
M       /var/tmp/
+       /var/tmp/passwd
```

Das Kommando zeigt alle Änderungen zwischen dem angegebenen Schnappschuss (in diesem Fall *mypool/var/tmp@my_recursive_snapshot*) und dem momentan aktuellen Dateisystem. Die erste Spalte zeigt die Art der Änderung an:

+	Das Verzeichnis oder die Datei wurde hinzugefügt.
-	Das Verzeichnis oder die Datei wurde gelöscht.
M	Das Verzeichnis oder die Datei wurde geändert.
R	Das Verzeichnis oder die Datei wurde umbenannt.

Vergleicht man die Ausgabe mit der Tabelle, wird klar, dass *passwd* hinzugefügt wurde, nachdem der Schnappschuss *mypool/var/tmp@my_recursive_snapshot* erstellt wurde. Das resultierte ebenfalls in einer Änderung am darüberliegenden Verzeichnis, das unter */var/tmp* eingehängt ist.

Zwei Schnappschüsse zu vergleichen ist hilfreich, wenn die Replikationseigenschaft von ZFS verwendet wird, um ein Dataset auf einen anderen Host zu Sicherungszwecken übertragen.

Zwei Schnappschüsse durch die Angabe des kompletten Namens des Datasets und dem Namen des Schnappschusses beider Datasets vergleichen:

```
# cp /var/tmp/passwd /var/tmp/passwd.copy
# zfs snapshot mypool/var/tmp@diff_snapshot
# zfs diff mypool/var/tmp@my_recursive_snapshot   mypool/var/tmp@diff_snapshot
M       /var/tmp/
+       /var/tmp/passwd
```

```
+       /var/tmp/passwd.copy
# zfs diff mypool/var/tmp@my_recursive_snapshot   mypool/var/tmp@after_cp
M       /var/tmp/
+       /var/tmp/passwd
```

Ein Administrator, der für die Sicherung zuständig ist, kann zwei Schnappschüsse miteinander vergleichen, die vom sendenden Host empfangen wurden, um festzustellen, welche Änderungen am Dataset vorgenommen wurden. Lesen Sie dazu den Abschnitt Replication um weitere Informationen zu erhalten.

20.4.5.3. Schnappschüsse zurückrollen

Wenn zumindest ein Schnappschuss vorhanden ist, kann dieser zu einem beliebigen Zeitpunkt zurückgerollt werden. In den meisten Fällen passiert dies, wenn der aktuelle Zustand des Datasets nicht mehr benötigt wird und eine ältere Version bevorzugt wird. Szenarien wie lokale Entwicklungstests, die fehlgeschlagen sind, defekte Systemaktualisierungen, welche die Funktionalität des Gesamtsystems einschränken oder die Anforderung, versehentlich gelöschte Dateien oder Verzeichnisse wiederherzustellen, sind allgegenwärtig. Glücklicherweise ist das zurückrollen eines Schnappschusses so leicht wie die Eingabe von `zfs rollback snapshotname`. Abhängig davon, wie viele Änderungen betroffen sind, wird diese Operation innerhalb einer gewissen Zeit abgeschlossen sein. Während dieser Zeit bleibt das Dataset in einem konsistenten Zustand, sehr ähnlich den ACID-Prinzipien, die eine Datenbank beim Zurückrollen entspricht. Während all dies passiert, ist das Dataset immer noch aktiv und erreichbar ohne dass eine Ausfallzeit nötig wäre. Sobald der Schnappschuss zurückgerollt wurde, besitzt das Dataset den gleichen Zustand, den es besaß, als der Schnappschuss angelegt wurde. Alle anderen Daten in diesem Dataset, die nicht Teil des Schnappschusses sind, werden verworfen. Einen Schnappschuss des aktuellen Zustandes des Datasets vor dem Zurückrollen anzulegen ist eine gute Idee, wenn hinterher noch Daten benötigt werden. Auf diese Weise kann der Benutzer vor und zurück zwischen den Schnappschüssen springen, ohne wertvolle Daten zu verlieren.

Im ersten Beispiel wird ein Schnappschuss aufgrund eines unvorsichtigen `rm`-Befehls zurückgerollt, der mehr Daten gelöscht hat, als vorgesehen.

```
# zfs list -rt all mypool/var/tmp
NAME                                 USED  AVAIL  REFER  MOUNTPOINT
mypool/var/tmp                       262K  93.2G   120K  /var/tmp
mypool/var/tmp@my_recursive_snapshot    88K      -   152K  -
mypool/var/tmp@after_cp               53.5K      -   118K  -
mypool/var/tmp@diff_snapshot             0      -   120K  -
% ls /var/tmp
passwd          passwd.copy
% rm /var/tmp/passwd*
% ls /var/tmp
vi.recover
%
```

Zu diesem Zeitpunkt bemerkt der Benutzer, dass zuviele Dateien gelöscht wurden und möchte diese zurück haben. ZFS bietet eine einfache Möglichkeit, diese durch zurückrollen zurück zu bekommen, allerdings nur, wenn Schnappschüsse von wichtigen Daten regelmäßig angelegt werden. Um die Dateien zurückzuerhalten und vom letzten Schnappschuss wieder zu beginnen, geben Sie ein:

```
# zfs rollback mypool/var/tmp@diff_snapshot
% ls /var/tmp
passwd          passwd.copy     vi.recover
```

Die Operation zum Zurückrollen versetzt das Dataset in den Zustand des letzten Schnappschusses zurück. Es ist ebenfalls möglich, zu einem Schnappschuss zurückzurollen, der viel früher angelegt wurde und es noch Schnappschüsse nach diesem gibt. Wenn Sie dies versuchen, gibt ZFS die folgende Warnung aus:

```
# zfs list -rt snapshot  mypool/var/tmp
AME                                  USED  AVAIL  REFER  MOUNTPOINT
mypool/var/tmp@my_recursive_snapshot    88K      -   152K  -
mypool/var/tmp@after_cp               53.5K      -   118K  -
mypool/var/tmp@diff_snapshot             0      -   120K  -
# zfs rollback mypool/var/tmp@my_recursive_snapshot
cannot rollback to 'mypool/var/tmp@my_recursive_snapshot': more recent snapshots exist
use '-r' to force deletion of the following snapshots:
```

```
mypool/var/tmp@after_cp
mypool/var/tmp@diff_snapshot
```

Diese Warnung bedeutet, dass noch Schnappschüsse zwischen dem momentanen Stand des Datasets und dem Schnappschuss, zu dem der Benutzer zurückrollen möchte, existieren. Um das Zurückrollen durchzuführen, müssen die Schnappschüsse gelöscht werden. ZFS kann nicht alle Änderungen zwischen verschiedenen Zuständen eines Datasets verfolgen, da Schnappschüsse nur gelesen werden können. ZFS wird nicht die betroffenen Schnappschüsse löschen, es sei denn, der Benutzer verwendet die Option `-r`, um anzugeben, dass dies die gewünschte Aktion ist. Falls dies der Fall ist und die Konsequenzen alle dazwischenliegenden Schnappschüsse zu verlieren verstanden wurden, kann der Befehl abgesetzt werden:

```
# zfs rollback -r mypool/var/tmp@my_recursive_snapshot
# zfs list -rt snapshot mypool/var/tmp
NAME                                     USED  AVAIL  REFER  MOUNTPOINT
mypool/var/tmp@my_recursive_snapshot     8K      -   152K  -
% ls /var/tmp
vi.recover
```

Die Ausgabe von `zfs list -t snapshot` bestätigt, dass die dazwischenliegenden Schnappschüsse als Ergebnis von `zfs rollback -r` entfernt wurden.

20.4.5.4. Individuelle Dateien aus Schnappschüssen wiederherstellen

Schnappschüsse sind unter einem versteckten Verzeichnis unter dem Eltern-Dataset eingehängt: `.zfs/snapshots/`*`snapshotname`*. Standardmäßig werden diese Verzeichnisse nicht von einem gewöhnlichen `ls -a` angezeigt. Obwohl diese Verzeichnisse nicht angezeigt werden, sind diese trotzdem vorhanden und der Zugriff darauf erfolgt wie auf jedes andere Verzeichnis. Die Eigenschaft `snapdir` steuert, ob diese Verzeichnisse beim Auflisten eines Verzeichnisses angezeigt werden oder nicht. Das Einstellen der Eigenschaft auf den Wert `visible` erlaubt es, diese in der Ausgabe von `ls` und anderen Kommandos, die mit Verzeichnisinhalten umgehen können, anzuzeigen.

```
# zfs get snapdir mypool/var/tmp
NAME            PROPERTY  VALUE    SOURCE
mypool/var/tmp  snapdir   hidden   default
% ls -a /var/tmp
.               ..              passwd          vi.recover
# zfs set snapdir=visible mypool/var/tmp
% ls -a /var/tmp
.               ..              .zfs            passwd          vi.recover
```

Einzelne Dateien lassen sich einfach auf einen vorherigen Stand wiederherstellen, indem diese aus dem Schnappschuss zurück in das Eltern-Dataset kopiert werden. Die Verzeichnisstruktur unterhalb von `.zfs/snapshot` enthält ein Verzeichnis, das exakt wie der Schnappschuss benannt ist, der zuvor angelegt wurde, um es einfacher zu machen, diese zu identifizieren. Im nächsten Beispiel wird angenommen, dass eine Datei aus dem versteckten `.zfs` Verzeichnis durch kopieren aus dem Schnappschuss, der die letzte Version dieser Datei enthielt, wiederhergestellt wird:

```
# rm /var/tmp/passwd
% ls -a /var/tmp
.               ..              .zfs            vi.recover
# ls /var/tmp/.zfs/snapshot
after_cp                my_recursive_snapshot
# ls /var/tmp/.zfs/snapshot/ after_cp
passwd          vi.recover
# cp /var/tmp/.zfs/snapshot/ after_cp/passwd /var/tmp
```

Als `ls .zfs/snapshot` ausgeführt wurde, war die `snapdir`-Eigenschaft möglicherweise nicht auf hidden gesetzt, trotzdem ist es immer noch möglich, den Inhalt dieses Verzeichnisses aufzulisten. Es liegt am Administrator zu entscheiden, ob diese Verzeichnisse angezeigt werden soll. Es ist möglich, diese für bestimmte Datasets anzuzeigen und für andere zu verstecken. Das Kopieren von Dateien oder Verzeichnissen aus diesem versteckten `.zfs/snapshot` Verzeichnis ist einfach genug. Jedoch führt der umgekehrte Weg zu einem Fehler:

```
# cp /etc/rc.conf /var/tmp/.zfs/snapshot/ after_cp/
cp: /var/tmp/.zfs/snapshot/after_cp/rc.conf: Read-only file system
```

Der Fehler erinnert den Benutzer daran, dass Schnappschüsse nur gelesen aber nicht mehr geändert werden können, nachdem diese angelegt wurden. Es können keine Dateien in diese Schnappschuss-Verzeichnisse kopiert oder daraus gelöscht werden, da dies sonst den Zustand des Datasets verändern würde, den sie repräsentieren.

Schnappschüsse verbrauchen Speicherplatz basierend auf der Menge an Änderungen, die am Eltern-Dataset durchgeführt wurden, seit der Zeit als der Schnappschuss erstellt wurde. Die Eigenschaft `written` eines Schnappschusses verfolgt, wieviel Speicherplatz vom Schnappschuss belegt wird.

Schnappschüsse werden zerstört und der belegte Platz wieder freigegeben durch den Befehl `zfs destroy` *`dataset@snapshot`*. Durch hinzufügen von `-r` werden alle Schnappschüsse rekursiv gelöscht, die den gleichen Namen wie das Eltern-Dataset besitzen. Mit der Option `-n -v` wird eine Liste von Schnappschüssen, die gelöscht werden würden, zusammen mit einer geschätzten Menge an zurückgewonnenem Speicherplatz angezeigt, ohne die eigentliche Zerstöroperation wirklich durchzuführen.

20.4.6. Klone verwalten

Ein Klon ist eine Kopie eines Schnappschusses, der mehr wie ein reguläres Dataset behandelt wird. Im Gegensatz zu Schnappschüssen kann man von einem Klon nicht nur lesen, er ist eingehängt und kann seine eigenen Eigenschaften haben. Sobald ein Klon mittels `zfs clone` erstellt wurde, lässt sich der zugrundeliegende Schnappschuss nicht mehr zerstören. Die Eltern-/Kindbeziehung zwischen dem Klon und dem Schnappschuss kann über `zfs promote` aufgelöst werden. Nachdem ein Klon auf diese Weise befördert wurde, wird der Schnappschuss zum Kind des Klons, anstatt des ursprünglichen Datasets. Dies wird die Art und Weise, wie der Speicherplatz berechnet wird, verändern, jedoch nicht den bereits belegten Speicher anpassen. Der Klon kann an einem beliebigen Punkt innerhalb der ZFS-Dateisystemhierarchie eingehängt werden, nur nicht unterhalb der ursprünglichen Stelle des Schnappschusses.

Um diese Klon-Funktionalität zu demonstrieren, wird dieses Beispiel-Dataset verwendet:

```
# zfs list -rt all camino/home/joe
NAME                    USED  AVAIL  REFER  MOUNTPOINT
camino/home/joe         108K   1.3G    87K  /usr/home/joe
camino/home/joe@plans    21K      -  85.5K  -
camino/home/joe@backup    0K      -    87K  -
```

Ein typischer Einsatzzweck für Klone ist das experimentieren mit einem bestimmten Dataset, während der Schnappschuss beibehalten wird für den Fall, dass etwas schiefgeht. Da Schnappschüsse nicht verändert werden können, wird ein Lese-/Schreibklon des Schnappschusses angelegt. Nachdem das gewünschte Ergebnis im Klon erreicht wurde, kann der Klon zu einem Dataset ernannt und das alte Dateisystem entfernt werden. Streng genommen ist das nicht nötig, da der Klon und das Dataset ohne Probleme miteinander koexistieren können.

```
# zfs clone camino/home/joe @backup camino/home/joenew
# ls /usr/home/joe*
/usr/home/joe:
backup.txz     plans.txt

/usr/home/joenew:
backup.txz     plans.txt
# df -h /usr/home
Filesystem          Size    Used   Avail Capacity  Mounted on
usr/home/joe        1.3G     31k    1.3G     0%    /usr/home/joe
usr/home/joenew     1.3G     31k    1.3G     0%    /usr/home/joenew
```

Nachdem ein Klon erstellt wurde, stellt er eine exakte Kopie des Datasets zu dem Zeitpunkt dar, als der Schnappschuss angelegt wurde. Der Klon kann nun unabhängig vom ursprünglichen Dataset geändert werden. Die einzige Verbindung zwischen den beiden ist der Schnappschuss. ZFS zeichnet diese Verbindung in der Eigenschaft namens `origin` auf. Sobald die Abhängigkeit zwischen dem Schnappschuss und dem Klon durch das Befördern des Klons mittels `zfs promote` entfernt wurde, wird auch die `origin`-Eigenschaft des Klons entfernt, da es sich nun um ein eigenständiges Dataset handelt. Dieses Beispiel demonstriert dies:

```
# zfs get origin camino/home/joenew
NAME                 PROPERTY  VALUE                    SOURCE
camino/home/joenew   origin    camino/home/joe@backup   -
```

```
# zfs promote camino/home/joenew
# zfs get origin camino/home/joenew
NAME                 PROPERTY  VALUE   SOURCE
camino/home/joenew   origin    -       -
```

Nachdem ein paar Änderungen, wie beispielsweise das Kopieren von `loader.conf` auf den beförderten Klon vorgenommen wurden, wird das alte Verzeichnis in diesem Fall überflüssig. Stattdessen kann der beförderte Klon diesen ersetzen. Dies kann durch zwei aufeinanderfolgende Befehl geschehen: `zfs destroy` auf dem alten Dataset und `zfs rename` auf dem Klon, um diesen genauso wie das alte Dataset zu benennen (es kann auch einen ganz anderen Namen erhalten).

```
# cp /boot/defaults/loader.conf /usr/home/joenew
# zfs destroy -f camino/home/joe
# zfs rename camino/home/joenew camino/home/joe
# ls /usr/home/joe
backup.txz     loader.conf     plans.txt
# df -h /usr/home
Filesystem          Size    Used   Avail Capacity  Mounted on
usr/home/joe        1.3G    128k    1.3G     0%    /usr/home/joe
```

Der geklonte Schnappschuss wird jetzt wie ein gewöhnliches Dataset behandelt. Es enthält alle Daten aus dem ursprünglichen Schnappschuss inklusive der Dateien, die anschließend hinzugefügt wurden, wie `loader.conf`. Klone können in unterschiedlichen Szenarien eingesetzt werden, um nützliche Eigenschaften für ZFS-Anwender zur Verfügung zu stellen. Zum Beispiel können Jails als Schnappschüsse bereitgestellt werden, die verschiedene Arten von installierten Anwendungen anbieten. Anwender können diese Schnappschüsse klonen und ihre eigenen Anwendungen nach Belieben hinzufügen. Sobald sie mit den Änderungen zufrieden sind, können die Klone zu vollständigen Datasets ernannt werden und dem Anwender zur Verfügung gestellt werden, als würde es sich um echte Datasets handeln. Das spart Zeit und Administrationsaufwand, wenn diese Jails auf diese Weise zur Verfügung gestellt werden.

20.4.7. Replikation

Daten auf einem einzigen Pool an einem Platz aufzubewahren, setzt diese dem Risiko aus, gestohlen oder Opfer von Naturgewalten zu werden, sowie menschlichem Versagen auszusetzen. Regelmäßige Sicherungen des gesamten Pools ist daher unerlässlich. ZFS bietet eine Reihe von eingebauten Serialisierungsfunktionen an, die in der Lage ist, eine Repräsentation der Daten als Datenstrom auf die Standardausgabe zu schreiben. Mit dieser Methode ist es nicht nur möglich, die Daten auf einen anderen Pool zu schicken, der an das lokale System angeschlossen ist, sondern ihn auch über ein Netzwerk an ein anderes System zu senden. Schnappschüsse stellen dafür die Replikationsbasis bereit (lesen Sie dazu den Abschnitt zu ZFS snapshots). Die Befehle, die für die Replikation verwendet werden, sind `zfs send` und `zfs receive`.

Diese Beispiele demonstrieren die Replikation von ZFS anhand dieser beiden Pools:

```
# zpool list
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
backup  960M    77K   896M     0%  1.00x  ONLINE  -
mypool  984M  43.7M   940M     4%  1.00x  ONLINE  -
```

Der Pool namens *`mypool`* ist der primäre Pool, auf den regelmäßig Daten geschrieben und auch wieder gelesen werden. Ein zweiter Pool, genannt *`backup`* wird verwendet, um als Reserve zu dienen im Falle, dass der primäre Pool nicht zur Verfügung steht. Beachten Sie, dass diese Ausfallsicherung nicht automatisch von ZFS durchgeführt wird, sondern manuell von einem Systemadministrator bei Bedarf eingerichtet werden muss. Ein Schnappschuss wird verwendet, um einen konsistenten Zustand des Dateisystems, das repliziert werden soll, zu erzeugen. Sobald ein Schnappschuss von *`mypool`* angelegt wurde, kann er auf den *`backup`*-Pool abgelegt werden. Nur Schnappschüsse lassen sich auf diese Weise replizieren. Änderungen, die seit dem letzten Schnappschuss entstanden sind, werden nicht mit repliziert.

```
# zfs snapshot mypool@backup1
# zfs list -t snapshot
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool@backup1             0      -  43.6M  -
```

Da nun ein Schnappschuss existiert, kann mit `zfs send` ein Datenstrom, der den Inhalt des Schnappschusses repräsentiert, erstellt werden. Dieser Datenstrom kann als Datei gespeichert oder von einem anderen Pool empfangen werden. Der Datenstrom wird auf die Standardausgabe geschrieben, muss jedoch in eine Datei oder in eine Pipe umgeleitet werden, sonst wird ein Fehler produziert:

```
# zfs send mypool@backup1
Error: Stream can not be written to a terminal.
You must redirect standard output.
```

Um ein Dataset mit `zfs send` zu replizieren, leiten Sie dieses in eine Datei auf dem eingehängten Backup-Pool um. Stellen Sie sicher, dass der Pool genug freien Speicherplatz besitzt, um die Größe des gesendeten Schnappschusses aufzunehmen. Das beinhaltet alle Daten im Schnappschuss, nicht nur die Änderungen zum vorherigen Schnappschuss.

```
# zfs send mypool@backup1 > /backup/backup1
# zpool list
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
backup  960M  63.7M   896M     6%  1.00x  ONLINE  -
mypool  984M  43.7M   940M     4%  1.00x  ONLINE  -
```

Das Kommando `zfs send` transferierte alle Daten im *backup1*-Schnappschuss auf den Pool namens *backup*. Erstellen und senden eines Schnappschusses kann automatisch von cron(8) durchgeführt werden.

Anstatt die Sicherungen als Archivdateien zu speichern, kann ZFS diese auch als aktives Dateisystem empfangen, was es erlaubt, direkt auf die gesicherten Daten zuzugreifen. Um an die eigentlichen Daten in diesem Strom zu gelangen, wird `zfs receive` benutzt, um den Strom wieder in Dateien und Verzeichnisse umzuwandeln. Das Beispiel unten kombiniert `zfs send` und `zfs receive` durch eine Pipe, um die Daten von einem Pool auf den anderen zu kopieren. Die Daten können direkt auf dem empfangenden Pool verwendet werden, nachdem der Transfer abgeschlossen ist. Ein Dataset kann nur auf ein leeres Dataset repliziert werden.

```
# zfs snapshot mypool@replica1
# zfs send -v mypool@replica1 | zfs receive backup/mypool
send from @ to mypool@replica1 estimated size is 50.1M
total estimated size is 50.1M
TIME        SENT   SNAPSHOT

# zpool list
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
backup  960M  63.7M   896M     6%  1.00x  ONLINE  -
mypool  984M  43.7M   940M     4%  1.00x  ONLINE  -
```

20.4.7.1. Inkrementelle Sicherungen

Die Unterschiede zwischen zwei Schnappschüssen kann `zfs send` ebenfalls erkennen und nur diese übertragen. Dies spart Speicherplatz und Übertragungszeit. Beispielsweise:

```
# zfs snapshot mypool@replica2
# zfs list -t snapshot
NAME                    USED  AVAIL  REFER  MOUNTPOINT
mypool@replica1         5.72M      -  43.6M  -
mypool@replica2             0      -  44.1M  -
# zpool list
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
backup  960M  61.7M   898M     6%  1.00x  ONLINE  -
mypool  960M  50.2M   910M     5%  1.00x  ONLINE  -
```

Ein zweiter Schnappschuss genannt *replica2* wurde angelegt. Dieser zweite Schnappschuss enthält nur die Änderungen, die zwischen dem jetzigen Stand des Dateisystems und dem vorherigen Schnappschuss, *replica1*, vorgenommen wurden. Durch `zfs send -i` und die Angabe des Schnappschusspaares wird ein inkrementeller Replikationsstrom erzeugt, welcher nur die Daten enthält, die sich geändert haben. Das kann nur erfolgreich sein, wenn der initiale Schnappschuss bereits auf der Empfängerseite vorhanden ist.

```
# zfs send -v -i mypool@replica1 mypool@replica2 | zfs receive /backup/mypool
```

```
send from @replica1 to mypool@replica2 estimated size is 5.02M
total estimated size is 5.02M
TIME        SENT   SNAPSHOT

# zpool list
NAME    SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
backup  960M  80.8M   879M     8%  1.00x  ONLINE  -
mypool  960M  50.2M   910M     5%  1.00x  ONLINE  -

# zfs list
NAME                         USED  AVAIL  REFER  MOUNTPOINT
backup                      55.4M   240G   152K  /backup
backup/mypool               55.3M   240G  55.2M  /backup/mypool
mypool                      55.6M  11.6G  55.0M  /mypool

# zfs list -t snapshot
NAME                                    USED  AVAIL  REFER  MOUNTPOINT
backup/mypool@replica1                  104K      -  50.2M  -
backup/mypool@replica2                     0      -  55.2M  -
mypool@replica1                         29.9K      -  50.0M  -
mypool@replica2                            0      -  55.0M  -
```

Der inkrementelle Datenstrom wurde erfolgreich übertragen. Nur die Daten, die verändert wurden, sind übertragen worden, anstatt das komplette *replica1*. Nur die Unterschiede wurden gesendet, was weniger Zeit und Speicherplatz in Anspruch genommen hat, statt jedesmal den gesamten Pool zu kopieren. Das ist hilfreich wenn langsame Netzwerke oder Kosten für die übertragene Menge Bytes in Erwägung gezogen werden müssen.

Ein neues Dateisystem, *backup/mypool*, ist mit allen Dateien und Daten vom Pool *mypool* verfügbar. Wenn die Option `-P` angegeben wird, werden die Eigenschaften des Datasets kopiert, einschließlich der Komprimierungseinstellungen, Quotas und Einhängepunkte. Wird die Option `-R` verwendet, so werden alle Kind-Datasets des angegebenen Datasets kopiert, zusammen mit ihren Eigenschaften. Senden und Empfangen kann automatisiert werden, so dass regelmäßig Sicherungen auf dem zweiten Pool angelegt werden.

20.4.7.2. Verschlüsselte Sicherungen über SSH senden

Datenströme über das Netzwerk zu schicken ist eine gute Methode, um Sicherungen außerhalb des Systems anzulegen. Jedoch ist dies auch mit einem Nachteil verbunden. Daten, die über die Leitung verschickt werden, sind nicht verschlüsselt, was es jedem erlaubt, die Daten abzufangen und die Ströme wieder zurück in Daten umzuwandeln, ohne dass der sendende Benutzer davon etwas merkt. Dies ist eine unerwünschte Situation, besonders wenn die Datenströme über das Internet auf ein entferntes System gesendet werden. SSH kann benutzt werden, um durch Verschlüsselung geschützte Daten über eine Netzwerkverbindung zu übertragen. Da ZFS nur die Anforderung hat, dass der Strom von der Standardausgabe umgeleitet wird, ist es relativ einfach, diesen durch SSH zu leiten. Um den Inhalt des Dateisystems während der Übertragung und auf dem entfernten System weiterhin verschlüsselt zu lassen, denken Sie über den Einsatz von PEFS nach.

Ein paar Einstellungen und Sicherheitsvorkehrungen müssen zuvor abgeschlossen sein. Es werden hier nur die nötigen Schritte für die `zfs send`-Aktion gezeigt. Weiterführende Informationen zu SSH, gibt es im Kapitel Abschnitt 14.10, „OpenSSH“.

Die folgende Konfiguration wird benötigt:

- Passwortloser SSH-Zugang zwischen dem sendenden und dem empfangenden Host durch den Einsatz von SSH-Schlüsseln.
- Normalerweise werden die Privilegien des `root`-Benutzers gebraucht, um Strom zu senden und zu empfangen. Das beinhaltet das Anmelden auf dem empfangenden System als `root`. Allerdings ist das Anmelden als `root` aus Sicherheitsgründen standardmäßig deaktiviert. Mit ZFS Delegation lassen sich nicht-`root`-Benutzer auf jedem System einrichten, welche die nötigen Rechte besitzen, um die Sende- und Empfangsoperation durchzuführen.
- Auf dem sendenden System:

```
# zfs allow -u someuser send,snapshot   mypool
```

- Um den Pool einzuhängen, muss der unprivilegierte Benutzer das Verzeichnis besitzen und gewöhnliche Benutzern muss die Erlaubnis gegeben werden, das Dateisystem einzuhängen. Auf dem empfangenden System nehmen Sie dazu die folgenden Einstellungen vor:

```
# sysctl vfs.usermount=1
vfs.usermount: 0 -> 1
# echo vfs.usermount=1 >> /etc/sysctl.conf
# zfs create recvpool/backup
# zfs allow -u someuser create,mount,receive recvpool/backup
# chown someuser /recvpool/backup
```

Der unprivilegierte Benutzer hat jetzt die Fähigkeit, Datasets zu empfangen und einzuhängen und das *home*-Dataset auf das entfernte System zu replizieren:

```
% zfs snapshot -r mypool/home@monday
% zfs send -R mypool/home@monday | ssh someuser@backuphost zfs recv -dvu recvpool/backup
```

Ein rekursiver Schnappschuss namens *monday* wird aus dem Dataset *home* erstellt, dass auf dem Pool *mypool* liegt. Es wird dann mit `zfs send -R` gesendet, um das Dataset, alle seine Kinder, Schnappschüsse, Klone und Einstellungen in den Strom mit aufzunehmen. Die Ausgabe wird an das wartende System *backuphost* mittels `zfs receive` durch SSH umgeleitet. Die Verwendung des Fully Qulified Domänennamens oder der IP-Adresse wird empfohlen. Die empfangende Maschine schreibt die Daten auf das *backup*-Dataset im *recvpool*-Pool. Hinzufügen der Option `-d` zu `zfs recv` überschreibt den Namen des Pools auf der empfangenden Seite mit dem Namen des Schnappschusses. Durch Angabe von `-u` wird das Dateisystem nicht auf der Empfängerseite eingehängt. Wenn `-v` enthalten ist, werden mehr Details zum Transfer angezeigt werden, einschließlich der vergangenen Zeit und der Menge an übertragenen Daten.

20.4.8. Dataset-, Benutzer- und Gruppenquotas

Dataset-Quotas werden eingesetzt, um den Speicherplatz einzuschränken, den ein bestimmtes Dataset verbrauchen kann. Referenz-Quotas funktionieren auf eine ähnliche Weise, jedoch wird dabei der Speicherplatz des Datasets selbst gezählt, wobei Schnappschüsse und Kind-Datasets dabei ausgenommen sind. Ähnlich dazu werden Benutzer- und Gruppen-Quotas dazu verwendet, um Benutzer oder Gruppen daran zu hindern, den gesamten Speicherplatz im Pool oder auf dem Dataset zu verbrauchen.

Um ein 10 GB großes Quota auf dem Dataset `storage/home/bob` zu erzwingen, verwenden Sie folgenden Befehl:

```
# zfs set quota=10G storage/home/bob
```

Um ein Referenzquota von 10 GB für `storage/home/bob` festzulegen, geben Sie ein:

```
# zfs set refquota=10G storage/home/bob
```

Um das Quota für `storage/home/bob` wieder zu entfernen:

```
# zfs set quota=none storage/home/bob
```

Das generelle Format ist `userquota@`*user*`=`*size* und der Name des Benutzers muss in einem der folgenden Formate vorliegen:

- POSIX-kompatibler Name wie *joe*.
- POSIX-numerische ID wie *789*.
- SID-Name wie *joe.bloggs@example.com*.
- SID-numerische ID wie *S-1-123-456-789* .

Um beispielsweise ein Benutzerquota von 50 GB für den Benutzer names *joe* zu erzwingen:

```
# zfs set userquota@joe=50G
```

Um jegliche Quotas zu entfernen:

```
# zfs set userquota@joe=none
```

Anmerkung

Benutzerquota-Eigenschaften werden nicht von `zfs get all` dargestellt. Nicht-`root`-Benutzer können nur ihre eigenen Quotas sehen, ausser ihnen wurde das `userquota`-Privileg zugeteilt. Benutzer mit diesem Privileg sind in der Lage, jedermanns Quota zu sehen und zu verändern.

Das generelle Format zum Festlegen einer Gruppenquota lautet: `groupquota@`*`group`*`=`*`size`*.

Um ein Quota für die Gruppe *`firstgroup`* von 50 GB zu setzen, geben Sie ein:

```
# zfs set groupquota@firstgroup=50G
```

Um eine Quota für die Gruppe *`firstgroup`*zu setzen oder sicherzustellen, dass diese nicht gesetzt ist, verwenden Sie stattdessen:

```
# zfs set groupquota@firstgroup=none
```

Genau wie mit der Gruppenquota-Eigenschaft, werden nicht-`root`-Benutzer nur die Quotas sehen, die den Gruppen zugeordnet ist, in denen Sie Mitglied sind. Allerdings ist `root` oder ein Benutzer mit dem `groupquota`-Privileg in der Lage, die Quotas aller Gruppen zu sehen und festzusetzen.

Um die Menge an Speicherplatz zusammen mit der Quota anzuzeigen, die von jedem Benutzer auf dem Dateisystem oder Schnappschuss verbraucht wird, verwenden Sie `zfs userspace`. Für Gruppeninformationen, nutzen Sie `zfs groupspace`. Für weitere Informationen zu unterstützten Optionen oder wie sich nur bestimmte Optionen anzeigen lassen, lesen Sie zfs(1).

Benutzer mit ausreichenden Rechten sowie `root` können das Quota für `storage/home/bob` anzeigen lassen:

```
# zfs get quota storage/home/bob
```

20.4.9. Reservierungen

Reservierungen garantieren ein Minimum an Speicherplatz, der immer auf dem Dataset verfügbar sein wird. Der reservierte Platz wird nicht für andere Datasets zur Verfügung stehen. Diese Eigenschaft kann besonders nützlich sein, um zu gewährleisten, dass freier Speicherplatz für ein wichtiges Dataset oder für Logdateien bereit steht.

Das generelle Format der `reservation`-Eigenschaft ist `reservation=`*`size`*. Um also eine Reservierung von 10 GB auf `storage/home/bob` festzulegen, geben Sie Folgendes ein:

```
# zfs set reservation=10G storage/home/bob
```

Um die Reservierung zu beseitigen:

```
# zfs set reservation=none storage/home/bob
```

Das gleiche Prinzip kann auf die `refreservation`-Eigenschaft angewendet werden, um eine Referenzreservierung mit dem generellen Format `refreservation=`*`size`* festzulegen.

Dieser Befehl zeigt die Reservierungen oder Referenzreservierungen an, die auf `storage/home/bob` existieren:

```
# zfs get reservation storage/home/bob
# zfs get refreservation storage/home/bob
```

20.4.10. Komprimierung

ZFS bietet transparente Komprimierung. Datenkomprimierung auf Blockebene während diese gerade geschrieben werden, spart nicht nur Plattenplatz ein, sondern kann auch den Durchsatz der Platte steigern. Falls Daten zu

25% komprimiert sind, jedoch die komprimierten Daten im gleichen Tempo wie ihre unkomprimierte Version, resultiert das in einer effektiven Schreibgeschwindigkeit von 125%. Komprimierung kann auch eine Alternative zu Deduplizierung darstellen, da es viel weniger zusätzlichen Hauptspeicher benötigt.

ZFS bietet mehrere verschiedene Kompressionsalgorithmen an, jede mit unterschiedlichen Kompromissen. Mit der Einführung von LZ4-Komprimierung in ZFS v5000, ist es möglich, Komprimierung für den gesamten Pool zu aktivieren, ohne die großen Geschwindigkeitseinbußen der anderen Algorithmen. Der größte Vorteil von LZ4 ist die Eigenschaft *früher Abbruch*. Wenn LZ4 nicht mindestens 12,5% Komprimierung im ersten Teil der Daten erreicht, wird der Block unkomprimiert geschrieben, um die Verschwendung von CPU-Zyklen zu vermeiden, weil die Daten entweder bereits komprimiert sind oder sich nicht komprimieren lassen. Für Details zu den verschiedenen verfügbaren Komprimierungsalgorithmen in ZFS, lesen Sie den Eintrag Komprimierung im Abschnitt Terminologie

Der Administrator kann die Effektivität der Komprimierung über eine Reihe von Dataset-Eigenschaften überwachen.

```
# zfs get used,compressratio,compression,logicalused    mypool/compressed_dataset
NAME        PROPERTY          VALUE     SOURCE
mypool/compressed_dataset  used              449G      -
mypool/compressed_dataset  compressratio     1.11x     -
mypool/compressed_dataset  compression       lz4       local
mypool/compressed_dataset  logicalused       496G      -
```

Dieses Dataset verwendet gerade 449 GB Plattenplatz (used-Eigenschaft. Ohne Komprimierung würde es stattdessen 496 GB Plattenplatz belegen (`logicalused`). Das ergibt eine Kompressionsrate von 1,11:1.

Komprimierung kann einen unerwarteten Nebeneffekt haben, wenn diese mit Benutzerquotas kombiniert wird. Benutzerquotas beschränken, wieviel Speicherplatz ein Benutzer auf einem Dataset verbrauchen kann. Jedoch basieren die Berechnungen darauf, wieviel Speicherplatz *nach der Komprimierung* belegt ist. Wenn also ein Benutzer eine Quota von10 GB besitzt und 10 GB von komprimierbaren Daten schreibt, wird dieser immer noch in der Lage sein, zusätzliche Daten zu speichern. Wenn später eine Datei aktualisiert wird, beispielsweise eine Datenbank, mit mehr oder weniger komprimierbaren Daten, wird sich die Menge an verfügbarem Speicherplatz ändern. Das kann in einer merkwürdigen Situation resultieren, in welcher der Benutzer nicht die eigentliche Menge an Daten (die Eigenschaft `logicalused`) überschreitet, jedoch die Änderung in der Komprimierung dazu führt, dass das Quota-Limit erreicht ist.

Kompression kann ebenso unerwartet mit Sicherungen interagieren. Quotas werden oft verwendet, um einzuschränken, wieviele Daten gespeichert werden können um sicherzustellen, dass ausreichend Speicherplatz für die Sicherung vorhanden ist. Wenn jedoch Quotas Komprimierung nicht berücksichtigen, werden womöglich mehr Daten geschrieben als in der unkomprimierten Sicherung Platz ist.

20.4.11. Deduplizierung

Wenn aktiviert, verwendet Deduplizierung die Prüfsumme jedes Blocks, um Duplikate dieses Blocks zu ermitteln. Sollte ein neuer Block ein Duplikat eines existierenden Blocks sein, dann schreibt ZFS eine zusätzliche Referenz auf die existierenden Daten anstatt des kompletten duplizierten Blocks. Gewaltige Speicherplatzeinsparungen sind möglich wenn die Daten viele Duplikate von Dateien oder wiederholte Informationen enthalten. Seien Sie gewarnt: Deduplizierung benötigt eine extrem große Menge an Hauptspeicher und die meistens Einsparungen können stattdessen durch das Aktivieren von Komprimierung erreicht werden.

Um Deduplizierung zu aktivieren, setzen Sie die `dedup`-Eigenschaft auf dem Zielpool:

```
# zfs set dedup=on  pool
```

Nur neu auf den Pool geschriebene Daten werden dedupliziert. Daten, die bereits auf den Pool geschrieben wurden, werden nicht durch das Aktivieren dieser Option dedupliziert. Ein Pool mit einer gerade aktivierten Deduplizierung wird wie in diesem Beispiel aussehen:

```
# zpool list
NAME  SIZE ALLOC  FREE CAP DEDUP HEALTH ALTROOT
pool 2.84G 2.19M 2.83G  0% 1.00x ONLINE -
```

Die Spalte `DEDUP` zeigt das aktuelle Verhältnis der Deduplizierung für diesen Pool an. Ein Wert von `1.00x` zeigt an, dass die Daten noch nicht dedupliziert wurden. Im nächsten Beispiel wird die Ports-Sammlung dreimal in verschiedene Verzeichnisse auf dem deduplizierten Pool kopiert.

```
# zpool list
for d in dir1 dir2 dir3; do
for> mkdir $d && cp -R /usr/ports $d &
for> done
```

Redundante Daten werden erkannt und dedupliziert:

```
# zpool list
NAME SIZE  ALLOC FREE CAP DEDUP HEALTH ALTROOT
pool 2.84G 20.9M 2.82G 0% 3.00x ONLINE -
```

Die `DEDUP`-Spalte zeigt einen Faktor von `3.00x`. Mehrere Kopien der Ports-Sammlung wurden erkannt und dedupliziert, was nur ein Drittel des Speicherplatzes benötigt. Das Potential für Einsparungen beim Speicherplatz ist enorm, wird jedoch damit erkauft, dass genügend Speicher zur Verfügung stehen muss, um die deduplizierten Blöcke zu verwalten.

Deduplizierung ist nicht immer gewinnbringend, besonders wenn die Daten auf dem Pool nicht redundant sind. ZFS kann potentielle Speicherplatzeinsparungen durch Deduplizierung auf einem Pool simulieren:

```
# zdb -S pool
Simulated DDT histogram:

bucket              allocated                       referenced
______   ______________________________   ______________________________
refcnt   blocks   LSIZE   PSIZE   DSIZE   blocks   LSIZE   PSIZE   DSIZE
------   ------   -----   -----   -----   ------   -----   -----   -----
     1    2.58M    289G    264G    264G    2.58M    289G    264G    264G
     2     206K   12.6G   10.4G   10.4G     430K   26.4G   21.6G   21.6G
     4    37.6K    692M    276M    276M     170K   3.04G   1.26G   1.26G
     8    2.18K   45.2M   19.4M   19.4M    20.0K    425M    176M    176M
    16      174   2.83M   1.20M   1.20M    3.33K   48.4M   20.4M   20.4M
    32       40   2.17M    222K    222K    1.70K   97.2M   9.91M   9.91M
    64        9     56K   10.5K   10.5K      865   4.96M    948K    948K
   128        2   9.50K      2K      2K      419   2.11M    438K    438K
   256        5   61.5K     12K     12K    1.90K   23.0M   4.47M   4.47M
    1K        2      1K      1K      1K    2.98K   1.49M   1.49M   1.49M
 Total    2.82M    303G    275G    275G    3.20M    319G    287G    287G

dedup = 1.05, compress = 1.11, copies = 1.00, dedup * compress / copies = 1.16
```

Nachdem `zdb -S` die Analyse des Pool abgeschlossen hat, zeigt es die Speicherplatzeinsparungen, die durch aktivierte Deduplizierung erreichbar sind, an. In diesem Fall ist `1.16` ein sehr schlechter Faktor, der größtenteils von Einsparungen durch Komprimierung beeinflusst wird. Aktivierung von Deduplizierung auf diesem Pool würde also keine signifikante Menge an Speicherplatz einsparen und ist daher nicht die Menge an Speicher wert, die nötig sind, um zu deduplizieren. Über die Formel *ratio = dedup * compress / copies* kann ein Systemadministrator die Speicherplatzbelegung planen und entscheiden, ob es sich lohnt, den zusätzlichen Hauptspeicher für die Deduplizierung anhand des späteren Workloads aufzuwenden. Wenn sich die Daten verhältnismäßig gut komprimieren lassen, sind die Speicherplatzeinsparungen sehr gut. Es wird empfohlen, in dieser Situation zuerst die Komprimierung zu aktivieren, da diese auch erhöhte Geschwindigkeit mit sich bringt. Aktivieren Sie Deduplizierung nur in solchen Fällen, bei denen die Einsparungen beträchtlich sind und genug Hauptspeicher zur Verfügung steht, um die DDT aufzunehmen.

20.4.12. ZFS und Jails

Um ein ZFS-Dataset einem Jail zuzuweisen, wird der Befehl `zfs jail` und die dazugehörige Eigenschaft `jailed` verwendet. Durch Angabe von `zfs jail` *`jailid`* wird ein Dataset dem spezifizierten Jail zugewiesen und kann mit `zfs unjail` wieder abgehängt werden. Damit das Dataset innerhalb der Jail kontrolliert werden kann, muss die

Eigenschaft `jailed` gesetzt sein. Sobald ein Dataset sich im Jail befindet, kann es nicht mehr länger auf dem Hostsystem eingehängt werden, da es Einhängepunkte aufweisen könnte, welche die Sicherheit des Systems gefährden.

20.5. Delegierbare Administration

Ein umfassendes System zur Berechtigungsübertragung erlaubt unprivilegierten Benutzern, ZFS-Administrationsaufgaben durchzuführen. Beispielsweise, wenn jedes Heimatverzeichnis eines Benutzers ein Dataset ist, können Benutzer das Recht darin erhalten, Schnappschüsse zu erstellen und zu zerstören. Einem Benutzer für die Sicherung kann die Erlaubnis eingeräumt werden, die Replikationseigenschaft zu verwenden. Einem Skript zum Sammeln von Speicherplatzverbrauch kann die Berechtigung gegeben werden, nur auf die Verbrauchsdaten aller Benutzer zuzugreifen. Es ist sogar möglich, die Möglichkeit zum Delegieren zu delegieren. Die Berechtigung zur Delegation ist für jedes Unterkommando und die meisten Eigenschaften möglich.

20.5.1. Delegieren, ein Dataset zu erstellen

`zfs allow` *`someuser`* `create` *`mydataset`* gibt dem angegebenen Benutzer die Berechtigung, Kind-Datasets unter dem ausgewählten Elterndataset anzulegen. Es gibt einen Haken: ein neues Dataset anzulegen beinhaltet, dass es eingehängt wird. Dies bedeutet, dass FreeBSDs `vfs.usermount` sysctl(8) auf `1` gesetzt wird, um nicht-root Benutzern zu erlauben, Dateisysteme einzubinden. Es gibt eine weitere Einschränkung um Missbrauch zu verhindern: nicht-`root` Benutzer müssen Besitzer des Einhängepunktes sein, an dem das Dateisystem eingebunden werden soll.

20.5.2. Delegationsberechtigung delegieren

`zfs allow` *`someuser`* `allow` *`mydataset`* gibt dem angegebenen Benutzer die Fähigkeit, jede Berechtigung, die er selbst auf dem Dataset oder dessen Kindern besitzt, an andere Benutzer weiterzugeben. Wenn ein Benutzer die `snapshot`- und die `allow`-Berechtigung besitzt, kann dieser dann die `snapshot`-Berechtigung an andere Benutzer delegieren.

20.6. Themen für Fortgeschrittene

20.6.1. Anpassungen

Eine Reihe von Anpassungen können vorgenommen werden, um ZFS unter verschiedenen Belastungen während des Betriebs bestmöglich einzustellen.

- *`vfs.zfs.arc_max`* - Maximale Größe des ARC. Die Voreinstellung ist der gesamte RAM weniger 1 GB oder die Hälfte vom RAM, je nachdem, was mehr ist. Allerdings sollte ein niedriger Wert verwendet werden, wenn das System weitere Dienste oder Prozesse laufen lässt, welche Hauptspeicher benötigen. Dieser Wert kann nur zur Bootzeit eingestellt werden und wird in `/boot/loader.conf` festgelegt.
- *`vfs.zfs.arc_meta_limit`* - Schränkt die Menge des ARC ein, welche für die Speicherung von Metadaten verwendet wird. Die Voreinstellung ist ein Viertel von `vfs.zfs.arc_max`. Diesen Wert zu erhöhen steigert die Geschwindigkeit, wenn die Arbeitslast Operationen auf einer großen Menge an Dateien und Verzeichnissen oder häufigen Metadatenoperationen beinhaltet. Jedoch bedeutet dies auch weniger Dateidaten, die in den ARC passen. Dieser Wert kann nur zur Bootzeit angepasst werden und wird in `/boot/loader.conf` gesetzt.
- *`vfs.zfs.arc_min`* - Minimale Größe des ARC. Der Standard beträgt die Hälfte von `vfs.zfs.arc_meta_limit`. Passen Sie diesen Wert an, um zu verhindern, dass andere Anwendungen den gesamten ARC verdrängen. Dieser Wert kann nur zur Bootzeit geändert und muss in `/boot/loader.conf` festgelegt werden.
- *`vfs.zfs.vdev.cache.size`* - Eine vorallokierte Menge von Speicher, die als Cache für jedes Gerät im Pool reserviert wird. Die Gesamtgröße von verwendetem Speicher ist dieser Wert multipliziert mit der Anzahl an Geräten. Nur zur Bootzeit kann dieser Wert angepasst werden und wird in `/boot/loader.conf` eingestellt.
- *`vfs.zfs.min_auto_ashift`* - Minimaler `ashift`-Wert (Sektorgröße), welche zur Erstellungszeit des Pools automatisch verwendet wird. Der Wert ist ein Vielfaches zur Basis Zwei. Der Standardwert von `9` repräsentiert `2^9 =`

`512`, eine Sektorgröße von 512 Bytes. Um *write amplification* zu vermeiden und die bestmögliche Geschwindigkeit zu erhalten, setzen Sie diesen Wert auf die größte Sektorgröße, die bei einem Gerät im Pool vorhanden ist.

Viele Geräte besitzen 4 KB große Sektoren. Die Verwendung der Voreinstellung `9` bei `ashift` mit diesen Geräten resultiert in einer write amplification auf diesen Geräten. Daten, welche in einem einzelnen 4 KB Schreibvorgang Platz finden würden, müssen stattdessen in acht 512-byte Schreibvorgänge aufgeteilt werden. ZFS versucht, die allen Geräten zugrundeliegende Sektorgröße während der Poolerstellung zu lesen, jedoch melden viele Geräte mit 4 KB Sektoren, dass ihre Sektoren aus Kompatibilitätsgründen 512 Bytes betragen. Durch das Setzen von `vfs.zfs.min_auto_ashift` auf `12` (`2^12 = 4096`) bevor der Pool erstellt wird, zwingt ZFS dazu, für diese Geräte 4 KB Blöcke für bessere Geschwindigkeit zu nutzen.

Erzwingen von 4 KB Blöcken ist ebenfalls hilfreich auf Pools bei denen Plattenaufrüstungen geplant sind. Zukünftige Platten werden wahrscheinlich 4 KB große Sektoren und der Wert von `ashift` lässt sich nach dem Erstellen des Pools nicht mehr ändern.

In besonderen Fällen ist die kleinere Blockgröße von 512-Byte vorzuziehen. Weniger Daten werden bei kleinen, zufälligen Leseoperationen übertragen, was besonders bei 512-Byte großen Platten für Datenbanken oder Plattenplatz für virtuelle Maschinen der Fall ist. Dies kann bessere Geschwindigkeit bringen, ganz besonders wenn eine kleinere ZFS record size verwendet wird.

- *`vfs.zfs.prefetch_disable`* - Prefetch deaktivieren. Ein Wert von `0` bedeutet aktiviert und `1` heißt deaktiviert. Die Voreinstellung ist `0`, außer, das System besitzt weniger als 4 GB RAM. Prefetch funktioniert durch das Lesen von grösseren Blöcken in den ARC als angefordert wurden, in der Hoffnung, dass diese Daten ebenfalls bald benötigt werden. Wenn die I/O-Last viele große Mengen von zufälligen Leseoperationen beinhaltet, ist das Deaktivieren von prefetch eine Geschwindigkeitssteigerung durch die Reduzierung von unnötigen Leseoperationen. Dieser Wert kann zu jeder Zeit über sysctl(8) angepasst werden.

- *`vfs.zfs.vdev.trim_on_init`* - Steuert, ob neue Geräte, die dem Pool hinzugefügt werden, das `TRIM`-Kommando ausführen sollen. Das beinhaltet die beste Geschwindigkeit und Langlebigkeit für SSDs, benötigt jedoch zusätzliche Zeit. Wenn das Gerät bereits sicher gelöscht wurde, kann durch deaktivieren dieser Option das Hinzufügen neuer Geräte schneller geschehen. Über sysctl(8) lässt sich dieser Wert jederzeit einstellen.

- *`vfs.zfs.vdev.max_pending`* - Begrenzt die Menge von ausstehenden I/O-Anfragen pro Gerät. Ein größerer Wert wird die Gerätewarteschlange für Befehle gefüllt lassen und möglicherweise besseren Durchsatz erzeugen. Ein niedrigerer Wert reduziert die Latenz. Jederzeit kann dieser Wert über sysctl(8) angepasst werden.

- *`vfs.zfs.top_maxinflight`* - Maximale Anzahl von ausstehenden I/Os pro darüberliegendem vdev. Begrenzt die Tiefe Kommandowarteschlange, um hohe Latenzen zu vermeiden. Das Limit ist pro darüberliegendem vdev, was bedeutet, dass das Limit für jeden mirror [527], RAID-Z [527], oder anderes vdev unabhängig gilt. Mit sysctl(8) kann dieser Wert jederzeit angepasst werden.

- *`vfs.zfs.l2arc_write_max`* - Begrenzt die Menge an Daten, die pro Sekunde in den L2ARC geschrieben wird. Durch diese Einstellung lässt sich die Lebensdauer von SSDs erhöhen, indem die Menge an Daten beschränkt wird, die auf das Gerät geschrieben wird. Dieser Wert ist über sysctl(8) zu einem beliebigen Zeitpunkt änderbar.

- *`vfs.zfs.l2arc_write_boost`* - Der Wert dieser Einstellung wird zu `vfs.zfs.l2arc_write_max` [524] addiert und erhöht die Schreibgeschwindigkeit auf die SSD bis der erste Block aus dem L2ARC verdrängt wurde. Diese „Turbo Warmup Phase“ wurde entwickelt, um den Geschwindigkeitsverlust eines leeren L2ARC nach einem Neustart zu reduzieren. Jederzeit kann dieser Wert mit sysctl(8) geändert werden.

- *`vfs.zfs.scrub_delay`* - Anzahl von Ticks an Verzögerung zwischen jedem I/O während eines `scrub`. Um zu gewährleisten, dass ein `scrub` nicht mit die normalen Vorgänge eines Pools beeinträchtigt. Wenn währenddessen andere I/Os durchgeführt werden, wird der `scrub` zwischen jedem Befehl verzögert. Dieser Wert regelt die Gesamtmenge von IOPS (I/Os Per Second), die von `scrub` generiert werden. Die Granularität der Einstellung ist bestimmt durch den Wert von `kern.hz`, welcher standardmäßig auf auf 1000 Ticks pro Sekunde eingestellt ist. Diese Einstellung kann geändert werden, was in einer unterschiedlich effektiven Limitierung der IOPS resultiert. Der Standardwert ist `4`, was ein Limit von 1000 ticks/sec / 4 = 250 IOPS ergibt. Ein Wert von *`20`* würde ein Limit von 1000 ticks/sec / 20 = 50 IOPS ergeben. Die `scrub`-Geschwindigkeit ist nur begrenzt, wenn es kürzlich Aktivi-

tät auf dem Pool gab, wie der Wert von `vfs.zfs.scan_idle` [525] verrät. Zu einem beliebigen Zeitpunkt kann über sysctl(8) eine Änderung an diesem Wert erfolgen.

- *`vfs.zfs.resilver_delay`* - Anzahl an Millisekunden Verzögerung, die zwischen jedem I/O während eines resilver eingefügt wird. Um zu versichern, dass ein resilver nicht die normalen Vorgänge auf dem Pool stört, wird dieser zwischen jedem Kommando verzögert, wenn andere I/Os auf dem Pool passieren. Dieser Wert steuert das Limit der Gesamt-IOPS (I/Os Pro Sekunde), die vom resilver erzeugt werden. Die Granularität der Einstellung wird durch den Wert von `kern.hz` bestimmt, welcher standardmäßig 1000 Ticks pro Sekunde beträgt. Diese Einstellung lässt sich ändern, was in einem unterschiedlich effizienten IOPS-Limit resultiert. Die Voreinstellung ist 2, was ein Limit von 1000 ticks/sec / 2 = 500 IOPS beträgt. Einen Pool wieder in den Zustand Online zu versetzen ist möglicherweise wichtiger wenn eine andere Platte den Pool in den Fault-Zustand versetzt, was Datenverlust zur Folge hat. Ein Wert von 0 wird der resilver-Operation die gleiche Priorität wie anderen Operationen geben, was den Heilungsprozess beschleunigt. Die Geschwindigkeit des resilver wird nur begrenzt, wenn es kürzlich andere Aktivitäten auf dem Pool gab, wie von `vfs.zfs.scan_idle` [525] festgestellt wird. Dieser Wert kann zu jeder Zeit über. sysctl(8) eingestellt werden.

- *`vfs.zfs.scan_idle`* - Anzahl an Millisekunden seit der letzten Operation bevor der Pool als im Leerlauf befindlich deklariert wird. Wenn sich der Pool im Leerlauf befindet, wird die Begrenzung für `scrub` und resilver deaktiviert. Dieser Wert kann mittels sysctl(8) jederzeit angepasst werden.

- *`vfs.zfs.txg.timeout`* - Maximale Anzahl von Sekunden zwischen Transaktionsgruppen (transaction group). Die momentane Transaktionsgruppe wird auf den Pool geschrieben und eine frische Transaktionsgruppe begonnen, wenn diese Menge an Zeit seit der vorherigen Transaktionsgruppe abgelaufen ist. Eine Transaktionsgruppe kann verfrüht ausgelöst werden, wenn genug Daten geschrieben werden. Der Standardwert beträgt 5 Sekunden. Ein größerer Wert kann die Lesegeschwindigkeit durch verzögern von asynchronen Schreibvorgängen verbessern, allerdings kann dies ungleiche Geschwindigkeiten hervorrufen, wenn eine Transaktionsgruppe geschrieben wird. Dieser Wert kann zu einem beliebigen Zeitpunkt mit sysctl(8) geändert werden.

20.6.2. ZFS auf i386

Manche der Eigenschaften, die von ZFS bereitgestellt werden, sind speicherintensiv und benötigen Anpassungen für die maximale Effizienz auf Systemen mit begrenztem RAM.

20.6.2.1. Hauptspeicher

Als absolutes Minimum sollte der gesamte verfügbare Hauptspeicher mindestens ein Gigabyte betragen. Die vorgeschlagene Menge an RAM ist bedingt durch die Poolgröße und welche Eigenschaften von ZFS verwendet werden. Eine Faustregel besagt, dass 1 GB RAM für jedes 1 TB Storage vorgesehen werden sollte. Wenn Deduplizierung zum Einsatz kommt, besagt die Regel, dass 5 GB RAM pro TB an Speicher, der dedupliziert werden soll, bereitgestellt sein muss. Obwohl manche Anwender ZFS mit weniger RAM einsetzen, stürzen Systeme häufiger wegen unzureichendem Hauptspeicher ab. Weitere Anpassungen sind unter Umständen nötig für Systeme mit weniger als die vorgeschlagene Menge an RAM.

20.6.2.2. Kernel-Konfiguration

Wegen des begrenzten Addressraumes der i386™-Plattform müssen ZFS-Anwendern auf der i386™-Architektur diese Option der Kernelkonfigurationsdatei hinzufügen, den Kernel erneut bauen und das System neu starten:

```
options         KVA_PAGES=512
```

Dies erweitert den Addressraum des Kernels, was es erlaubt, die Einstellung `vm.kvm_size` hinter die momentan vorgegebene Grenze von 1 GB oder das Limit von 2 GB für PAE zu bringen. Um den passenden Wert für diese Option zu finden, teilen Sie den gewünschten Addressraum in Megabyte durch vier. In diesem Beispiel beträgt sie `512` für 2 GB.

20.6.2.3. Loader-Anpassungen

Der `kmem`-Addressraum kann auf allen FreeBSD-Architekturen erhöht werden. Auf einem Testsystem mit 1 GB physischen Speichers wurden mit diesen Optionen in `/boot/loader.conf` und einem anschließenden Systemneustart Erfolge erzielt:

```
vm.kmem_size="330M"
vm.kmem_size_max="330M"
vfs.zfs.arc_max="40M"
vfs.zfs.vdev.cache.size="5M"
```

Für eine detailliertere Liste an Empfehlungen für ZFS-bezogene Einstellungen, lesen Sie http://wiki.freebsd.org/ZFSTuningGuide.

20.7. Zusätzliche Informationen

- FreeBSD Wiki - ZFS
- FreeBSD Wiki - ZFS Tuning
- Illumos Wiki - ZFS
- Oracle Solaris ZFS Administration Guide
- ZFS Evil Tuning Guide
- ZFS Best Practices Guide
- Calomel Blog - ZFS Raidz Performance, Capacity und Integrity

20.8. ZFS-Eigenschaften und Terminologie

ZFS ist ein fundamental anderes Dateisystem aufgrund der Tatsache, dass es mehr als ein Dateisystem ist. ZFS kombiniert die Rolle eines Dateisystems mit dem Volumemanager, was es ermöglicht, zusätzliche Speichermedien zu einem laufenden System hinzuzufügen und diesen neuen Speicher sofort auf allen auf dem Pool existierenden Dateisystemen zur Verfügung zu haben. Durch die Kombination von traditionell getrennten Rollen ist ZFS in der Lage, Einschränkungen, die zuvor RAID-Gruppen daran gehindert hatten, zu wachsen. Jedes Gerät auf höchster Ebene in einem Pool wird ein *vdev* genannt, was eine einfache Platte oder eine RAID-Transformation wie ein Spiegel oder RAID-Z-Verbund sein kann. ZFS-Dateisysteme (*datasets* genannt), haben jeweils Zugriff auf den gesamten freien Speicherplatz des gesamten Pools. Wenn Blöcke aus diesem Pool allokiert werden, verringert sich auch der freie Speicherplatz für jedes Dateisystem. Dieser Ansatz verhindert die allgegenwärtige Falle von umfangreichen Partitionen, bei denen freier Speicherplatz über alle Partitionen hinweg fragmentiert wird.

zpool

Ein Speicher-*Pool* ist der grundlegendste Baustein von ZFS. Ein Pool besteht aus einem oder mehreren vdevs, was die zugrundeliegenden Geräte repräsentiert, welche die Daten speichern. Ein Pool wird dann verwendet, um ein oder mehrere Dateisysteme (Datasets) oder Blockgeräte (Volumes) zu erstellen. Diese Datasets und Volumes teilen sich den im Pool verfügbaren freien Speicherplatz. Jeder Pool wird eindeutig identifiziert durch einen Namen und eine GUID. Die verfügbaren Eigenschaften werden durch die ZFS-Versionsnummer des Pool bestimmt.

Anmerkung

FreeBSD 9.0 und 9.1 enthalten Unterstützung für ZFS Version 28. Spätere Versionen setzen ZFS Version 5000 mit Feature Flags ein. Das neue Feature Flag System

	erlaubt eine größere Kompatibilität mit anderen Implementierungen von ZFS.
vdev Arten	Ein Pool besteht aus einem oder mehreren vdevs, die selbst eine einfache Platte oder im Fall von RAID eine Gruppe von Platten darstellt. Wenn mehrere vdevs eingesetzt werden, verteilt ZFS die Daten über die vdevs, um die Geschwindigkeit zu steigern und den verfügbaren Platz zu maximieren. • *Festplatte* - Der einfachste Typ von vdev ist ein Standard-Blockgerät. Dies kann die komplette Platte (wie `/dev/ada0` oder `/dev/da0`) oder auch eine Partition (`/dev/ada0p3`) sein. Auf FreeBSD gibt es keine Geschwindigkeitseinbußen bei der Verwendung einer Partition anstatt einer kompletten Platte. Dies unterscheidet sich von den Empfehlungen, welche in der Solaris Dokumentation gegeben werden. • *File* - Zusätzlich zu Festplatten können ZFS-Pools aus regulären Dateien aufgebaut sein, was besonders hilfreich ist, um zu testen und zu experimentieren. Verwenden Sie den kompletten Pfad zu der Datei als Gerätepfad im Befehl zpool create. Alle vdevs müssen mindestens 128 MB groß sein. • *Mirror* - Wenn ein Spiegel erstellt wird, verwenden Sie das Schlüsselwort `mirror`, gefolgt von der Liste an Mitgliedsgeräten für den Spiegel. Ein Spiegel besteht aus zwei oder mehr Geräten und sämtliche Daten werden auf alle Geräte, die Mitglied des Spiegels sind, geschrieben. Ein Spiegel-vdev wird nur soviele Daten speichern, wie das kleinste Gerät im Verbund aufnehmen kann. Ein Spiegel-vdev kann den Verlust von allen Mitgliedsgeräten bis auf eines verkraften, ohne irgendwelche Daten zu verlieren. **Anmerkung** Ein reguläre einzelne vdev-Platte kann jederzeit zu einem Spiegel-vdev über das Kommando `zpool attach` aktualisiert werden. • *RAID-Z* - ZFS implementiert RAID-Z, eine Varianten des RAID-5-Standards, der bessere Verteilung der Parität bietet und das „RAID-5 write hole“ eliminiert, bei dem die Daten und Parität nach einem unerwarteten Neustart inkonsistent werden können. ZFS unter-

stützt drei Stufen von RAID-Z, die unterschiedliche Arten von Redundanz im Austausch gegen niedrigere Stufen von verwendbarem Speicher. Diese Typen werden RAID-Z1 bis RAID-Z3 genannt, basierend auf der Anzahl der Paritätsgeräte im Verbund und der Anzahl an Platten, die ausfallen können, während der Pool immer noch normal funktioniert.

In einer RAID-Z1-Konfiguration mit vier Platten, bei der jede 1 TB besitzt, beträgt der verwendbare Plattenplatz 3 TB und der Pool wird immer noch im Modus degraded weiterlaufen, wenn eine Platte davon ausfällt. Wenn eine zusätzliche Platte ausfällt, bevor die defekte Platte ersetzt wird, können alle Daten im Pool verloren gehen.

Eine Konfiguration von acht Platten zu je 1 TB als RAID-Z3 wird 5 TB verwendbaren Speicher bieten und in der Lage sein, weiterhin zu funktionieren, wenn drei Platten ausgefallen sind. Sun™ empfiehlt nicht mehr als neun Platten in einem einzelnen vdev. Wenn die Konfiguration mehr Platten aufweist, wird empfohlen, diese in getrennten vdevs aufzuteilen, so dass die Daten des Pools zwischen diesen aufgeteilt werden.

Eine Konfiguration von zwei RAID-Z2-vdevs, bestehend aus jeweils 8 Platten würde etwa einem RAID-60-Verbund entsprechen. Der Speicherplatz einer RAID-Z-Gruppe ist ungefähr die Größe der kleinsten Platte multipliziert mit der Anzahl von nicht-Paritätsplatten. Vier 1 TB Platten in einem RAID-Z1 besitzt eine effektive Größe von ungefähr 3 TB und ein Verbund von acht 1 TB-Platten als RAID-Z3 enthält 5 TB verfügbarer Plattenplatz.

- *Spare* - ZFS besitzt einen speziellen Pseudo-vdev Typ, um einen Überblick über die verfügbaren hot spares zu behalten. Beachten Sie, dass hot spares nicht automatisch eingesetzt werden. Diese müssen manuell konfiguriert werden, um ein ausgefallenes Gerät über `zfs replace` zu ersetzen.

- *Log* - ZFS Log-Geräte, auch bezeichnet als ein ZFS Intent Log (ZIL) verschieben das Intent Log von den regulären Geräten im Pool auf ein dediziertes Gerät, typischerweise eine SSD. Ein dediziertes Log-Gerät zu besitzen kann die Geschwindigkeit von Anwendungen mit einer großen Anzahl von synchronen Schreibvorgängen, besonders Datenbanken, signifikant steigern. Log-Geräte können gespiegelt werden, jedoch wird RAID-Z nicht unterstützt. Werden mehrere Log-Geräte verwendet, so werden Schreibvorgänge gleichmäßig unter diesen aufgeteilt.

	• *Cache* - Ein Cache-vdev einem Pool hinzuzufügen, erhöht den Speicher des L2ARC Caches. Cache-Geräte lassen sich nicht spiegeln. Da ein Cache-Gerät nur zusätzliche Kopien von existierenden Daten speichert, gibt es kein Risiko, Daten zu verlieren.
Transaktionsgruppe (Transaction Group, TXG)	Transaktionsgruppen sind die Art und Weise, wie geänderte Blöcke zusammen gruppiert und letztendlich auf den Pool geschrieben werden. Transaktionsgruppen sind die atomare Einheit, welche ZFS verwendet, um Konsistenz zu gewährleisten. Jeder Transaktionsgruppe wird eine einzigartige, fortlaufende 64-Bit Identifikationsnummer zugewiesen. Es kann bis zu drei aktive Transaktionsgruppen gleichzeitig geben, wobei sich jede davon in einem der folgenden drei Zustände befinden kann: • *Open (Offen)* - Wenn eine neue Transaktionsgruppe erstellt wird, befindet diese sich im Zustand offen und akzeptiert neue Schreibvorgänge. Es ist immer eine Transaktionsgruppe in diesem Zustand, jedoch kann die Transaktionsgruppe neue Schreibvorgänge ablehnen, wenn diese ein Limit erreicht hat. Sobald eine offene Transaktionsgruppe an das Limit stößt oder das `vfs.zfs.txg.timeout` [525] wurde erreicht, geht die Transaktionsgruppe in den nächsten Zustand über. • *Quiescing (Stilllegen)* - Ein kurzer Zustand, der es noch ausstehenden Operationen erlaubt, zum Abschluss zu kommen, währenddessen das Erstellen einer neuen Transaktionsgruppe jedoch nicht blockiert wird. Sobald alle Transaktionen in der Gruppe abgeschlossen sind, geht die Transaktionsgruppen in den letzten Zustand über. • *Syncing (Sychronisieren)* - Alle Daten in der Transaktionsgruppe werden auf das Speichermedium geschrieben. Dieser Prozess wird wiederum andere Daten wie Metadaten und space maps verändern, die ebenfalls auf das Speichermedium geschrieben werden müssen. Der Prozess des Synchronisierens beinhaltet mehrere Durchläufe. Der erste Prozess, welches der größte, gefolgt von den Metadaten, ist, beinhaltet alle geänderten Datenblöcke und kann mehrere Durchläufe benötigen, um zum Ende zu gelangen. Da das Allokieren von Speicher für die Datenblöcke neue Metadaten generiert, kann der Synchronisationsprozess nicht beendet werden, bis ein Durchlauf fertig ist, der keinen zusätzlichen Speicher allokiert. Der Synchronisierungszustand ist der Zustand, in dem auch *synctasks* abgeschlossen werden. Synctasks sind administrative Operationen, wie das Erstellen oder zerstören von Schnappschüssen und Datasets, welche den Überblock verändern, wenn sie abgeschlossen sind. Sobald der Synchronisationszustand abge-

	schlossen ist, geht die Transaktionsgruppe aus dem Stilllegungszustand über in den Synchronisationszustand. Alle administrativen Funktionen, wie Schnappschüsse werden als Teil einer Transaktionsgruppe geschrieben. Wenn ein synctask erstellt ist, wird dieser der momentan geöffneten Transaktionsgruppe hinzugefügt und diese Gruppe wird so schnell wie möglich in den Synchronisationszustand versetzt, um die Latenz von administrativen Befehlen zu reduzieren.
Adaptive Replacement Cache (ARC)	ZFS verwendet einen Adaptive Replacement Cache (ARC), anstatt eines traditionellen Least Recently Used (LRU) Caches. Ein LRU-Cache ist eine einfache Liste von Elementen im Cache, sortiert nach der letzten Verwendung jedes Elements in der Liste. Neue Elemente werden an den Anfang der Liste eingefügt. Wenn der Cache voll ist, werden Elemente vom Ende der Liste verdrängt, um Platz für aktivere Objekte zu schaffen. Ein ARC besteht aus vier Listen: derjenigen der Most Recently Used (MRU) und Most Frequently Used (MFU) Objekte, plus einer sogenannten ghost list für jede von beiden. Diese Ghost Lists verfolgen die kürzlich verdrängten Objekte, um zu verhindern, dass diese erneut in den Cache aufgenommen werden. Dies erhöht die Trefferrate (hit ratio) des Caches, indem verhindert wird, dass Elemente, die in der Vergangenheit nur ab und zu benutzt wurden, wieder im Cache landen. Ein weiterer Vorteil der Verwendung sowohl einer MRU und einer MFU ist, dass das Scannen eines gesamten Dateisystems normalerweise alle Daten aus einem MRU- oder LRU-Cache verdrängt, um dem gerade frisch zugegriffenem Inhalt den Vorzug zu geben. Mit ZFS gibt es also eine MFU, die nur die am häufigsten verwendeten Elemente beinhaltet und der Cache von am meisten zugegriffenen Blöcken bleibt erhalten.
L2ARC	L2ARC ist die zweite Stufe des Caching-Systems von ZFS. Der Haupt-ARC wird im RAM abgelegt. Da die Menge an verfügbarem RAM meist begrenzt ist, kann ZFS auch cache vdevs [529] verwenden. Solid State Disks (SSDs) werden oft als diese Cache-Geräte eingesetzt, aufgrund ihrer höheren Geschwindigkeit und niedrigeren Latenz im Vergleich zu traditionellen drehenden Speichermedien wie Festplatten. Der Einsatz des L2ARC ist optional, jedoch wird durch die Verwendung eine signifikante Geschwindigkeitssteigerung bei Lesevorgängen bei Dateien erzielt, welche auf der SSD zwischengespeichert sind, anstatt von der regulären Platte gelesen werden zu müssen. L2ARC kann ebenfalls die Deduplizierung beschleunigen, da eine DDT, welche nicht in den RAM passt, jedoch in den L2ARC wesentlich schneller sein wird als eine DDT, die von der Platte gelesen werden muss. Die Häufigkeit, in der Daten zum Cache-Gerät hinzugefügt werden, ist begrenzt, um zu verhindern, dass eine SSD frühzeitig durch zu viele Schreibvorgän-

	ge aufgebraucht ist. Bis der Cache voll ist (also der erste Block verdrängt wurde, um Platz zu schaffen), wird das Schreiben auf den L2ARC begrenzt auf die Summe der Schreibbegrenzung und das Bootlimit, sowie hinterher auf das Schreiblimit. Ein paar sysctl(8)-Werte steuert diese Limits. `vfs.zfs.l2arc_write_max` steuert, wie viele Bytes in den Cache pro Sekunde geschrieben werden, während `vfs.zfs.l2arc_write_boost` zu diesem Limit während der „Turbo Warmup Phase“ hinzuaddiert wird (Write Boost).
ZIL	ZIL beschleunigt synchrone Transaktionen durch die Verwendung von Speichermedien wie SSDs, welche schneller sind als diejenigen, welche Teil des Speicherpools sind. Wenn eine Anwendung einen synchronen Schreibvorgang anfordert (eine Garantie, dass die Daten sicher auf den Platten gespeichert wurden anstatt nur zwischengespeichert zu sein, um später geschrieben zu werden), werden die Daten auf den schnelleren ZIL-Speicher geschrieben und dann später auf die regulären Festplatten. Dies reduziert die Latenz sehr und verbessert die Geschwindigkeit. Nur synchrone Vorgänge wie die von Datenbanken werden durch den Einsatz eines ZIL profitieren. Reguläre, asynchrone Schreibvorgänge wie das Kopieren von Dateien wird den ZIL überhaupt nicht verwenden.
Copy-On-Write	Im Gegensatz zu traditionellen Dateisystemen werden beim Überschreiben von Daten bei ZFS die neuen Daten an einen anderen Block geschrieben, anstatt die alten Daten an der gleichen Stelle zu überschreiben. Nur wenn dieser Schreibvorgang beendet wurde, werden die Metadaten aktualisiert, um auf die neue Position zu verweisen. Im Falle eines kurzen Schreibvorgangs (ein Systemabsturz oder Spannungsverlust während eine Datei geschrieben wird) sind die gesamten Inhalte der Originaldatei noch vorhanden und der unvollständige Schreibvorgang wird verworfen. Das bedeutet auch, dass ZFS nach einem unvorhergesehenen Ausfall keinen fsck(8) benötigt.
Dataset	*Dataset* ist der generische Begriff für ein ZFS-Dateisystem, Volume, Schnappschüsse oder Klone. Jedes Dataset besitzt einen eindeutigen Namen in der Form *`poolname/path@snapshot`* Die Wurzel des Pools ist technisch gesehen auch ein Dataset. Kind-Datasets werden hierarchisch wie Verzeichnisse benannt. Beispielsweise ist *`mypool/home`* das Heimatdataset, ein Kind von *`mypool`* und erbt die Eigenschaften von diesem. Dies kann sogar noch erweitert werden durch das Erstellen von *`mypool/home/user`* . Dieses Enkelkind-Dataset wird alle Eigenschaften von den Eltern und Großeltern erben. Eigenschaften auf einem Kind können die geerbten Standardwerte der Eltern und Großeltern ändern und überschreiben. Die Verwaltung von Datasets und dessen Kindern lässt sich delegieren.

Dateisystem	Ein ZFS-Dataset wird meistens als ein Dateisystem verwendet. Wie jedes andere Dateisystem kann auch ein ZFS-Dateisystem irgendwo in der Verzeichnishierarchie eingehängt werden und enthält seine eigenen Dateien und Verzeichnisse mit Berechtigungen, Flags und anderen Metadaten.
Volume	Zusätzlich zu regulären Dateisystem-Datasets, kann ZFS auch Volumes erstellen, die Blockgeräte sind. Volumes besitzen viele der gleichen Eigenschaften, inklusive copy-on-write, Schnappschüsse, Klone und Prüfsummen. Volumes sind nützlich, um andere Dateisystemformate auf ZFS aufzusetzen, so wie UFS Virtualisierung, oder das Exportieren von iSCSI-Abschnitten.
Snapshot (Schnappschuss)	Das copy-on-write (COW)-Entwicklung von ZFS erlaubt das Erstellen von beinahe sofortigen, konsistenten Schnappschüssen mit beliebigen Namen. Nachdem ein Schnappschuss von einem Dataset angelegt oder ein rekursiver Schnappschuss eines Elterndatasets, welcher alle Kinddatasets enthält, erstellt wurde, werden neue Daten auf neue Blöcke geschrieben, jedoch die alten Blöcke nicht wieder als freier Speicher zurückgewonnen. Der Schnappschuss enthält die Originalversion des Dateisystems und das aktive Dateisystem besitzt alle Änderungen, die seit dem Schnappschuss erstellt wurden. Kein zusätzlicher Platz wird benötigt. Werden neue Daten auf das aktive Dateisystem geschrieben, werden neue Blöcke allokiert, um diese Daten zu speichern. Die scheinbare Größe des Schnappschusses wird wachsen, da die Blöcke nicht mehr länger im aktiven Dateisystem, sondern nur noch im Schnappschuss Verwendung finden. Diese Schnappschüsse können nur lesend eingehängt werden, um vorherige Versionen von Dateien wiederherzustellen. Ein rollback eines aktiven Dateisystems auf einen bestimmten Schnappschuss ist ebenfalls möglich, was alle Änderungen, die seit dem Anlegen des Schnappschusses vorgenommen wurden, wieder Rückgängig macht. Jeder Block im Pool besitzt einen Referenzzähler, der verfolgt, wieviele Schnappschüsse, Klone, Datasets oder Volumes diesen Block nutzen. Wenn Dateien und Schnappschüsse gelöscht werden, verringert dies auch den Referenzzähler. Wenn ein Block nicht mehr länger referenziert wird, kann er als freier Speicher wieder genutzt werden. Schnappschüsse können auch mit hold markiert werden. Wenn versucht wird, einen solchen Schnappschuss zu zerstören, wird stattdessen ein `EBUSY`-Fehler ausgegeben. Jeder Schnappschuss kann mehrere holds besitzen, jeder mit einem eindeutigen Namen. Das Kommando release entfernt diese, damit der Schnappschuss gelöscht werden kann. Schnappschüsse lassen sich auf Volumes ebenfalls anlegen, allerdings können diese nur geklont oder zurückgerollt werden, nicht jedoch unabhängig eingehängt.
Clone (Klone)	Schnappschüsse können auch geklont werden. Ein Klon stellt eine veränderbare Version eines Schnappschusses

	dar, was es ermöglicht, das Dateisystem als neues Dataset aufzuspalten. Genau wie bei einem Schnappschuss verbraucht ein Klon keinen zusätzlichen Platz. Wenn neue Daten auf einen Klon geschrieben und neue Blöcke allokiert werden, wächst auch die Größe des Klons. Wenn Blöcke im geklonten Dateisystem oder Volume überschrieben werden, verringert sich auch der Referenzzähler im vorherigen Block. Der Schnappschuss, auf dem der Klon basiert kann nicht gelöscht werden, weil der Klon darauf eine Abhängigkeit besitzt. Der Schnappschuss stellt den Elternteil dar und der Klon das Kind. Klone lassen sich *promoted* (befördern), was die Abhängigkeit auflöst und den Klon zum Elternteil macht und den vorherigen Elternteil das Kind. Diese Operation benötigt keinen zusätzlichen Plattenplatz. Da die Menge an verwendetem Speicher vom Elternteil und dem Kind vertauscht wird, betrifft dies eventuell vorhandene Quotas und Reservierungen.
Checksum (Prüfsumme)	Jeder Block, der allokiert wird erhält auch eine Prüfsumme. Der verwendete Prüfsummenalgorithmus ist eine Eigenschaft jedes Datasets, siehe dazu `set`. Die Prüfsumme jedes Blocks wird transparent validiert wenn er gelesen wird, was es ZFS ermöglicht, stille Verfälschung zu entdecken. Wenn die gelesenen Daten nicht mit der erwarteten Prüfsumme übereinstimmen, wird ZFS versuchen, die Daten aus jeglicher verfügbarer Redundanz (wie Spiegel oder RAID-Z) zu rekonstruieren. Eine Überprüfung aller Prüfsummen kann durch das Kommando `scrub` ausgelöst werden. Prüfsummenalgorithmen sind: • `fletcher2` • `fletcher4` • `sha256` Die `fletcher`-Algorithmen sind schneller, aber dafür ist `sha256` ein starker kryptographischer Hash und besitzt eine viel niedrigere Chance auf Kollisionen zu stoßen mit dem Nachteil geringerer Geschwindigkeit. Prüfsummen können deaktiviert werden, dies wird aber nicht empfohlen.
Compression	Jedes Dataset besitzt eine compression-Eigenschaft, die standardmäßig ausgeschaltet ist. Diese Eigenschaft kann auf eine Reihe von Kompressionsalgorithmen eingestellt werden. Dadurch werden alle neuen Daten, die auf das Dataset geschrieben werden, komprimiert. Neben einer Reduzierung von verbrauchtem Speicher wird oft der Lese- und Schreibdurchsatz erhöht, weil weniger Blöcke gelesen oder geschrieben werden müssen. • *LZ4* - Wurde in der ZFS Poolversion 5000 (feature flags) hinzugefügt und LZ4 ist jetzt der empfohlene Kompressionsalgorithmus. LZ4 komprimiert unge-

	fähr 50% schneller als LZJB, wenn er auf komprimierbaren Daten angewendet wird und ist über dreimal schneller, wenn unkomprimierbare Daten vorliegen. LZ4 entkomprimiert auch ungefähr 80% schneller als LZJB. Auf modernen CPUs, kann LZ4 oft über 500 MB/s komprimieren und entkomprimiert (pro einzelnem CPU-Kern) bei über 1.5 GB/s. **Anmerkung** LZ4-Komprimierung ist nur verfügbar nach FreeBSD 9.2. • *LZJB* - Der Standardkompressionsalgorithmus wurde von Jeff Bonwick, einem der ursprünglichen Entwickler von ZFS, entworfen. LZJB bietet gute Komprimierung mit weniger CPU-Überhang im Vergleich zu GZIP. In der Zukunft wird der Standardkompressionsalgorithmus wahrscheinlich auf LZ4 gewechselt. • *GZIP* - Ein populärer Stromkompressionsalgorithmus ist auch in ZFS verfügbar. Einer der Hauptvorteile von der Verwendung von GZIP ist seine konfigurierbare Komprimierungsstufe. Wenn die Eigenschaft `compress` gesetzt wird, kann der Administrator die Stufe der Komprimierung wählen, die von `gzip1`, der kleinsten Komprimierungsstufe, bis zu `gzip9`, der höchsten Komprimierungsstufe, reicht. Dies erlaubt es dem Administrator zu steuern, wieviel CPU-Zeit für eingesparten Plattenplatz eingetauscht werde soll. • *ZLE* - Zero Length Encoding ist ein besonderer Kompressionsalgorithmus, welcher nur fortlaufende Aneinanderreihungen von Nullen komprimiert. Dieser Komprimierungsalgorithmus ist nur sinnvoll, wenn das Dataset viele große Blöcke von Nullen aufweist.
Copies	Wenn die Eigenschaft `copies` auf einen Wert grösser als 1 gesetzt wird, weist das ZFS an, mehrere Kopien eines Blocks im Dateisystem oder Volume anzulegen. Diese Eigenschaft auf einem wichtigen Dataset einzustellen sorgt für zusätzliche Redundanz, aus der ein Block wiederhergestellt werden kann, der nicht mehr mit seiner Prüfsumme übereinstimmt. In Pools ohne Redundanz ist die copies-Eigenschaft die einzige Form von Redundanz. Die Eigenschaft kann einen einzelnen schlechten Sektor oder andere Formen von kleineren Verfälschungen wiederherstellen, schützt jedoch nicht den Pool vom Verlust einer gesamten Platte.
Deduplizierung	Prüfsummen ermöglichen es, Duplikate von Blöcken zu erkennen, wenn diese geschrieben werden. Mit Deduplizierung erhöht sich der Referenzzähler eines existie-

	renden, identischen Blocks, was Speicherplatz einspart. Um Blockduplikate zu erkennen, wird im Speicher eine Deduplizierungstabelle (DDT) geführt. Die Tabelle enthält eine Liste von eindeutigen Prüfsummen, die Position dieser Blöcke und einen Referenzzähler. Werden neue Daten geschrieben, wird die Prüfsumme berechnet und mit der Liste verglichen. Wird eine Übereinstimmung gefunden, wird der existierende Block verwendet. Der SHA256-Prüfsummenalgorithmus wird mit Deduplizierung benutzt, um einen sicheren kryptographischen Hash zu bieten. Deduplizierung lässt sich konfigurieren. Wenn `dedup` auf `on` steht, wird angenommen, dass eine übereinstimmende Prüfsumme bedeutet, dass die Daten identisch sind. Steht `dedup` auf `verify`, werden die Daten in den beiden Blöcken Byte für Byte geprüft, um sicherzustellen, dass diese wirklich identisch sind. Wenn die Daten nicht identisch sind, wird die Kollision im Hash vermerkt und die beiden Blöcke separat gespeichert. Da die DDT den Hash jedes einzigartigen Blocks speichern muss, benötigt sie eine große Menge an Speicher. Eine generelle Faustregel besagt, dass 5-6 GB RAM pro 1 TB deduplizierter Daten benötigt werden. In Situationen, in denen es nicht praktikabel ist, genug RAM vorzuhalten, um die gesamte DDT im Speicher zu belassen, wird die Geschwindigkeit stark darunter leiden, da die DDT von der Platte gelesen werden muss, bevor jeder neue Block geschrieben wird. Deduplizierung kann den L2ARC nutzen, um die DDT zu speichern, was einen guten Mittelweg zwischen schnellem Systemspeicher und langsameren Platten darstellt. Bedenken Sie, dass durch die Verwendung von Komprimierung meistens genauso große Platzersparnis möglich ist, ohne den zusätzlichen Hauptspeicherplatzbedarf.
Scrub (Bereinigung)	Anstatt einer Konsistenzprüfung wie fsck(8) verwendet ZFS `scrub`. `scrub` liest alle Datenblöcke, die auf dem Pool gespeichert sind und prüft deren Prüfsumme gegen die als richtig in den Metadaten gespeicherte Prüfsumme. Eine periodische Prüfung aller im Pool gespeicherten Daten versichert, dass verfälschte Blöcke rekonstruiert werden können, bevor dies nötig ist. Ein Scrub wird nicht nach einem unsauberen Herunterfahren benötigt, wird jedoch einmal alle drei Monate angeraten. Die Prüfsumme von jedem Block wird verifiziert, wenn Blöcke während des normalen Betriebs gelesen werden, jedoch stellt ein Scrub sicher, dass sogar weniger häufig verwendete Blöcke auf stille Verfälschungen hin untersucht werden. Datenintegrität wird dadurch erhöht, besonders wenn es sich um Archivspeichersituationen handelt. Die relative Priorität des `scrub` lässt sich mit `vfs.zfs.scrub_delay` [524] anpassen, um zu verhindern, dass der scrub die Geschwindigkeit von anderen Anfragen auf dem Pool beeinträchtigt.

Dataset Quotas	ZFS bietet sehr schnelle und akkurate Dataset-, Benutzer- und Gruppenspeicherplatzbuchhaltung, zusätzlich zu Quotas und Speicherplatzreservierungen. Dies gibt dem Administrator feingranulare Kontrolle darüber, wie Speicherplatz allokiert und die Reservierung für kritische Dateisysteme vorgenommen wird ZFS unterstützt verschiedene Arten von Quotas: die Dataset-Quota, die Referenzquota (refquota), die Benutzerquota und die Gruppenquota sind verfügbar. Quotas beschränken die Menge an Speicherplatz, welche ein Dataset, seine Kinder, einschließlich Schnappschüsse des Datasets, deren Kinder und die Schnappschüsse von diesen Datasets, verbrauchen können.  **Anmerkung** Quotas können nicht auf Volumes gesetzt werden, da die Eigenschaft `volsize` als eine implizite Quota agiert.
Referenzquota	Ein Referenzquota beschränkt die Menge an Speicherplatz, die ein Dataset verbrauchen kann durch das Erzwingen einer harten Grenze. Jedoch beinhaltet diese harte Grenze nur Speicherplatz, die das Dataset referenziert und beinhaltet nicht den Speicher, der von Kindern, wie Dateisystemen oder Schnappschüssen, verbraucht wird.
Benutzerquota	Benutzerquotas sind hilfreich, um die Menge an Speicherplatz, die ein bestimmter Benutzer verbrauchen kann, einzuschränken.
Gruppenquota	Die Gruppenquota beschränkt die Menge an Speicherplatz, die eine bestimmte Gruppe verbrauchen darf.
Dataset-Reservierung	Die Eigenschaft `reservation` ermöglicht es, ein Minimum an Speicherplatz für ein bestimmtes Dataset und dessen Kinder zu garantieren. Wenn eine Reservierung von 10 GB auf `storage/home/bob` gesetzt ist und ein anderes Dataset versucht, allen freien Speicherplatz zu verwenden, bleiben zumindest noch 10 GB an Speicher reserviert. Wenn von `storage/home/bob` ein Schnappschuss angelegt wird, wird dieser von der Reservierung abgezogen und zählt damit dagegen. Die Eigenschaft `refreservation` funktioniert auf ähnliche Weise, jedoch *exkludiert* diese Kinder wie Schnappschüsse. Reservierungen jeder Art sind in vielen Situationen nützlich, so wie bei der Planung und dem Testen der richtigen Speicherplatzallokation in einem neuen System oder durch die Zusicherung, dass genug Speicher-

	platz auf Dateisystemen für Audio-Logs oder Systemwiederherstellungsprozeduren und Dateien verfügbar ist.
Referenzreservierung	Die Eigenschaft `refreservation` ermöglicht es, ein Minimum an Speicherplatz für die Verwendung eines bestimmten Datasets zu garantieren, *exklusiv* dessen Kinder. Das bedeutet, dass wenn eine 10 GB-Reservierung auf `storage/home/bob` vorhanden ist und ein anderes Dataset versucht, alle freien Speicherplatz aufzubrauchen, sind zumindest noch 10 GB Speicher reserviert. Im Gegensatz zu einer regulären Reservierung wird der Speicher von Schnappschüssen und Kinddataset nicht gegen die Reservierung gezählt. Beispielsweise, wenn ein Schnappschuss von `storage/home/bob` angelegt wird, muss genug Plattenplatz außerhalb der Menge an `refreservation` vorhanden sein, damit die Operation erfolgreich durchgeführt wird. Kinder des Hauptdatasets werden nicht in die Menge an `refreservation` gezählt und dringen auf diese Weise auch nicht in den gesetzten Speicher ein.
Resilver	Wenn eine Platte ausfällt und ersetzt wird, muss die neue Platte mit den Daten gefüllt werden, die verloren gegangen sind. Der Prozess der Verwendung der Paritätsinformationen, welche über die übrigen Platten verteilt sind, um die fehlenden Daten zu berechnen und auf die neue Platte zu übertragen, wird *resilvering* genannt.
Online	Ein Pool oder vdev im Zustand `Online` besitzt alle verbundenen Mitgliedsgeräte und ist voll funktionsfähig. Individuelle Geräte im Zustand `Online` funktionieren normal.
Offline	Individuelle Geräte lassen sich vom Administrator in den Zustand `Offline` versetzen, wenn es ausreichend Redundanz gibt, um zu verhindern, dass der Pool oder das vdev in den Zustand Faulted versetzt wird. Ein Administrator kann eine Platte vor einem Austausch offline nehmen oder um es leichter zu machen, diese zu identifizieren.
Degraded	Ein Pool oder vdev im Zustand `Degraded` hat eine oder mehrere Platten, welche getrennt wurden oder ausgefallen sind. Der Pool kann immer noch verwendet werden, doch wenn noch weitere Geräte ausfallen, kann der Pool nicht wiederhergestellt werden. Die fehlenden Geräte anzuschließen oder die defekten Platten zu ersetzen wird den Pool wieder in den Zustand Online versetzen, nachdem die angeschlossenen oder neuen Geräte den Resilver-Prozess abgeschlossen haben.
Faulted	Ein Pool oder vdev im Zustand `Faulted` funktioniert nicht länger. Die Daten darauf sind nicht mehr länger verfügbar. Ein Pool oder vdev geht in den Zustand `Faulted` über, wenn die Anzahl der fehlenden oder defekten Geräte die Redundanzstufe im vdev überschreiten. Wenn fehlende Geräte angeschlossen werden, geht der Pool wieder in den Zustand Online. Wenn es nicht genügend Redundanz gibt, um die Anzahl an defekten Plat-

ten zu kompensieren, sind die Inhalte des Pools verloren und müssen von der Sicherung wiederhergestellt werden.

Kapitel 21. Dateisystemunterstützung

Geschrieben von Tom Rhodes.
Übersetzt von Benedict Reuschling und Daniel Seuffert.

21.1. Übersicht

Dateisysteme sind ein wesentlicher Bestandteil von Betriebssystemen. Sie erlauben es Benutzern, Dateien zu laden und zu speichern, ermöglichen den Zugriff auf Daten und machen Festplatten überhaupt erst nützlich. Betriebssysteme unterscheiden sich normalerweise bei dem mitgelieferten Dateisystem. Traditionell ist dies unter FreeBSD das Unix File System UFS, welches mit UFS2 modernisiert wurde. Seit FreeBSD 7.0 steht auch das Z-Dateisystem (ZFS) als natives Dateisystem zur Verfügung. Hierzu finden Sie in Kapitel 20, *Das Z-Dateisystem (ZFS)* weitere Informationen.

FreeBSD unterstützt auch eine Vielzahl weiterer Dateisysteme, um auf Daten von anderen Betriebssystemen lokal zuzugreifen, beispielsweise Daten auf USB-Speichermedien, Flash-Speichern und Festplatten. Dazu gehört die Unterstützung für das Linux® Extended File System (EXT) und das Reiser Dateisystem.

Es gibt verschiedene Stufen der Unterstützung in FreeBSD für diese unterschiedlichen Dateisysteme. Manche benötigen ein geladenes Kernelmodul, andere die Installation bestimmter Werkzeuge. Einige Dateisysteme haben volle Unterstützung für Lese- und Schreibzugriffe, während auf andere nur-lesend zugegriffen werden kann.

Nachdem Sie dieses Kapitel gelesen haben, wissen Sie:

- Den Unterschied zwischen nativen und unterstützten Dateisystemen.
- Welche Dateisysteme von FreeBSD unterstützt werden.
- Wie man fremde Dateisysteme aktiviert, konfiguriert, darauf zugreift und diese verwendet.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- Grundlagen von UNIX® und FreeBSD verstehen (Kapitel 4, *Grundlagen des UNIX Betriebssystems*).
- Mit den Grundlagen der Konfiguration und dem Bauen des Kernels vertraut sein (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).
- Problemlos Software von Drittherstellern in FreeBSD installieren können (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).
- Sich ein wenig mit Festplatten, Speicher und Gerätenamen in FreeBSD auskennen (Kapitel 18, *Speichermedien*).

21.2. Linux® Dateisysteme

FreeBSD bietet integrierte Unterstützung für einige Linux®-Dateisysteme. Dieser Abschnitt demonstriert, wie der Support aktiviert und die unterstützten Linux®-Dateisysteme eingehangen werden.

21.2.1. ext2

Seit FreeBSD 2.2 ist eine Kernel-Unterstützung für das ext2-Dateisystem vorhanden. In FreeBSD 8.x und früheren Versionen wurde der Code noch unter der GPL lizensiert. Der Code wurde neu geschrieben und steht seit FreeBSD 9.0 unter der BSD-Lizenz.

Der ext2fs(5)-Treiber erlaubt dem FreeBSD Kernel sowohl Lese-, als auch Schreibzugriffe auf ext2-Dateisysteme.

Anmerkung

Dieser Treiber kann auch für den Zugriff auf ext3 und ext4 Dateisysteme verwendet werden. Allerdings werden ext3-Journale, erweiterte Attribute und Inodes größer 128 Byte nicht unterstützt. ext4 wird nur-lesend unterstützt.

Um auf ein ext-Dateisystem zuzugreifen, muss zuerst das entsprechende Kernelmodul geladen werden:

```
# kldload ext2fs
```

Mounten Sie anschließend das ext-Volume unter Angabe des FreeBSD Partitionsnamens und eines existierenden Mountpunktes. Dieses Beispiel hängt `/dev/ad1s1` nach `/mnt` ein:

```
# mount -t ext2fs /dev/ad1s1 /mnt
```

21.2.2. ReiserFS

FreeBSD bietet Nur-Lese-Unterstützung für das Reiser-Dateisystem ReiserFS.

Den Treiber für reiserfs(5) laden Sie wie folgt:

```
# kldload reiserfs
```

Mounten Sie anschließend das auf `/dev/ad1s1` befindliche ReiserFS-Volume:

```
# mount -t reiserfs /dev/ad1s1 /mnt
```

Kapitel 22. Virtualisierung

Beigetragen von Murray Stokely.
Übersetzt von Oliver Peter.

22.1. Übersicht

Virtualisierungssoftware erlaubt es, mehrere Betriebssysteme gleichzeitig auf dem selben Computer laufen zu lassen. Derartige Softwaresysteme für PCs setzen in der Regel ein Host-Betriebssystem voraus, auf dem die Virtualisierungssoftware läuft und unterstützen eine nahezu beliebige Anzahl von Gast-Betriebssystemen.

Nachdem Sie dieses Kapitel gelesen haben,

- Kennen Sie den Unterscheid zwischen einem Host-Betriebssystem und einem Gast-Betriebssystem.
- Können Sie FreeBSD auf einem Intel®-basierenden Apple® Mac® installieren.
- Können Sie FreeBSD unter Microsoft® Windows® und Virtual PC installieren.
- Wissen Sie, wie man ein virtualisiertes FreeBSD-System für optimale Leistung konfiguriert.

Bevor Sie dieses Kapitel lesen, sollten Sie

- Die Grundlagen von UNIX® und FreeBSD verstehen (Kapitel 4, *Grundlagen des UNIX Betriebssystems*).
- FreeBSD installieren können (Kapitel 2, *FreeBSD 8.X (und älter) installieren*).
- Wissen, wie man seine Netzwerkverbindung konfiguriert (Kapitel 31, *Weiterführende Netzwerkthemen*).
- Software Dritter installieren können (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

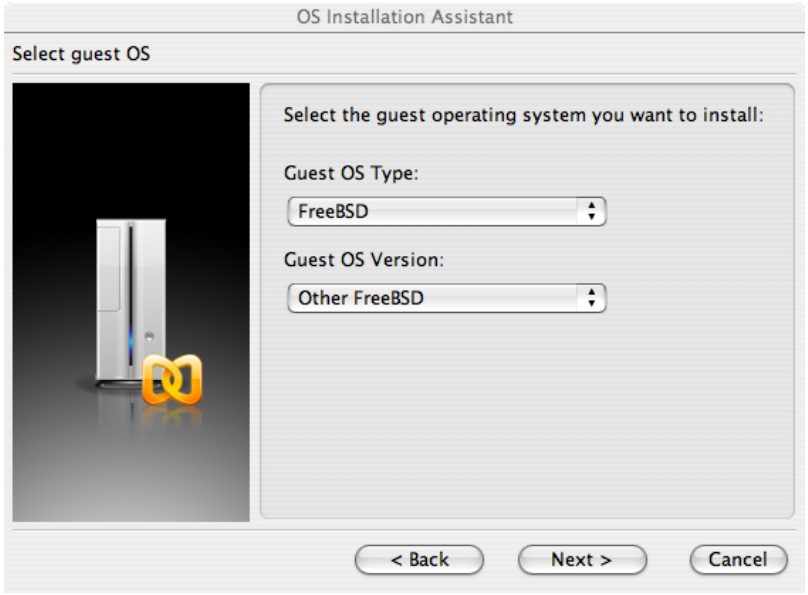
22.2. FreeBSD als Gast-Betriebssystem

22.2.1. Parallels unter MacOS X

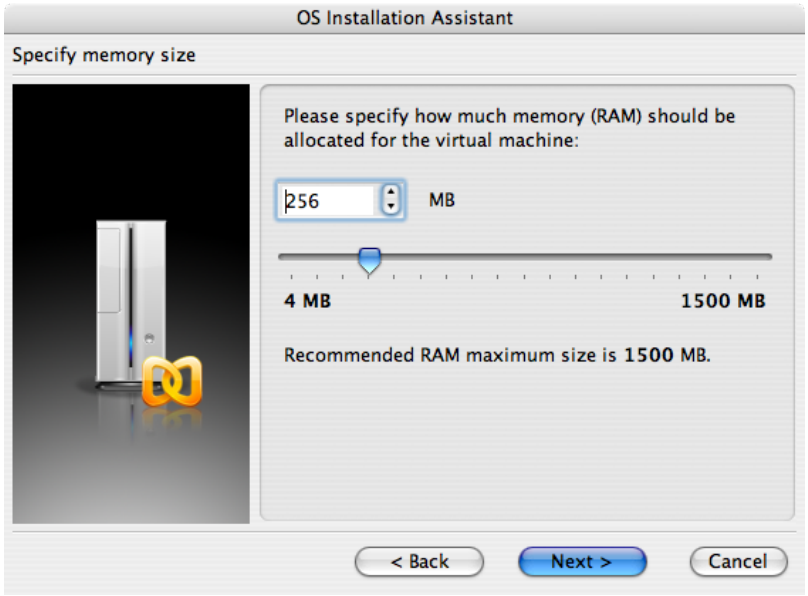
Parallels Desktop für Mac® ist ein kommerzielles Softwareprodukt, welches für Intel®-basierende Apple® Mac®-Computer mit Mac OS® X 10.4.6 oder höher verfügbar ist. FreeBSD wird von diesem Softwarepaket als Gast-Betriebssystem vollständig unterstützt. Nach der Installation von Parallels auf Mac OS® X konfigurieren Sie als erstes eine virtuelle Maschine, in der Sie danach das gewünschte Gast-Betriebssystem (in unserem Fall FreeBSD) installieren.

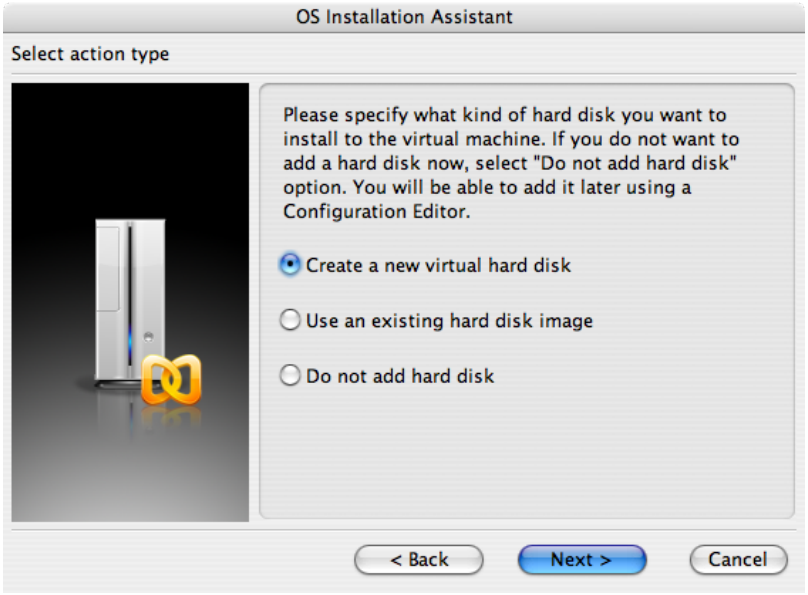
22.2.1.1. Installation von FreeBSD unter Parallels/Mac OS® X

Der erste Schritt bei der Installation von FreeBSD unter Parallels/Mac OS® X ist es, eine virtuelle Maschine zu konfigurieren, in der Sie FreeBSD installieren können. Dazu wählen Sie bei der Frage nach dem Guest OS Type FreeBSD aus:

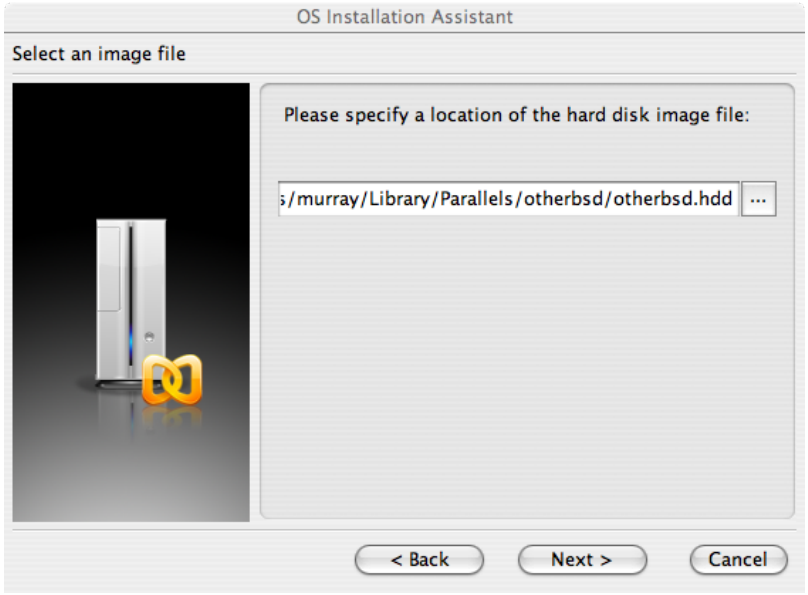


Danach legen Sie geeignete Größen für Festplatten- und Arbeitsspeicher für die zu erstellende FreeBSD-Instanz fest. 4 GB Plattenplatz sowie 512 MB RAM sind in der Regel für die Arbeit unter Parallels ausreichend:

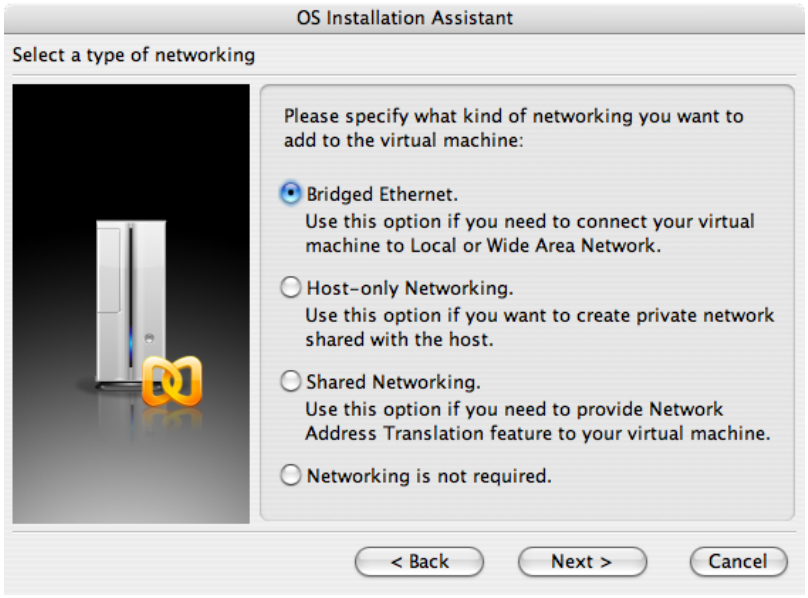


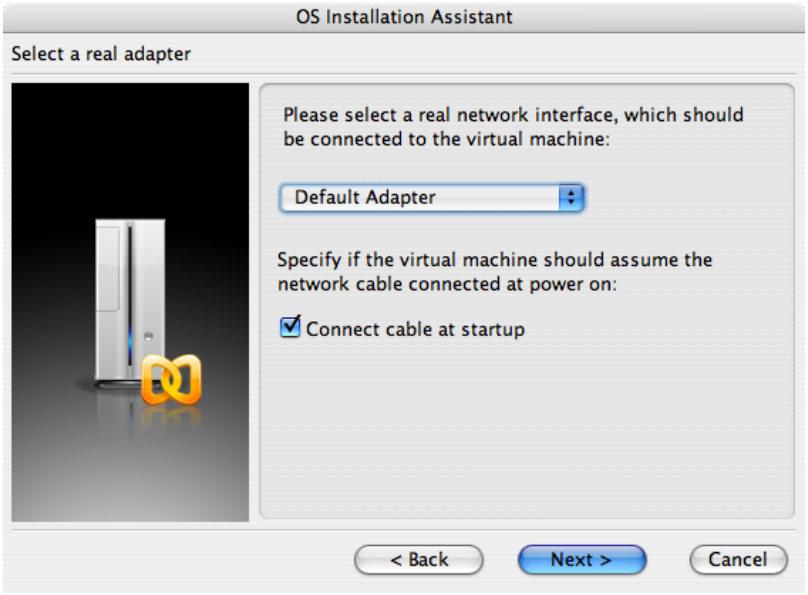

OS Installation Assistant
Select action type
Please specify what kind of hard disk you want to install to the virtual machine. If you do not want to add a hard disk now, select "Do not add hard disk" option. You will be able to add it later using a Configuration Editor.
Create a new virtual hard disk
Use an existing hard disk image
Do not add hard disk
< Back
Next >
Cancel


OS Installation Assistant
Specify hard disk options
Please specify a preferred virtual disk size:
8000
MB
Select disk format:
Expanding (recommended).
Disk image file is small initially and grows as you add more data to the virtual machine. This disk format takes less time to create and saves disk space on the host.
Plain.
Disk image file consumes all the allocated space right from the start. It takes more time to create but allows guest OS to operate faster.
< Back
Next >
Cancel

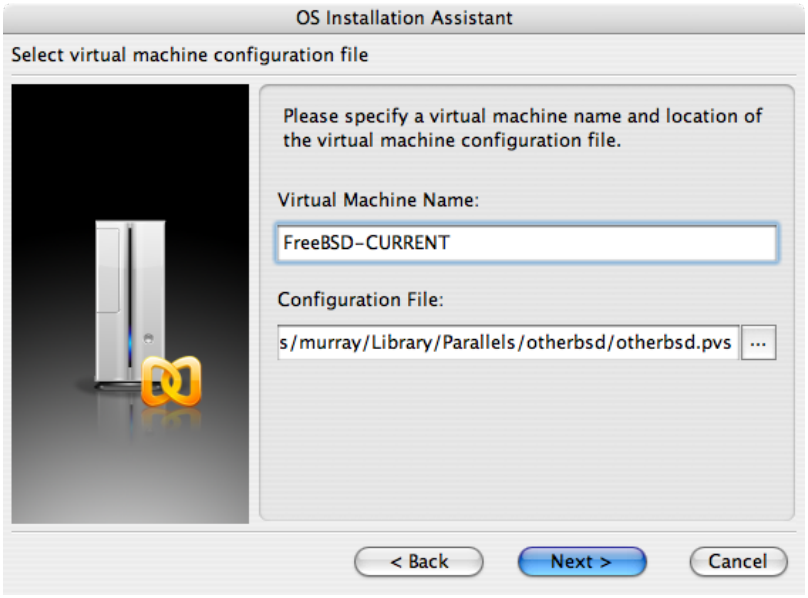


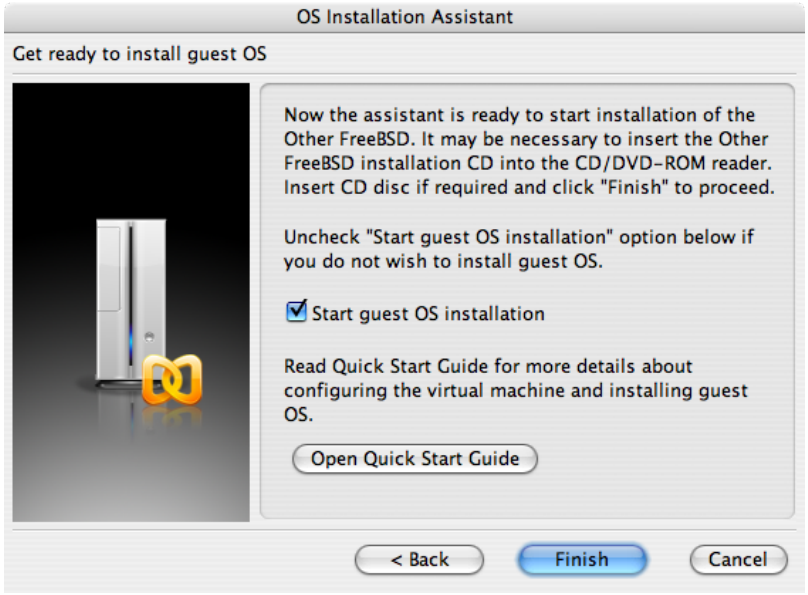
Wählen Sie den gewünschten Netzwerktyp aus und konfigurieren Sie Ihre Netzwerkverbindung:



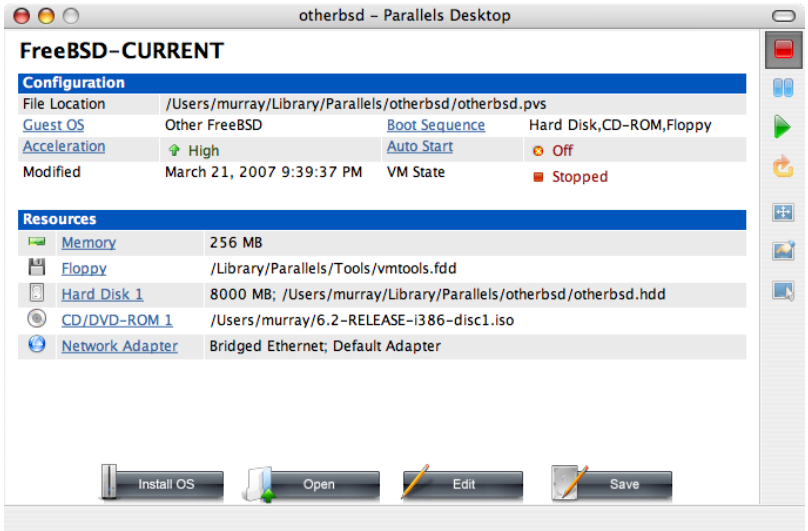


Speichern Sie Ihre Eingaben, um die Konfiguration abzuschließen:

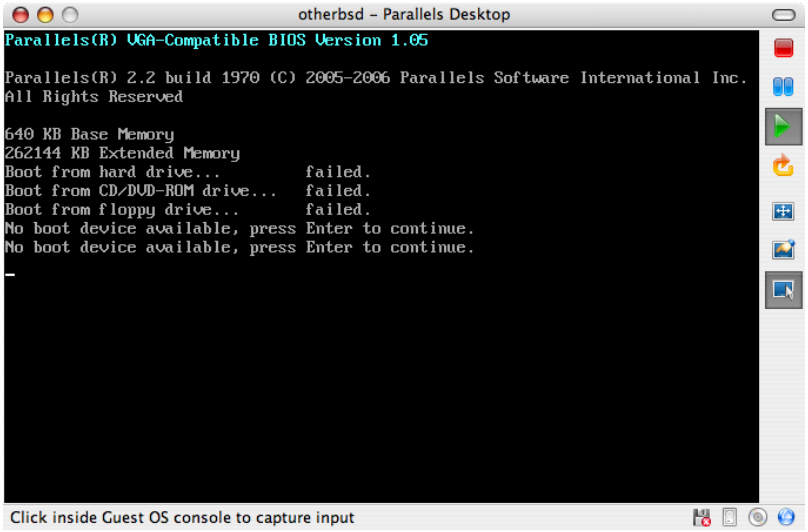




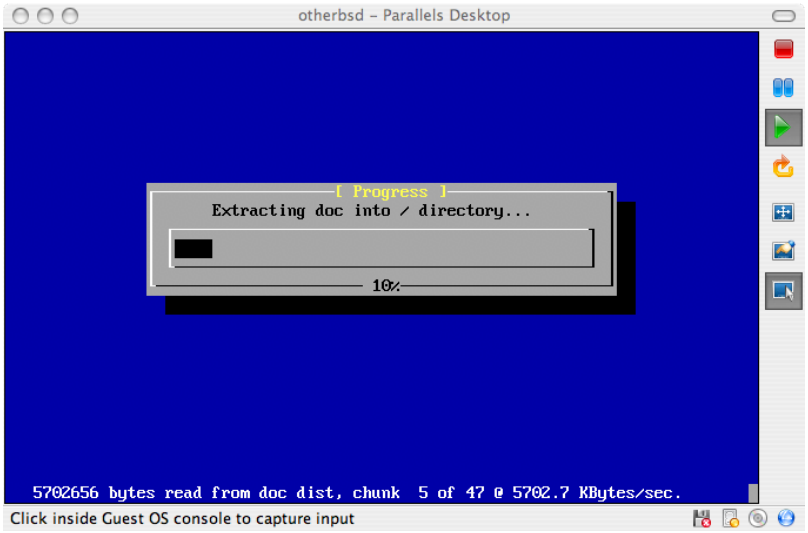
Nachdem Sie die virtuelle Maschine erstellt haben, installieren Sie im nächsten Schritt FreeBSD in dieser virtuellen Maschine. Dazu verwenden Sie am besten eine offizielle FreeBSD-CDROM oder Sie laden von einem offiziellen FTP-Server ein ISO-Abbild auf Ihren Mac® herunter. Danach klicken Sie auf das Laufwerksymbol in der rechten unteren Ecke des Parallels-Fensters, um ihr virtuelles Laufwerk mit dem ISO-Abbild oder mit dem physikalischen CD-ROM-Laufwerk ihres Computers zu verknüpfen.



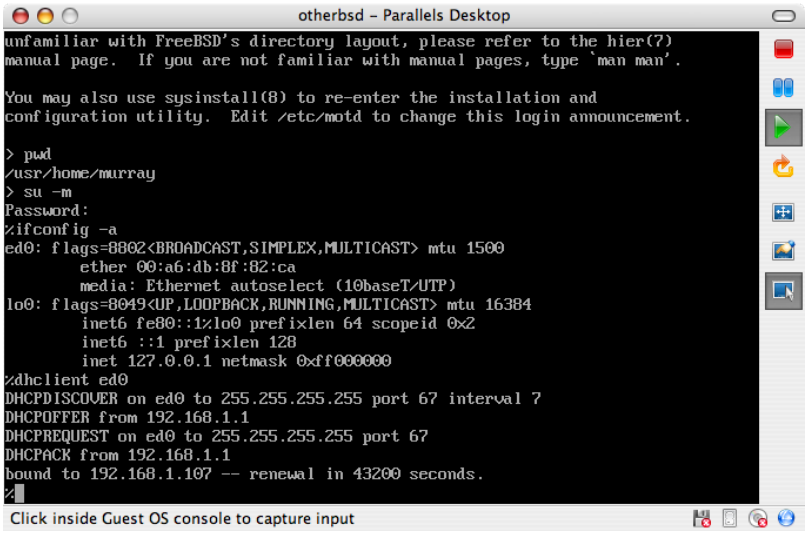
Nachdem Sie diese Verknüpfung hergestellt haben, starten sie die virtuelle FreeBSD-Maschine neu, indem Sie wie gewohnt auf das Symbol "Neustarten" klicken. Parallels startet nun ein Spezial-BIOS, das zuerst prüft, ob Sie eine CD-ROM eingelegt haben (genau so, wie es auch ein echtes BIOS machen würde).



In unserem Fall findet das BIOS ein FreeBSD-Installationsmedium und beginnt daher eine normale Installation mit sysinstall (wie in Kapitel 2, *FreeBSD 8.X (und älter) installieren* des Handbuchs beschreiben).



Nachdem die Installation abgeschlossen ist, können Sie die virtuelle Maschine starten.



22.2.1.2. FreeBSD für den Einsatz unter Parallels/Mac OS® X optimieren

Nachdem Sie FreeBSD erfolgreich unter Mac OS® X mit Parallels installiert haben, sollten Sie ihr virtuelles FreeBSD-System für virtualisierte Operationen optimieren:

1. Setzen der Bootloader-Variablen

   Die wichtigste Änderung ist es, die Variable `kern.hz` zu verkleinern, um so die CPU-Auslastung in der Parallels-Umgebung zu verringern.

   ```
   kern.hz=100
   ```

   Ohne diese Einstellung kann ein unbeschäftigtes FreeBSD unter Parallels trotzdem rund 15 Prozent der CPU-Leistung eines Single Prozessor iMac®'s verbrauchen. Nach dieser Änderung reduziert sich dieser Wert auf etwa 5 Prozent.

2. Erstellen einer neuen Kernelkonfigurationsdatei

   Sie können alle SCSI-, FireWire- und USB-Laufwerks-Treiber entfernen. Parallels stellt einen virtuellen Netzwerkadapter bereit, der den ed(4)-Treiber verwendet. Daher können alle Netzwerkgeräte bis auf ed(4) und miibus(4) aus dem Kernel entfernt werden.

3. Netzwerkbetrieb einrichten

   Die einfachste Netzwerkkonfiguration ist der Einsatz von DHCP, um Ihre virtuelle Maschine mit dem gleichen lokalen Netzwerk, in dem sich der Host-Mac® befindet, zu verbinden. Dazu fügen Sie die Zeile `ifconfig_ed0="DHCP"` in die Datei `/etc/rc.conf` ein. Weitere Informationen zur Konfiguration des Netzwerks unter FreeBSD finden Sie im Kapitel 31, *Weiterführende Netzwerkthemen* des Handbuchs.

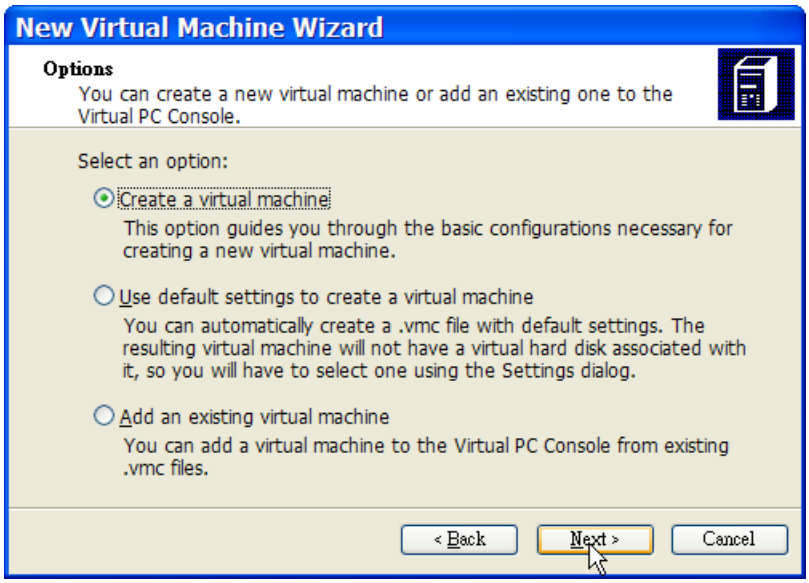
22.2.2. Virtual PC unter Windows®

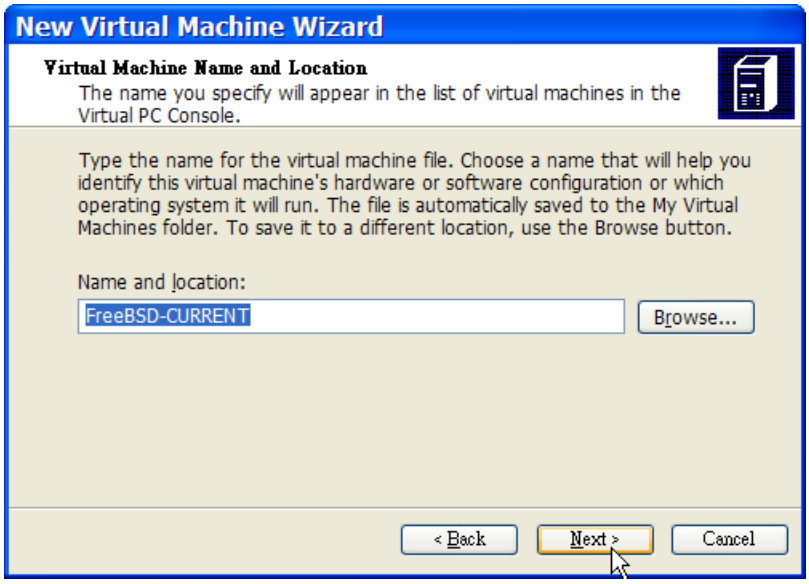
Übersetzt von Johann Kois.

Virtual PC für Windows® wird von Microsoft® kostenlos zum Download angeboten. Die Systemanforderungen für dieses Programm finden Sie hier. Nachdem Sie Virtual PC unter Microsoft® Windows® installiert haben, müssen Sie eine virtuelle Maschine konfigurieren und das gewünschte Betriebssystem installieren.

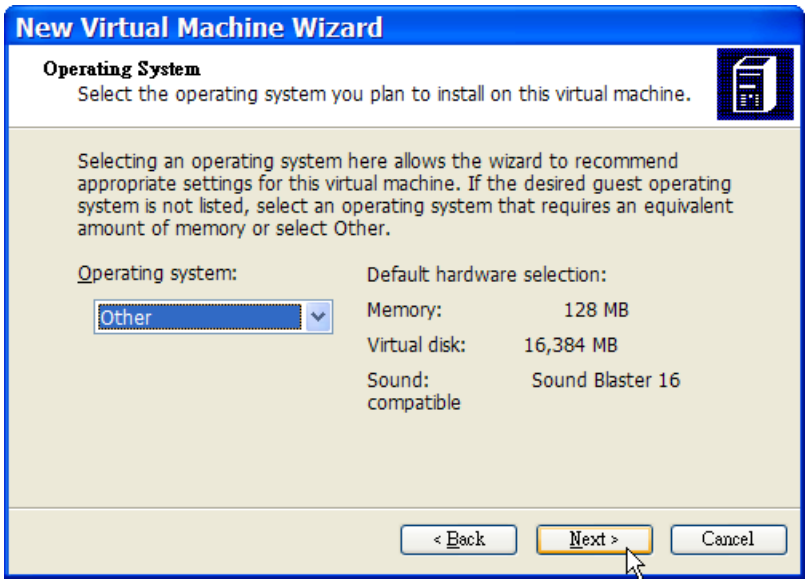
22.2.2.1. FreeBSD in Virtual PC/Microsoft® Windows® installieren

Der erste Schritt zur Installation von FreeBSD in Microsoft® Windows®/Virtual PC ist es, eine neue virtuelle Maschine zu erstellen, in die Sie FreeBSD installieren können. Dazu wählen Sie die Option Create a virtual machine, wenn Sie danach gefragt werden:

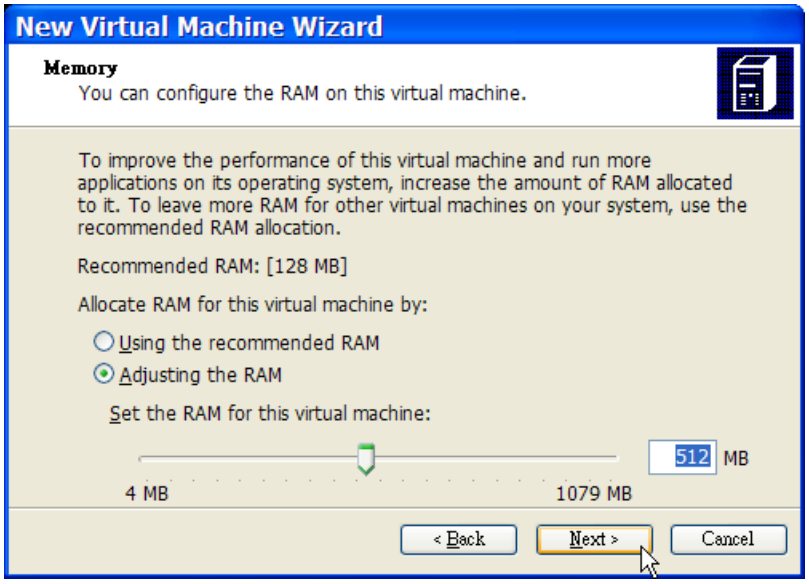


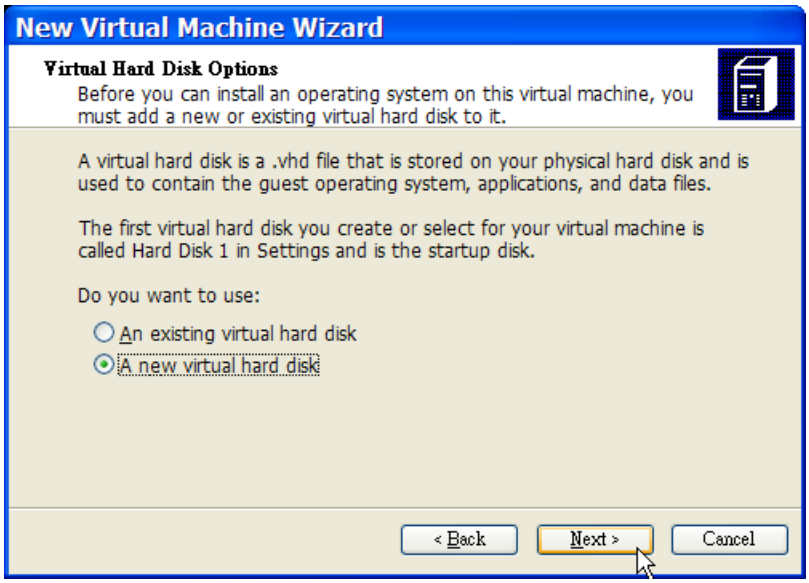


Bei der Frage nach dem Operating system wählen Sie Other:

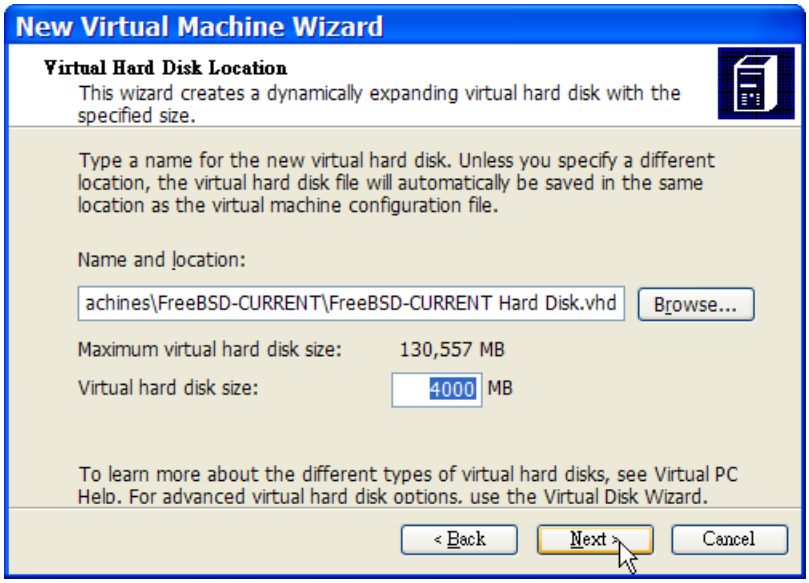


Danach müssen Sie den von Ihnen gewüschten Plattenplatz sowie die Größe des Hauptspeichers angeben. 4 GB Plattenplatz sowie 512 MB RAM sollten für die Installation von FreeBSD in Virtual PC ausreichend sein:

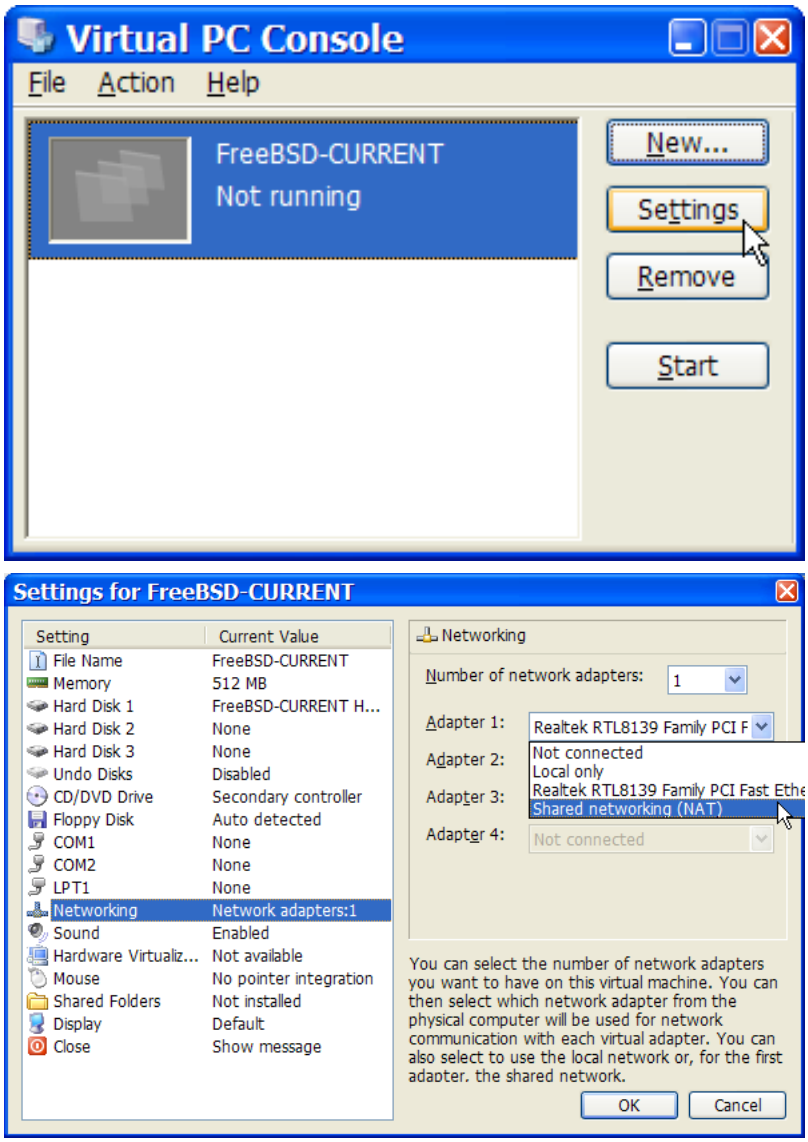




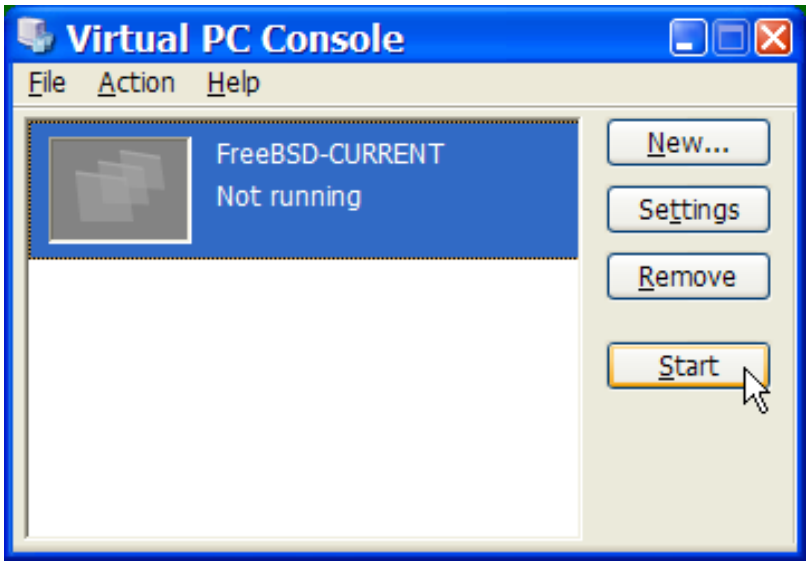
Speichern Sie Ihre Eingaben und beenden Sie die Konfiguration:

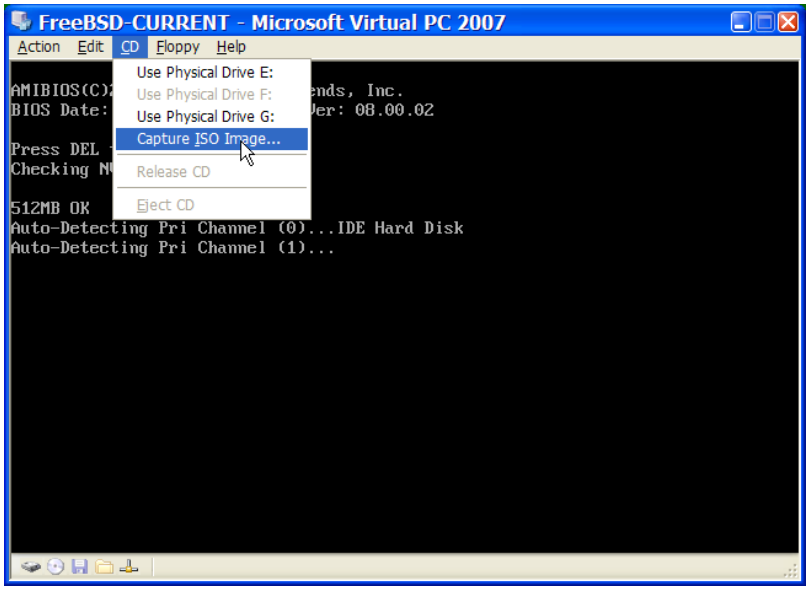


Wählen Sie nun die für FreeBSD erstellte virtuelle Maschine aus und klicken Sie auf Settings, um das Netzwerk zu konfigurieren:

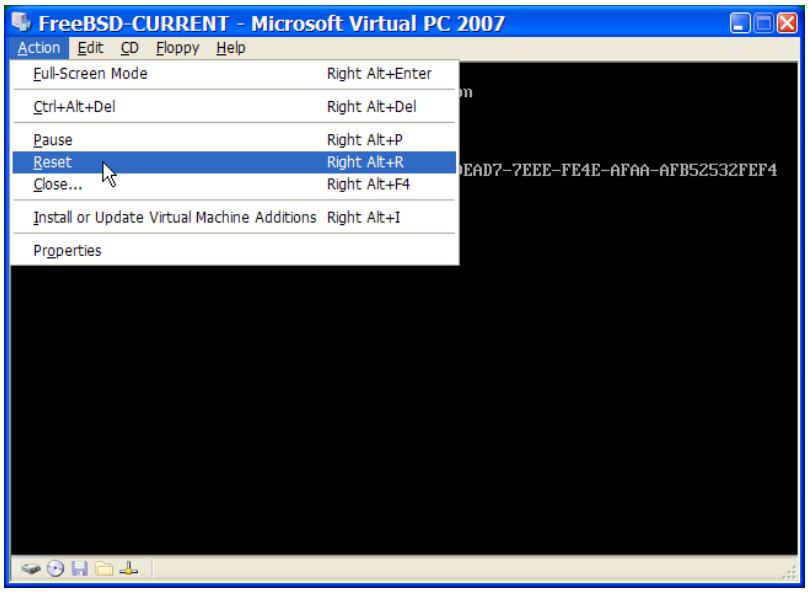


Nun können Sie FreeBSD installieren. Dazu verwenden Sie am besten eine offizielle FreeBSD-CD-ROM oder ein ISO-Image, das Sie von einem offiziellen FreeBSD-FTP-Server heruntergeladen haben. Wenn Sie ein ISO-Image auf Ihrer Festplatte gespeichert haben, oder eine FreeBSD-CD-ROM in Ihr CD-Laufwerk eingelegt haben, doppelklicken Sie auf die virtuelle Maschine, die Sie für FreeBSD angelegt haben. Danach klicken Sie auf CD und wählen die Option Capture ISO Image... im Virtual PC-Fenster. Danach können Sie im folgenden Fenster das CD-Laufwerk mit Ihrem physikalischen CD-Laufwerk oder mit dem ISO-Image verknüpfen.

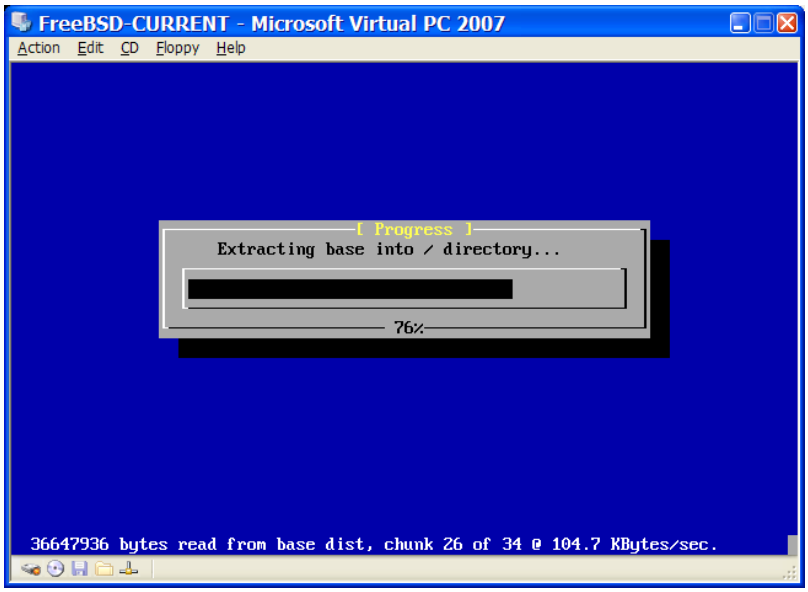




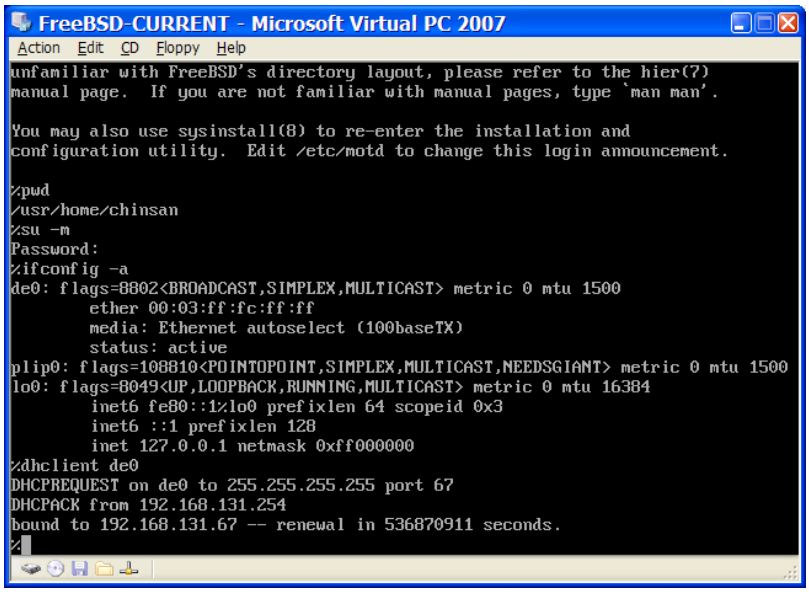
Danach starten Sie die virtuelle Maschine neu, indem Sie zuerst auf Action und danach auf Reset klicken. Virtual PC startet Ihre virtuelle Maschine nun neu und prüft zuerst, ob die virtuelle Maschine über ein CD-Laufwerk verfügt.



Da dies hier der Fall ist, beginnt nun eine normale, auf sysinstall basierende Installation, die in Kapitel 2, *FreeBSD 8.X (und älter) installieren* beschrieben wird. Sie können FreeBSD nun installieren. Verzichten Sie an dieser Stelle aber unbedingt auf die X11-Konfiguration.



Nachdem die Installation abgeschlossen ist, entfernen Sie die CD-ROM aus dem Laufwerk (oder lösen die Verknüpfung zum ISO-Image). Danach starten Sie die virtuelle Maschine neu, um FreeBSD zu starten.



22.2.2.2. FreeBSD in Microsoft® Windows®/Virtual PC konfigurieren

Nachdem Sie FreeBSD auf Ihrem Microsoft® Windows®-System erfolgreich unter Virtual PC installiert haben, sollten Sie ihr virtuelles FreeBSD noch anpassen, um eine optimale Funktion zu gewährleisten.

1. Setzen der Bootloader-Variablen

   Die wichtigste Änderung ist es, die Variable `kern.hz` zu verkleinern, um so die CPU-Auslastung in der Virtual PC-Umgebung zu verringern. Dazu fügen Sie die folgende Zeile in die Datei `/boot/loader.conf` ein:

   ```
   kern.hz=100
   ```

   Ohne diese Einstellung kann ein unbeschäftigtes FreeBSD unter Virutal PC trotzdem rund 40 Prozent der CPU-Leistung eines Ein-Prozessor-Systems verbrauchen. Nach dieser Änderung reduziert sich dieser Wert auf etwa 5 Prozent.

2. Erstellen einer neuen Kernelkonfigurationsdatei

   Sie können alle SCSI-, FireWire- und USB-Laufwerks-Treiber entfernen. Virtual PC stellt einen virtuellen Netzwerkadapter bereit, der den de(4)-Treiber verwendet. Daher können alle Netzwerkgeräte bis auf de(4) und miibus(4) aus dem Kernel entfernt werden.

3. Das Netzwerk einrichten

   Die einfachste Netzwerkkonfiguration ist der Einsatz von DHCP, um Ihre virtuelle Maschine mit dem gleichen lokalen Netzwerk, in dem sich Ihr Host-Microsoft® Windows® befindet, zu verbinden. Dazu fügen Sie die Zeile `ifconfig_de0="DHCP"` in die Datei `/etc/rc.conf` ein. Weitere Informationen zur Konfiguration des Netzwerks unter FreeBSD finden Sie im Kapitel 31, *Weiterführende Netzwerkthemen* des Handbuchs.

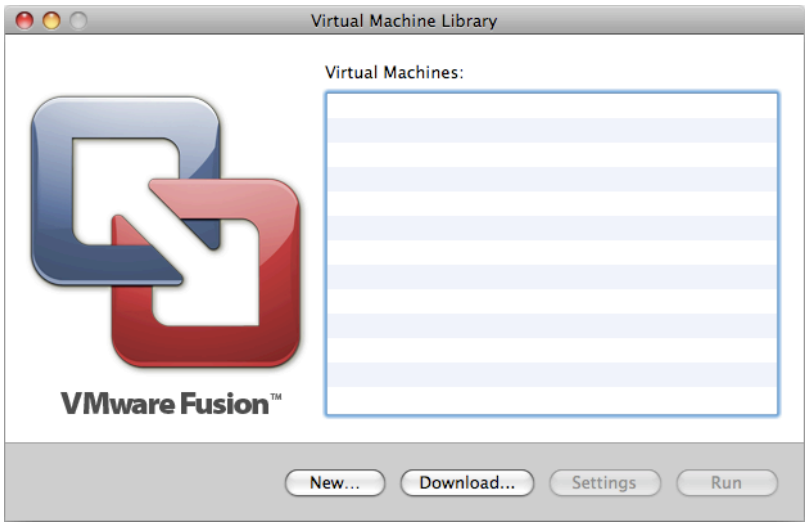
22.2.3. VMware unter MacOS

Übersetzt von Johann Kois.

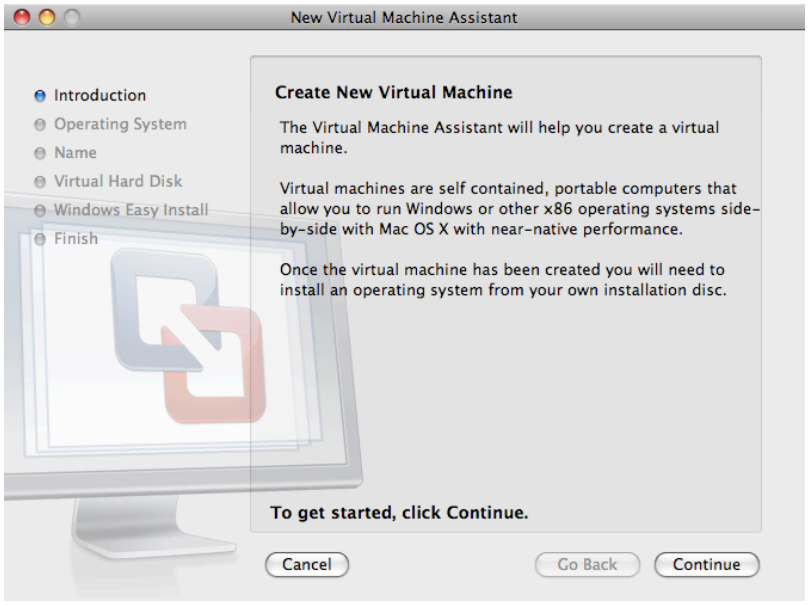
VMware Fusion für Mac® ist ein kommerzielles Programm, das für Intel® basierte Apple® Mac®-Computer mit Mac OS® 10.4.9 oder neuer erhältlich ist. FreeBSD wird von diesem Produkt vollständig als Gast-Betriebssystem unterstützt. Nachdem Sie VMware Fusion unter Mac OS® X installiert haben, können Sie das gewünschte Gastbetriebssystem (in unserem Fall FreeBSD) installieren.

22.2.3.1. FreeBSD in VMware/Mac OS® X installieren

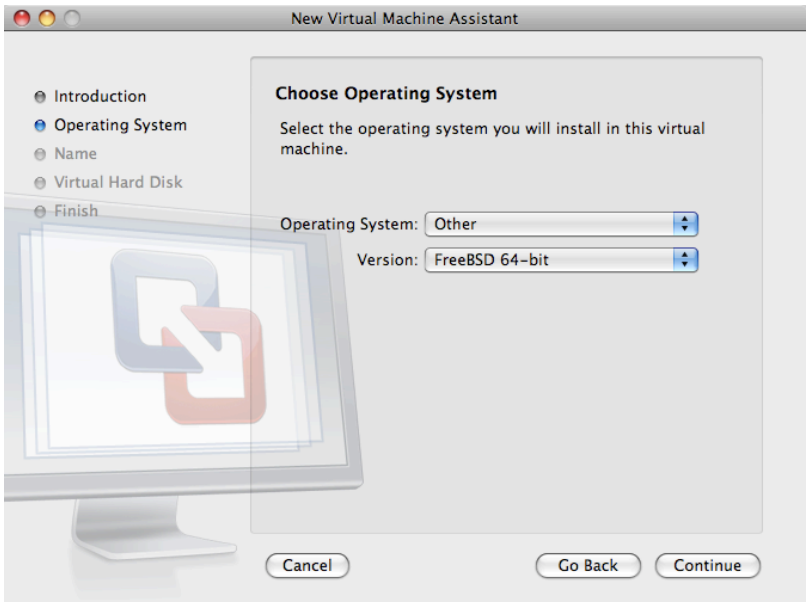
Zuerst müssen Sie VMware Fusion starten, um eine virtuelle Maschine zu erstellen. Dazu wählen Sie die Option "New":



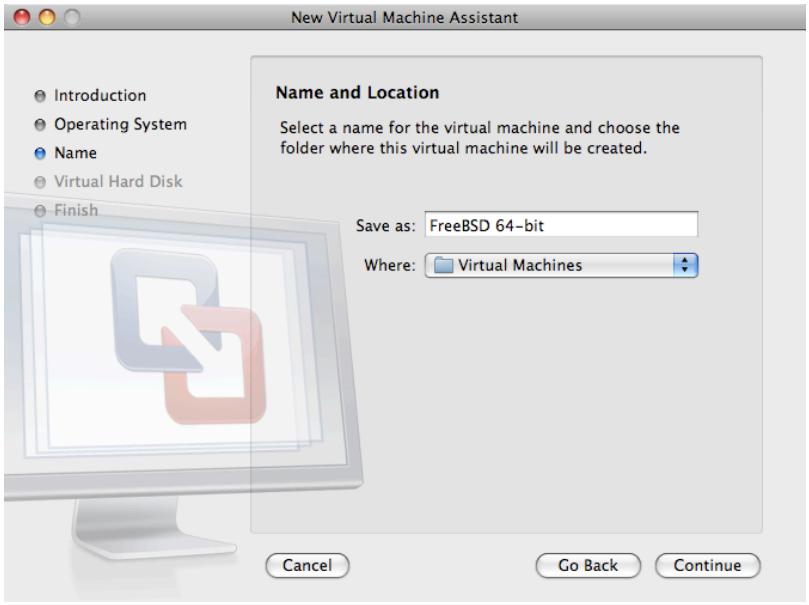
Dadurch wird ein Assistent gestartet, der Ihnen bei der Erzeugung einer neuen virtuellen Maschine behilflich ist. Clicken Sie auf "Continue", um den Prozess zu starten:



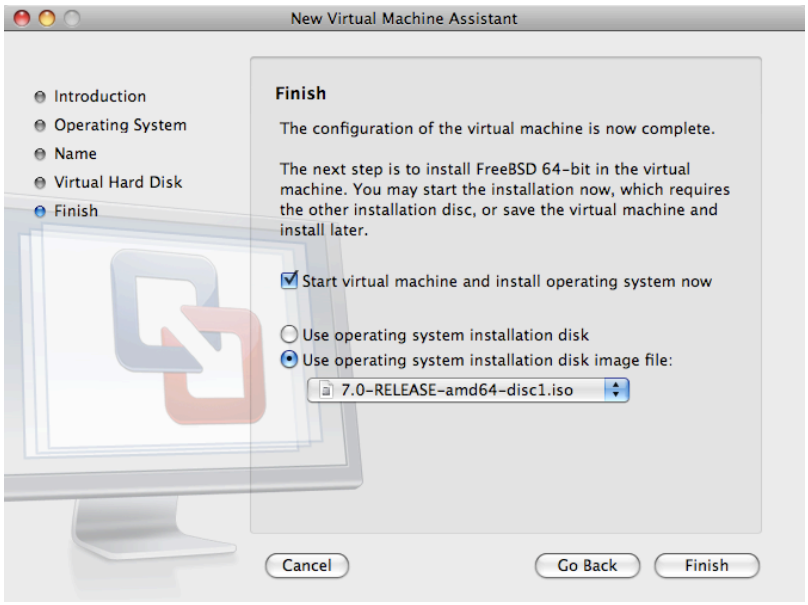
Wählen Sie Other als das Operating System, danach FreeBSD oder FreeBSD 64-bit, je nach dem, welche Version Sie installieren wollen, wenn Sie nach der zu installierenden Version gefragt werden:



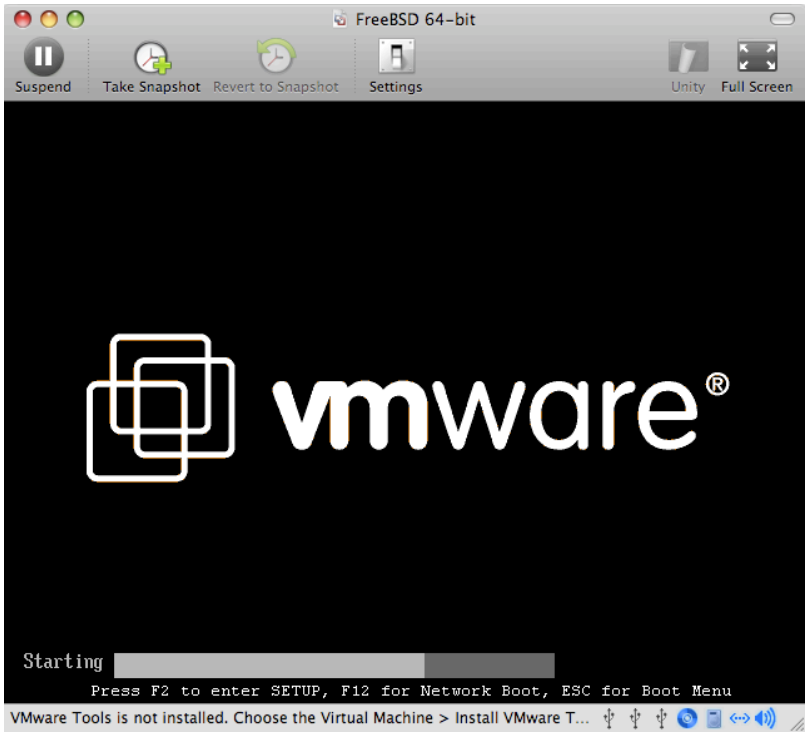
Vergeben Sie einen Namen für virtuelle Maschine an und legen Sie den Speicherort fest:



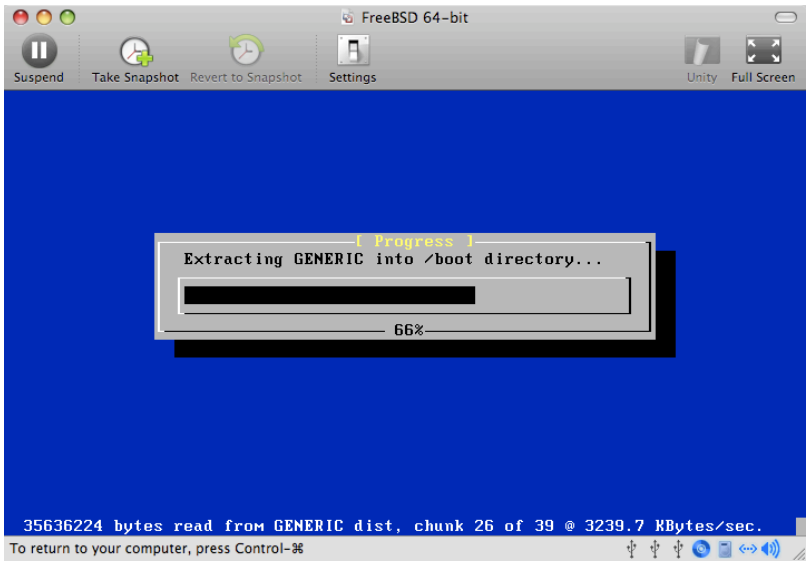
Legen Sie die Größe Ihrer virtuellen Festplatte fest:



Nachdem Sie auf "Finish" geklickt haben, wird die virtuelle Maschine gestartet:



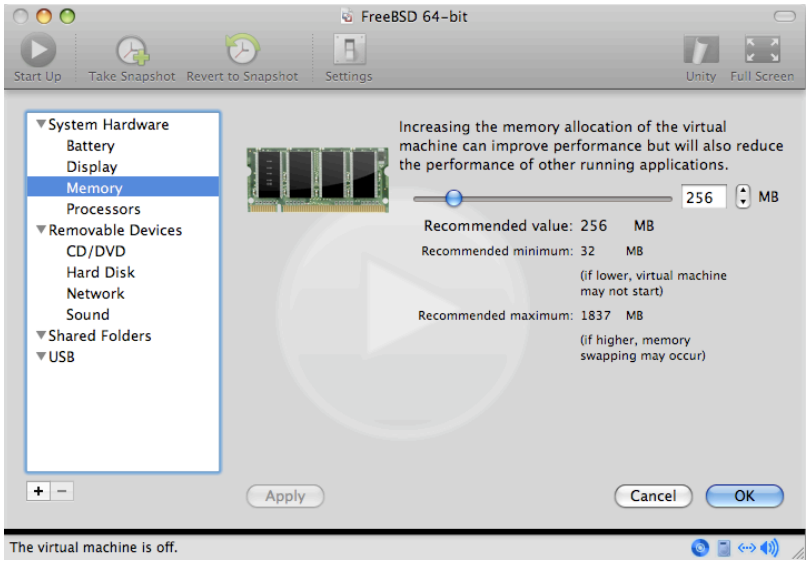
Nun können Sie FreeBSD wie gewohnt installieren (lesen Sie dazu auch Kapitel 2, *FreeBSD 8.X (und älter) installieren* des Handbuchs):



Nachdem die Installation abgeschlossen ist, können Sie noch verschiedene Parameter der virtuellen Maschine, etwa den Speicherverbrauch, konfigurieren:

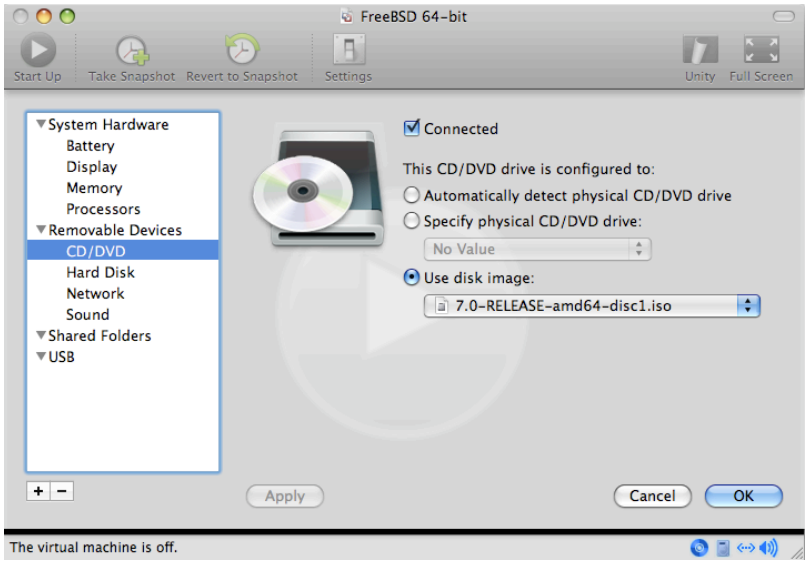
Anmerkung

Die Hardware der virtuellen Maschine kann nicht geändert werden, solange die virtuelle Maschine läuft.

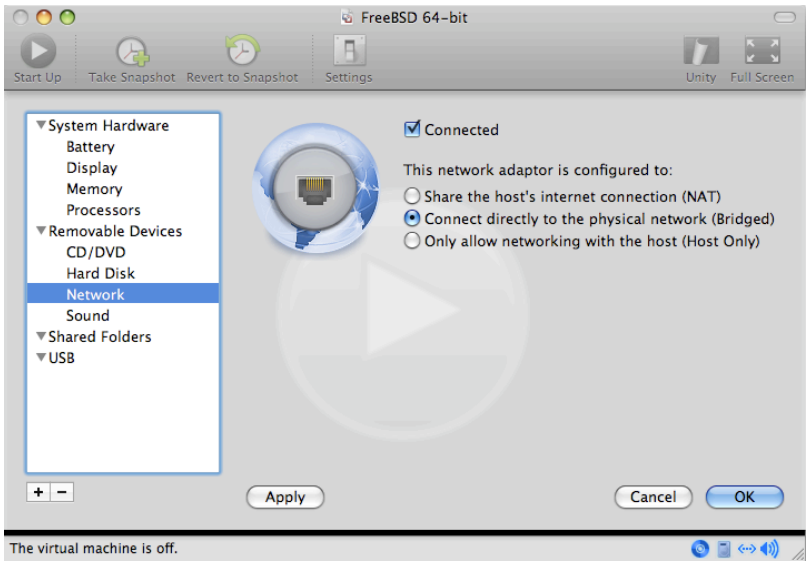


Die Anzahl der CPUs der virtuellen Maschine:

Den Status des CD-Laufwerks. Sie können das CD-Laufwerk von der virtuellen Maschine lösen, wenn Sie es nicht benötigen.



Zuletzt sollten Sie noch festlegen, wie sich die virtuelle Maschine mit dem Netzwerk verbinden soll. Sollen neben dem Gastsystem auch andere Rechner auf Ihre virtuelle Maschine zugreifen können, müssen Sie die Option Connect directly to the physical network (Bridged) wählen. Ist dies nicht der Fall, sollten Sie die Option Share the host's internet connection (NAT) wählen. In dieser Einstellung kann die virtuelle Maschine zwar auf auf das Internet zugreifen, andere Rechner dürfen aber nicht auf die virtuelle Maschine zugreifen.



Nachdem Sie die Konfiguration abgeschlossen haben, können Sie FreeBSD starten.

22.2.3.2. FreeBSD unter Mac OS® X/VMware konfigurieren

Nachdem Sie FreeeBSD erfolgreich unter VMware für Mac OS® X installiert haben, sollten Sie ihr virtuelles FreeBSD noch anpassen, um eine optimale Funktion zu gewährleisten.

1. Die wichtigste Änderung ist es, die Variable `kern.hz` zu verkleinern, um so die CPU-Auslastung in der VMware-Umgebung zu verringern.

   ```
   kern.hz=100
   ```

   Ohne diese Einstellung kann ein unbeschäftigtes FreeBSD unter VMware trotzdem rund 15 Prozent der CPU-Leistung eines Single Prozessor iMac®'s verbrauchen. Nach dieser Änderung reduziert sich dieser Wert auf etwa 5 Prozent.

2. Erstellen einer neuen Kernelkonfigurationsdatei

   Sie können alle FireWire- und USB-Laufwerks-Treiber entfernen. VMware stellt einen virtuellen Netzwerkadapter bereit, der den em(4)-Treiber verwendet. Daher können alle Netzwerkgeräte bis auf em(4) und miibus(4) aus dem Kernel entfernt werden.

3. Netzwerkbetrieb einrichten

   Die einfachste Netzwerkkonfiguration ist der Einsatz von DHCP, um Ihre virtuelle Maschine mit dem gleichen lokalen Netzwerk, in dem sich der Host-Mac® befindet, zu verbinden. Dazu fügen Sie die Zeile `ifconfig_em0="DHCP"` in die Datei `/etc/rc.conf` ein. Weitere Informationen zur Konfiguration des Netzwerks unter FreeBSD finden Sie im Kapitel 31, *Weiterführende Netzwerkthemen* des Handbuchs.

22.2.4. VirtualBox™ Gasterweiterungen auf einem FreeBSD Gast

Die VirtualBox™ Gasterweiterungen bieten Unterstützung für:

- Gemeinsame Zwischenablage
- Mauszeiger-Integration
- Zeitsynchronisation mit dem Host
- Skalierung von Fenstern

- Nahtloser Modus

Anmerkung

Die folgenden Kommandos werden im FreeBSD Gastsystem ausgeführt.

Installieren Sie das Paket emulators/virtualbox-ose-additions in das FreeBSD Gastsystem:

```
# cd /usr/ports/emulators/virtualbox-ose-additions
# make install clean
```

Fügen Sie folgende Einträge in `/etc/rc.conf` hinzu:

```
vboxguest_enable="YES"
vboxservice_enable="YES"
```

Wenn ntpd(8) oder ntpdate(8) verwendet wird um die Uhrzeit zu synchronisieren, dann sollte die Synchronisierung mit dem Host deaktiviert werden:

```
vboxservice_flags"--disable-timesync"
```

`vboxvideo_drv` sollte von `Xorg -configure` automatisch erkannt werden. Alternativ kann auch manuell ein entsprechender Eintrag für die VirtualBox™ Grafikkarte in `xorg.conf` hinzugefügt werden:

```
Section "Device"
   Identifier "Card0"
   Driver "vboxvideo"
   VendorName "InnoTek Systemberatung GmbH"
   BoardName "VirtualBox Graphics Adapter"
EndSection
```

Um `vboxmouse_drv` zu verwenden, muss `xorg.conf` ebenfalls angepasst werden:

```
Section "InputDevice"
   Identifier "Mouse0"
   Driver "vboxmouse"
EndSection
```

Benutzer von HAL sollten die Datei `/usr/local/etc/hal/fdi/policy/90-vboxguest.fdi` erstellen oder sie aus `/usr/local/share/hal/fdi/policy/10osvendor/90-vboxguest.fdi` kopieren:

```
<?xml version="1.0" encoding="utf-8"?>
<!--
# Sun VirtualBox
# Hal driver description for the vboxmouse driver
# $Id: chapter.xml,v 1.33 2012-03-17 04:53:52 eadler Exp $

>>>>>>>>Copyright (C) 2008-2009 Sun Microsystems, Inc.

 This file is part of VirtualBox Open Source Edition (OSE, as
 available from http://www.virtualbox.org. This file is free software;
 you can redistribute it and/or modify it under the terms of the GNU
 General Public License (GPL) as published by the Free Software
 Foundation, in version 2 as it comes in the "COPYING" file of the
 VirtualBox OSE distribution. VirtualBox OSE is distributed in the
 hope that it will be useful, but WITHOUT ANY WARRANTY of any kind.

 Please contact Sun Microsystems, Inc., 4150 Network Circle, Santa
 Clara, CA 95054 USA or visit http://www.sun.com if you need
 additional information or have any questions.
-->
```

```
<deviceinfo version="0.2">
  <device>
    <match key="info.subsystem" string="pci">
      <match key="info.product" string="VirtualBox guest Service">
        <append key="info.capabilities" type="strlist">input</append>
>>>>>>>><append key="info.capabilities" type="strlist">input.mouse</append>
        <merge key="input.x11_driver" type="string">vboxmouse</merge>
>>>>>>>><merge key="input.device" type="string">/dev/vboxguest</merge>
      </match>
    </match>
  </device>
</deviceinfo>
```

22.3. FreeBSD als Host-Betriebssystem

Übersetzt von Benedict Reuschling und Christoph Sold.

Seit einigen Jahren wurde FreeBSD nicht offiziell von irgendeiner der verfügbaren Virtualisierungslösungen als Host-Betriebssystem unterstützt. Viele Anwender verwenden aber noch ältere VMware-Versionen (z.B. emulators/vmware3), welches die Linux®-Kompatibilitätsschicht nutzt. Kurz nach der Veröffentlichung von FreeBSD 7.2 erschien VirtualBox™ als Open-Source Edition (OSE) von Sun™ in der Ports-Sammlung als ein direkt auf FreeBSD lauffähiges Programm.

VirtualBox™ ist ein vollständiges Virtualisierungspaket, das aktiv weiterentwickelt wird und für die meisten Betriebssysteme einschliesslich Windows®, Mac OS®, Linux® und FreeBSD zur Verfügung steht. Es kann sowohl Windows® als auch UNIX®-ähnliche Gastsysteme betreiben. Es ist als Open Source und als proprietäre Edition erhältlich. Die wichtigste Einschränkung der OSE aus Anwendersicht ist die fehlende USB-Unterstützung. Weitere Unterschiede können von der „Editions“-Seite des VirtualBox™-Wikis, das unter `http://www.virtualbox.org/wiki/Editions` zu finden ist, entnommen werden. Momentan steht nur OSE unter FreeBSD zur Verfügung.

22.3.1. VirtualBox™ installieren

VirtualBox™ steht als FreeBSD-Port in emulators/virtualbox-ose bereit. Da VirtualBox™ sehr aktiv weiterentwickelt wird, vergewissern Sie sich, dass Ihre Ports-Sammlung aktuell ist. Benutzen Sie die folgenden Befehle um VirtualBox™ zu installieren:

```
# cd /usr/ports/emulators/virtualbox-ose
# make install clean
```

Eine nützliche Option im Konfigurationsdialog ist die `GuestAdditions`-Programmsammlung. Diese stellen eine Reihe von nützlichen Eigenschaften in den Gastbetriebssystemen zur Verfügung, wie beispielsweise Mauszeigerintegration (was es ermöglicht, die Maus zwischen dem Host und dem Gast zu teilen ohne eine spezielle Tastenkombination für diesen Wechsel zu drücken), sowie schnelleres Rendern von Videos, besonders in Windows® Gästen. Diese Gastzusätze sind im Devices-Menü zu finden, nachdem die Installation des Gastbetriebssystem abgeschlossen ist.

Ein paar Konfigurationsänderungen sind notwendig, bevor VirtualBox™ das erste Mal gestartet wird. Der Port installiert ein Kernelmodul in `/boot/modules` , das in den laufenden Kernel geladen werden muss:

```
# kldload vboxdrv
```

Um sicherzustellen, dass das Modul immer nach einem Neustart geladen wird, fügen Sie die folgende Zeile in die Datei `/boot/loader.conf` ein:

```
vboxdrv_load="YES"
```

Um die Kernelmodule für die Unterstützung von Netzwerkbrücken oder *Host-Only* Netzwerken zu laden, fügen Sie folgendes in `/etc/rc.conf` ein und starten Sie den Computer neu:

```
vboxnet_enable="YES"
```

Die Gruppe `vboxusers` wird während der Installation von VirtualBox™ angelegt. Alle Benutzer, die Zugriff auf VirtualBox™ haben sollen, müssen in diese Gruppe aufgenommen werden. Der `pw`-Befehl kann benutzt werden, um neue Mitglieder hinzuzufügen:

```
# pw groupmod vboxusers -m yourusername
```

Damit Netzwerkbrücken funktionieren, müssen die in der Voreinstellung eingeschränkten Berechtigungen für `/dev/vboxnetctl` angepasst werden:

```
# chown root:vboxusers /dev/vboxnetctl
# chmod 0600 /dev/vboxnetctl
```

Um die Berechtigungen dauerhaft zu speichern, fügen Sie folgende Einträge in `/etc/devfs.conf` hinzu:

```
own     vboxnetctl root:vboxusers
perm    vboxnetctl 0600
```

Um VirtualBox™ zu starten, wählen Sie entweder den Eintrag Sun VirtualBox aus dem Menü der graphischen Benutzeroberfläche, oder geben Sie den folgenden Befehl in ein Terminal ein:

```
% VirtualBox
```

Besuchen Sie die offizielle Webseite von VirtualBox™ unter `http://www.virtualbox.org` , um weitere Informationen zur Konfiguration und Verwendung zu erhalten. Da der FreeBSD-Port noch recht neu ist, befindet er sich noch unter ständiger Entwicklung. Um die aktuellen Nachrichten und Anleitungen zur Fehlerbehebung zu erhalten, besuchen Sie die entsprechende Seite im FreeBSD-Wiki unter `http://wiki.FreeBSD.org/VirtualBox` .

22.3.2. USB Unterstützung für VirtualBox™

Anmerkung

Für die folgenden Schritte wird mindestens VirtualBox™ 4.0.0 vorausgesetzt.

Um auf USB-Geräte lesend und schreibend zuzugreifen, müssen Benutzer Mitglied der Gruppe `operator` sein:

```
# pw groupmod operator -m jerry
```

Fügen Sie dann folgenden Eintrag in `/etc/devfs.rules` hinzu. Erstellen Sie die Datei, wenn sie nicht bereits existiert:

```
[system=10]
add path 'usb/*' mode 0660 group operator
```

Um diese neuen Regeln zu laden, fügen Sie folgenden Eintrag in `/etc/rc.conf` hinzu:

```
devfs_system_ruleset="system"
```

Danach starten Sie devfs neu:

```
# service devfs restart
```

USB kann nun im Gastsystem aktiviert werden. Die USB-Geräte sollten nun in den VirtualBox™ Einstellungen sichtbar sein.

22.3.3. Host CD/DVD-Zugriff in VirtualBox™

Das Kernelmodul `atapicam` muss durch die folgenden Einträge in `/boot/loader.conf` geladen werden:

```
atapicam_load="YES"
```

Damit die CD/DVD-Funktionen von VirtualBox™ funktionieren, muss HAL in `/etc/rc.conf` aktiviert und anschließend gestartet werden:

```
hald_enable="YES"
```

```
# service hald start
```

Damit die CD/DVD-Funktionen von Benutzern verwendet werden können, benötigen diese Zugriff auf `/dev/xpt0` , `/dev/cd`*N* und `/dev/pass`*N*. Fügen Sie folgende Einträge in `/etc/devfs.conf` hinzu:

```
perm cd0 0600
perm xpt0 0660
perm pass0 0660
```

Kapitel 23. Lokalisierung – I18N/ L10N einrichten und benutzen

Beigesteuert von Andrey Chernov.
Überarbeitet von Michael C. Wu.
Übersetzt von Alexander Langer und Martin Heinen.

23.1. Übersicht

FreeBSD ist ein über die ganze Welt verteiltes Projekt. Dieses Kapitel behandelt die Internationalisierung und Lokalisierung von FreeBSD, mit denen nicht englisch sprechende Benutzer FreeBSD an ihre Bedürfnisse anpassen können. Die Internationalisierung betrifft sowohl die System- als auch die Anwendungsebene, daher wird im Laufe des Texts auf genauere Anwendungsdokumentationen verwiesen.

Nachdem Sie dieses Kapitel durchgearbeitet haben, werden Sie wissen

- wie verschiedene Sprachen und Lokalisierungen in modernen Betriebssystemen codiert werden,
- wie Sie die Locale Ihrer Login-Shell setzen,
- wie Sie die Konsole für nicht-englische Sprachen konfigurieren,
- wie Sie das X Window System mit verschiedenen Sprachen benutzen,
- wo Sie mehr Informationen über das Erstellen von I18N-konformen Anwendungen erhalten.

Bevor Sie dieses Kapitel lesen, sollten Sie

- wissen, wie Sie zusätzliche Anwendungen installieren (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

23.2. Grundlagen

23.2.1. Was ist I18N/L10N?

Entwickler kürzen das Wort *internationalization* (englisch für Internationalisierung) mit I18N ab, weil sich zwischen dem ersten und letzten Buchstaben des Worts 18 Buchstaben befinden. L10N benutzt die gleiche Namensgebung und ist eine Abkürzung des Worts *localization* (englisch für Lokalisierung). Mit I18N/L10N-Methoden, -Protokollen und -Anwendungen können Benutzer eine Sprache ihrer Wahl verwenden.

I18N-Anwendungen werden mit Hilfe von I18N-Bibliotheken programmiert. Diese erlauben es Entwicklern, eine einfache Sprachdatei zu schreiben und Menüs und Texte an jede Sprache anzupassen. Wir möchten Programmierern empfehlen, für ihre eigenen Anwendungen auf diese Techniken zurückzugreifen.

23.2.2. Wieso soll ich I18N/L10N benutzen?

I18N/L10N wird immer dann benutzt, wenn Sie Daten in anderen Sprachen als Englisch anzeigen, eingeben oder verarbeiten möchten.

23.2.3. Welche Sprachen werden von I18N unterstützt?

I18N und L10N sind nichts FreeBSD spezifisches. Momentan können Sie unter den meisten der verbreitetsten Sprachen der Welt wählen, unter anderen Chinesisch, Japanisch, Koreanisch, Französisch, Russisch und Deutsch.

23.3. Lokale Anpassungen benutzen

In seiner ganzen Schönheit ist L10N nichts, was auf FreeBSD alleine beschränkt ist, im Gegenteil, es ist eine Konvention, an die sich viele Programme für verschiedene Betriebssysteme halten. Wir möchten Sie anregen, FreeBSD bei der Unterstützung dieser Konvention zu helfen.

Lokale Anpassungen werden durch die Angabe von drei Werten erreicht: dem Sprachcode, dem Ländercode und der Codierung. Die Zusammenfassung dieser Werte wird „Locale“ genannt und sieht wie folgt aus:

```
Sprachcode _Ländercode .Codierung
```

23.3.1. Sprach- und Ländercodes

Um FreeBSD (oder ein anderes UNIX® System, das I18N unterstützt) an lokale Gegebenheiten und Sprachen anzupassen, muss der Benutzer herausfinden, welche Codes für sein Land und seine Sprache benutzt werden. Ländercodes geben den Anwendungen dabei vor, welche Variation einer bestimmten Sprache zu benutzen ist. Eine Variation von Deutsch wäre zum Beispiel de_CH, das eine lokale Anpassung an das in der Schweiz gesprochene Deutsch meint. Außerdem benutzen Webbrowser, SMTP/POP Server, Webserver usw. diese, um Entscheidungen über die Sprache zu fällen. Im Folgenden sind einige Beispiele für Sprach- und Ländercodes aufgelistet:

Sprachcode/Ländercode	Beschreibung
en_US	Englisch - USA
ru_RU	Russisch für Russland
zh_TW	Traditionelles Chinesisch für Taiwan

Eine vollständige Liste der verfügbaren Lokalisierungen erhalten Sie durch die Eingabe von:

```
% locale -a
```

23.3.2. Codierungen

Einige Sprachen benutzen Codierungen, die nicht dem 7-Bit breitem ASCII-Standard entsprechen, wie 8-Bit Codierungen, Wide- oder Multibyte Zeichen (multibyte(3) geht darauf näher ein). Ältere Anwendungen erkennen diese Zeichen nicht und halten sie fälschlicherweise für Steuerzeichen. Neuere Anwendungen erkennen für gewöhnlich 8-Bit Zeichen. Es hängt allerdings von der Implementierung ab, ob man eine Anwendung neu kompilieren muss, um in den Genuss von lokalen Zeichensätzen zu kommen, oder ob man es sie nur nachträglich konfigurieren muss. Um es möglich zu machen, Wide- oder Multibyte-Zeichen einzugeben und zu verarbeiten, unterstützt die FreeBSD-Ports-Sammlung verschiedene Sprachen für diverse Programme. Bitte konsultieren Sie die I18N-Dokumentation des entsprechenden FreeBSD-Ports.

In den meisten Fällen muss der Benutzer in die Dokumentation des Programms schauen, um herauszufinden, wie man es entsprechend für die eigene Sprache und den eigenen Zeichensatz konfiguriert, oder welche Optionen beim Übersetzen anzugeben sind.

Einige Dinge, die man im Hinterkopf behalten sollte, sind:

- Sprachbezogene C-char Zeichensätze[1] (siehe multibyte(3)), zum Beispiel ISO8859-1, ISO8859-15, KOI8-R, CP437.
- Wide- oder Multibyte-Codierungen, zum Beispiel EUC, Big5.

Eine aktuelle Liste der Zeichensätze ist in der IANA Registry. verfügbar.

[1]Mit C-char Zeichensätzen werden Zeichensätze bezeichnet, die zur Codierung den C-Datentyp `char` verwenden.

Anmerkung

Ab FreeBSD 4.5 werden X11-kompatible Codierungen verwendet.

23.3.3. I18N-Anwendungen

Im FreeBSD-Ports- und Paket-System werden I18N-Anwendungen mit einem `I18N` im Namen gekennzeichnet, damit man sie leicht identifizieren kann. Trotzdem kann es vorkommen, dass die benötigte Sprache nicht immer unterstützt wird.

23.3.4. Einstellen der Locale

Zum Aktivieren der Lokalisierung reicht es, die Umgebungsvariable `LANG` in Ihrer Login-Shell auf den Wert der Locale zu setzen und die Variable zu exportieren. Dies geschieht normalerweise in Ihrer `~/.login_conf` oder der Startdatei Ihrer Shell (`~/.profile`, `~/.bashrc` , `~/.cshrc`). Wenn `LANG` gesetzt ist, brauchen die speziellen Variablen wie `LC_CTYPE` oder `LC_CTIME` in der Regel nicht gesetzt zu werden. Sie sollten sprachbezogene FreeBSD-Dokumentation zu Rate ziehen, wenn Sie mehr Informationen wünschen.

Setzen Sie die zwei folgenden Umgebungsvariablen in Ihren Konfigurationsdateien:

- `LANG` für Funktionen der POSIX® setlocale(3) Familie
- `MM_CHARSET` gibt den den MIME Zeichensatz von Anwendungen an

Damit ist die Locale für die Shell, jede Anwendung und X11 eingestellt.

23.3.4.1. Verfahren zum Einstellen der Locale

Es gibt zwei Wege, die Locale zu setzen, die im Folgenden beschrieben werden. Die erste und empfohlene Methode ist, die Umgebungsvariablen in der Login-Klasse zu setzen, die zweite ist, sie in den Startdateien der Shell zu setzen.

23.3.4.1.1. Lokalisierung in der Login-Klasse

Wenn Sie diese Methode verwenden, werden die Umgebungsvariablen für die Locale und den MIME Zeichensatz einmal für alle Shells, anstatt einzeln für jede Shell, gesetzt. Die Lokalisierung kann von einem Benutzer selbst oder von einem Administrator mit Superuser-Rechten für alle eingestellt werden.

23.3.4.1.1.1. Einrichten als Benutzer

`.login_conf` im Heimatverzeichnis eines Benutzers sollte mindestens die folgenden Einträge enthalten, damit beide Variablen für den Gebrauch der Latin-1 Codierung gesetzt werden:

```
me:\
 :charset=ISO-8859-1:\
 :lang=de_DE.ISO8859-1:
```

Damit traditionelles Chinesisch (BIG-5 Codierung) verwendet werden kann, sind in `.login_conf` die nachstehenden Ergänzungen vorzunehmen. Einige Programme behandeln die Lokalisierung für Chinesisch, Japanisch und Koreanisch falsch, daher müssen mehr Variablen als üblich gesetzt werden:

```
#Users who do not wish to use monetary units or time formats
#of Taiwan can manually change each variable
```

```
me:\
 :lang=zh_TW.Big5:\
 :setenv=LC_ALL=zh_TW.Big5:\
 :setenv=LC_COLLATE=zh_TW.Big5:\
 :setenv=LC_CTYPE=zh_TW.Big5:\
 :setenv=LC_MESSAGES=zh_TW.Big5:\
 :setenv=LC_MONETARY=zh_TW.Big5:\
 :setenv=LC_NUMERIC=zh_TW.Big5:\
 :setenv=LC_TIME=zh_TW.Big5:\
 :charset=big5:\
 :xmodifiers="@im=gcin": #Set gcin as the XIM Input Server
```

Weitere Informationen entnehmen Sie bitte login.conf(5).

23.3.4.1.1.2. Einrichten als Administrator

Stellen Sie sicher, dass in der Login-Klasse der Benutzer in `/etc/login.conf` die richtige Sprache eingestellt ist. Die folgenden Einstellungen müssen in `/etc/login.conf` vorgenommen werden:

```
Sprache|Account-Typ-Beschreibung :\
 :charset=MIME_Zeichensatz :\
 :lang=Locale:\
 :tc=default:
```

Die für Latin-1 erforderlichen Einträge sehen wie folgt aus:

```
german|German Users Accounts:\
 :charset=ISO-8859-1:\
 :lang=de_DE.ISO8859-1:\
 :tc=default:
```

Bevor Sie die Login-Klasse eines Benutzers ändern, müssen Sie den folgenden Befehl ausführen:

```
# cap_mkdb /etc/login.conf
```

Erst danach werden Ihre Änderungen in `/etc/login.conf` im System sichtbar.

Ändern der Login-Klasse mit vipw(8)

Wenn Sie neue Accounts mit `vipw` anlegen, erstellen Sie Einträge in folgender Art:

```
user:password:1111:11:Sprache:0:0:Benutzername:/home/user:/bin/sh
```

Ändern der Login-Klasse mit adduser(8)

Wenn Sie neue Accounts mit `adduser` anlegen, stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

- Geben Sie in `/etc/adduser.conf` mit `defaultclass = Sprache` eine Sprache vor. In diesem Fall müssen Sie für Benutzer anderer Sprachen eine andere Login-Klasse angeben.
- Geben Sie die Sprache jedes Mal ein, wenn Sie dazu von adduser(8) aufgefordert werden:

  ```
  Enter login class: default []:
  ```

- Sie können die Login-Klasse auch auf der Kommandozeile von adduser(8) übergeben:

  ```
  # adduser -class Sprache
  ```

Ändern der Login-Klasse mit pw(8)

Wenn Sie neue Accounts mit pw(8) anlegen, benutzen Sie die folgende Kommandozeile:

```
# pw useradd Account -L Sprache
```

23.3.4.1.2. Lokalisierung in den Startdateien der Shells

> **Anmerkung**
>
> Da Sie jede Shell unterschiedlich einrichten müssen, sollten Sie diese Methode nicht verwenden. Benutzen Sie stattdessen bitte Login-Klassen.

Um die Locale und den MIME Zeichensatz anzugeben, setzen Sie die unten aufgeführten Variablen in den Startdateien der Shells (`/etc/profile` und `/etc/csh.login`). In den folgenden Beispielen verwenden wir die deutsche Sprache.

Einstellungen in `/etc/profile`:

```
LANG=de_DE.ISO8859-1; export LANG
MM_CHARSET=ISO-8859-1; export MM_CHARSET
```

Einstellungen in `/etc/csh.login`:

```
setenv LANG de_DE.ISO8859-1
setenv MM_CHARSET ISO-8859-1
```

Alternativ können Sie die Einstellungen in den Vorgabedateien der Shells vornehmen. Die oben gezeigten Einstellungen aus `/etc/profile` tragen Sie dann in `/usr/share/skel/dot.profile` und die Einstellungen aus `/etc/csh.login` in `/usr/share/skel/dot.login` ein.

Die Einstellungen für X11 in `$HOME/.xinitrc` sind von der verwendeten Login-Shell abhängig. Mit Bourne Shells verwenden Sie den folgenden Eintrag:

```
LANG=de_DE.ISO8859-1; export LANG
```

Mit C-Shells verwenden Sie den nachstehenden Eintrag:

```
setenv LANG de_DE.ISO8859-1
```

23.3.5. Einrichten der Konsole

Wenn Sie C-char Zeichensätze[1] verwenden, müssen Sie die richtigen Zeichensätze für die gewählte Sprache in `/etc/rc.conf` angeben:

```
font8x16=Zeichensatz
font8x14=Zeichensatz
font8x8=Zeichensatz
```

Dabei ist *`Zeichensatz`* der Name der passenden Datei aus `/usr/share/syscons/fonts` ohne die Endung `.fnt`.

Setzen Sie bei Bedarf die richtige Tasten- und Bildschirmzuordnung (keymap und screenmap). Dies können Sie in `sysinstall` einstellen, indem Sie Configure und dann Console wählen. Sie können die Zuordnungen aber auch direkt in `/etc/rc.conf` angeben:

```
scrnmap=screenmap_name
keymap=keymap_name
keychange="fkey_number sequence "
```

`screenmap_name` ist der Name einer Datei aus `/usr/share/syscons/scrnmaps` ohne die Endung `.scm`. Eine Bildschirmzuordnung und der zugehörige Zeichensatz verbreitert die Zeichenmatrix von VGA Karten im Pseudographik Modus von 8 Bit auf 9 Bit. Sie wird benötigt, wenn der Zeichensatz des Bildschirms 8 Bit verwendet.

Lesen Sie den nächsten Absatz, wenn Sie in `/etc/rc.conf` den moused Dæmon mit der nachstehenden Anweisung aktiviert haben:

```
moused_enable="YES"
```

Der Mauszeiger des syscons(4) Treibers belegt in der Voreinstellung den Bereich von 0xd0 bis 0xd3 des Zeichensatzes. Wenn dieser Bereich ebenfalls von der eingestellten Sprache benötigt wird, müssen Sie den Mauszeiger verschieben. Dazu fügen Sie die folgende Zeile in Ihre Kernelkonfigurationsdatei ein:

```
mousechar_start=3
```

keymap_name ist der Name einer Datei aus `/usr/share/syscons/keymaps` ohne die Endung `.kbd`. Welche Tastenzuordnung Sie benutzen müssen, können Sie ohne einen Neustart mit kbdmap(1) ausprobieren.

Mit `keychange` können die Funktionstasten so programmiert werden, dass Sie zu dem ausgesuchten Terminal passen. Die Sequenzen der Funktionstasten können nicht in Tastenzuordnungen definiert werden.

Stellen Sie sicher, dass der richtige Terminaltyp für die `ttyv*` Konsolen in `/etc/ttys` angegeben ist. Momentan sind die folgenden Terminaltypen definiert:

Zeichensatz	Terminaltyp
ISO8859-1 oder ISO8859-15	`cons25l1`
ISO8859-2	`cons25l2`
ISO8859-7	`cons25l7`
KOI8-R	`cons25r`
KOI8-U	`cons25u`
CP437 (VGA default)	`cons25`
US-ASCII	`cons25w`

Mit Wide- oder Multibyte-Zeichensätzen müssen Sie den richtigen Port aus dem Verzeichnis `/usr/ports/Sprache` verwenden. Einige Ports erscheinen als Konsolen werden aber vom System als serielle vtty's betrachtet. Achten Sie daher darauf, dass Sie genügend vtty's für X11 und die Pseudo-seriellen Konsolen definiert haben. Nachstehend finden Sie eine unvollständige Liste der Ports, die eine andere Sprache als Englisch auf der Konsole verwenden:

Sprache	Port
traditionelles Chinesisch (BIG-5)	chinese/big5con
Japanisch	japanese/kon2-16dot oder japanese/mule-freewnn
Koreanisch	korean/han

23.3.6. Einrichten von X11

Obwohl X11 nicht Teil des FreeBSD Projects ist, stellen wir hier einige Hinweise für FreeBSD-Benutzer zusammen. Weitere Details entnehmen Sie bitte der Xorg Website oder der Dokumentation Ihres X11 Servers.

Anwendungsspezifische I18N-Einstellungen (Zeichensätze, Menüs, usw.) können Sie in `~/.Xresources` vornehmen.

23.3.6.1. Zeichensätze

Installieren Sie den Xorg-Server (x11-servers/xorg-server) und die TrueType® Zeichensätze Ihrer Sprache. Wenn Sie die Locale gesetzt haben, sollten die Menüs in Ihrer Sprache erscheinen.

23.3.6.2. Eingabe von nicht-englischen Zeichen

Das X11 Input Method (XIM) Protokoll ist ein neuer Standard für alle X11-Clients. Jede X11-Anwendung sollte als XIM-Client, der Eingaben von einem XIM-Server entgegen nimmt, implementiert sein. XIM-Server sind für verschiedene Sprachen erhältlich.

23.3.7. Einrichten eines Druckers

Drucker verfügen normalerweise schon über einige C-char Zeichensätze[1]. Wide- oder Multibyte-Zeichensätze müssen gesondert eingerichtet werden. Wir empfehlen Ihnen, dazu apsfilter zu benutzen. Weiterhin können Sie mit sprachspezifischen Konvertern Ihre Dokumente auch in PostScript® oder PDF umwandeln.

23.3.8. Kernel und Dateisysteme

Das FreeBSD-Dateisystem (FFS) unterstützt 8-Bit, so dass es mit C-char Zeichensätzen[1] (siehe multibyte(3)) verwendet werden kann. Der Zeichensatz wird allerdings nicht im Dateisystem gespeichert, das heißt es werden nur die 8-Bit Werte gespeichert und die Codierung wird nicht berücksichtigt. Offiziell werden Wide- oder Multibyte-Zeichensätze noch nicht unterstützt, für einige Zeichensätze existieren Patche, die eine solche Unterstützung aktivieren. Sie sind allerdings nicht im Quelltext enthalten, da sie nur schwer pflegbare Übergangslösungen sind. Die Patche und weitere Informationen erhalten Sie auf den Webseiten der betreffenden Sprache.

Das MS-DOS® Dateisystem von FreeBSD kann von MS-DOS®- und Unicode-Zeichensätzen nach frei wählbaren FreeBSD Zeichensätzen konvertieren. Weitere Details entnehmen Sie bitte mount_msdosfs(8).

23.4. I18N-Programme übersetzen

Viele FreeBSD-Ports besitzen I18N-Unterstützung, einige davon enthalten `-I18N` im Namen. Für diese und viele andere Programme ist keine spezielle Konfiguration notwendig.

Einige Anwendungen wie MySQL müssen allerdings speziell für einen Zeichensatz in ihrem `Makefile` konfiguriert werden. Normalerweise wird dazu das `Makefile` angepasst oder configure mit einem speziellen Parameter aufgerufen.

23.5. Lokalisierung für einzelne Sprachen

23.5.1. Russisch (KOI8-R Codierung)

Beigetragen von Andrey Chernov.

Weitere Informationen über die KOI8-R Codierung erhalten Sie auf der Webseite KOI8-R References (Russian Net Character Set).

23.5.1.1. Einrichten der Locale

Fügen Sie die folgenden Zeilen in `~/.login_conf` ein:

```
me:My Account:\
 :charset=KOI8-R:\
 :lang=ru_RU.KOI8-R:
```

Weitere Erklärungen finden Sie in Einstellen der Locale.

23.5.1.2. Einrichten der Konsole

- Fügen Sie folgende Zeile in `/etc/rc.conf` ein:

  ```
  mousechar_start=3
  ```

- Nehmen Sie zusätzlich die folgenden Einstellungen in `/etc/rc.conf` auf:

  ```
  keymap="ru.koi8-r"
  scrnmap="koi8-r2cp866"
  font8x16="cp866b-8x16"
  font8x14="cp866-8x14"
  ```

```
font8x8="cp866-8x8"
```

- Benutzen Sie `cons25r` als Terminaltyp für jeden `ttyv*` Eintrag in `/etc/ttys` .

Weitere Beispiele finden Sie in Einrichten der Konsole.

23.5.1.3. Einrichten eines Druckers

Die meisten Drucker mit russischen Zeichen besitzen die Codetabelle CP866, so dass ein spezielles Programm zur Übersetzung von KOI8-R nach CP866 benötigt wird. Zu diesem Zweck ist `/usr/libexec/lpr/ru/koi2alt` im Basissystem enthalten. Der Eintrag für einen Drucker mit russischer Sprachunterstützung in `/etc/printcap` sieht wie folgt aus:

```
lp|Russian local line printer:\
 :sh:of=/usr/libexec/lpr/ru/koi2alt:\
 :lp=/dev/lpt0:sd=/var/spool/output/lpd:lf=/var/log/lpd-errs:
```

Näheres erfahren Sie in printcap(5).

23.5.1.4. MS-DOS® Dateisystem und russische Dateinamen

Russische Dateinamen auf MS-DOS® Dateisystemen werden mit dem folgenden Eintrag in `/etc/fstab` erkannt:

```
/dev/ad0s2      /dos/c  msdos   rw,-Wkoi2dos,-Lru_RU.KOI8-R 0 0
```

Die Option `-L` legt die Locale fest. Die Option `-W` legt die Zeichenumwandlung fest. Stellen Sie sicher, dass `/usr` eingehangen ist, bevor Sie die MS-DOS®-Partition einhängen, da die Tabellen zur Zeichenumwandlung in `/usr/libdata/msdosfs` liegen. Weitere Informationen erhalten Sie in der Hilfeseite mount_msdosfs(8).

23.5.1.5. Einrichten von X11

1. Richten Sie zunächst die normale Lokalisierung ein.

2. Wenn Sie Xorg verwenden, installieren Sie den Port x11-fonts/xorg-fonts-cyrillic.

   Im Abschnitt `"Files"` von `/etc/X11/xorg.conf` fügen Sie den folgende Eintrag *vor* allen anderen `FontPath` Einträgen ein:

   ```
   FontPath   "/usr/local/lib/X11/fonts/cyrillic"
   ```

   Anmerkung

   Zusätzliche kyrillische Schriftarten finden Sie in der Ports-Sammlung.

3. Die Unterstützung für eine russische Tastatur aktivieren Sie im `"Keyboard"` Abschnitt von `xorg.conf`:

   ```
   Option "XkbLayout"   "us,ru"
   Option "XkbOptions"  "grp:toggle"
   ```

   Stellen Sie zudem sicher, dass `XkbDisable` deaktiviert (auskommentiert) ist.

   Beim Einsatz von `grp:toggle` können Sie mit Right Alt (Alt Gr) zwischen dem RUS- und LAT-Modus wechseln, verwenden Sie hingegen `grp:ctrl_shift_toggle`, so erfolgt der Wechsel mit Ctrl+Shift. Für `grp:caps_toggle` ist zum Wechseln des RUS/LAT-Modus CapsLock zuständig. Die alte Funktion von CapsLock steht nur im LAT-Modus mit der Tastenkombination Shift+CapsLock zur Verfügung. `grp:caps_toggle` funktioniert aus unbekannten Gründen unter Xorg nicht.

Wenn Ihre Tastatur Windows®-Tasten besitzt und nicht-alphanumerische Tasten im RUS-Modus nicht funktionieren, fügen Sie die folgende Zeile in `xorg.conf` ein:

```
Option "XkbVariant" ",winkeys"
```

Anmerkung

Die russische XKB-Tastatur funktioniert vielleicht nicht mit nicht-lokalisierten Anwendungen.

Anmerkung

Lokalisierte Anwendungen sollten mindestens die Funktion `XtSetLanguageProc (NULL, NULL, NULL);` frühzeitig aufrufen.

Weitere Informationen über die Lokalisierung von X11-Anwendungen erhalten Sie auf der Webseite KOI8-R for X Window.

23.5.2. Traditionell chinesische Lokalisierung für Taiwan

Das taiwanesische FreeBSD Project stellt ein Tutorium unter `http://netlab.cse.yzu.edu.tw/~statue/freebsd/zh-tut/` zur Verfügung, das viele chinesische Anwendungen benutzt. Der Editor des `FreeBSD Chinese HOWTOs` ist Shen Chuan-Hsing <statue@freebsd.sinica.edu.tw>.

Chuan-Hsing Shen <statue@freebsd.sinica.edu.tw> hat mithilfe des Tutoriums die Chinese FreeBSD Collection (CFC) geschaffen. Die Pakete und Skripten stehen unter `ftp://freebsd.csie.nctu.edu.tw/pub/taiwan/CFC/` .

23.5.3. Deutsche Lokalisierung (für alle ISO 8859-1 Sprachen)

Von Slaven Rezic <eserte@cs.tu-berlin.de> stammt ein Tutorium, das die Benutzung von Umlauten mit FreeBSD beschreibt. Das Tutorium ist in Deutsch verfasst und unter `http://user.cs.tu-berlin.de/~eserte/FreeBSD/doc/umlaute/umlaute.html` verfügbar.

23.5.4. Griechische Lokalisierung

Nikos Kokkalis <nickkokkalis@gmail.com> hat einen ganzen Artikel über die Griechisch-Unterstützung in FreeBSD geschrieben. Er ist als Teil der offiziellen FreeBSD Dokumentation auf Griechisch erhältlich unter http://www.freebsd.org/doc/el_GR.ISO8859-7/articles/greek-language-support/index.html. Bitte beachten Sie, dass dies *nur* für Griechisch gilt.

23.5.5. Japanische und koreanische Lokalisierung

Informationen über die japanische Lokalisierung entnehmen Sie bitte `http://www.jp.FreeBSD.org/` , Informationen über die koreanische Lokalisierung erhalten Sie unter `http://www.kr.FreeBSD.org/` .

23.5.6. Nicht-englische FreeBSD-Dokumentation

Teile vor FreeBSD Dokumentation wurden in andere Sprachen übersetzt. Folgen Sie bitte den Links auf der FreeBSD-Webseite oder schauen Sie in `/usr/share/doc` nach.

Kapitel 24. FreeBSD aktualisieren

Umstrukturiert und aktualisiert von Jim Mock.
Im Original von Jordan Hubbard, Poul-Henning Kamp, John Polstra und Nik Clayton.
Übersetzt von Martin Heinen.

24.1. Übersicht

FreeBSD wird zwischen einzelnen Releases ständig weiter entwickelt. Manche Leute bevorzugen die offiziellen Release-Versionen, während andere wiederum lieber auf dem aktuellen Stand der Entwicklung bleiben möchten. Wie dem auch sei, sogar offizielle Release-Versionen werden oft mit Sicherheitsaktualisierungen und anderen kritischen Fehlerbereinigungen versorgt. Unabhängig von der eingesetzten Version bringt FreeBSD alle nötigen Werkzeuge mit, um das System aktuell zu halten und es innerhalb verschiedener Versionen zu aktualisieren. Dieses Kapitel beschreibt, wie man einem Entwicklungssystem folgen kann, sowie die grundlegenden Werkzeuge um FreeBSD zu aktualisieren.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie

- wissen, welche Werkzeuge verwendet werden können, um das System und die Port-Sammlung zu aktualisieren.
- wissen, wie das System mit freebsd-update, Subversion oder CTM aktualisiert wird.
- wissen, wie man das aktuell installierte System mit einer ursprünglichen Version vergleicht.
- wissen, wie die installierte Dokumentation mit Subversion oder Dokumentations-Ports aktualisiert wird.
- den Unterschied zwischen den beiden Entwicklungszweigen FreeBSD-STABLE und FreeBSD-CURRENT kennen.
- wissen, wie das komplette Basissystem neu gebaut und installiert wird.

Bevor Sie dieses Kapitel lesen, sollten Sie

- das Netzwerk richtig konfiguriert haben (Kapitel 31, *Weiterführende Netzwerkthemen*).
- wissen, wie Software Dritter installiert wird (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

Anmerkung

In diesem Kapitel wird `svn` verwendet, um die FreeBSD Quellen zu beziehen und zu aktualisieren. Um es zu verwenden, muss zuerst der Port oder das Paket devel/subversion installiert werden.

24.2. FreeBSD-Update

Geschrieben von Tom Rhodes.
Basierend auf bereitgestellten Mitschriften von Colin Percival.
Übersetzt von Benedict Reuschling.

Das Einspielen von Sicherheitsaktualisierungen ist ein wichtiger Bestandteil bei der Wartung von Computersoftware, besonders wenn es um das Betriebssystem geht. Für lange Zeit war dieser Prozess unter FreeBSD nicht einfach. Fehlerbehebungen mussten auf den Quellcode angewendet werden, danach wurde der Code zu neuen Binärdateien übersetzt und schliesslich mussten diese Dateien neu installiert werden.

Das ist seit längerem nicht mehr der Fall, da FreeBSD jetzt ein Werkzeug namens `freebsd-update` enthält. Dieses Werkzeug bringt zwei getrennte Funktionen mit sich. Die erste Funktion ermöglicht die Anwendung von Sicher-

heitsaktualisierungen im Binärformat auf das FreeBSD Basissystem, ohne dieses neu zu übersetzen und zu installieren. Die zweite Funktion unterstützt Aktualisierungen zwischen Haupt- und Unterversionen.

Anmerkung

Binäre Aktualisierungen sind für alle Architekturen und Releases verfügbar, die aktuell vom FreeBSD Security Team betreut werden. Vor der Aktualisierung auf eine neue Release-Version sollten die aktuellen Ankündigungen zu dem Release gelesen werden, da diese wichtige Informationen zu der gewünschten Version enthalten. Release Ankündigungen finden Sie unter `http://www.FreeBSD.org/releases/` .

Wenn eine `crontab` existiert, welche die Eigenschaften von freebsd-update(8) verwendet, muss diese deaktiviert werden, bevor die folgende Aktion gestartet wird.

24.2.1. Die Konfigurationsdatei

Manche Anwender möchten sicherlich Einstellungen in der Standard-Konfigurationsdatei unter `/etc/freebsd-update.conf` vornehmen, um bessere Kontrolle über den gesamten Prozess zu besitzen. Die Optionen sind gut dokumentiert, jedoch benötigen die folgenden ein paar zusätzliche Erklärungen:

```
# Components of the base system which should be kept updated.
Components src world kernel
```

Dieser Parameter kontrolliert, welche Teile von FreeBSD auf dem aktuellen Stand gehalten werden sollen. In der Voreinstellung wird der Quellcode, das gesamte Basissystem sowie der Kernel aktualisiert. Die Komponenten sind die gleichen wie während der Installation. Das hinzufügen von `world/games` erlaubt es, Aktualisierungen für Spiele anzuwenden. Die Verwendung von `src/bin` erlaubt es, den Quellcode in `src/bin` aktuell zu halten.

Die beste Einstellung ist, diese Option so zu belassen, da eine Änderung es bedingt, dass man als Benutzer jede Komponente auflisten muss, die aktualisiert werden soll. Dies könnte katastrophale Folgen nach sich ziehen, da der Quellcode und die Binärdateien dadurch nicht mehr synchron wären.

```
# Paths which start with anything matching an entry in an IgnorePaths
# statement will be ignored.
IgnorePaths
```

Fügen Sie Pfade wie `/bin` oder `/sbin` hinzu, um diese speziellen Verzeichnisse während des Aktualisierungsprozesses unberührt zu lassen. Diese Option kann verwendet werden, um zu verhindern, dass `freebsd-update` lokale Änderungen überschreibt.

```
# Paths which start with anything matching an entry in an UpdateIfUnmodified
# statement will only be updated if the contents of the file have not been
# modified by the user (unless changes are merged; see below).
UpdateIfUnmodified /etc/ /var/ /root/ /.cshrc /.profile
```

Aktualisiert nur unmodifizierte Konfigurationsdateien in den angegebenen Verzeichnissen. Jede Änderung, die der Benutzer daran vorgenommen hat, wird die automatische Aktualisierung dieser Dateien ungültig machen. Es gibt eine weitere Option `KeepModifiedMetadata`, die `freebsd-update` instruiert, die Änderungen während der Zusammenführung zu speichern.

```
# When upgrading to a new FreeBSD release, files which match MergeChanges
# will have any local changes merged into the version from the new release.
MergeChanges /etc/ /var/named/etc/
```

Eine Liste von Verzeichnissen mit Konfigurationsdateien, in denen `freebsd-update` Zusammenführungen versuchen soll. Dieser Verschmelzungsprozess von Dateien ist eine Serie von diff(1)-Korrekturen, ähnlich wie mergemaster(8), aber mit weniger Optionen. Die Änderungen werden entweder akzeptiert, oder öffnen einen Editor,

oder `freebsd-update` bricht ab. Im Zweifelsfall sichern Sie `/etc` und akzeptieren einfach die Änderungen. Lesen Sie Abschnitt 24.7.12.1, „`mergemaster`“, um Informationen über `mergemaster` zu erhalten.

```
# Directory in which to store downloaded updates and temporary
# files used by FreeBSD Update.
# WorkDir /var/db/freebsd-update
```

In diesem Verzeichnis werden alle Korrekturen und temporären Dateien abgelegt. Im Falle einer Versionsaktualisierung sollte diesem Verzeichnis mindestens ein Gigabyte Festplattenspeicher zur Verfügung stehen.

```
# When upgrading between releases, should the list of Components be
# read strictly (StrictComponents yes) or merely as a list of components
# which *might* be installed of which FreeBSD Update should figure out
# which actually are installed and upgrade those (StrictComponents no)?
# StrictComponents no
```

Wenn diese Option auf `yes` gesetzt ist, wird `freebsd-update` annehmen, dass die `Components`-Liste vollständig ist und nicht versuchen, Änderungen ausserhalb dieser Liste zu tätigen. Tatsächlich wird `freebsd-update` versuchen, jede Datei zu aktualisieren, die zu der `Components`-Liste gehört.

24.2.2. Sicherheitsaktualisierungen

Sicherheitsaktualisierungen für FreeBSD können wie folgt heruntergeladen und installiert werden:

```
# freebsd-update fetch
# freebsd-update install
```

Wenn während Aktualisierung Korrekturen am Kernel angewendet werden, muss das System neu gestartet werden, damit der korrigierte Kernel gebootet wird. Andernfalls sollte das System aktualisiert sein und `freebsd-update` kann als nächtlicher cron(8)-Job laufen, indem folgender Eintrag in `/etc/crontab` hinzugefügt wird:

```
@daily                                  root    freebsd-update cron
```

Dieser Eintrag besagt, dass `freebsd-update` einmal am Tag ausgeführt wird. Wenn es über `cron` ausgeführt wird, prüft `freebsd-update` lediglich, ob Aktualisierungen vorliegen. Wenn Korrekturen existieren, werden diese automatisch auf die lokale Festplatte heruntergeladen, aber nicht eingespielt. Der `root`-Benutzer bekommt eine Nachricht, damit die Korrekturen überprüft und manuell installiert werden können.

Wenn etwas schief geht, kann `freebsd-update` den letzten Satz von Änderungen mit folgendem Befehl rückgängig machen:

```
# freebsd-update rollback
```

Sobald dieser Vorgang abgeschlossen ist, sollte das System neu gestartet werden, wenn der Kernel oder ein beliebiges Kernelmodul geändert wurde. Dies ermöglicht es FreeBSD, die neuen Binärdateien in den Hauptspeicher zu laden.

Das `freebsd-update` -Werkzeug kann nur den `GENERIC` -Kernel automatisch aktualisieren. Wenn ein angepasster Kernel verwendet wird, muss dieser neu erstellt und installiert werden, nachdem `freebsd-update` den Rest der Aktualisierungen durchgeführt hat. Allerdings wird `freebsd-update` den `GENERIC` -Kernel in `/boot/GENERIC` erkennen und aktualisieren, selbst wenn dies nicht der aktuell verwendete Kernel des Systems ist.

Anmerkung

Es ist eine gute Idee, immer eine Kopie des `GENERIC` -Kernels in `/boot/GENERIC` aufzubewahren. Das wird bei der Diagnose von verschiedenen Problemen eine grosse Hilfe sein, sowie bei der Durchführung von Versionsaktualisierungen mit `freebsd-update` , wie in Abschnitt 24.2.3, „Aktualisierungen an Haupt- und Unterversionen“ beschrieben ist.

Solange die Standardkonfiguration in `/etc/freebsd-update.conf` nicht geändert wurde, wird `freebsd-update` die aktualisierten Quellcodedateien des Kernels zusammen mit dem Rest der Neuerungen installieren. Die erneute Übersetzung und Installation eines neuen, angepassten Kernels kann dann auf die übliche Art und Weise durchgeführt werden.

Anmerkung

Die Aktualisierungen, die über `freebsd-update` verteilt werden, betreffen nicht immer den Kernel. Es ist nicht notwendig, den angepassten Kernel neu zu erstellen, wenn die Kernelquellen nicht durch die Ausführung von `freebsd-update install` geändert wurden. Allerdings wird `freebsd-update` immer `/usr/src/sys/conf/newvers.sh` aktualisieren. Der aktuelle Patch-Level, der mit der `-p`-Nummer bei `uname -r` ausgegeben wird, wird aus dieser Datei ausgelesen. Die Neuinstallation des angepassten Kernels, selbst wenn sich daran nichts geändert hat, erlaubt es uname(1), den aktuellen Patch-Level des Systems korrekt wiederzugeben. Dies ist besonders hilfreich, wenn mehrere Systeme gewartet werden, da es eine schnelle Einschätzung der installierten Aktualisierungen in jedem einzelnen System ermöglicht.

24.2.3. Aktualisierungen an Haupt- und Unterversionen

Aktualisierungen einer Unterversion zur nächsten in FreeBSD ist beispielsweise die Aktualisierung von FreeBSD 9.0 auf FreeBSD 9.1. In der Regel funktionieren die installierten Anwendungen weiterhin problemlos nach der Aktualisierung einer Unterversion.

Eine Aktualisierung der Hauptversion ist besipielsweise die Aktualisierung von FreeBSD 8.X auf FreeBSD 9.X. Dieser Prozess entfernt alte Objekt-Dateien und Bibliotheken, was dazu führt, dass die meisten Anwendungen von Drittherstellern nicht mehr funktionieren. Nach der Aktualisierung auf eine neue Hauptversion wird empfohlen, dass alle installierten Ports entweder entfernt und neu installiert werden, oder mit einem Werkzeug wie ports-mgmt/portmaster aktualisiert werden. Um die Neuerstellung aller installierten Anwendungen zu erwzingen, benutzen Sie folgenden Befehl:

```
# portmaster -f
```

Dies sorgt dafür, dass alles korrekt neu installiert wird. Beachten Sie, dass das Setzen der `BATCH`-Umgebungsvariable auf `yes` während dieses Prozesses auf jede Eingabe mit `ja` antwortet, was es nicht mehr notwendig macht, manuell eingreifen zu müssen.

24.2.3.1. Umgang mit angepassten Kerneln

Wenn ein angepasster Kernel verwendet wird, ist der Aktualisierungsprozess ein wenig aufwändiger und das Vorgehen variiert je nach Version von FreeBSD.

24.2.3.1.1. Angepasste Kernel unter FreeBSD 8.X und früher

Eine Kopie des `GENERIC` -Kernel wird benötigt und sollte in `/boot/GENERIC` abgelegt sein. Wenn der `GENERIC` -Kernel nicht im System vorhanden ist, kann er über eine der folgenden Methoden bezogen werden:

- Wenn ein angepasster Kernel erstmalig gebaut wurde, ist der Kernel in `/boot/kernel.old` in Wirklichkeit der `GENERIC` -Kernel. Benennen Sie dieses Verzeichnis in `/boot/GENERIC` um.

- Angenommen, ein direkter Zugriff auf die Maschine ist möglich, so kann eine Kopie des `GENERIC` -Kernels von den Installationsmedien installiert werden. Benutzen Sie dazu folgende Befehle:

```
# mount /cdrom
# cd /cdrom/X.Y-RELEASE/kernels
# ./install.sh GENERIC
```

Ersetzen Sie `X.Y-RELEASE` durch die aktuelle Version des verwendeten Releases. Der `GENERIC` -Kernel wird standardmäßig in `/boot/GENERIC` installiert.

- Falls alle obigen Schritte fehlschlagen, kann der `GENERIC` -Kernel folgendermaßen aus den Quellen neu gebaut und installiert werden:

```
# cd /usr/src
# env DESTDIR=/boot/GENERIC make kernel __MAKE_CONF=/dev/null SRCCONF=/dev/null
# mv /boot/GENERIC/boot/kernel/* /boot/GENERIC
# rm -rf /boot/GENERIC/boot
```

Damit dieser Kernel als `GENERIC` -Kernel von `freebsd-update` erkannt wird, darf die `GENERIC` -Konfigurationsdatei in keiner Weise geändert worden sein. Es wird ebenfalls empfohlen, dass dieser ohne irgendwelche speziellen Optionen erstellt wird.

Der Neustart in den `GENERIC` -Kernel ist zu diesem Zeitpunkt nicht notwendig.

24.2.3.1.2. Angepasste Kernel unter FreeBSD 9.X und später

- Wenn ein angepasster Kernel erstmalig gebaut wurde, ist der Kernel in `/boot/kernel.old` in Wirklichkeit der `GENERIC` -Kernel. Benennen Sie einfach dieses Verzeichnis in `/boot/GENERIC` um.

- Angenommen, ein direkter Zugriff auf die Maschine ist möglich, so kann eine Kopie des `GENERIC` -Kernels von den Installationsmedien installiert werden. Benutzen Sie dazu folgende Befehle:

```
# mount /cdrom
# cd /cdrom/usr/freebsd-dist
# tar -C/ -xvf kernel.txz boot/kernel/kernel
```

- Wenn die oben genannten Optionen nicht verwendet werden können, kann der `GENERIC` -Kernel aus den Quellen neu gebaut und installiert werden:

```
# cd /usr/src
# make kernel __MAKE_CONF=/dev/null SRCCONF=/dev/null
```

Damit dieser Kernel als `GENERIC` -Kernel von `freebsd-update` erkannt wird, darf die `GENERIC` -Konfigurationsdatei in keiner Weise geändert worden sein. Es wird ebenfalls empfohlen, dass dieser ohne irgendwelche speziellen Optionen erstellt wird.

Der Neustart in den `GENERIC` -Kernel ist zu diesem Zeitpunkt nicht notwendig.

24.2.3.2. Die Aktualisierung durchführen

Aktualisierungen an Haupt- und Unterversionen können durchgeführt werden, wenn man `freebsd-update` eine Release-Version als Ziel übergibt. Beispielsweise wird das folgende Kommando das System auf FreeBSD 9.1 aktualisieren:

```
# freebsd-update -r 9.1-RELEASE upgrade
```

Nachdem das Kommando empfangen wurde, überprüft `freebsd-update` die Konfigurationsdatei und das aktuelle System, um die nötigen Informationen für die Systemaktualisierung zu sammeln. Eine Bildschirmausgabe wird anzeigen, welche Komponenten erkannt und welche nicht erkannt wurden. Zum Beispiel:

```
Looking up update.FreeBSD.org mirrors... 1 mirrors found.
Fetching metadata signature for 9.0-RELEASE from update1.FreeBSD.org... done.
Fetching metadata index... done.
Inspecting system... done.

The following components of FreeBSD seem to be installed:
kernel/smp src/base src/bin src/contrib src/crypto src/etc src/games
src/gnu src/include src/krb5 src/lib src/libexec src/release src/rescue
```

```
src/sbin src/secure src/share src/sys src/tools src/ubin src/usbin
world/base world/info world/lib32 world/manpages

The following components of FreeBSD do not seem to be installed:
kernel/generic world/catpages world/dict world/doc world/games
world/proflibs

Does this look reasonable (y/n)? y
```

An diesem Punkt wird `freebsd-update` versuchen, alle notwendigen Dateien für die Aktualisierung herunter zu laden. In manchen Fällen wird der Benutzer mit Fragen konfrontiert, um festzustellen, was installiert werden soll oder auf welche Art und Weise fortgesetzt werden soll.

Wenn ein angepasster Kernel benutzt wird, produziert der vorherige Schritt eine Warnung ähnlich zu der folgenden:

```
WARNING: This system is running a "
MYKERNEL" kernel, which is not a
kernel configuration distributed as part of FreeBSD 9.0-RELEASE.
This kernel will not be updated: you MUST update the kernel manually
before running "/usr/sbin/freebsd-update install"
```

Diese Warnung kann an dieser Stelle problemlos ignoriert werden. Der aktualisierte `GENERIC` -Kernel wird als ein Zwischenschritt im Aktualisierungsprozess verwendet.

Nachdem alle Korrekturen auf das lokale System heruntergeladen wurden, werden diese eingespielt. Dieser Prozess kann eine gewisse Zeit in Anspruch nehmen, abhängig von der Geschwindigkeit und Auslastung der Maschine. Konfigurationsdateien werden ebenfalls zusammengefügt. Dieser Teil der Prozedur verlangt einige Benutzereingaben, da eine Datei möglicherweise von Hand zusammengefasst werden muss oder ein Editor erscheint auf dem Bildschirm zum manuellen bearbeiten. Die Ergebnisse von jeder erfolgreichen Zusammenfassung werden dem Benutzer angezeigt, während der Prozess weiter läuft. Eine fehlgeschlagene oder ignorierte Zusammenfassung wird den Prozess sofort beenden. Benutzer sollten eine Sicherung von `/etc` anlegen und wichtige Dateien später manuell vereinen, beispielsweise `master.passwd` oder `group`.

Anmerkung

Das System ist zu diesem Zeitpunkt noch nicht verändert worden, da alle Korrekturen und Vereinigungen in einem anderen Verzeichnis vorgenommen wurden. Wenn alle Korrekturen erfolgreich eingespielt, alle Konfigurationsdateien zusammengefügt wurden und es den Anschein hat, dass der Prozess problemlos verlaufen wird, müssen die Änderungen vom Anwender noch angewendet und auf die Platte geschrieben werden:

```
# freebsd-update install
```

Der Kernel und die Module werden zuerst aktualisiert. Zu diesem Zeitpunkt muss die Maschine neu gestartet werden. Wenn das System einen angepassten Kernel verwendet, benutzen Sie nextboot(8), um den Kernel für den nächsten Neustart auf `/boot/GENERIC` zu setzen:

```
# nextboot -k GENERIC
```

Warnung

Bevor das System mit dem `GENERIC` -Kernel neu gestartet wird, vergewissern Sie sich, dass für den Neustart alle benötigten Treiber enthalten sind. Falls auf die Maschine aus der Ferne zugegriffen wird, stellen Sie sicher, dass das System ordnungsgemäß an das Netzwerk angeschlossen ist. Achten Sie besonders darauf, dass wenn der angepasste Kernel Funktionalität

beinhaltet, die normalerweise von Kernelmodulen zur Verfügung gestellt werden, dass diese temporär über `/boot/loader.conf` in den `GENERIC`-Kernel übernommen werden. Zudem wird empfohlen, nicht benötigte Dienste, eingehängte Platten und verbundene Netzlaufwerke zu deaktivieren, bis der Aktualisierungsprozess abgeschlossen ist.

Die Maschine sollte nun mit dem aktualisierten Kernel neu gestartet werden:

```
# shutdown -r now
```

Sobald das System wieder online ist, muss `freebsd-update` erneut gestartet werden. Der Zustand des Prozesses wurde zuvor gesichert und deshalb wird `freebsd-update` nicht von vorne beginnen, jedoch alle alten gemeinsam genutzten Bibliotheken und Objektdateien löschen.

```
# freebsd-update install
```

Anmerkung

Abhängig davon, ob irgendwelche Bibliotheksversionen erhöht wurden, kann es sein, dass nur zwei Installationsphasen anstatt drei durchlaufen werden.

24.2.3.3. Neubau der Ports nach einer Aktualisierung auf eine Hauptversion

Nach der Aktualisierung auf eine Hauptversion, muss jegliche Drittanbieter-Software neu erstellt und installiert werden. Dies ist notwendig, da die installierte Software möglicherweise Abhängigkeiten zu Bibliotheken enthält, die während der Aktualisierung entfernt wurden. Dieser Prozess kann mit einem Werkzeug wie ports-mgmt/portmaster automatisiert werden:

```
# portmaster -f
```

Sobald dies abgeschlossen ist, beenden Sie den Aktualisierungsprozess mit einem letzten Aufruf von `freebsd-update`. Geben Sie den folgenden Befehl ein, um alle losen Enden des Aktualisierungsprozesses miteinander zu verknüpfen:

```
# freebsd-update install
```

Wenn der `GENERIC`-Kernel temporär Verwendung fand, ist dies der richtige Zeitpunkt, einen neuen, angepassten Kernel zu bauen und über die übliche Methode zu installieren.

Booten Sie anschließend die Maschine in die neue FreeBSD-Version. Der Prozess ist damit abgeschlossen.

24.2.4. Vergleich des Systemzustands

`freebsd-update` kann verwendet werden, um den Zustand der installierten FreeBSD-Version gegenüber einer bekannten und funktionierenden Kopie zu vergleichen. Diese Option vergleicht die aktuelle Version von Systemwerkzeugen, Bibliotheken und Konfigurationsdateien. Um diesen Vergleich zu starten, geben Sie den folgenden Befehl ein:

```
# freebsd-update IDS >> outfile.ids
```

Warnung

Obwohl der Befehlsname IDS lautet, ist dies kein Ersatz für ein echtes Intrusion Detection System wie security/snort. Da `freebsd-update` seine Daten auf Platte ablegt, ist die Möglichkeit von Verfälschungen offensichtlich. Obwohl diese Möglichkeit durch die Verwendung

> von `kern.securelevel` oder die Ablage von Daten auf einem Nur-Lese Dateisystem eingedämmt werden kann, besteht eine bessere Lösung darin, das System gegen ein gesichertes Medium, wie eine DVD oder einen externen, separat aufbewahrten USB-Plattenspeicher, zu vergleichen.

Das System wird nun überprüft, und eine lange Liste von Dateien zusammen mit den sha256(1)-Hashwerten, sowohl der von der Release-Version bekannte Wert als auch der des aktuell installierten Systems, in `outfile.ids` geschrieben.

Die Zeilen in der Ausgabe sind extrem lang, aber das Ausgabeformat kann einfach verarbeitet werden. Um beispielsweise eine Liste von allen Dateien zu erhalten, die sich vom aktuellen Release unterscheiden, geben Sie das folgende Kommando ein:

```
# cat outfile.ids | awk '{ print $1 }' | more
/etc/master.passwd
/etc/motd
/etc/passwd
/etc/pf.conf
```

Diese Beispielausgabe wurde abgeschnitten, da noch viele weitere Dateien vorhanden sind. Einige Dateien wurden auf natürliche Art verändert. `/etc/passwd` wurde beispielsweise geändert, um Benutzer zum System hinzuzufügen. Andere Dateien, wie Kernelmodule, unterscheiden sich, weil `freebsd-update` diese aktualisiert hat. Um bestimmte Dateien oder Verzeichnisse auszuschließen, fügen Sie diese an die `IDSIgnorePaths`-Option in `/etc/freebsd-update.conf` an.

Diese Vorgehensweise kann als Teil einer ausgeklügelten Aktualisierungsmethode benutzt werden, unabhängig von der zuvor angesprochenen Variante.

24.3. Portsnap: Ein Werkzeug zur Aktualisierung der Ports-Sammlung

Geschrieben von Tom Rhodes.
Basierend auf bereitgestellten Mitschriften von Colin Percival.
Übersetzt von Benedict Reuschling.

Das Basissystem von FreeBSD enthält portsnap(8) zum Aktualisieren der Ports-Sammlung. Dieses Programm verbindet sich mit einem entfernten Rechner, überprüft den Sicherungsschlüssel und lädt eine neue Kopie der Ports-Sammlung herunter. Der Schlüssel wird verwendet, um die Integrität aller heruntergeladenen Dateien zu prüfen. Um die aktuellsten Dateien der Ports-Sammlung herunter zu laden, geben Sie das folgende Kommando ein:

```
# portsnap fetch
Looking up portsnap.FreeBSD.org mirrors... 9 mirrors found.
Fetching snapshot tag from geodns-1.portsnap.freebsd.org... done.
Fetching snapshot metadata... done.
Updating from Tue May 22 02:12:15 CEST 2012 to Wed May 23 16:28:31 CEST 2012.
Fetching 3 metadata patches.. done.
Applying metadata patches... done.
Fetching 3 metadata files... done.
Fetching 90 patches.....10....20....30....40....50....60....70....80....90. done.
Applying patches... done.
Fetching 133 new ports or files... done.
```

Dieses Beispiel zeigt, dass portsnap(8) mehrere Korrekturen für die aktuellen Ports-Daten gefunden und verifiziert hat. Es zeigt auch, dass das Programm zuvor schon einmal gestartet wurde. Wäre es das erste Mal, würde nur die Ports-Sammlung heruntergeladen werden.

Wenn portsnap(8) die `fetch`-Operation erfolgreich abgeschlossen hat, befinden sich die Ports-Sammlung und die dazugehörigen Korrekturen, welche die Überprüfung bestanden haben, auf dem lokalen System. Wenn `portsnap`

das erste Mal ausgeführt wird, muss `portsnap extract` aufgerufen werden, um die Ports-Sammlung zu installieren:

```
# portsnap extract
/usr/ports/.cvsignore
/usr/ports/CHANGES
/usr/ports/COPYRIGHT
/usr/ports/GIDs
/usr/ports/KNOBS
/usr/ports/LEGAL
/usr/ports/MOVED
/usr/ports/Makefile
/usr/ports/Mk/bsd.apache.mk
/usr/ports/Mk/bsd.autotools.mk
/usr/ports/Mk/bsd.cmake.mk
...
```

Um eine bereits installierte Ports-Sammlung zu aktualisieren, verwenden Sie `portsnap update`:

```
# portsnap update
```

Der Prozess ist jetzt abgeschlossen und Anwendungen können mittels der aktuellen Ports-Sammlung installiert oder aktualisiert werden.

Die Operationen `fetch` und `extract` oder `update` können auch nacheinander ausgeführt werden:

```
# portsnap fetch update
```

Dieser Befehl lädt die aktuelle Version der Ports-Sammlung herunter und aktualisiert anschließend die lokale Version unter `/usr/ports` .

24.4. Aktualisieren der Dokumentationssammlung

Übersetzt von Benedict Reuschling.

Dokumentation ein wichtiger Bestandteil des FreeBSD Betriebssystems. Obwohl eine aktuelle Version der FreeBSD Dokumentation jederzeit auf der FreeBSD Webseite verfügbar ist, verfügen manche Benutzer nur über eine langsame oder überhaupt keine Netzwerkverbindung. Es gibt mehrere Möglichkeiten, die lokale Kopie der Dokumentation durch die aktuelle FreeBSD-Dokumentationssammlung zu aktualisieren.

24.4.1. Verwenden von Subversion um die Dokumentation zu aktualisieren

Die Dokumentationsquellen von FreeBSD können mittels svn aktualisiert werden. Dieser Abschnitt beschreibt:

- Die Installation der Dokumentations-Werkzeugsammlung, welche die Werkzeuge enthält, die nötig sind, um die FreeBSD Dokumentation aus den Quellen neu zu erstellen.
- Das Herunterladen einer Kopie der Dokumentationsquellen nach `/usr/doc` , unter Verwendung von svn.
- Den Bau der FreeBSD Dokumentation aus den Quellen und die Installation unter `/usr/share/doc` .
- Manche der Optionen zum Erstellen, die vom System zum Bauen der Dokumentation unterstützt werden, z.B. die Optionen welche nur ein paar der unterschiedlichen Sprachübersetzungen der Dokumentation erstellen oder die Optionen, die ein bestimmtes Ausgabeformat auswählen.

24.4.2. svn und die Werkzeugsammlung der Dokumentation installieren

Die Erstellung der FreeBSD Dokumentation aus den Quellen benötigt eine große Anzahl an Werkzeugen, die nicht Teil des FreeBSD Basissystems sind, da sie eine große Menge Plattenplatz verbrauchen und nicht von allen Free-

BSD-Anwendern benötigt werden. Sie sind daher nur für diejenigen Benutzer sinnvoll, die aktiv neue Dokumentation für FreeBSD schreiben oder häufig die Dokumentation aus den Quellen aktualisieren.

Alle benötigten Werkzeuge, einschließlich svn sind im Meta-Port textproc/docproj vorhanden, der vom FreeBSD Documentation Project entwickelt wurde.

Anmerkung

Wenn Sie die Dokumentation nicht als PostScript® oder PDF benötigen, können Sie alternativ die Installation des textproc/docproj-nojadetex-Ports in Erwägung ziehen. Diese Version der Dokumentations-Werkzeugsammlung enthält alles ausser das teTeX-Textsatzsystem. teTeX ist eine sehr grosse Sammlung an Werkzeugen, deshalb ist es vernünftig, deren Installation auszulassen, wenn die Ausgabe von PDF nicht unbedingt gebraucht wird.

24.4.3. Die Dokumentationsquellen aktualisieren

In diesem Beispiel wird svn verwendet, um eine saubere Kopie der Dokumentationsquellen über das HTTPS-Protokoll zu holen:

```
# svn checkout https://svn.freebsd.org/doc/head /doc/head /usr/doc
```

Benutzen Sie dazu einen der Spiegel aus Subversion Mirror Sites.

Es dauert eine Weile, wenn die Dokumentationsquellen das allererste Mal heruntergeladen werden. Lassen Sie es laufen, bis es fertig ist.

Zukünftige Aktualisierungen der Dokumentationsquellen können wie folgt durchgeführt werden:

```
# svn update /usr/doc
```

Nachdem die Quellen einmal ausgecheckt wurden, wird durch `/usr/doc/Makefile` ein alternativer Weg unterstützt, die Dokumentation zu aktualisieren. Geben Sie dazu die folgenden Befehle ein:

```
# cd /usr/doc
# make update
```

24.4.4. Einstellbare Optionen der Dokumentationsquellen

Das System zum aktualisieren und erstellen der FreeBSD-Dokumentationssammlung unterstützt ein paar Optionen, welche den Prozess der Aktualisierung von Teilen der Dokumentation oder einer bestimmten Übersetzung erleichtert. Diese Optionen können entweder systemweit in `/etc/make.conf` gesetzt, oder als Kommandozeilenoptionen an make(1) übergeben werden.

Zu den Optionen gehören:

`DOC_LANG`
: Eine Liste von Sprachen und Kodierungen, die gebaut und installiert werden sollen, z.B. `en_US.ISO8859-1` , um nur die englische Dokumentation zu erhalten.

`FORMATS`
: Ein einzelnes Format oder eine Liste von Ausgabeformaten, das gebaut werden soll. Momentan werden `html` , `html-split` , `txt`, `ps`, `pdf`, und `rtf` unterstützt.

`DOCDIR`
: Wohin die Dokumentation installiert werden soll. Der Standardpfad ist `/usr/share/doc` .

Für weitere `make`-Variablen, die als systemweite Optionen in FreeBSD unterstützt werden, lesen Sie make.conf(5).

Für weitere `make`-Variablen, die vom System zum Erstellen der FreeBSD-Dokumentation unterstützt werden, lesen Sie die Fibel für neue Mitarbeiter des FreeBSD-Dokumentationsprojekts.

24.4.5. Die FreeBSD-Dokumentation aus den Quellen installieren

Sobald ein aktueller Schnappschuss der Dokumentationsquellen nach `/usr/doc` heruntergeladen wurde, ist alles bereit für eine Aktualisierung der bestehenden Dokumentation.

Eine komplette Aktualisierung aller Sprachen, definiert in `DOC_LANG`, kann durch folgende Eingabe erreicht werden:

```
# cd /usr/doc
# make install clean
```

Wenn nur eine Aktualisierung einer bestimmten Sprache gewünscht wird, kann make(1) in einem sprachspezifischen Unterverzeichnis von `/usr/doc` aufgerufen werden:

```
# cd /usr/doc/en_US.ISO8859-1
# make update install clean
```

Die zu installierenden Ausgabeformate können durch das Setzen von `FORMATS` angegeben werden:

```
# cd /usr/doc
# make FORMATS='html html-split' install clean
```

Informationen zum Bearbeiten und Einreichen von Korrekturen finden Sie in der Fibel für neue Mitarbeiter des FreeBSD-Dokumentationsprojekts.

24.4.6. Verwendung von Dokumentations-Ports

Basierend auf der Arbeit von Marc Fonvieille.

Im vorherigen Abschnitt wurde eine Methode gezeigt, wie die FreeBSD-Dokumentation aus den Quellen gebaut werden kann. Allerdings sind quellbasierte Aktualisierungen möglicherweise nicht für alle FreeBSD-Systeme geeignet oder praktikabel. Das Erstellen der Dokumentationsquellen benötigt eine große Anzahl an Werkzeugen, Programmen und Hilfsmitteln, die *documentation toolchain*, einen gewissen Grad an Vertrautheit mit svn, ausgecheckte Quellen von einem Repository, sowie ein paar manuelle Schritte, um diese ausgecheckten Quellen zu bauen. Dieser Abschnitt beschreibt eine alternative Methode, in der die Ports-Sammlung verwendet wird und die es ermöglicht:

- vorgefertigte Schnappschüsse der Dokumentation herunterzuladen und zu installieren, ohne vorher die Werkzeugsammlung der Dokumentation installieren zu müssen.
- die Dokumentationsquellen herunterzuladen und durch das Ports-System erstellen zu lassen, was die Schritte zum Auschecken und Erstellen etwas erleichtert.

Diese beiden Methoden der Aktualisierung der FreeBSD-Dokumentation werden durch eine Menge von *Dokumentations-Ports* unterstützt, die von Documentation Engineering Team <doceng@FreeBSD.org> monatlich aktualisiert wird. Diese sind in der FreeBSD Ports-Sammlung unter der Kategorie docs gelistet (http://www.freshports.org/docs/).

24.4.6.1. Erstellen und Installieren von Dokumentations-Ports

Die Dokumentations-Ports nutzen das Ports-System, um das Erstellen von Dokumentation wesentlich einfacher zu machen. Es automatisiert den Prozess des Auscheckens der Dokumentationsquellen, aufrufen von make(1) mit den passenden Umgebungsvariablen und Kommandozeilenoptionen und macht die Installation und Deinstallation von Dokumentation so einfach wie die Installation von jedem anderen Port oder Paket.

Anmerkung

Als zusätzliche Eigenschaft zeichnen sie eine Abhängigkeit zur *Dokumentations-Werkzeugsammlung* auf, wenn die Dokumentations-Ports lokal erstellt werden, weshalb diese auch automatisch mitinstalliert wird.

Die Dokumentations-Ports sind wie folgt organisiert:

- Der „Master-Port“, misc/freebsd-doc-en, der alle englischen Dokumentations-Ports installiert.
- Der „Alles-in-Einem-Port“, misc/freebsd-doc-all, welcher die komplette Dokumentation in allen verfügbaren Sprachen erstellt und installiert.
- Es gibt noch einen „Slave-Port“ für jede Übersetzung, beispielsweise misc/freebsd-doc-hu für Dokumentation in ungarischer Sprache.

Um die englische Dokumentation zu bauen im getrennten HTML-Format in `/usr/local/share/doc/freebsd` zu installieren, installieren Sie den folgenden Port:

```
# cd /usr/ports/misc/freebsd-doc-en
# make install clean
```

24.4.6.1.1. Gebräuchliche Schalter und Optionen

Es gibt viele Optionen, die das Standardverhalten der Dokumentations-Ports verändern. Dazu gehören:

`WITH_HTML`
: Erstellt das HTML-Format mit einer einzigen HTML-Datei pro Dokument. Die formatierte Dokumentation wird als Datei mit dem Namen `article.html`, oder gegebenenfalls `book.html`, zuzüglich der Bilder gespeichert.

`WITH_PDF`
: Erstellt das Adobe® Portable Document Format (PDF). Die formatierte Dokumentation wird als Datei mit dem Namen `article.pdf`, oder gegebenenfalls als `book.pdf` gespeichert.

`DOCBASE`
: Legt den Pfad fest, wohin die Dokumentation installiert werden soll. Die Voreinstellung ist `/usr/local/share/doc/freebsd` .

Anmerkung

Der Standardpfad zum Verzeichnis unterscheidet sich von dem Verzeichnis, das von svn verwendet wird. Das liegt daran, dass Ports üblicherweise in `/usr/local` installiert werden. Dies kann durch die Verwendung von `PREFIX` überschrieben werden.

Dieses Beispiel verwendet Variablen, um die ungarische Dokumentation als PDF zu installieren:

```
# cd /usr/ports/misc/freebsd-doc-hu
# make -DWITH_PDF DOCBASE=share/doc/freebsd/hu install clean
```

24.4.6.2. Verwendung von Dokumentations-Paketen

Das Erstellen der Dokumentations-Ports aus den Quellen, wie im vorherigen Abschnitt beschrieben, benötigt die lokale Installation der Dokumentations-Werkzeugsammlung und ein wenig Festplattenspeicher für das Bauen

der Ports. Sollten die Ressourcen zum Bauen der Dokumentations-Werkzeugsammlung nicht zur Verfügung stehen, oder weil das erstellen zuviel Plattenplatz benötigen würde, ist es trotzdem möglich, bereits zuvor gebaute Schnappschüsse der Dokumentations-Ports zu installieren.

Documentation Engineering Team <doceng@FreeBSD.org> erstellt monatliche Schnappschüsse der Dokumentations-Pakete von FreeBSD. Diese Binärpakete können mit jedem der mitgelieferten Paketwerkzeuge installiert werden, beispielsweise pkg_add(1), pkg_delete(1) und so weiter.

Anmerkung

Wenn Binärpakete zu Einsatz kommen, wird die FreeBSD-Dokumentation in *allen* verfügbaren Formaten in der gegebenen Sprache installiert.

Zum Beispiel installiert das folgende Kommando das aktuelle, vorgefertigte Paket der ungarischen Dokumentation:

```
# pkg_add -r hu-freebsd-doc
```

Anmerkung

Pakete verwenden ein Format, welches sich von dem Namen des dazugehörigen Ports unterscheidet: *lang*`-freebsd-doc` . *lang* entspricht hier der Kurzform des Sprachcodes, z.B. `hu` für Ungarisch, oder `zh_cn` für vereinfachtes Chinesisch.

24.4.6.3. Dokumentations-Ports aktualisieren

Dokumentations-Ports können wie jeder andere Port aktualisiert werden. Beispielsweise aktualisiert das folgende Kommando die installierte ungarische Dokumentation mittels ports-mgmt/portmaster unter Verwendung von Paketen:

```
# portmaster -PP hu-freebsd-doc
```

24.5. Einem Entwicklungszweig folgen

FreeBSD besitzt zwei Entwicklungszweige: FreeBSD-CURRENT und FreeBSD-STABLE. Dieser Abschnitt beschreibt beide Zweige und erläutert, wie Sie ein System auf dem aktuellen Stand eines Zweiges halten. Zuerst wird FreeBSD-CURRENT vorgestellt, dann FreeBSD-STABLE.

24.5.1. FreeBSD-CURRENT

FreeBSD-CURRENT ist die Spitze der Entwicklung von FreeBSD. Benutzer von FreeBSD-CURRENT sollten über sehr gute technische Fähigkeiten verfügen und in der Lage sein, schwierige Probleme alleine zu lösen. Wenn FreeBSD neu für Sie ist, verwenden Sie besser FreeBSD-STABLE.

24.5.1.1. Was ist FreeBSD-CURRENT?

FreeBSD-CURRENT besteht aus den neuesten Quellen des FreeBSD-Systems. Es enthält Sachen, an denen gerade gearbeitet wird, experimentelle Änderungen und Übergangsmechanismen, die im nächsten offiziellen Release der Software enthalten sein können oder nicht. Obwohl FreeBSD-CURRENT täglich von vielen Entwicklern gebaut wird, gibt es Zeiträume, in denen sich das System nicht bauen lässt. Diese Probleme werden so schnell wie möglich behoben, aber ob Sie mit FreeBSD-CURRENT Schiffbruch erleiden oder die gewünschten Verbesserungen erhalten, kann von dem Zeitpunkt abhängen, an dem der Quelltext synchronisiert wurde.

24.5.1.2. Wer braucht FreeBSD-CURRENT?

FreeBSD-CURRENT wird hauptsächlich für drei Interessengruppen zur Verfügung gestellt:

1. Entwickler, die an einem Teil des Quellbaums arbeiten und daher über die aktuellen Quellen verfügen müssen.
2. Tester, die bereit sind, Zeit in das Lösen von Problemen zu investieren und sicherstellen, dass FreeBSD-CURRENT so stabil wie möglich bleibt. Diese Tester machen Vorschläge zu Änderungen oder der generellen Entwicklung von FreeBSD und stellen Patches bereit, um diese Vorschläge zu realisieren.
3. Für Leute, die die Entwicklung im Auge behalten wollen, oder die Quellen zu Referenzzwecken benutzen wollen. Auch diese Gruppe macht Vorschläge oder steuert Quellcode bei.

24.5.1.3. Was FreeBSD-CURRENT nicht ist!

1. Der schnellste Weg, neue Funktionen vor dem offiziellen Release auszuprobieren. Bedenken Sie, dass neue Funktionen noch nicht im vollen Umfang getestet wurden und daher höchstwahrscheinlich Fehler enthalten.
2. Ein schneller Weg, um an Fehlerbehebungen (engl. *bug fixes*) zu kommen. Jede Fehlerbehebung führt mit gleicher Wahrscheinlichkeit neue Fehler ein, mit der sie alte behebt.
3. In keiner Weise „offiziell unterstützt“.

24.5.1.4. Benutzen von FreeBSD-CURRENT

1. Lesen Sie die Mailinglisten freebsd-current und svn-src-head. Dies ist *notwendig*, um die Kommentare über den akutellen Status des Systems und wichtige Mitteilungen zum aktuellen Zustand von FreeBSD-CURRENT zu erfahren.

   Die svn-src-head Mailingliste erfasst die Commit-Logs für jede Änderung und enthält alle relevanten Informationen zu möglichen Seiteneffekten.

   Um diese Listen zu abonnieren, besuchen Sie http://lists.FreeBSD.org/mailman/listinfo, klicken Sie auf die gewünschte Liste und folgen Sie den Anweisungen. Wenn Sie die Änderungen am gesamten Quellbaum verfolgen möchten, abonnieren Sie die svn-src-all Liste.

2. Beschaffen Sie sich die Quellen von einem FreeBSD-Spiegel, mit einer der folgenden Methoden:

   a. Benutzen Sie svn, um den gewünschten Entwicklungs- oder Release-Zweig auszuchecken. Dies ist die empfohlene Methode für den Zugang zur Entwicklung von FreeBSD. Checken Sie den -CURRENT Quelltext aus dem `head`-Zweig von einem der Subversion Mirror Sites aus. Aufgrund der Größe des Repositories ist es empfehlenswert, nur die gewünschten Teilbäume auszuchecken.

   b. Benutzen Sie CTM, wenn Sie über eine schlechte Internet-Anbindung verfügen. CTM ist eine Option, aber es ist nicht so zuverlässig wie Subversion. Aus diesem Grund ist Subversion die empfohlene Methode für jedes System mit Internet-Anbindung.

3. Wenn Sie die Quellen einsetzen und nicht nur darin lesen wollen, besorgen Sie sich die *kompletten* Quellen von FreeBSD-CURRENT und nicht nur ausgesuchte Teile. Der Grund hierfür ist, dass die verschiedenen Teile der Quellen voneinander abhängen. Es ist ziemlich sicher, dass Sie in Schwierigkeiten geraten, wenn Sie versuchen, nur einen Teil der Quellen zu übersetzen.

   Lesen Sie `/usr/src/Makefile` sehr aufmerksam, bevor Sie FreeBSD-CURRENT übersetzen. Wenn Sie FreeBSD das erste Mal aktualisieren, sollten Sie sowohl einen Kernel als auch das System neu installieren. Lesen Sie die Mailingliste FreeBSD-CURRENT und `/usr/src/UPDATING` , um über Änderungen im Installationsverfahren, die manchmal vor der Einführung eines neuen Releases notwendig sind, informiert zu sein.

4. Seien Sie aktiv! Benutzer von FreeBSD-CURRENT werden aufgefordert ihre Verbesserungsvorschläge oder Fehlerbehebungen einzureichen. Verbesserungsvorschläge, die Code enthalten, werden übrigens begeistert entgegengenommen!

24.5.2. FreeBSD-STABLE

24.5.2.1. Was ist FreeBSD-STABLE?

FreeBSD-STABLE ist der Entwicklungszweig, auf dem Releases erstellt werden. Dieser Zweig ändert sich langsamer als FreeBSD-CURRENT und alle Änderungen hier sollten zuvor in FreeBSD-CURRENT ausgetestet sein. Beachten Sie, dass dies *immer noch* ein Entwicklungszweig ist und daher zu jedem Zeitpunkt die Quellen von FreeBSD-STABLE verwendbar sein können oder nicht. FreeBSD-STABLE ist Teil des Entwicklungsprozesses und nicht für Endanwender gedacht.

24.5.2.2. Wer braucht FreeBSD-STABLE?

Wer daran interessiert ist den FreeBSD-Entwicklungsprozess zu verfolgen oder dazu beizutragen, insbesondere im Hinblick auf das nächste Hauptversion, der sollte es in Erwägung ziehen, FreeBSD-STABLE zu benutzen.

Auch wenn sicherheitsrelevante Fehlerbehebungen in den FreeBSD-STABLE Zweig einfließen, müssen Sie deswegen noch lange nicht FreeBSD-STABLE verfolgen. Jeder FreeBSD Sicherheitshinweis beschreibt für jedes betroffene Release, wie der sicherheitsrelevante Fehler behoben wird. [1]

Obwohl wir versuchen sicherzustellen, dass der FreeBSD-STABLE Zweig sich jederzeit übersetzen lässt und lauffähig ist, können wir dafür keine Garantie übernehmen. Auch wenn Neuentwicklungen in FreeBSD-CURRENT stattfinden, ist es jedoch so, dass mehr Leute FreeBSD-STABLE anstelle von FreeBSD-CURRENT benutzen und es daher unvermeidlich ist, dass Fehler und Grenzfälle erst in FreeBSD-STABLE auffallen.

Aus diesen Gründen empfehlen wir Ihnen *nicht* blindlings FreeBSD-STABLE zu benutzen. Es ist besonders wichtig, dass FreeBSD-STABLE zuerst sorgfältig in einer Testumgebung getestet wird, bevor die Produktion auf FreeBSD-STABLE migriert.

Benutzer, die keine Ressourcen haben, um diese Tests durchzuführen wird empfohlen, das aktuelle FreeBSD-Release zu verwenden und den binären Update-Mechanismus zu nutzen, um auf neue Releases zu migrieren.

24.5.2.3. Benutzen von FreeBSD-STABLE

1. Lesen Sie Mailingliste freebsd-stable, damit Sie über Abhängigkeiten beim Bau von FreeBSD-STABLE und Sachen, die besondere Aufmerksamkeit erfordern, informiert sind. Umstrittene Fehlerbehebungen oder Änderungen werden von den Entwicklern auf dieser Liste bekannt gegeben. Dies erlaubt es den Benutzern, Einwände gegen die vorgeschlagenen Änderungen vorzubringen.

   Abonnieren Sie die passende SVN-Liste für den jeweiligen Branch, den Sie verfolgen. Wenn Sie beispielsweise den Zweig 9-STABLE verfolgen, lesen Sie die svn-src-stable-9. Diese Liste enthält zu jeder Änderung das Commit-Log, das Informationen zu möglichen Seiteneffekten enthält.

   Um diese Listen zu abonnieren, besuchen Sie die Seite http://lists.FreeBSD.org/mailman/listinfo. Klicken Sie auf die gewünschte Liste und folgenden Sie den Anweisungen. Wenn Sie daran interessiert sind, Änderungen am gesamten Quellbaum zu verfolgen, abonnieren Sie svn-src-all.

2. Wenn Sie ein neues System installieren und dazu einen der monatlich aus FreeBSD-STABLE erzeugten Snapshots verwenden wollen, sollten Sie zuerst die Snapshot Website auf aktuelle Informationen überprüfen. Alternativ können Sie auch das neueste FreeBSD-STABLE-Release von den Spiegeln beziehen und das System nach den folgenden Anweisungen aktualisieren.

   Es stehen mehrere Methoden zur Verfügung, um ein System mit einem älteren Release von einem der FreeBSD-Spiegel zu aktualisieren.

   a. Benutzen Sie svn, um den gewünschten Entwicklungs- oder Release-Zweig auszuchecken. Dies ist die empfohlene Methode für den Zugang zur Entwicklung von FreeBSD. Die Zweige umfassen `head`, für den aktuellen Entwicklungszweig, sowie weitere Zweige die auf der Release Engineering Seite beschrieben sind, wie beispielsweise `stable/9` oder `releng/9.0` . Das bevorzugte URL-Präfix für Subversion zum Auschecken des

[1] Eine vollständige Beschreibung der Sicherheitspolitik für alte FreeBSD Releases entnehmen Sie http://www.FreeBSD.org/security/.

Basissystems ist `http://svn.freebsd.org/base/` . Aufgrund der Größe des Repositories ist es empfehlenswert, nur die gewünschten Teilbäume auszuchecken.

b. Wenn Sie über keine schnelle Internet-Anbindung verfügen, sollten Sie die Nutzung von CTM in Betracht ziehen.

3. Benutzen Sie Subversion, wenn Sie schnellen Zugriff auf die Quellen brauchen und die Bandbreite keine Rolle spielt, andernfalls benutzen Sie CTM.

4. Lesen Sie `/usr/src/Makefile` sehr aufmerksam, bevor Sie FreeBSD-STABLE übersetzen. Wenn Sie FreeBSD das erste Mal aktualisieren, sollten Sie sowohl einen Kernel als auch das System neu installieren. Lesen Sie die Mailingliste FreeBSD-STABLE und `/usr/src/UPDATING` , um über Änderungen im Installationsverfahren, die manchmal vor der Einführung eines neuen Releases notwendig sind, informiert zu sein.

24.6. Synchronisation der Quellen

Sie können eine Internet-Verbindung (oder E-Mail) dazu nutzen, Teile von FreeBSD, wie die Quellen zu einzelnen Projekten, oder das Gesamtsystem, aktuell zu halten. Die primären Dienste dafür sind Subversion und CTM.

Warnung

Obwohl es möglich ist, nur Teile des Quellbaums zu aktualisieren, ist die einzige unterstützte Migrationsprozedur, den kompletten Quellbaum zu aktualisieren und alles neu zu übersetzen. Dazu zählen alle Userland-Programme in `/bin` und `/sbin`, sowie die Kernelquellen. Wird hingegen nur ein Teil der Quellen, zum Beispiel nur der Kernel oder nur die Programme aus dem Userland aktualisiert, treten Probleme auf, die von Übersetzungsfehlern über Kernel-Panics bis hin zu Beschädigung von Daten reichen können.

Subversion benutzt die *Pull*-Methode [2], um die Quellen zu aktualisieren. Der Benutzer, oder ein `cron`-Skript, ruft das Programm `svn` auf, das die Quellen aktualisiert. Subversion ist die empfohlene Methode, um die lokalen Quellen zu aktualisieren. Mit beiden Methoden erhalten Sie aktuelle Updates zu einem genau von Ihnen bestimmten Zeitpunkt. Es ist einfach, die Prozedur auf bestimmte Dateien oder Verzeichnisse einschränken. Die Updates werden zur Laufzeit generiert.

CTM vergleicht die Quellen nicht mit denen auf einem Server. Stattdessen läuft auf dem Server ein Skript, das Änderungen an Dateien gegenüber seinem vorigen Lauf bemerkt, die Änderungen komprimiert, mit einer Sequenznummer versieht und für das Verschicken per E-Mail kodiert. Dabei werden nur druckbare ASCII-Zeichen verwendet. Wenn Sie diese „CTM-Deltas“ erhalten haben, können Sie sie mit ctm_rmail(1) benutzen, welches die Deltas dekodiert, verifiziert und dann die Änderungen an den Quellen vornimmt. Dieses Verfahren ist viel effizienter als Subversion und erzeugt auch weniger Last auf den Servern, da es die *Push*-Methode [3] verwendet.

Es gibt noch weitere Unterschiede. Wenn ein Benutzer unabsichtlich Teile des Archivs löschen, wird das von Subversion erkannt und repariert. CTM leistet das nicht. Wenn ein Benutzer Teile des Quellbaums gelöscht hat und keine Sicherung besitzt, muss er von neuem, das heißt vom letzten „Basis-Delta“, starten und die Änderungen wieder mit CTM nachziehen.

24.7. Das komplette Basissystem neu bauen

[2] Von engl. *to pull* = *ziehen*. Der Client holt sich bei dieser Methode die Dateien ab.

[3] Von engl. *to push* = *schieben*. Der Server schickt dem Client die Dateien.

Sobald der lokalen Quellbaum mit einer bestimmten FreeBSD Version, z.B. FreeBSD-STABLE oder FreeBSD-CURRENT synchronisiert wurde, kann dieser dazu benutzt werden das System neu zu bauen.

Erstellen Sie eine Sicherungskopie!

Es kann nicht oft genug betont werden, wie wichtig es ist, das System zu sichern, *bevor* die nachfolgenden Schritte ausgeführt werden. Obwohl der Neubau des Systems eine einfache Aufgabe ist, kann dennoch vorkommen, dass Fehler im Quellbaum dazu führen, dass das System nicht mehr bootet.

Stellen Sie sicher, dass Sie eine Sicherung erstellt haben und über ein startfähiges Installationsmedium verfügen. Wahrscheinlich werden die Medien nicht benötigt, aber gehen Sie auf Nummer sicher!

Abonnieren Sie die richtige Mailingliste

Die FreeBSD-STABLE und FreeBSD-CURRENT Zweige befinden sich in *ständiger Entwicklung*. Die Leute, die zu FreeBSD beitragen, sind Menschen und ab und zu machen sie Fehler.

Manchmal sind diese Fehler harmlos und lassen das System eine Warnung ausgeben. Die Fehler können allerdings auch katastrophal sein und dazu führen, dass das System nicht mehr booten kann, oder Dateisysteme beschädigt werden.

Wenn Probleme auftauchen, wird ein „heads up“ an die passende Mailingliste geschickt, welches das Problem erklärt und die betroffenen Systeme benennt. Eine „all clear“ Meldung wird versendet, wenn das Problem gelöst ist.

Benutzer, die FreeBSD-STABLE oder FreeBSD-CURRENT benutzen und nicht die Mailinglisten FreeBSD-STABLE beziehungsweise FreeBSD-CURRENT lesen, bringen sich nur unnötig in Schwierigkeiten.

Verwenden Sie nicht `make world`

Einige ältere Dokumentationen empfehlen `make world` für den Neubau. Das Kommando überspringt jedoch wichtige Schritte und sollte nur von Experten verwendet werden. In fast allen Fällen ist `make world` falsch. Benutzen Sie stattdessen die nachstehende Anleitung.

24.7.1. Richtig aktualisieren

Bevor das System aktualisiert wird, lesen Sie `/usr/src/UPDATING` , um die für die Quellcodeversion nötigen Aufgaben zu erledigen, bevor das System neu gebaut wird. Danach kann das System mit den folgenden Schritten aktualisiert werden.

Bei den hier dargestellten Aktualisierungsschritten wird davon ausgegangen, dass momentan eine alte FreeBSD-Version verwendet wird, die aus einem alten Compiler, Kernel, sowie einem alten Basissystem und veralteten Konfigurationsdateien besteht. Mit „Basissystem“ sind hier die zentralen Binärdateien, Bibliotheken und Entwicklerdateien gemeint. Der Compiler ist Teil des „Basissystems“, beinhaltet aber ein paar Besonderheiten.

Es wird außerdem davon ausgegangen, dass bereits die Quellen für ein neues System bezogen wurden. Falls die Quellen nicht auf dem aktuellen Stand sind, lesen Sie Abschnitt 24.6, „Synchronisation der Quellen“, um detaillierte Hilfe über die Aktualisierung der Quellen zu erhalten.

Die Aktualisierung des Systems aus den Quellen ist ein wenig ausgetüftelter als es zunächst den Anschein hat. Die Entwickler von FreeBSD haben es über die Jahre für Nötig befunden, den vorgeschlagenen Ablauf ziemlich stark zu verändern, da neue Arten von unvermeidlichen Abhängigkeiten mit der Zeit ans Licht kamen. Der übrige Teil dieses Abschnitts beschreibt die Überlegungen hinter der aktuell empfohlenen Aktualisierungsreihenfolge.

Jede erfolgreiche Aktualisierung muss sich mit den folgenden Sachverhalten auseinandersetzen:

- Der alte Compiler ist aufgrund von Fehlern möglicherweise nicht in der Lage, den neuen Kernel zu übersetzen. Deshalb sollte der neue Kernel mit dem neuen Compiler übersetzt werden, was bedeutet, dass der neue Compiler vor dem neuen Kernel gebaut werden muss. Das bedeutet nicht unbedingt, dass der neue Compiler auch *installiert* werden muss, bevor der neue Kernel gebaut wird.
- Das neue Basissystem benötigt eventuell neue Eigenschaften des Kernels. Also muss der neue Kernel installiert sein, bevor das neue Basissystem installiert wird.

Diese ersten beiden Sachverhalte sind die Grundlage für die zentrale Sequenz von `buildworld`, `buildkernel`, `installkernel` und `installworld`, die in den folgenden Abschnitten beschrieben wird. Weitere Gründe für diese Vorgehensweise sind hier aufgeführt:

- Das alte Basissystem wird möglicherweise nicht korrekt mit dem neuen Kernel funktionieren, weshalb das neue Basissystem sofort nach der Installation des neuen Kernels installiert werden muss.
- Manche Änderungen an der Konfiguration müssen erledigt worden sein, bevor das neue Basissystem installiert wird, jedoch können andere die Funktionalität des alten Basissystems beeinträchtigen. Aus diesem Grund sind zwei verschiedene Schritte notwendig, um eine Aktualisierung der Konfiguration durchzuführen.
- Der Aktualisierungsprozess ersetzt zum Grossteil Dateien oder fügt neue hinzu, bestehende Dateien werden nicht gelöscht. In wenigen Ausnahmefällen kann dies Probleme verursachen. Aus diesem Grund wird der Aktualisierungsprozess manchmal bestimmte Dateien zum manuellen Löschen vorschlagen. Dies wird eventuell in der Zukunft automatisch durchgeführt.

Diese Bedenken haben zu der folgenden Reihenfolge geführt. Beachten Sie, dass der genaue Ablauf für bestimmte Aktualisierungen zusätzliche Schritte nach sich zieht, jedoch sollte der Kernprozess davon nicht beeinträchtigt werden:

1. `make buildworld`

   Dieser Schritt übersetzt zuerst den neuen Compiler und ein paar damit zusammenhängende Werkzeuge und verwendet dann den neuen Compiler, um den Rest des Basissystems zu erstellen. Das Ergebnis landet dann in `/usr/obj` .

2. `make buildkernel`

   Dieser Ansatz nutzt den *neuen* Compiler, der in `/usr/obj` abgelegt ist, um vor falschen Compiler-Kernel-Kombinationen zu schützen.

3. `make installkernel`

   Platziert den neuen Kernel und Kernelmodule auf der Platte, was es erlaubt, mit dem frisch aktualisierten Kernel zu starten.

4. Starten Sie das System neu in den Single-User-Modus.

   Der Single-User-Modus minimiert Probleme mit der Aktualisierung von Programmen, die bereits gestartet sind. Ebenso minimiert es Probleme, die mit der Verwendung des alten Basissystems und des neuen Kernels zu tun haben könnten.

5. `mergemaster -p`

   Dieser Schritt aktualisiert ein paar initiale Konfigurationsdateien als Vorbereitung für das neue Basissystem. Beispielsweise fügt es neue Benutzergruppen zum System oder neue Benutzernamen in die Passwortdatenbank hinzu. Dies wird oftmals benötigt, wenn neue Gruppen oder bestimmte Systembenutzerkonten seit der letzten Aktualisierung hinzu gekommen sind, so dass der `installworld`-Schritt in der Lage ist, auf dem neu installierten System die Benutzer oder Systemgruppennamen ohne Probleme zu verwenden.

6. `make installworld`

   Kopiert das Basissystem aus `/usr/obj` . Der neue Kernel und das neue Basissystem sind jetzt auf der Platte installiert.

7. `mergemaster`

   Aktualisiert die verbleibenden Konfigurationsdateien, da nun das neue Basissystem auf der Platte ist.

8. Starten Sie das System neu.

   Ein kompletter Systemneustart ist notwendig, um den neuen Kernel und das neue Basissystem mit den neuen Konfigurationsdateien zu laden.

Beachten Sie, dass wenn Sie von einem Release des gleichen FreeBSD-Zweigs auf ein aktuelleres Release des gleichen Zweigs, z.B. von 9.0 auf 9.1, aktualisieren, dann ist diese Vorgehensweise nicht unbedingt notwendig, da Sie nur sehr unwahrscheinlich in ungünstige Kombinationen zwischen Compiler, Kernel, Basissystem und den Konfigurationsdateien geraten werden. Die ältere Vorgehensweise von `make world` , gefolgt von der Erstellung und Installation des neuen Kernels funktioniert möglicherweise gut genug, um kleinere Aktualisierungen vorzunehmen.

Wenn Sie allerdings zwischen Hauptversionen aktualisieren wollen und befolgen diese Schritte nicht, sollten Sie sich auf Probleme gefasst machen.

Es ist auch wichtig zu wissen, dass viele Aktualisierungen spezielle und zusätzliche Schritte benötigen, wie beispielsweise das umbenennen oder löschen von bestimmten Dateien vor installworld. Lesen Sie `/usr/src/UPDATING` gründlich, besonders am Ende, wo die aktuell vorgeschlagene Aktualisierungssequenz explizit aufgelistet ist.

Diese Prozedur hat sich mit der Zeit weiterentwickelt, da die Entwickler es für unmöglich erachtet haben, bestimmte Arten von Kombinationsproblemen vollständig auszuschliessen. Hoffentlich wird die aktuelle Aktualisierungsprozedur für lange Zeit stabil bleiben.

Als Zusammenfassung ist hier nochmal die aktuell vorgeschlagene Vorgehensweise für die Aktualisierung von FreeBSD aus den Quellen aufgelistet:

```
# cd /usr/src
# make buildworld
# make buildkernel
# make installkernel
# shutdown -r now
```

Anmerkung

Es gibt einige, sehr seltene Situationen, in denen Sie `mergemaster -p` zusätzlich ausführen müssen, bevor Sie das System mit `buildworld` bauen. Diese Situationen werden in `UPDATING` beschrieben. Solche Situationen treten aber in der Regel nur dann auf, wenn das FreeBSD-System um eine oder mehrere Hauptversionen aktualisiert wird.

Nachdem installkernel erfolgreich abgeschlossen wurde, starten Sie das System durch die Eingabe von boot -s am Loaderprompt im Single-User-Modus. Danach führen Sie die folgenden Kommandos aus:

```
# mount -u /
# mount -a -t ufs
# adjkerntz -i
# mergemaster -p
# cd /usr/src
# make installworld
# mergemaster
# reboot
```

Lesen Sie bitte weiter

Die folgenden Abschnitte beschreiben detailliert die einzelnen Schritte, insbesondere wenn eine angepasste Kernelkonfiguration verwendet wird.

24.7.2. Lesen Sie /usr/src/UPDATING

Lesen Sie vor der Aktualisierung /usr/src/UPDATING . Die Datei enthält wichtige Informationen zu potentiellen Problemen, und gibt die Reihenfolge vor, in der bestimmte Kommandos gestartet werden müssen. Die Anweisungen in UPDATING sind aktueller als die in diesem Handbuch. Im Zweifelsfall folgen Sie bitte den Anweisungen aus UPDATING .

Wichtig

Das Lesen von UPDATING ersetzt nicht das Abonnieren der richtigen Mailingliste. Die beiden Voraussetzungen ergänzen sich, es reicht nicht aus, nur eine zu erfüllen.

24.7.3. Überprüfen Sie /etc/make.conf

Die verfügbaren make(1)-Optionen werden in make.conf(5) und /usr/share/examples/etc/make.conf dargestellt. Diese Einstellungen können in /etc/make.conf hinzugefügt werden, um das Verhalten von make(1) beim Übersetzen von Programmen zu beeinflussen. Änderungen an einigen Einstellungen können weitreichende und unerwartete Auswirkungen nach sich ziehen. Lesen Sie die Kommentare in diesen beiden Ressourcen und beachten Sie, dass die Standardwerte aus einer Kombination von Leistung und Sicherheit gewählt wurden.

Die in /etc/make.conf gesetzten Optionen wirken sich bei jedem Aufruf von make(1) aus, einschließlich der Übersetzung von Programmen aus der Ports-Sammlung, vom Benutzer geschriebene C-Programme oder beim Bau des FreeBSD-Betriebssystems.

24.7.4. /etc/src.conf überprüfen

/etc/src.conf kontrolliert den Bau des Betriebssystems aus dem Quellcode. Im Gegensatz zu /etc/make.conf greifen die Optionen in /etc/src.conf nur dann, wenn das FreeBSD Betriebssystem selbst gebaut wird. Die vielen Optionen für diese Datei werden in src.conf(5) beschrieben. Seien Sie vorsichtig mit dem Entfernen von scheinbar nicht mehr benötigten Kernelmodulen und Optionen. Manchmal gibt es unerwartete oder subtile Wechselwirkungen.

24.7.5. Aktualisieren Sie die Dateien in /etc

/etc enthält den Großteil der Konfigurationsdateien des Systems und Skripten, die beim Start des Systems ausgeführt werden. Einige dieser Skripten ändern sich bei einer Migration auf eine neue FreeBSD-Version.

Einige der Konfigurationsdateien, wie beispielsweise `/etc/group` , werden für den Normalbetrieb des Systems gebraucht.

Es gab Fälle, in denen die Installationsroutine von `make installworld` auf bestimmte Accounts oder Gruppen angewiesen war. Bei einer Aktualisierung ist es jedoch wahrscheinlich, dass diese Accounts oder Gruppen noch nicht existieren. In einigen Fällen prüft `make buildworld` ob die Accounts oder Gruppen vorhanden sind.

Um dieses Problem zu umgehen, rufen Sie mergemaster(8) im prä-buildworld-Modus auf, der mit `-p` aktiviert wird. In diesem Modus werden nur Dateien verglichen, die für den Erfolg von `buildworld` oder `installworld` essentiell sind.

Tipp

Um im System nach Dateien zu suchen die der Gruppe gehören, die umbenannt oder gelöscht werden soll:

```
# find / -group GID -print
```

Dieses Kommando zeigt alle Dateien an, die der Gruppe *GID* gehören. Dies kann entweder ein Gruppenname oder eine numerische ID sein.

24.7.6. Wechseln Sie in den Single-User-Modus

Sie können das System im Single-User-Modus übersetzen. Bei der Installation des Systems werden viele wichtige Dateien, wie die Standard-Systemprogramme, die Bibliotheken und Include-Dateien, verändert. Sie bringen sich in Schwierigkeiten, wenn Sie diese Dateien auf einem laufenden System verändern, besonders dann, wenn zu dieser Zeit Benutzer auf dem System aktiv sind.

Bei dieser Methode übersetzen Sie das System im Mehrbenutzermodus und wechseln anschließend für die Installation in den Single-User-Modus. Wenn Sie diese Methode benutzen wollen, warten Sie mit den folgenden Schritten, bis der Bau des Systems abgeschlossen ist. Wechseln Sie dann in den Single-User-Modus, um `installkernel` oder `installworld` auszuführen.

Mit dem folgenden Kommando kann ein laufendes System in den Single-User-Modus gebracht werden:

```
# shutdown now
```

Alternativ können Sie das System mit der Option „single user“ in den Single-User-Modus booten. Geben Sie dann die folgenden Befehle am Single-User-Modus Shell-Prompt ein:

```
# fsck -p
# mount -u /
# mount -a -t ufs
# swapon -a
```

Die Kommandos überprüfen die Dateisysteme, hängen `/` wieder beschreibbar ein, hängen dann alle anderen UFS Dateisysteme aus `/etc/fstab` ein und aktivieren den Swap-Bereich.

Anmerkung

Zeigt die CMOS-Uhr die lokale Zeit und nicht GMT an (dies erkennen Sie daran, dass date(1) die falsche Zeit und eine falsche Zeitzone anzeigt), setzen Sie das folgende Kommando ab:

```
# adjkerntz -i
```

Dies stellt sicher, dass die Zeitzone richtig eingestellt ist.

24.7.7. Entfernen Sie /usr/obj

Die neu gebauten Teile des Systems werden in der Voreinstellung unter /usr/obj gespeichert. Die Verzeichnisse dort spiegeln die Struktur unter /usr/src .

Um den make buildworld Prozess zu beschleunigen und Ärger aufgrund von Abhängigkeiten zu vermeiden, können Sie dieses Verzeichnis entfernen.

Einige Dateien unter /usr/obj haben vielleicht die immutable-Option gesetzt, die zuvor mit chflags(1) entfernt werden muss:

```
# cd /usr/obj
# chflags -R noschg *
# rm -rf *
```

24.7.8. Übersetzen der Quellen des Basissystems

24.7.8.1. Sichern der Ausgaben

Es ist ratsam, die Ausgaben von make(1) in einer Datei zu sichern. Wenn etwas schief geht, kann eine Kopie der Fehlermeldung zu einer der FreeBSD-Mailinglisten gesendet werden.

Dazu können Sie einfach das Kommando script(1) benutzen, dem Sie beim Aufruf als Parameter den Dateinamen für die Ausgaben mitgeben. Setzen Sie das Kommando unmittelbar vor dem Neubau ab und geben Sie **exit** ein, wenn der Bau abgeschlossen ist:

```
# script /var/tmp/mw.out
Script started, output file is /var/tmp/mw.out
# make TARGET
… Ausgaben des Kommandos …
# exit
Script done, …
```

Sichern Sie die Ausgaben *nicht* in /tmp, da dieses Verzeichnis beim nächsten Reboot aufgeräumt werden kann. Ein geeigneteres Verzeichnis ist /var/tmp , oder das Heimatverzeichnis von root.

24.7.8.2. Übersetzen des Basissystems

Wechseln Sie in das Verzeichnis, in dem die Quellen liegen (in der Voreinstellung ist das /usr/src):

```
# cd /usr/src
```

Benutzen Sie make(1), um das Basissystem neu zu bauen. Dieses Kommando liest Anweisungen aus einem Makefile, wechles beschreibt, wie die Programme, aus denen FreeBSD besteht, zu bauen sind und in welcher Reihenfolge diese zu bauen sind.

Ein typischer Aufruf von make sieht wie folgt aus:

```
# make -x -DVARIABLE target
```

In diesem Beispiel ist *-x* eine Option, die an make(1) weitergegeben wird. Eine Liste gültiger Optionen finden Sie in make(1).

Das Verhalten eines Makefiles wird von Variablen bestimmt. Mit *-DVARIABLE* setzen Sie eine Variable. Diese Variablen sind dieselben, die auch in /etc/make.conf gesetzt werden, dies ist nur ein alternativer Weg, Variablen zu setzen.

Um zu verhindern, dass die „profiled“ Bibliotheken gebaut werden, rufen Sie make wie folgt auf:

```
# make -DNO_PROFILE  target
```

Dieser Aufruf entspricht dem folgenden Eintrag in `/etc/make.conf` :

```
NO_PROFILE=    true     #    Avoid compiling profiled libraries
```

Jedes `Makefile` definiert einige „Ziele“, die festlegen, was genau zu tun ist. Mit *`target`* wählen Sie eins dieser Ziele aus.

Einige Ziele im `Makefile` werden verwendet, um den Bauprozess in eine Reihe von Einzelschritten zu unterteilen.

Im Regelfall müssen make(1) keine Parameter mitgegeben werden, so dass die Kommandozeile wie folgt aussehen wird:

```
# make target
```

`target` steht dabei für die verschiedenen Ziele. Das erste Ziel sollte immer `buildworld` sein.

Mit `buildworld` wird ein kompletter Baum unterhalb von `/usr/obj` gebaut, der mit `installworld` auf dem System installiert werden kann.

Über separate Optionen zu verfügen, ist aus mehreren Gründen nützlich. Erstens können Sie das System gefahrlos auf einem laufenden System bauen, da die Bauprozedur vom Rest des Systems isoliert ist. Das System lässt sich im Mehrbenutzermodus ohne negative Seiteneffekte bauen. Die Installation mit `installworld` sollte aber immer noch im Single-User-Modus erfolgen.

Zweitens kann NFS benutzt werden, um mehrere Maschinen in einem Netzwerk zu aktualisieren. Um die Maschinen `A`, `B` und `C` zu aktualisieren, lassen Sie `make buildworld` und `make installworld` auf `A` laufen. Auf den Maschinen `B` und `C` können Sie die Verzeichnisse `/usr/src` und `/usr/obj` von `A` einhängen und brauchen dort nur noch `make installworld` auszuführen, um die Bauresultate zu installieren.

Obwohl das Ziel `world` noch existiert, sollte es wirklich nicht mehr benutzt werden.

Benutzen Sie stattdessen:

```
# make buildworld
```

Mit `-j` können Sie `make` anweisen, mehrere Prozesse zu starten. Besonders effektiv ist das auf Mehrprozessor-Systemen. Da aber der Übersetzungsprozess hauptsächlich von I/O statt der CPU bestimmt wird, ist diese Option auch auf Einprozessor-Systemen nützlich.

Auf einem typischen Einprozessor-System können Sie den folgenden Befehl eingeben:

```
# make -j4 buildworld
```

make(1) wird dann bis zu vier Prozesse gleichzeitig laufen lassen. Erfahrungsberichte aus den Mailinglisten zeigen, dass dieser Aufruf typischerweise den besten Geschwindigkeitsgewinn bringt.

Wenn Sie ein Mehrprozessor-System besitzen und SMP im Kernel konfiguriert ist, probieren Sie Werte zwischen 6 und 10 aus.

24.7.8.3. Laufzeiten

Die Laufzeit eines Baus wird von vielen Faktoren beeinflusst, ein aktuelles System benötigt aber etwa zwei Stunden um FreeBSD-STABLE zu bauen. Der Bau von FreeBSD-CURRENT dauert etwas länger.

24.7.9. Übersetzen und Installation des Kernels

Kompilieren Sie einen neuen Kernel, um den vollen Nutzen aus dem System zu ziehen. Praktisch gesehen ist das sogar notwendig, da sich einige Datenstrukturen geändert haben und Programme wie ps(1) und top(1) nur mit einem Kernel zusammen arbeiten, der auch zu dem entsprechenden Quellcode passt.

Am einfachsten und sichersten bauen Sie dazu den `GENERIC` Kernel. Obwohl der `GENERIC` Kernel vielleicht nicht alle Geräte unterstützt, sollte er alles enthalten, um das System in den Single-User-Modus zu booten. Dies ist auch ein guter Test, um zu sehen, dass das System ordnungsgemäß funktioniert. Nachdem das System mit `GENERIC` gebootet wurde und sichergestellt ist, dass das System funktioniert, kann ein neuer Kernel basierend auf einer angepassten Konfigurationsdatei erstellt werden.

In FreeBSD müssen Sie das Basissystem neu bauen, bevor Sie einen neuen Kernel erstellen.

Anmerkung

Verwenden Sie `KERNCONF=`*`MYKERNEL`*, um einen Kernel mit einer vorhandenen, angepassten Konfigurationsdatei zu erstellen:

```
# cd /usr/src
# make buildkernel KERNCONF= MYKERNEL
# make installkernel KERNCONF= MYKERNEL
```

Wenn `kern.securelevel` einen Wert größer als `1` besitzt *und* der Kernel mit `noschg` oder ähnlichen Optionen geschützt ist, müssen Sie `installkernel` im Single-User-Modus ausführen. Andernfalls laufen diese beiden Kommandos problemlos im Mehrbenutzermodus. Weitere Informationen über `kern.securelevel` finden Sie in init(8). Optionen, die auf Dateien gesetzt werden können, werden in chflags(1) detailliert erläutert.

24.7.10. Booten Sie in den Single-User-Modus

Booten Sie in den Single-User-Modus, um zu prüfen ob der neue Kernel funktioniert. Folgen Sie dazu den Anweisungen aus Abschnitt 24.7.6, „Wechseln Sie in den Single-User-Modus“.

24.7.11. Installation des Systems

Nun kann das neue System mit `installworld` installiert werden:

```
# cd /usr/src
# make installworld
```

Anmerkung

Wenn mit `make buildworld` Variablen verwendet werden, müssen dieselben Variablen auch bei `make installworld` angegeben werden. Auf die anderen Optionen trifft das nur bedingt zu: `-j` darf mit `installworld` nicht benutzt werden.

Haben Sie zum Bauen die folgende Kommandozeile verwendet:

```
# make -DNO_PROFILE buildworld
```

dann installieren Sie das Ergebnis mit:

```
# make -DNO_PROFILE installworld
```

Andernfalls würde das System bei der Installation versuchen, die „profiled“ Bibliotheken, die aber gar nicht gebaut wurden, zu installieren.

24.7.12. Aktualisieren der von `make installworld` ausgelassenen Dateien

Neue oder geänderte Konfigurationsdateien aus einigen Verzeichnissen, besonders `/etc`, `/var` und `/usr` werden bei der Installationsprozedur nicht berücksichtigt.

Diese Dateien können einfach mit mergemaster(8) aktualisiert werden. Sichern Sie `/etc` für den Fall, dass während der Aktualisierung etwas schief geht.

24.7.12.1. `mergemaster`

Beigetragen von Tom Rhodes.

mergemaster(8) ist ein Bourne-Shell Skript, das dabei behilflich ist die Unterschiede zwischen den Konfigurationsdateien in `/etc` und denen im Quellbaum unter `/usr/src/etc` zu finden. `mergemaster` ist der empfohlene Weg, die Systemkonfiguration mit dem Quellbaum abzugleichen.

Um zu beginnen, rufen Sie `mergemaster` auf. Ausgehend von `/` wird `mergemaster` einen virtuellen Root-Baum aufbauen und darin die neuen Konfigurationsdateien ablegen. Diese Dateien werden dann mit den auf dem System installierten Dateien verglichen. Unterschiede zwischen den Dateien werden im diff(1)-Format dargestellt. Neue oder geänderte Zeilen werden mit + gekennzeichnet. Zeilen die gelöscht oder ersetzt werden, sind mit - gekennzeichnet. Das Anzeigeformat wird in diff(1) genauer erklärt.

mergemaster(8) zeigt Ihnen jede geänderte Datei an und Sie haben die Wahl, die neue Datei (in `mergemaster` wird sie temporäre Datei genannt) zu löschen, sie unverändert zu installieren, den Inhalt der neuen Datei mit dem Inhalt der alten Datei abzugleichen, oder die diff(1) Ausgabe noch einmal zu sehen.

Wenn Sie die temporäre Datei löschen, geht `mergemaster` davon aus, dass Sie die aktuelle Datei unverändert behalten möchten. Wählen Sie die Option nur dann, wenn Sie keinen Grund sehen, die aktuelle Datei zu ändern.

Wenn Sie die temporäre Datei installieren, wird Ihre aktuelle Datei mit der neuen Datei überschrieben. Sie sollten alle unveränderten Konfigurationsdateien auf diese Weise aktualisieren.

Wenn Sie sich entschließen den Inhalt beider Dateien abzugleichen, wird ein Texteditor aufgerufen, in dem Sie beide Dateien nebeneinander betrachten können. Mit der Taste l übernehmen Sie die aktuelle Zeile der links dargestellten Datei, mit der Taste r übernehmen Sie die Zeile der rechts dargestellten Datei. Das Ergebnis ist eine Datei, die aus Teilen der beiden ursprünglichen Dateien besteht und installiert werden kann. Dieses Verfahren wird gewöhnlich bei veränderten Dateien genutzt.

Haben Sie sich entschieden die Differenzen noch einmal anzuzeigen, zeigt mergemaster(8) dieselbe Ausgabe, die bereits vor der Eingabeaufforderung ausgegeben wurde.

Wenn mergemaster(8) alle Systemdateien abgearbeitet hat, werden weitere Optionen abgefragt. Sie werden unter Umständen gefragt, ob die Passwort-Datei neu gebaut werden soll. Am Ende haben Sie die Möglichkeit, die restlichen temporären Dateien zu löschen.

24.7.12.2. Manueller Abgleich der Konfigurationsdateien

Wenn Sie den Abgleich lieber selbst ausführen wollen, beachten Sie bitte, dass Sie nicht einfach die Dateien aus `/usr/src/etc` nach `/etc` kopieren können. Einige dieser Dateien müssen zuerst *installiert* werden, bevor sie benutzt werden können. Das liegt daran, dass `/usr/src/etc` keine exakte Kopie von `/etc` ist. Zudem gibt es Dateien, die sich in `/etc` befinden aber nicht in `/usr/src/etc` .

Wenn Sie, wie empfohlen, `mergemaster` benutzen, können Sie direkt in den nächsten Abschnitt wechseln.

Am einfachsten ist es, wenn Sie die neuen Dateien in ein temporäres Verzeichnis installieren und sie nacheinander auf Differenzen zu den bestehenden Dateien durchsehen.

Sichern Sie die Inhalte von `/etc`

Es wird empfohlen, zuerst das bestehende `/etc` an einen sicheren Ort zu kopieren:

```
# cp -Rp /etc /etc.old
```

Mit -R wird rekursiv kopiert und -p erhält die Attribute der kopierten Dateien, wie Zugriffszeiten und Eigentümer.

Als nächstes müssen Sie ein temporäres Verzeichnis für die Installation der neuen Dateien in /etc erstellen. /var/tmp/root ist eine gute Wahl:

```
# mkdir /var/tmp/root
# cd /usr/src/etc
# make DESTDIR=/var/tmp/root distrib-dirs distribution
```

Die obigen Kommandos bauen die nötige Verzeichnisstruktur auf und installieren die neuen Dateien in diese Struktur. Unterhalb von /var/tmp/root wurden einige leere Verzeichnisse angelegt, die Sie am besten wie folgt entfernen:

```
# cd /var/tmp/root
# find -d . -type d | xargs rmdir 2>/dev/null
```

Dadurch werden alle leeren Verzeichnisse entfernt. Um die Warnungen über nicht leere Verzeichnisse zu unterdrücken, wurde die Standardfehlerausgabe nach /dev/null umgeleitet.

/var/tmp/root enthält nun alle Dateien, die unterhalb von / installiert werden sollten. Sie müssen nun jede dieser Dateien mit den schon existierenden Dateien des Systems vergleichen.

Einige der installierten Dateien unter /var/tmp/root beginnen mit einem „.“. Verwenden Sie ls -a um sicherzustellen, dass Sie alle derartigen Dateien finden.

Benutzen Sie diff(1), um zwei Dateien zu vergleichen:

```
# diff /etc/shells /var/tmp/root/etc/shells
```

Dieses Kommando zeigt die Unterschiede zwischen der installierten Version von /etc/shells und der neuen Version in /var/tmp/root/etc/shells . Entscheiden Sie anhand der Unterschiede, ob Sie beide Dateien abgleichen, oder die alte Version durch die neue Version ersetzen wollen.

Versehen Sie das temporäre Verzeichnis /var/tmp/root mit einem Zeitstempel

Wenn das System oft neu gebaut wird, muss auch /etc genauso oft aktualisiert werden. Dies kann mit der Zeit ein bisschen mühsam werden.

Um diesen Prozess zu beschleunigen, behalten Sie eine Kopie der Dateien, die zuletzt nach /etc installiert wurden.

1. Folgen Sie der normalen Prozedur um das System zu bauen. Wenn Sie /etc und die anderen Verzeichnisse aktualisieren wollen, geben Sie dem temporären Verzeichnis einen Namen, der das aktuelle Datum enthält.

   ```
   # mkdir /var/tmp/root-20130214
   # cd /usr/src/etc
   # make DESTDIR=/var/tmp/root-20130214 \
       distrib-dirs distribution
   ```

2. Gleichen Sie die Änderungen entsprechend der Anleitung von oben ab. Wenn Sie fertig sind, entfernen Sie das Verzeichnis /var/tmp/root-20130214 *nicht*.

3. Nachdem die neuen Quellen heruntergeladen und gebaut haben, folgen Sie Schritt 1. Erstellen Sie ein neues Verzeichnis mit einem aktuellen Datum. Dieses Beispiel verwendet `/var/tmp/root-20130221` .

4. Vergleichen Sie die Unterschiede, die sich in einer Woche ergeben haben, indem Sie diff(1) rekursiv anwenden:

```
# cd /var/tmp
# diff -r root-20130214 root-20130221
```

   Üblicherweise sind diese Differenzen kleiner, als die Differenzen zwischen `/var/tmp/root-20130221/etc` und `/etc`. Da die angezeigten Differenzen kleiner sind, ist es jetzt einfacher den Abgleich der Dateien in `/etc` durchzuführen.

5. Wenn Sie fertig sind, können Sie das ältere der beiden `/var/tmp/root-*` Verzeichnisse entfernen:

```
# rm -rf /var/tmp/root-20130214
```

6. Wiederholen Sie diesen Prozess jedes Mal wenn Sie Dateien in `/etc` abgleichen müssen.

Benutzen Sie date(1), um die Verzeichnisnamen automatisch zu erzeugen:

```
# mkdir /var/tmp/root-`date "+%Y%m%d"`
```

24.7.13. Das System neu starten

Nachdem Sie sich davon überzeugt haben, dass alle Dateien an der richtigen Stelle sind, starten Sie das System mit shutdown(8) neu:

```
# shutdown -r now
```

Herzlichen Glückwunsch! Sie haben gerade erfolgreich ein FreeBSD System aktualisiert.

Es ist leicht einen Teil des Systems wiederherzustellen, für den Fall, dass Ihnen ein kleiner Fehler unterlaufen ist. Wenn beispielsweise während des Updates oder Abgleichs `/etc/magic` aus Versehen gelöscht wurde, wird file(1) nicht mehr funktionieren. In diesem Fall kann das Problem mit dem folgenden Kommando behoben werden:

```
# cd /usr/src/usr.bin/file
# make all install
```

24.7.14. Fragen

F: Muss ich wirklich immer alles neu bauen, wenn sich etwas geändert hat?

A: Darauf gibt es keine einfache Antwort. Was zu tun ist, hängt von den Änderungen ab. Es lohnt wahrscheinlich nicht, alles neu zu bauen, wenn sich bei einem svn-Lauf nur die folgenden Dateien geändert haben:

```
src/games/cribbage/instr.c
src/games/sail/pl_main.c
src/release/sysinstall/config.c
src/release/sysinstall/media.c
src/share/mk/bsd.port.mk
```

   In diesem Fall können Sie in die entsprechenden Unterverzeichnisse wechseln und dort `make all install` ausführen. Wenn sich allerdings etwas Wichtiges, wie `src/lib/libc/stdlib` , geändert hat, sollten Sie die Welt oder mindestens die statisch gelinkten Teile des Systems neu bauen.

Letztendlich ist das Ihre Entscheidung. Sie sind vielleicht damit zufrieden, das System alle zwei Wochen neu zu bauen und in der Zwischenzeit die anfallenden Änderungen zu sammeln. Wenn Sie sich zutrauen, alle Abhängigkeiten zu erkennen, bauen Sie vielleicht auch nur die geänderten Sachen neu.

Das hängt auch noch davon ab, wie oft Sie ein Update durchführen wollen und ob Sie FreeBSD-STABLE oder FreeBSD-CURRENT benutzen.

F: Der Bau bricht mit vielen Signal 11-Fehlern (oder anderen Signalnummern) ab. Was ist da passiert?

A:

Normalerweise zeigen diese Meldungen Hardwarefehler an. Ein Neubau der Welt ist ein guter Belastungstest für die Hardware und zeigt oft Probleme mit dem Speicher auf. Dies äußert sich darin, dass der Compiler mit seltsamen Signalen abbricht.

Es liegt garantiert ein Hardwarefehler vor, wenn `make` neu gestartet wird und an einer anderen Stelle abbricht.

In diesem Fall können nur einzelne Komponenten des Systems getauscht werden, um zu bestimmen, welche Komponente den Fehler verursacht.

F: Kann `/usr/obj` entfernt werden, wenn ich fertig bin?

A: Kurze Antwort: Ja.

In `/usr/obj` werden alle Dateien abgelegt, die während der Übersetzungsphase erstellt wurden. Dieses Verzeichnis wird in einem der ersten Schritte von `make buildworld` entfernt. Es macht daher wenig Sinn, dieses Verzeichnis zu behalten. Zudem wird ungefähr 2 GB Plattenspeicher freigegeben, wenn dieses Verzeichnis gelöscht wird.

Erfahrene Benutzer können `make buildworld` anweisen, diesen Schritt zu überspringen. Nachfolgende Bauprozeduren werden dadurch erheblich schneller, da die meisten Quelldateien nicht mehr neu übersetzt werden müssen. Dafür können aber subtile Abhängigkeitsprobleme entstehen, die dazu führen, dass der Bau auf merkwürdige Weise abbrechen kann. Dies führt häufig zu unnötigen Diskussionen auf den FreeBSD Mailinglisten, wenn sich jemand über einen kaputten Bau beschwert, aber nicht sieht, dass er Probleme hat, weil er eine Abkürzung genommen hat.

F: Kann ein abgebrochener Bau weitergeführt werden?

A: Das hängt davon ab, wieweit der Bauprozess fortgeschritten ist.

Üblicherweise werden durch `make buildworld` essentielle Werkzeuge, wie gcc(1) und make(1), und die Systembibliotheken neu erstellt. Die neu erstellten Werkzeuge und Bibliotheken werden dann benutzt, um sich selbst noch einmal zu bauen, und wieder installiert. Anschließend wird das Gesamtsystem, einschließlich der normalen Benutzerprogramme wie ls(1) und grep(1), mit den neu erstellten Systemdateien gebaut.

Während der letzten Phase können Sie relativ gefahrlos folgende Kommandos ausführen:

```
… Fehler beheben …
# cd /usr/src
# make -DNO_CLEAN all
```

Diese Variablen verhindern, dass `make buildworld` die vorher erstellten Dateien löscht.

Das Sie sich im letzten Schritt der Bauprozedur befinden, erkennen Sie daran, dass Sie in der Ausgabe von `make buildworld` die folgenden Zeilen finden:

```
--------------------------------------------------------------
Building everything..
--------------------------------------------------------------
```

Wenn diese Meldung nicht angezeigt wird, oder Sie sich nicht sicher sind, dann ist es besser, noch einmal ganz von Vorne anzufangen.

F: Wie kann ich den Bauprozess beschleunigen?

A:
- Bauen Sie im Single-User-Modus.
- Legen Sie `/usr/src` und `/usr/obj` in getrennte Dateisysteme auf unterschiedliche Festplatten. Benutzen Sie nach Möglichkeit auch getrennte Platten-Controller.
- Alternativ können diese Dateisysteme mit ccd(4) auf mehrere Festplatten verteilt werden.
- Deaktivieren Sie den Bau der „profiled“-Bibliotheken, indem Sie `NO_PROFILE=true` in `/etc/make.conf` aufnehmen.
- Benutzen Sie `make` zusammen mit `-j`*n*, um mehrere Prozesse parallel laufen zu lassen. Normalerweise beschleunigt dies den Bauprozess auf Einprozessor- und Mehrprozessorsystemen.
- Das Dateisystem `/usr/src` kann mit der Option `noatime` eingehangen werden. Dies verhindert, dass die Zugriffszeiten der Dateien aktualisiert werden.

```
# mount -u -o noatime /usr/src
```

Warnung

Das Beispiel geht davon aus, dass sich `/usr/src` auf einem separaten Dateisystem befindet. Wenn es Teil des `/usr` Dateisystems ist, muss dieses Dateisystem als Mountpoint angegeben werden.

- Das Dateisystem, in dem sich `/usr/obj` befindet, kann mit `async` eingehangen werden, so dass Schreibzugriffe auf die Platte asynchron stattfinden. Das heißt ein Schreibzugriff ist sofort beendet, die Daten werden allerdings erst einige Sekunden später geschrieben. Dadurch können Schreibzugriffe zusammengefasst werden, was einen erheblichen Geschwindigkeitszuwachs mit sich bringen kann.

Warnung

Beachten Sie, dass dies das Dateisystem anfälliger für Fehler macht. Im Fall eines Stromausfalls besteht eine erhöhte Wahrscheinlichkeit, dass das Dateisystem beim Start der Maschine zerstört ist.

Wenn `/usr/obj` das einzige Verzeichnis auf auf diesem Dateisystem ist, stellt das kein Problem dar. Wenn sich allerdings auf diesem Dateisystem noch andere wertvolle Daten befinden, stellen Sie sicher, dass Sie über aktuelle Sicherungen verfügen.

```
# mount -u -o async /usr/obj
```

Warnung

Ersetzen Sie `/usr/obj` durch den Mountpoint des entsprechenden Dateisystems, wenn es sich nicht auf einem eigenen Dateisystem befindet.

F: Was mache ich, wenn etwas nicht funktioniert?

A: Stellen Sie sicher, dass sich in Ihrer Umgebung keine Reste eines vorherigen Baus befinden:

```
# chflags -R noschg /usr/obj/usr
# rm -rf /usr/obj/usr
# cd /usr/src
# make cleandir
# make cleandir
```

Ja, `make cleandir` muss wirklich zweimal aufgerufen werden.

Danach starten Sie den Bauprozess wieder mit `make buildworld`.

Wenn Sie immer noch Probleme haben, schicken Sie die Fehlermeldungen und die Ausgabe von `uname -a` an die Mailingliste 'Fragen und Antworten zu FreeBSD' <de-bsd-questions@de.FreeBSD.org>. Bereiten Sie sich darauf vor, weitere Fragen zu Ihrer Umgebung zu beantworten.

24.8. Veraltete Dateien, Verzeichnisse und Bibliotheken löschen

Basiert auf Notizen von Anton Shterenlikht.

Aufgrund der ständigen Weiterentwicklung von FreeBSD kann es dazu kommen, dass Dateien und deren Inhalte obsolet werden, weil deren Funktionalität entweder in anderen Dateien implementiert wurde, sich die Versionsnummer der Bibliothek geändert hat oder deren Funktion nicht mehr benötigt wird. Dies kann sowohl Dateien und Verzeichnisse, aber auch Bibliotheken betreffen. Diese veralteten Dateien sollten daher entfernt werden, wenn das System aktualisiert wird. Der Vorteil besteht darin, dass das System von nicht mehr benötigten Dateien befreit wird. Falls die obsolete Bibliothek Sicherheits- oder Stabilitätsprobleme aufweist, sollte das System ebenfalls aktualisiert werden, um das System sicher zu halten und/oder durch die fehlerhafte Bibliothek verursachte Systemabstürze zu vermeiden. Veraltete Dateien, Verzeichnisse und Bibliotheken sind in der Datei `/usr/src/ObsoleteFiles.inc` aufgelistet. Verwenden Sie die folgenden Anweisungen, um diese Dateien während der Systemaktualisierung zu entfernen.

Folgen Sie den Anweisungen von Abschnitt 24.7.1, „Richtig aktualisieren“. Nachdem Sie `make installworld` sowie `mergemaster` erfolgreich ausgeführt haben, überprüfen Sie das System auf veraltete Dateien und Bibliotheken:

```
# cd /usr/src
# make check-old
```

Werden dabei veraltete Dateien gefunden, können diese mit dem folgenden Kommando entfernt werden:

```
# make delete-old
```

Tipp

Weitere interessante *targets* finden Sie in `/usr/src/Makefile` .

Bei jeder Datei wird nachgefragt, ob Sie diese wirklich löschen wollen. Es ist aber auch möglich, alle Dateien automatisch löschen zu lassen. Dies erreichen Sie, indem Sie die Umgebungsvariable `BATCH_DELETE_OLD_FILES` setzen:

```
# make -DBATCH_DELETE_OLD_FILES delete-old
```

Alternativ können Sie auch `yes` einsetzen und somit die Antwort `yes` an die einzelnen Abfragen weiterreichen:

```
# yes | make delete-old
```

Warnung

Das Löschen veralteter Dateien kann dazu führen, dass Programme, die auf diese Dateien angewiesen sind, nicht mehr funktionieren. Dies gilt insbesondere für veraltete Bibliotheken. In den meisten Fällen ist es dann notwendig, Programme, Ports und Bibliotheken, welche die veraltete Bibliothek verwenden, neu zu bauen, bevor Sie den Befehl `make delete-old-libs` ausführen.

Die Ports-Sammlung enthält Werkzeuge, die Ihnen beim Prüfen von Bibliothek-Abhängigkeiten helfen können: sysutils/libchk sowie sysutils/bsdadminscripts.

Veraltete Bibliotheken können zu Konflikten mit neueren Bibliotheken führen und beispielsweise folgende Meldungen verursachen:

```
/usr/bin/ld: warning: libz.so.4, needed by /usr/local/lib/libtiff.so, may conflict with ↺
libz.so.5
/usr/bin/ld: warning: librpcsvc.so.4, needed by /usr/local/lib/libXext.so, may conflict ↺
with librpcsvc.so.5
```

Um diese Probleme zu lösen, müssen Sie zuerst herausfinden, welcher Port die Bibliothek installiert hat:

```
# pkg_info -W  /usr/local/lib/libtiff.so
/usr/local/lib/libtiff.so was installed by package tiff-3.9.4
# pkg_info -W /usr/local/lib/libXext.so
/usr/local/lib/libXext.so was installed by package libXext-1.1.1,1
```

Danach deinstallieren Sie den Port und bauen ihn neu, um ihn danach erneut zu installieren. Dieser Vorgang kann durch den Einsatz von ports-mgmt/portmaster automatisiert werden. Nachdem alle Ports neu gebaut wurden und keine alten alten Bibliotheken mehr verwenden werden, können Sie die alten Bibliotheken endgültig entfernen:

```
# make delete-old-libs
```

24.9. Installation mehrerer Maschinen

Beigetragen von Mike Meyer.

Wenn Sie mehrere Maschinen alle auf dem gleichen Stand halten wollen, ist es eine Verschwendung von Ressourcen, die Quellen auf jeder Maschine vorzuhalten und zu übersetzen. Die Lösung dazu ist, eine Maschine den Großteil der Arbeit durchführen zu lassen und den anderen Maschinen das Ergebnis mit NFS zur Verfügung zu stellen. Dieser Abschnitt zeigt Ihnen wie das geht.

24.9.1. Voraussetzungen

Stellen Sie zuerst eine Liste der Maschinen zusammen, die auf demselben Stand sein sollen. Wir nennen diese Maschinen die *Baugruppe*. Jede dieser Maschinen kann mit einem eigenen Kernel laufen, doch sind die Programme des Userlands auf allen Maschinen gleich. Wählen Sie aus der Baugruppe eine Maschine aus, auf der der Bau durchgeführt wird, den *Bau-Master*. Dies sollte eine Maschine sein, die über die nötigen Ressourcen für `make buildworld` und `make installworld` verfügt. Sie brauchen auch eine *Testmaschine*, auf der Sie die Updates testen, bevor Sie sie in Produktion installieren. Dies sollte eine Maschine, eventuell der Bau-Master, sein, die über einen längeren Zeitraum nicht zur Verfügung stehen kann.

Alle Maschinen der Baugruppe müssen `/usr/obj` und `/usr/src` von derselben Maschine an gleichem Ort einhängen. Idealerweise befinden sich die beiden Verzeichnisse auf dem Bau-Master auf verschiedenen Festplatten, sie können allerdings auch auf dem Bau-Master über NFS zur Verfügung gestellt werden. Wenn Sie mehrere Baugruppen haben, sollte sich `/usr/src` auf einem Bau-Master befinden und über NFS für den Rest der Maschinen zur Verfügung gestellt werden.

Stellen Sie sicher, dass `/etc/make.conf` und `/etc/src.conf` auf allen Maschinen einer Baugruppe mit der Datei des Bau-Masters übereinstimmt. Der Bau-Master muss jeden Teil des Systems bauen, den irgendeine Maschine der Baugruppe benötigt. Auf dem Bau-Master müssen in `/etc/make.conf` alle zu bauenden Kernel mit der Variablen `KERNCONF` bekannt gegeben werden. Geben Sie dabei den Kernel des Bau-Masters zuerst an. Für jeden zu bauenden Kernel muss auf dem Bau-Master die entsprechende Konfigurationsdatei unter `/usr/src/sys/`*`arch`*`/conf` abgelegt werden.

24.9.2. Installation des Basissystems

Bauen Sie auf dem Bau-Master, wie in Abschnitt 24.7.8.2, „Übersetzen des Basissystems“ beschrieben, den Kernel und die Welt, installieren Sie aber nichts. Wechseln Sie auf die Testmaschine und installieren Sie den gerade gebauten Kernel. Wenn diese Maschine `/usr/src` und `/usr/obj` über NFS bekommt, müssen Sie das Netzwerk im Single-User-Modus aktivieren und die beiden Dateisysteme einhängen. Am einfachsten ist dies, wenn Sie auf der Testmaschine vom Mehrbenutzermodus mit `shutdown now` in den Single-User-Modus wechseln. Sie können dann mit der normalen Prozedur den neuen Kernel und das System installieren und anschließend `mergemaster` laufen lassen. Wenn Sie damit fertig sind, können Sie die Maschine wieder in den Mehrbenutzermodus booten.

Nachdem Sie sichergestellt haben, dass die Testmaschine einwandfrei funktioniert, wiederholen Sie diese Prozedur für jede Maschine in der Baugruppe.

24.9.3. Die Ports-Sammlung

Dasselbe Verfahren können Sie auch für die Ports-Sammlung anwenden. Zuerst müssen alle Maschinen einer Baugruppe `/usr/ports` von derselben Maschine über NFS zur Verfügung gestellt bekommen. Setzen Sie dann ein Verzeichnis für die Quellen auf, das sich alle Maschinen teilen. Dieses Verzeichnis können Sie in `/etc/make.conf` mit der Variablen `DISTDIR` angeben. Das Verzeichnis sollte für den Benutzer beschreibbar sein, auf den der Benutzer `root` vom NFS Subsystem abgebildet wird. Jede Maschine sollte noch `WRKDIRPREFIX` auf ein lokales Bauverzeichnis setzen. Wenn Sie vorhaben, Pakete zu bauen und zu verteilen, sollten Sie `PACKAGES` auf ein Verzeichnis mit den gleichen Eigenschaften wie `DISTDIR` setzen.

Kapitel 25. DTrace

Written by Tom Rhodes.
Übersetzt von Benedict Reuschling und Christoph Sold.

25.1. Überblick

DTrace, auch bekannt als Dynamic Tracing, wurde von Sun™ als ein Werkzeug zur Analyse von Performance-Problemen in Produktiv- und Entwicklungssystemen entwickelt. Zusätzlich zur Diagnose von Performance-Problemen kann DTrace auch verwendet werden, um bei der Untersuchung und Behebung von unerwartetem Verhalten im FreeBSD-Kernel und den Anwenderprogrammen zu helfen.

DTrace ist ein bemerkenswertes Werkzeug zur Profilerstellung, mit einer beeindruckenden Palette von Eigenschaften zur Diagnose von Systemereignissen. Es kann auch dazu verwendet werden, bestehende Skripte ablaufen zu lassen, um einen Nutzen aus deren Möglichkeiten zu ziehen. Nutzer können mittels der Programmiersprache D von DTrace ihre eigenen Hilfsmittel schreiben, was es ermöglicht, die eigenen Profile nach Ihren Bedürfnissen anzupassen.

Die DTrace-Implementierung in FreeBSD bietet experimentelle Unterstützung für DTrace im Userland. Userland DTrace erlaubt es Anwendern, function boundary tracing für Anwendungsprogramme über den `pid`-Provider hinweg vorzunehmen und um statische Sonden in Anwendungsprogramme für die spätere Aufzeichnung einzufügen. Manche Ports, wie beispielsweise databases/postgres-server und lang/php5 besitzen eine DTrace-Option, um statische Sonden zu aktivieren. FreeBSD 10.0-RELEASE besitzt eine verhältnismässig gute Unterstützung für DTrace im Anwendungsbereich, wird jedoch noch nicht als produktionsreif angesehen. Insbesondere ist es möglich, dass Programme während der Aufzeichnung abstürzen.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie Folgendes wissen:

- Was DTrace ist und welche Funktionen es zur Verfügung stellt.
- Unterschiede zwischen der Solaris™ DTrace Implementierung und derjenigen, die FreeBSD bereitstellt.
- Wie man DTrace auf FreeBSD aktiviert und verwendet.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- UNIX® und FreeBSD Grundlagen verstehen (Kapitel 4, *Grundlagen des UNIX Betriebssystems*).
- Vertraut sein mit Sicherheitsaspekten und wie diese FreeBSD betreffen (Kapitel 14, *Sicherheit*).

Warnung

Diese Funktion ist als experimentell anzusehen. Manche Einstellungen enthalten möglicherweise nicht alle Funktionalitäten, andere Teile könnten gar nicht laufen. Mit der Zeit, wenn diese Funktion als für den Produktivbetrieb geeignet erscheint, wird auch diese Dokumentation geändert, um diesem Umstand gerecht zu werden.

25.2. Unterschiede in der Implementierung

Obwohl DTrace in FreeBSD sehr ähnlich zu dem in Solaris™ ist, existieren doch Unterschiede. Der Hauptunterschied besteht darin, dass in FreeBSD DTrace als eine Menge von Kernelmodulen implementiert ist und DTrace nicht verwendet werden kann, bis diese Module geladen wurden. Um alle nötigen Module zu laden, geben Sie ein:

```
# kldload dtraceall
```

Beginnend mit FreeBSD 10.0-RELEASE werden die Module automatisch geladen, sobald `dtrace` aufgerufen wird.

FreeBSD verwendet die Kerneloption `DDB_CTF` , um die Unterstützung im Kernel für das Laden von CTF-Daten aus Kernelmodulen und dem Kernel selbst zu ermöglichen. CTF ist das Compact C Type Format von Solaris™, welches eine reduzierte Form von Debug-Informationen kapselt, ähnlich zu DWARF und den antiken Stabs. Diese CTF-Daten werden dem Binärcode von den `ctfconvert` und `ctfmerge` Befehlen den Werkzeugen zum Bauen des Systems hinzugefügt. Das `ctfconvert`-Dienstprogramm parst die vom Compiler erstellten DWARF ELF Debug-Abschnitte und `ctfmerge` vereint CTF ELF-Abschnitte aus Objekten, entweder in ausführbare Dateien oder Shared-Libraries.

Einige Provider in FreeBSD unterscheiden sich von der Solaris™-Implementierung. Am deutlichsten wird das beim `dtmalloc`-Provider, welcher das Aufzeichnen von `malloc()` nach Typen im FreeBSD-Kernel ermöglicht. Manche der Provider in Solaris™ wie `cpc` und `mib` sind in FreeBSD nicht vorhanden. Diese können in zukünftigen FreeBSD-Versionen auftauchen. Weiterhin sind manche der Provider in beiden Betriebssystemen nicht zueinander kompatibel, in dem Sinne daß deren Sonden unterschiedliche Argumenttypen aufweisen. Dadurch können D-Skripte, die unter Solaris™ geschrieben wurden, evtl. unter FreeBSD funktionieren oder auch nicht, umgekehrt ist das genauso.

In FreeBSD darf DTrace wegen unterschiedlicher Sicherheitskonzepte nur von `root` verwendet werden. Solaris™ besitzt ein paar Audit-Funktionen auf den unteren Ebenen, die noch nicht in FreeBSD implementiert sind. Deshalb kann nur `root` auf `/dev/dtrace/dtrace` zugreifen.

Zum Schluss muss noch erwähnt werden, dass die DTrace-Software unter die CDDL Lizenz fällt. Die `Common Development and Distribution License` wird von FreeBSD mitgeliefert, sehen Sie sich dazu `/usr/src/cddl/contrib/opensolaris/OPENSOLARIS.LICENSE` an, oder lesen Sie die Online-Version unter `http://www.opensolaris.org/os/licensing.`

Diese Lizenz bedeutet, dass ein FreeBSD-Kernel mit den DTrace-Optionen immer noch BSD-lizenziert ist; allerdings tritt die CDDL in Kraft, wenn Module in Binärform vertrieben werden oder die Binärdateien geladen werden.

25.3. Die DTrace Unterstützung aktivieren

In FreeBSD 9.2 und 10.0 ist die Unterstützung von DTrace im `GENERIC` -Kernel bereits eingebaut. Nutzer von früheren Versionen sollten die folgenden Zeilen in eine eigene Kernelkonfigurationsdatei einfügen und den Kernel mittels der Anleitung in Kapitel 9, *Konfiguration des FreeBSD-Kernels* neu übersetzen:

```
options         KDTRACE_HOOKS
options         DDB_CTF
options         DEBUG=-g
```

Besitzer der AMD64-Architektur werden wahrscheinlich noch die folgende Zeile zur Kernelkonfigurationsdatei hinzufügen:

```
options         KDTRACE_FRAME
```

Diese Option liefert die Unterstützung für die FBT-Eigenschaft. DTrace wird auch ohne diese Option funktionieren; jedoch wird dann Function Boundary Tracing nur eingeschränkt unterstützt.

Sobald FreeBSD in den neuen Kernel gebootet oder die DTrace-Kernelmodule mittels `kldload dtraceall` geladen wurden, benötigt das System Unterstützung für die Korn-Shell, da DTrace mehrere Dienstprogramme enthält, die in `ksh` implementiert sind. Vergewissern Sie sich, dass das Paket oder der Port shells/ksh93 installiert ist. Es ist auch möglich, diese Werkzeuge unter shells/pdksh oder shells/mksh laufen zu lassen.

Zum Schluss sollten Sie noch den aktuellen DTrace-Werkzeugsatz beschaffen. Die DTrace-Werkzeugsammlung enthält gebrauchsfertige Skripte, um Systeminformationen zu sammeln. Es gibt Skripte zum Überprüfen von offenen Dateien, Speicher- und CPU-Gebrauch und noch viel mehr. FreeBSD 10 installiert ein paar dieser Skripte in `/usr/share/dtrace`. Für andere FreeBSD-Versionen oder um die volle DTrace-Werkzeugsammlung zu installieren, verwenden Sie den sysutils/DTraceToolkit Port oder das Paket.

Anmerkung

Die Skripte in `/usr/share/dtrace` wurden speziell für FreeBSD portiert. Nicht alle Skripte in der DTrace-Werkzeugsammlung werden in FreeBSD unverändert funktionieren und manche Skript benötigen einigen Aufwand, damit diese auf FreeBSD funktionieren.

Der DTrace-Werkzeugsatz beinhaltet viele Skripte in der speziellen Sprache von DTrace. Diese Sprache wird die D-Sprache genannt und ist sehr ähnlich zu C++. Eine detaillierte Beschreibung dieser Sprache sprengt den Rahmen dieses Dokuments. In `http://wikis.oracle.com/display/DTrace/Documentation` wird diese Sprache ausführlich beschrieben.

25.4. DTrace verwenden

DTrace-Skripte bestehen aus einer Liste von einer oder mehreren *Sonden* oder Instrumentationspunkten, an denen jede Sonde mit einer Aktion verknüpft ist. Jedesmal, wenn die Bedingung für eine Sonde zutrifft, wird die verknüpfte Aktion ausgeführt. Beispielsweise könnte eine Aktion ausgeführt werden, wenn eine Datei geöffnet, ein Prozess gestartet oder eine Codezeile ausgeführt wird. Die Aktion könnte die Protokollierung von Informationen sein oder die Änderung von Kontextvariablen. Das Lesen und Schreiben von Kontextvariablen erlaubt es den Sonden, Informationen auszutauschen und kooperativ die Korrelation bestimmter Ereignisse zu analysieren.

Um alle Sonden anzuzeigen, kann der Administrator nun den folgenden Befehl eingeben:

```
# dtrace -l | more
```

Jede Sonde besitzt eine `ID`, einen `PROVIDER` (dtrace oder fbt), ein `MODULE` und einen `FUNCTION NAME`. Lesen Sie für weitere Informationen zu diesem Kommando dtrace(1).

Die Beispiele in diesem Abschnitt geben einen Überblick, wie man zwei dieser voll funktionsfähigen Skripte aus der DTrace-Werkzeugsammlung verwendet: die Skripte `hotkernel` und `procsystime`.

Das `hotkernel` Skript wurde entworfen, um zu identifizieren, welche Funktion die meiste Kernelzeit beansprucht. Es wird es Ausgaben ähnlich der Folgenden produzieren:

```
# cd /usr/share/dtrace/toolkit
# ./hotkernel
Sampling... Hit Ctrl-C to end.
```

Verwenden Sie wie angegeben die Tastenkombination Ctrl+C drücken, um den Prozess zu stoppen. Nach dem Abbruch wird das Skript eine Liste von Kernelfunktionen und Zeitmessungen ausgeben, aufsteigend sortiert nach den Zeiten:

```
kernel`_thread_lock_flags                               2   0.0%
0xc1097063                                              2   0.0%
kernel`sched_userret                                    2   0.0%
kernel`kern_select                                      2   0.0%
kernel`generic_copyin                                   3   0.0%
kernel`_mtx_assert                                      3   0.0%
kernel`vm_fault                                         3   0.0%
kernel`sopoll_generic                                   3   0.0%
kernel`fixup_filename                                   4   0.0%
kernel`_isitmyx                                         4   0.0%
kernel`find_instance                                    4   0.0%
kernel`_mtx_unlock_flags                                5   0.0%
kernel`syscall                                          5   0.0%
kernel`DELAY                                            5   0.0%
0xc108a253                                              6   0.0%
```

```
kernel`witness_lock                                     7   0.0%
kernel`read_aux_data_no_wait                            7   0.0%
kernel`Xint0x80_syscall                                 7   0.0%
kernel`witness_checkorder                               7   0.0%
kernel`sse2_pagezero                                    8   0.0%
kernel`strncmp                                          9   0.0%
kernel`spinlock_exit                                   10   0.0%
kernel`_mtx_lock_flags                                 11   0.0%
kernel`witness_unlock                                  15   0.0%
kernel`sched_idletd                                   137   0.3%
0xc10981a5                                          42139  99.3%
```

Dieses Skript funktioniert auch mit Kernelmodulen. Um diese Eigenschaft zu verwenden, starten Sie das Skript mit `-m`:

```
# ./hotkernel -m
Sampling... Hit Ctrl-C to end.
^C
MODULE                                              COUNT   PCNT
0xc107882e                                              1   0.0%
0xc10e6aa4                                              1   0.0%
0xc1076983                                              1   0.0%
0xc109708a                                              1   0.0%
0xc1075a5d                                              1   0.0%
0xc1077325                                              1   0.0%
0xc108a245                                              1   0.0%
0xc107730d                                              1   0.0%
0xc1097063                                              2   0.0%
0xc108a253                                             73   0.0%
kernel                                                874   0.4%
0xc10981a5                                         213781  99.6%
```

Das `procsystime` Skript fängt die Systemaufruf-Zeiten für eine gegebene Prozess-ID (PID) oder einen Prozessnamen ab und gibt diese aus. Im folgenden Beispiel wurde eine neue Instanz von `/bin/csh` erzeugt. Dann wurde `procsystime` ausgeführt und verbleibt so, während ein paar Befehle in die andere Instanz von `csh` eingegeben werden. Dies sind die Ergebnisse dieses Versuchs:

```
# ./procsystime -n csh
Tracing... Hit Ctrl-C to end...
^C

Elapsed Times for processes csh,

         SYSCALL          TIME (ns)
          getpid               6131
       sigreturn               8121
           close              19127
           fcntl              19959
             dup              26955
         setpgid              28070
            stat              31899
       setitimer              40938
           wait4              62717
        sigaction              67372
     sigprocmask             119091
    gettimeofday             183710
           write             263242
          execve             492547
           ioctl             770073
           vfork            3258923
      sigsuspend            6985124
            read         3988049784
```

Wie aus der Ausgabe ersichtlich ist, verbraucht der `read()`-Systemaufruf die meiste Zeit in Nanosekunden, während der Systemaufruf `getpid()` hingegen am schnellsten läuft.

Teil IV. Netzwerke

FreeBSD ist eins der meist benutzten Betriebssysteme für leistungsfähige Netzwerkserver. Die Kapitel in diesem Teil behandeln die nachstehenden Themen:

- Serielle Datenübertragung
- PPP und PPP over Ethernet
- Electronic-Mail
- Den Betrieb von Netzwerkdiensten
- Firewalls
- Weiterführende Netzwerkthemen

Diese Kapitel sollten Sie lesen, wenn Sie die Informationen darin benötigen. Sie brauchen Sie nicht in einer bestimmten Reihenfolge zu lesen, noch müssen Sie die Kapitel lesen, bevor Sie anfangen, FreeBSD zu benutzen.

Inhaltsverzeichnis

Kapitel 26. Serielle Datenübertragung

Übersetzt von Martin Heinen.

26.1. Übersicht

UNIX® Systeme unterstützten schon immer die serielle Datenübertragung. Tatsächlich wurden Ein- und Ausgaben auf den ersten UNIX® Maschinen über serielle Leitungen durchgeführt. Seit der Zeit, in der ein durchschnittlicher Terminal aus einem seriellen Drucker mit 10 Zeichen/Sekunde und einer Tastatur bestand, hat sich viel verändert. Dieses Kapitel behandelt einige Möglichkeiten, serielle Datenübertragung unter FreeBSD zu verwenden.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie Folgendes wissen:

- Wie Sie Terminals an ein FreeBSD-System anschließen.
- Wie Sie sich mit einem Modem auf entfernte Rechner einwählen.
- Wie Sie entfernten Benutzern erlauben, sich mit einem Modem in ein FreeBSD-System einzuwählen.
- Wie Sie ein FreeBSD-System über eine serielle Konsole booten.

Bevor Sie dieses Kapitel lesen, sollten Sie

- einen angepassten Kernel konfigurieren und installieren können.
- Berechtigungen und Prozesse unter FreeBSD verstehen.
- Zugriff auf die Handbücher der seriellen Komponenten haben, die mit FreeBSD verwendet werden sollen.

26.2. Begriffe und Hardware

Die folgenden Begriffe werden oft verwendet, wenn es um serielle Kommunikation geht:

bps
: Bits pro Sekunde (bps) ist die Einheit für die Übertragungsgeschwindigkeit.

DEE (DTE)
: Eine Datenendeinrichtung (Data Terminal Equipment) ist einer der beiden Endpunkte bei der seriellen Kommunikation. Zum Beispiel ein Computer.

DÜE (DCE)
: Datenübertragungseinrichtung (Data Communications Equipment) ist der andere Endpunkt bei der seriellen Kommunikation. Typischerweise ein Modem.

RS-232
: Der originale Standard, der serielle Datenübertragung definiert. Er wird heutzutage als TIA-232 bezeichnet.

In diesem Abschnitt wird der Begriff „Baud“ nicht für Übertragungsgeschwindigkeiten gebraucht. Baud bezeichnet elektrische Zustandswechsel pro Zeiteinheit, die Taktfrequenz, während „bps“ der *richtige* Begriff für die Übertragungsgeschwindigkeit ist.

Um ein Modem oder einen Terminal an ein FreeBSD-System anzuschließen, muss der Computer über eine serielle Schnittstelle verfügen. Zusätzlich wird das passende Kabel benötigt, um das Gerät mit der Schnittstelle zu verbinden. Benutzer, die mit seriellen Geräten und den nötigen Kabeln schon vertraut sind, können diesen Abschnitt überspringen.

26.2.1. Kabel und Schnittstellen

Es gibt verschiedene serielle Kabel. Die zwei häufigsten sind Nullmodemkabel und Standard-RS-232-Kabel. Die Dokumentation der Hardware sollte beschreiben, welcher Kabeltyp benötigt wird.

Ein Nullmodemkabel verbindet einige Signale, wie die Betriebserde, eins zu eins, andere Signale werden getauscht: Die Sende- und Empfangsleitungen werden zum Beispiel gekreuzt.

Nullmodemkabel für die Anbindung eines Terminals können auch selbst hergestellt werden. Die folgende Tabelle enthält die Signalnamen von RS-232C sowie die Pinbelegung für einen Stecker vom Typ DB-25. Obwohl der Standard eine direkte Verbindung von Pin 1 zu Pin 1 (*Protective Ground*) vorschreibt, ist diese in vielen Fällen nicht vorhanden. Einige Terminals benötigen nur die Pins 2, 3 und 7 für eine korrekte Funktion, während andere eine unterschiedliche Konfiguration als die in den folgenden Beispielen gezeigte benötigen.

Tabelle 26.1. Nullmodemkabel vom Typ DB-25-zu-DB-25

Signal	Pin #		Pin #	Signal
SG	7	verbunden mit	7	SG
TD	2	verbunden mit	3	RD
RD	3	verbunden mit	2	TD
RTS	4	verbunden mit	5	CTS
CTS	5	verbunden mit	4	RTS
DTR	20	verbunden mit	6	DSR
DTR	20	verbunden mit	8	DCD
DSR	6	verbunden mit	20	DTR
DCD	8	verbunden mit	20	DTR

Die folgenden zwei Schemata werden heutzutage ebenfalls häufig eingesetzt:

Tabelle 26.2. Nullmodemkabel vom Typ DB-9-zu-DB-9

Signal	Pin #		Pin #	Signal
RD	2	verbunden mit	3	TD
TD	3	verbunden mit	2	RD
DTR	4	verbunden mit	6	DSR
DTR	4	verbunden mit	1	DCD
SG	5	verbunden mit	5	SG
DSR	6	verbunden mit	4	DTR
DCD	1	verbunden mit	4	DTR
RTS	7	verbunden mit	8	CTS
CTS	8	verbunden mit	7	RTS

Tabelle 26.3. Nullmodemkabel vom Typ DB-9-zu-DB-25

Signal	Pin #		Pin #	Signal
RD	2	verbunden mit	2	TD
TD	3	verbunden mit	3	RD
DTR	4	verbunden mit	6	DSR
DTR	4	verbunden mit	8	DCD
SG	5	verbunden mit	7	SG

Signal	Pin #		Pin #	Signal
DSR	6	verbunden mit	20	DTR
DCD	1	verbunden mit	20	DTR
RTS	7	verbunden mit	5	CTS
CTS	8	verbunden mit	4	RTS

Anmerkung

Wird ein Pin eines Kabels mit zwei Pins des anderen Kabels verbunden, werden dazu in der Regel zuerst die beiden Pins mit einem kurzem Draht verbunden. Danach wird dieser Draht mit dem Pin des anderen Endes verbunden.

Die eben besprochenen Schemata scheinen die beliebtesten zu sein. Weitere Varianten verbinden SG mit SG, TD mit RD, RTS und CTS mit DCD, DTR mit DSR, und umgekehrt.

Ein Standard-RS-232C-Kabel verbindet alle Signale direkt. Das Signal „Transmitted Data“ wird mit dem Signal „Transmitted Data“ der Gegenstelle verbunden. Dieses Kabel wird benötigt, um ein Modem mit einem FreeBSD-System zu verbinden. Manche Terminals benötigen dieses Kabel ebenfalls.

Über serielle Schnittstellen werden Daten zwischen dem FreeBSD-System und dem Terminal übertragen. Dieser Abschnitt beschreibt die verschiedenen Schnittstellen und wie sie unter FreeBSD angesprochen werden.

Da es verschiedene Schnittstellen gibt, sollte vor dem Kauf oder Selbstbau eines Kabels sichergestellt werden, dass dieses zu den Schnittstellen des Terminals und des FreeBSD-Systems passt.

Die meisten Terminals besitzen DB-25-Stecker. Personal Computer haben DB-25- oder DB-9-Stecker. Eine serielle Multiportkarte hat vielleicht RJ-12- oder RJ-45-Anschlüsse.

Die Dokumentation der Geräte sollte Aufschluss über den Typ der benötigten Anschlüsse geben. Oft hilft es, wenn Sie sich den Anschluss einfach ansehen.

Unter FreeBSD wird jede serielle Schnittstelle (Port) über einen Eintrag in `/dev` angesprochen. Es gibt dort zwei verschiedene Einträge:

- Schnittstellen für eingehende Verbindungen werden `/dev/ttyu`*`N`* genannt. Dabei ist *`N`* die Nummer der Schnittstelle, deren Zählung bei Null beginnt. Allgemein wird diese Schnittstelle für Terminals benutzt. Diese Schnittstelle funktioniert nur, wenn ein „Data Carrier Detect“ Signal (DCD) vorliegt.
- Für ausgehende Verbindungen wird `/dev/cuad`*`N`* verwendet. Dieser Port wird normalerweise nur von Modems genutzt. Er kann allerdings auch für Terminals benutzt werden, die das „Data Carrier Detect“ Signal nicht unterstützen.

Wenn ein Terminal an die erste serielle Schnittstelle (`COM1`) angeschlossen ist, wird er über `/dev/ttyu0` angesprochen. Wenn er an der zweiten seriellen Schnittstelle (COM2) angeschlossen ist, verwenden Sie `/dev/ttyu1` , usw.

26.2.2. Kernelkonfiguration

In der Voreinstellung benutzt FreeBSD vier serielle Schnittstellen, die unter MS-DOS® als `COM1`, `COM2`, `COM3` und `COM4` bekannt sind. Momentan unterstützt FreeBSD einfache Multiportkarten, wie bspw. die BocaBoard 1008 und 2016 und bessere wie die von Digiboard und Stallion Technologies. In der Voreinstellung sucht der Kernel allerdings nur nach den Standardanschlüssen.

Um zu überprüfen, ob der Kernel die seriellen Schnittstellen erkennt, achten Sie auf die Meldungen beim Booten, oder schauen sich diese später mit `/sbin/dmesg` an. Achten Sie auf Meldungen die mit `uart` beginnen:

```
# /sbin/dmesg | grep 'uart'
```

Wenn der Kernel nicht alle seriellen Schnittstellen erkennt, müssen Sie `/boot/device.hints` konfigurieren. Wenn Sie diese Datei editieren, können Sie die Einträge für Geräte, die auf dem System nicht vorhanden sind, auskommentieren oder komplett entfernen.

Anmerkung

`port IO_COM1` ist ein Ersatz für `port 0x3f8`, `IO_COM2` bedeutet `port 0x2f8`, `IO_COM3` bedeutet `port 0x3e8` und `IO_COM4` steht für `port 0x2e8`. Die angegebenen IO-Adressen sind genau wie die Interrupts 4, 3, 5 und 9 üblich für serielle Schnittstellen. Beachten Sie, dass sich normale serielle Schnittstellen auf ISA-Bussen *keine* Interrupts teilen können. Multiportkarten besitzen zusätzliche Schaltkreise, die es allen 16550As auf der Karte erlauben, sich einen oder zwei Interrupts zu teilen.

26.2.3. Gerätedateien

Die meisten Geräte im Kernel werden durch Gerätedateien in `/dev` angesprochen. Die `sio` Geräte werden durch `/dev/ttyuN` für eingehende Verbindungen und durch `/dev/cuadN` für ausgehende Verbindungen angesprochen. Zum Initialisieren der Geräte stellt FreeBSD die Dateien `/dev/ttyuN.init` und `/dev/cuadN.init` zur Verfügung. Zusätzlich existieren Dateien für das Sperren von Gerätedateien (*Locking*). Dabei handelt es sich um die Dateien `/dev/ttyuN.lock` und `/dev/cuadN.lock`. Diese Dateien werden benutzt, um Kommunikationsparameter beim Öffnen eines Ports vorzugeben. Für Modems, die zur Flusskontrolle `RTS/CTS` benutzen, kann damit `crtscts` gesetzt werden. Die Geräte `/dev/ttyldN` und `/dev/cualaN` (locking devices) werden genutzt, um bestimmte Parameter festzuschreiben und vor Veränderungen zu schützen. Weitere Informationen zu Terminals finden Sie in termios(4), sio(4) erklärt die Dateien zum Initialisieren und Sperren der Geräte, stty(1) beschreibt schließlich Terminal-Einstellungen.

26.2.4. Konfiguration der seriellen Schnittstelle

Anwendungen benutzen normalerweise die Geräte `ttyuN` oder `cuadN`. Das Gerät besitzt einige Voreinstellungen für Terminal-I/O, wenn es von einem Prozess geöffnet wird. Mit dem folgenden Kommando können Sie sich diese Einstellungen ansehen:

```
# stty -a -f /dev/ttyu1
```

Wenn diese Einstellungen verändert werden, bleiben sie nur solange wirksam, bis das Gerät geschlossen wird. Wenn das Gerät danach wieder geöffnet wird, sind die Voreinstellungen wieder wirksam. Um die Voreinstellungen dauerhaft zu ändern, öffnen Sie das Gerät, das zum Initialisieren dient und verändern dessen Einstellungen. Um beispielsweise für `ttyu5` den `CLOCAL` Modus, 8-Bit Kommunikation und `XON/XOFF` Flusssteuerung einzuschalten, setzen Sie das folgende Kommando ab:

```
# stty -f /dev/ttyu5.init clocal cs8 ixon ixoff
```

In `/etc/rc.d/rc.serial` werden die systemweiten Voreinstellungen für serielle Geräte vorgenommen.

Um zu verhindern, dass Einstellungen von Anwendungen verändert werden, können Sie die Geräte zum Festschreiben von Einstellungen („locking devices“) benutzen. Wenn sie beispielsweise die Geschwindigkeit von `ttyu5` auf 57600 bps festlegen wollen, benutzen Sie das folgende Kommando:

```
# stty -f /dev/ttyld5 57600
```

Eine Anwendung, die `ttyu5` öffnet, kann nun nicht mehr die Geschwindigkeit ändern und muss 57600 bps benutzen.

Die Geräte zum Initialisieren und Festschreiben von Einstellungen sollten selbstverständlich nur von `root` beschreibbar sein.

26.3. Terminals

Beigetragen von Sean Kelly.

Wenn Sie sich nicht an der Konsole oder über ein Netzwerk an ein FreeBSD-System anmelden können, sind Terminals ein bequemer und kostengünstiger Weg, um auf ein System zuzugreifen. Dieser Abschnitt beschreibt wie Sie Terminals mit FreeBSD benutzen.

Das ursprüngliche UNIX® System besaß keine Konsolen. Zum Anmelden und Starten von Programmen wurden stattdessen Terminals benutzt, die an den seriellen Schnittstellen des Rechners angeschlossen waren.

Die Möglichkeit, über eine serielle Schnittstelle eine Anmeldesitzung herzustellen, existiert heute noch in fast jedem UNIX®-artigen Betriebssystem, einschließlich FreeBSD. Der Einsatz eines Terminals, das an einem freien seriellen Port angeschlossen ist, ermöglicht es dem Benutzer sich anzumelden und dort jedes Textprogramm zu starten, das normalerweise an der Konsole oder in einem `xterm` Fenster ausgeführt wird.

Viele Terminals können an einem FreeBSD-System angeschlossen werden. Ein alter Computer kann als Terminal an ein leistungsfähiges FreeBSD-System angeschlossen werden. Damit kann ein Einzelarbeitsplatz in ein leistungsfähiges Mehrbenutzersystem verwandelt werden.

FreeBSD unterstützt drei Arten von Anschlüssen:

Dumb-Terminals
: Dumb-Terminals (unintelligente Datenstationen) sind Geräte, die über die serielle Schnittstelle mit einem Rechner verbunden werden. Sie werden „unintelligent“ genannt, weil sie nur Text senden und empfangen und keine Programme laufen lassen können. Alle Programme, wie Texteditoren, Compiler oder Spiele befinden sich auf dem Rechner, der mit dem Terminal verbunden ist.

  Es gibt viele Dumb-Terminals, die von verschiedenen Herstellern produziert werden, und so gut wie jeder der verschiedenen Terminals sollte mit FreeBSD zusammenarbeiten. Manche High-End Geräte verfügen sogar über Grafikfähigkeiten, die allerdings nur von spezieller Software genutzt werden kann.

  Dumb-Terminals sind in Umgebungen beliebt, in denen keine Grafikanwendungen benötigt werden.

Computer, die als Terminal fungieren
: Jeder Computer kann die Funktion eines Dumb-Terminals, der ja nur Text senden und empfangen kann, übernehmen. Dazu wird lediglich das richtige Kabel benötigt und eine *Terminalemulation*, die auf dem Computer läuft.

  Diese Konfiguration ist sehr nützlich. Wenn ein Benutzer zum Beispiel gerade an der FreeBSD-Konsole arbeitet, kann ein anderer Benutzer einen weniger leistungsstarken Computer, der als Terminal mit dem FreeBSD-System verbunden ist, benutzen, um dort gleichzeitig im Textmodus zu arbeiten.

  Bereits im Basissystem sind mindestens zwei Werkzeuge vorhanden, die Sie zur Arbeit über eine serielle Konsole einsetzen können: cu(1) sowie tip(1).

  Um sich von einem FreeBSD-System aus über eine serielle Verbindung mit einem anderen System zu verbinden, geben Sie folgenden Befehl ein:

  ```
  # cu -l serial-port-device
  ```

  „serial-port-device“ ist hier der Name der Gerätedatei, die einer bestimmten seriellen Schnittstelle des Systems zugewiesen ist. Diese Gerätedateien werden `/dev/cuadN` genannt.

  Der Buchstabe „N“ muss dabei durch die Nummer des seriellen Ports des Systems ersetzt werden.

Anmerkung

Beachten Sie, dass die Numerierung dieser Daten unter FreeBSD mit Null und nicht mit Eins beginnt. Die Schnittstelle „COM1“ entspricht daher `/dev/cuad0` unter FreeBSD.

Anmerkung

In der Ports-Sammlung finden sich weitere Programme, wie beispielsweise comms/minicom, mit denen eine Verbindung über eine serielle Schnittstelle hergestellt werden kann.

X-Terminals
: X-Terminals sind die ausgereiftesten der verfügbaren Terminals. Sie werden nicht mit der seriellen Schnittstelle sondern mit einem Netzwerk, wie dem Ethernet, verbunden. Diese Terminals sind auch nicht auf den Textmodus beschränkt, sondern können jede X-Anwendung darstellen.

  Die Einrichtung und Verwendung von X-Terminals wird in diesem Abschnitt *nicht* beschrieben.

26.3.1. Konfiguration

Dieser Abschnitt beschreibt, wie Sie ein FreeBSD-System konfigurieren müssen, um sich an einem Terminal anzumelden. Dabei wird vorausgesetzt, dass der Kernel bereits die serielle Schnittstelle, die mit dem Terminal verbunden ist, unterstützt. Weiterhin sollte der Terminal schon angeschlossen sein.

Der `init` Prozess ist für das Initialisieren des Systems und den Start von Prozessen zum Zeitpunkt des Systemstarts verantwortlich. Unter anderem liest `init /etc/ttys` ein und startet für jeden verfügbaren Terminal einen `getty` Prozess. `getty` wiederum fragt beim Anmelden den Benutzernamen ab und startet `login`.

Um Terminals auf einem FreeBSD-System einzurichten, führen Sie folgenden Schritte als `root` durch:

1. Fügen Sie einen Eintrag in `/etc/ttys` für die serielle Schnittstelle aus `/dev` ein, falls dieser nicht bereits vorhanden ist.
2. Geben Sie `/usr/libexec/getty` als auszuführendes Programm an. Als Parameter für *`getty`* geben Sie den passenden Verbindungstyp aus `/etc/gettytab` an.
3. Geben Sie den Terminaltyp an.
4. Aktivieren Sie den Anschluss.
5. Geben Sie die Sicherheit des Anschlusses an.
6. Veranlassen Sie `init /etc/ttys` erneut zu lesen.

Optional können Sie in `/etc/gettytab` auch einen auf Ihre Zwecke angepassten Terminaltyp erstellen. gettytab(5) und getty(8) enthalten dazu weitere Informationen.

26.3.1.1. Hinzufügen eines Eintrags in `/etc/ttys`

In `/etc/ttys` werden alle Terminals aufgeführt, an denen eine Anmeldung auf dem FreeBSD-System möglich ist. Hier findet sich zum Beispiel ein Eintrag für die erste virtuelle Konsole `/dev/ttyv0` , der es Benutzern ermöglicht, sich dort anzumelden. Die Datei enthält weitere Einträge für andere virtuelle Konsolen, serielle Schnittstellen und Pseudoterminals. Um einen Terminal zu konfigurieren, fügen Sie einen Eintrag für den Namen des Gerätes aus `/dev` ohne das Präfix `/dev` hinzu. Zum Beispiel wird `/dev/ttyv0` als `ttyv0` aufgeführt.

In der Voreinstellung enthält `/etc/ttys` Einträge für die ersten vier seriellen Schnittstellen: `ttyu0` bis `ttyu3`. Wird an eine von diesen Schnittstellen ein Terminal angeschlossen, braucht in dieser Datei kein weiter Eintrag hinzugefügt werden.

Beispiel 26.1. Einträge in `/etc/ttys` hinzufügen

Dieses Beispiel konfiguriert zwei Terminals: Einen Wyse-50 und einen alten 286 IBM PC, der mit Procomm einen VT-100 Terminal emuliert. Der Wyse-Terminal ist mit der zweiten seriellen Schnittstelle verbunden und der 286 mit der sechsten seriellen Schnittstelle, einem Anschluss auf einer Multiportkarte. Die entsprechenden Einträge in `/etc/ttys` würden dann wie folgt aussehen:

```
ttyu1❶  "/usr/libexec/getty std.38400"❷  wy50❸  on❹  insecure❺
ttyu5   "/usr/libexec/getty std.19200"  vt100  on  insecure
```

❶ Das erste Feld gibt normalerweise den Namen der Gerätedatei aus `/dev` an.

❷ Im zweiten Feld wird das auszuführende Kommando, normal ist das getty(8), angegeben. `getty` initialisiert und öffnet die Verbindung, setzt die Geschwindigkeit und fragt den Benutzernamen ab. Danach führt es login(1) aus.

`getty` akzeptiert einen optionalen Parameter auf der Kommandozeile, den Verbindungstyp, der die Eigenschaften der Verbindung, wie die Geschwindigkeit und Parität, festlegt. Die Typen und die damit verbundenen Eigenschaften liest `getty` aus `/etc/gettytab`.

`/etc/gettytab` enthält viele Einträge sowohl für neue wie auch alte Terminalverbindungen. Die meisten Einträge, die mit `std` beginnen, sollten mit einem festverdrahteten Terminal funktionieren. Für jede Geschwindigkeit zwischen 110 bps und 115200 bps gibt es einen `std` Eintrag. Weitere Informationen dazu finden Sie in gettytab(5).

Wenn Sie den Verbindungstyp in `/etc/ttys` eintragen, stellen Sie sicher, dass die Kommunikationseinstellungen auch mit denen des Terminals übereinstimmen.

In diesem Beispiel verwendet der Wyse-50 keine Parität und 38400 bps, der 286 PC benutzt ebenfalls keine Parität und arbeitet mit 19200 bps.

❸ Das dritte Feld gibt den Terminaltyp an, der normalerweise mit diesem Anschluss verbunden ist. Für Einwählverbindungen wird oft `unknown` oder `dialup` benutzt, da sich die Benutzer praktisch mit beliebigen Terminals oder Emulatoren anmelden können. Bei festverdrahteten Terminals ändert sich der Typ nicht, so dass in diesem Feld ein richtiger Typ aus der termcap(5) Datenbank angegeben werden kann.

In diesem Beispiel benutzt der Wyse-50 den entsprechenden Typ aus termcap(5), der 286 PC wird als VT-100, den er ja emuliert, angegeben.

❹ Das vierte Feld gibt an, ob der Anschluss aktiviert werden soll. Ist das Feld auf `on` gesetzt, startet `init` das Programm, das im zweiten Feld angegeben ist. Normalerweise ist dies `getty`. Wenn das Feld auf `off`gesetzt wird, wird `getty` nicht ausgeführt und folglich kann sich niemand an dem betreffenden Terminal anmelden.

❺ Das letzte Feld gibt die Sicherheit des Anschlusses an. Wenn hier `secure` angegeben wird, darf sich `root`, oder jeder Account mit der UID `0` über diese Verbindung anmelden. Wenn `insecure` angegeben wird, dürfen sich nur unprivilegierte Benutzer anmelden. Diese können später mit su(1) oder einem ähnlichen Mechanismus zu `root` wechseln.

Es wird dringend empfohlen `insecure` zu verwenden, sogar für Terminals hinter verschlossenen Türen. Es ist ganz einfach sich mit `su` anzumelden, wenn Superuser-Rechte benötigt werden.

26.3.1.2. `init` zwingen, `/etc/ttys` erneut zu lesen

Nachdem Änderungen in `/etc/ttys` vorgenommen wurden, schicken Sie `init` ein SIGHUP-Signal (hangup), um es zu veranlassen, seine Konfigurationsdatei neu zu lesen:

```
# kill -HUP 1
```

Anmerkung

Da `init` immer der erste Prozess auf einem System ist, besitzt es immer die Prozess-ID `1`.

Wenn alles richtig eingerichtet ist, alle Kabel angeschlossen und die Terminals eingeschaltet sind, sollte für jeden Terminal ein `getty` Prozess laufen und auf jedem Terminal sollte eine Anmeldeaufforderung zu sehen sein.

26.3.2. Fehlersuche

Selbst wenn Sie den Anweisungen akribisch gefolgt sind, kann es immer noch zu Fehlern beim Einrichten eines Terminals kommen. Hier eine Liste der häufigsten Symptome, sowie einige mögliche Lösungen:

Wenn kein Anmeldeprompt erscheint, stellen Sie sicher, dass der Terminal verbunden und eingeschaltet ist. Wenn ein PC als Terminal fungiert, überprüfen Sie, dass die Terminalemulation auf den richtigen Schnittstellen läuft.

Stellen Sie sicher, dass Sie das richtige Kabel verwenden und dass das Kabel fest mit dem Terminal und dem FreeBSD-Rechner verbunden ist.

Stellen Sie sicher, dass die Einstellungen für die Geschwindigkeit (bps) und Parität auf dem FreeBSD-System und dem Terminal gleich sind. Wenn der Terminal einen Bildschirm besitzt, überprüfen Sie die richtige Einstellung von Helligkeit und Kontrast. Wenn der Terminal druckt, stellen Sie die ausreichende Versorgung mit Papier und Tinte sicher.

Überprüfen Sie mit `ps`, dass der `getty` Prozess für den Terminal läuft:

```
# ps -axww|grep getty
```

Für jeden Terminal sollte ein Eintrag vorhanden sein. Aus dem folgenden Beispiel ist zu erkennen, dass `getty` auf der zweiten seriellen Schnittstelle `tyyd1` läuft und den Verbindungstyp `std.38400` aus `/etc/gettytab` benutzt:

```
22189  d1  Is+    0:00.03 /usr/libexec/getty std.38400 ttyu1
```

Wenn `getty` nicht läuft, überprüfen Sie, ob der Anschluss in `/etc/ttys` aktiviert ist. Denken Sie daran `kill -HUP 1` auszuführen, nachdem `/etc/ttys` geändert wurde.

Wenn `getty` läuft, aber der Terminal immer noch kein Anmeldeprompt ausgibt, oder am Anmeldeprompt nichts eingegeben werden kann, kann es sein, dass der Terminal oder Kabel keinen Hardware-Handshake unterstützt. Ändern Sie dann den Eintrag `std.38400` in `/etc/ttys` zu `3wire.38400` . Nachdem Sie `/etc/ttys` geändert haben, setzen Sie `kill -HUP 1` ab. Der Eintrag `3wire` besitzt ähnliche Eigenschaften wie der Eintrag `std`, ignoriert aber den Hardware-Handshake. Wenn Sie den Eintrag `3wire` verwenden, muss vielleicht die Geschwindigkeit verkleinert oder die Software-Flusssteuerung aktiviert werden, um Pufferüberläufe zu vermeiden.

Wenn nur unverständliche Zeichen erscheinen, stellen Sie sicher, dass die Einstellungen für die Geschwindigkeit (bps) und Parität auf dem FreeBSD-System und dem Terminal gleich sind. Kontrollieren Sie den `getty` Prozess und stellen Sie sicher, dass der richtige Verbindungstyp aus `/etc/gettytab` benutzt wird. Wenn das nicht der Fall ist, editieren Sie `/etc/ttys` und setzen das Kommando `kill-HUP 1` ab.

Wenn Zeichen doppelt und eingegebene Passwörter im Klartext erscheinen, stellen Sie den Terminal oder die Terminalemulation von „half duplex“ oder „local echo“ auf „full duplex“ um.

26.4. Einwählverbindungen

Beigetragen von Guy Helmer.
Mit Anmerkungen von Sean Kelly.

Das Einrichten von Einwählverbindungen auf FreeBSD-Systemen ähnelt dem Anschließen von Terminals, nur dass anstelle eines Terminals ein Modem verwendet wird. FreeBSD unterstützt sowohl externe als auch interne Modems.

Externe Modems sind für Einwählverbindungen besser geeignet, da sie die Konfiguration in nicht flüchtigem RAM speichern können. Zudem verfügen Sie über Leuchtanzeigen, die den Status wichtiger RS-232 Signale anzeigen.

Interne Modems verfügen normalerweise nicht über nicht flüchtiges RAM und lassen sich meist nur über DIP-Schalter konfigurieren. Selbst wenn ein internes Modem Leuchtanzeigen besitzt, sind diese meist schwer einzusehen, wenn das Modem eingebaut ist.

Mit einem externen Modem muss das passende Kabel verwendet werden. Ein Standard RS-232C Kabel, bei dem die folgenden Signale miteinander verbunden sind, sollte ausreichen:

Tabelle 26.4. Signalnamen

Abkürzung	Bedeutung
RD	Received Data
TD	Transmitted Data
DTR	Data Terminal Ready
DSR	Data Set Ready
DCD	Data Carrier Detect (dadurch erkennt RS-232 das Signal *Received Line*)
SG	Signal Ground
RTS	Request to Send
CTS	Clear to Send

Ab Geschwindigkeiten von 2400 bps benötigt FreeBSD die Signale RTS und CTS für die Flusssteuerung. Das Signal CD zeigt an, ob ein Träger vorliegt, das heißt ob die Verbindung aufgebaut ist oder beendet wurde. DTR zeigt an, dass das Gerät betriebsbereit ist. Es gibt einige Kabel, bei denen nicht alle nötigen Signale verbunden sind. Wenn Probleme dieser Art auftreten, dass zum Beispiel die Sitzung nicht beendet wird, obwohl die Verbindung beendet wurde, kann das an einem solchen Kabel liegen.

Wie andere UNIX® Betriebssysteme auch, benutzt FreeBSD Hardwaresignale, um festzustellen, ob ein Anruf beantwortet wurde, eine Verbindung beendet wurde, oder um die Verbindung zu schließen und das Modem zurückzusetzen. FreeBSD vermeidet es, dem Modem Kommandos zu senden, oder den Statusreport des Modems abzufragen.

26.4.1. Schnittstellenbausteine

FreeBSD unterstützt EIA RS-232C (CCITT V.24) serielle Schnittstellen, die auf den NS8250, NS16450, NS16550 oder NS16550A Bausteinen basieren. Die Bausteine der Serie 16550 verfügen über einen 16 Byte großen Puffer, der als FIFO angelegt ist. Wegen Fehler in der FIFO-Logik kann der Puffer in einem 16550 Baustein allerdings nicht genutzt werden, das heißt der Baustein muss als 16450 betrieben werden. Bei allen Bausteinen ohne Puffer und dem 16550 Baustein muss jedes Byte einzeln von dem Betriebssystem verarbeitet werden, was Fehler bei hohen Geschwindigkeiten oder großer Systemlast erzeugt. Es sollten daher nach Möglichkeit serielle Schnittstellen, die auf 16550A Bausteinen basieren, eingesetzt werden.

26.4.2. Überblick

Wie bei Terminals auch, startet `init` für jede serielle Schnittstelle, die eine Einwählverbindung zur Verfügung stellt, einen `getty` Prozess. Wenn das Modem beispielsweise an `/dev/ttyu0` angeschlossen ist, sollte in der Ausgabe von `ps ax` eine Zeile wie die folgende erscheinen:

```
 4850 ??  I      0:00.09 /usr/libexec/getty V19200 ttyu0
```

Wenn sich ein Benutzer einwählt und die Verbindung aufgebaut ist, zeigt das Modem dies durch das CD Signal (Carrier Detect) an. Der Kernel merkt, dass ein Signal anliegt und weist `getty` an, die Schnittstelle zu öffnen. Dann sendet `getty` das Anmeldeprompt mit der ersten für die Verbindung vereinbarten Geschwindigkeit und wartet auf eine Antwort. Wenn die Antwort unverständlich ist, weil zum Beispiel die Geschwindigkeit des Modems von `getty`s Geschwindigkeit abweicht, versucht `getty` die Geschwindigkeit solange anzupassen, bis es eine verständliche Antwort erhält.

Nachdem der Benutzer seinen Benutzernamen eingegeben hat, führt `getty` `/usr/bin/login` aus, welches das Passwort abfragt und danach die Shell des Benutzers startet.

26.4.3. Konfigurationsdateien

Drei Konfigurationsdateien in `/etc` steuern, ob eine Einwahl in das FreeBSD-System möglich ist. `/etc/gettytab`, konfiguriert den `/usr/libexec/getty` Dæmon. In `/etc/ttys` wird festgelegt, auf welchen Schnittstellen `/sbin/init` einen `getty` Prozess startet. Schließlich bietet `/etc/rc.d/serial` die Möglichkeit, Schnittstellen zu initialisieren.

Es gibt zwei Ansichten darüber, wie Modems für Einwählverbindungen unter UNIX® zu konfigurieren sind. Zum einen kann die Geschwindigkeit zwischen dem Modem und dem Computer fest eingestellt werden. Sie ist damit unabhängig von der Geschwindigkeit, mit der sich der entfernte Benutzer einwählt. Dies hat den Vorteil, dass der entfernte Benutzer das Anmeldeprompt sofort bekommt. Der Nachteil bei diesem Verfahren ist, dass das System die tatsächliche Geschwindigkeit der Verbindung nicht kennt. Damit können bildschirmorientierte Programme wie Emacs ihren Bildschirmaufbau nicht an langsame Verbindungen anpassen, um die Antwortzeiten zu verbessern.

Die andere Möglichkeit besteht darin, die Geschwindigkeit der RS-232 Schnittstelle des lokalen Modems an die Geschwindigkeit des entfernten Modems anzupassen. Bei einer V.32bis (14400 bps) Verbindung kann das lokale Modem die RS-232 Schnittstelle mit 19200 bps betreiben, während bei einer Verbindung mit 2400 bps die RS-232 Schnittstelle mit 2400 bps betrieben wird. Da `getty` die Verbindungsgeschwindigkeit des Modems nicht kennt, startet es den Anmeldevorgang mit der Ausgabe von `login:` und wartet auf eine Antwort. Wenn der Benutzer der Gegenstelle nun nur unverständliche Zeichen erhält, muss er solange Enter drücken, bis das Anmeldeprompt erscheint. Solange die Geschwindigkeiten nicht übereinstimmen, sind die Antworten der Gegenstelle für `getty` ebenfalls unverständlich. In diesem Fall wechselt `getty` zur nächsten Geschwindigkeit und gibt wieder `login:` aus. In aller Regel erhält der Benutzer der Gegenstelle nach ein bis zwei Tastendrücken eine erkennbare Anmeldeaufforderung. Diese Anmeldeprozedur sieht nicht so sauber wie die Methode mit einer festen Geschwindigkeit aus, bietet dem Benutzer einer langsamen Verbindung allerdings den Vorteil, dass sich bildschirmorientierte Programme an die Geschwindigkeit anpassen können.

Im Folgenden wird die Konfiguration für beide Methoden besprochen, doch die Methode der angepassten Geschwindigkeit wird bei der Diskussion bevorzugt.

26.4.3.1. `/etc/gettytab`

Mit `/etc/gettytab` wird getty(8) im Stil von termcap(5) konfiguriert. Das Format dieser Datei und die Bedeutung der Einträge wird in gettytab(5) beschrieben.

Wenn die Modemgeschwindigkeit vorgeben wird, sollten Anpassungen in `/etc/gettytab` nicht erforderlich sein.

Wenn jedoch die Geschwindigkeit angepasst werden soll, erstellen Sie einen Eintrag in `/etc/gettytab`, um `getty` die Geschwindigkeit für das Modem mitzuteilen. Für ein 2400 bps Modem kann der vorhandene `D2400` Eintrag benutzt werden.

```
#
# Fast dialup terminals, 2400/1200/300 rotary (can start either way)
#
D2400|d2400|Fast-Dial-2400:\
        :nx=D1200:tc=2400-baud:
3|D1200|Fast-Dial-1200:\
        :nx=D300:tc=1200-baud:
5|D300|Fast-Dial-300:\
        :nx=D2400:tc=300-baud:
```

Wird ein Modem mit einer höheren Geschwindigkeit eingesetzt, müssen weitere Einträge in `/etc/gettytab` erstellt werden. Dieses Beispiel zeigt einen Eintrag für ein 14400 bps Modem mit einer Geschwindigkeit bis zu 19200 bps:

```
#
# Additions for a V.32bis Modem
#
um|V300|High Speed Modem at 300,8-bit:\
        :nx=V19200:tc=std.300:
un|V1200|High Speed Modem at 1200,8-bit:\
        :nx=V300:tc=std.1200:
uo|V2400|High Speed Modem at 2400,8-bit:\
        :nx=V1200:tc=std.2400:
up|V9600|High Speed Modem at 9600,8-bit:\
        :nx=V2400:tc=std.9600:
uq|V19200|High Speed Modem at 19200,8-bit:\
        :nx=V9600:tc=std.19200:
```

Die damit erzeugten Verbindungen verwenden 8 Bit und keine Parität.

Im obigen Beispiel startet die Geschwindigkeit bei 19200 bps (eine V.32bis Verbindung) und geht dann über 9600 bps (V.32), 400 bps, 1200 bps und 300 bps wieder zurück zu 19200 bps. Das Schlüsselwort `nx=` (*next table*) sorgt für das zyklische Durchlaufen der Geschwindigkeiten. Jede Zeile zieht zudem noch mit `tc=` (*table continuation*) die Vorgabewerte für die jeweilige Geschwindigkeit an.

Wenn Sie ein 28800 bps Modem besitzen und/oder Kompression mit einem 14400 bps Modem benutzen wollen, brauchen Sie höhere Geschwindigkeiten als 19200 bps. Das folgende Beispiel startet mit 57600 bps:

```
#
# Additions for a V.32bis or V.34 Modem
# Starting at 57600 bps
#
vm|VH300|Very High Speed Modem at 300,8-bit:\
        :nx=VH57600:tc=std.300:
vn|VH1200|Very High Speed Modem at 1200,8-bit:\
        :nx=VH300:tc=std.1200:
vo|VH2400|Very High Speed Modem at 2400,8-bit:\
        :nx=VH1200:tc=std.2400:
vp|VH9600|Very High Speed Modem at 9600,8-bit:\
        :nx=VH2400:tc=std.9600:
vq|VH57600|Very High Speed Modem at 57600,8-bit:\
        :nx=VH9600:tc=std.57600:
```

Wenn Sie eine langsame CPU oder ein stark ausgelastetes System besitzen und sich kein 16550A im System befindet, erhalten Sie bei 57600 bps vielleicht sio Fehlermeldungen der Form „silo overflow“.

26.4.3.2. `/etc/ttys`

`/etc/ttys` wurde bereits in Beispiel 26.1, „Einträge in `/etc/ttys` hinzufügen“ besprochen. Die Konfiguration für Modems ist ähnlich, allerdings braucht `getty` ein anderes Argument und es muss ein anderer Terminaltyp angegeben werden. Der Eintrag für beide Methoden (feste und angepasste Geschwindigkeit) hat die folgende Form:

```
ttyu0   "/usr/libexec/getty xxx"   dialup on
```

Das erste Feld der obigen Zeile gibt die Gerätedatei für diesen Eintrag an. `ttyu0` bedeutet, dass `getty` mit `/dev/ttyu0` arbeitet. Das zweite Feld `"/usr/libexec/getty xxx"` gibt das Kommando an, das `init` für dieses Gerät startet (*xxx* wird durch einen passenden Eintrag aus `/etc/gettytab` ersetzt). Die Vorgabe für den Terminaltyp, hier `dialup`, wird im dritten Feld angegeben. Das vierte Feld, `on`, zeigt `init` an, dass die Schnittstelle aktiviert ist. Im fünften Feld könnte noch `secure` angegeben werden, um Anmeldungen von `root` zu erlauben, doch sollte das wirklich nur für physikalisch sichere Terminals, wie die Systemkonsole, aktiviert werden.

Die Vorgabe für den Terminaltyp, `dialup` im obigen Beispiel, hängt von lokalen Gegebenheiten ab. Traditionell wird `dialup` für Einwählverbindungen verwendet, so dass die Benutzer in ihren Anmeldeskripten den Terminaltyp auf ihren Terminal abstimmen können, wenn der Typ auf `dialup` gesetzt ist. Wenn Sie nur VT102 Terminals oder Emulatoren einsetzen, können Sie den Terminaltyp hier auch fest auf `vt102` setzen.

Nachdem `/etc/ttys` geändert wurde, muss `init` ein HUP Signal schicken, damit es die Datei wieder einliest:

```
# kill -HUP 1
```

Stellen Sie sicher, dass das Modem richtig konfiguriert und angeschlossen ist, bevor Sie das Signal an `init` schicken.

Das Argument von `getty` muss in diesem Fall eine feste Geschwindigkeit vorgeben. Der Eintrag für ein Modem, das fest auf 19200 bps eingestellt ist, könnte wie folgt aussehen:

```
ttyu0   "/usr/libexec/getty std.19200"   dialup on
```

Wenn das Modem auf eine andere Geschwindigkeit eingestellt ist, setzen Sie anstelle von `std.19200` einen passenden Eintrag der Form `std.speed` ein. Stellen Sie sicher, dass dies auch ein gültiger Verbindungstyp aus `/etc/gettytab` ist.

Das Argument von `getty` muss hier auf einen der Einträge aus `/etc/gettytab` zeigen, der zu einer Kette von Einträgen gehört, die die zu probierenden Geschwindigkeiten beschreiben. Wenn Sie dem obigen Beispiel gefolgt sind und zusätzliche Einträge in `/etc/gettytab` erzeugt haben, können Sie die folgende Zeile verwenden:

```
ttyu0   "/usr/libexec/getty V19200"   dialup on
```

26.4.3.3. `/etc/rc.d/serial`

Modems, die höhere Geschwindigkeiten unterstützen, zum Beispiel V.32, V.32bis und V.34 Modems, benutzen Hardware-Flusssteuerung (`RTS/CTS`). Für die entsprechenden Schnittstellen können Sie die Flusssteuerung mit `stty` in `/etc/rc.d/serial` einstellen.

Um beispielsweise die Hardware-Flusssteuerung für die Geräte zur Ein- und Auswahl der zweiten seriellen Schnittstelle (`COM2`) zu aktivieren, benutzen Sie die Dateien zur Initialisierung der entsprechenden Geräte und fügen die folgenden Zeilen in `/etc/rc.d/serial` hinzu:

```
# Serial port initial configuration
stty -f /dev/ttyu1.init crtscts
stty -f /dev/cuad1.init crtscts
```

26.4.4. Modemkonfiguration

Für ein Modem, das seine Konfiguration in nicht flüchtigem RAM speichert, wird ein Terminalprogramm wie Telix unter MS-DOS® oder `tip` unter FreeBSD benötigt, um die Parameter einzustellen. Verbinden Sie sich mit derselben Geschwindigkeit, die `getty` zuerst benutzen würde, mit dem Modem und treffen Sie folgende Einstellungen:

- DCD ist eingeschaltet, wenn das Trägersignal des entfernten Modems erkannt wird.
- Im Betrieb liegt DTR an. Bei einem Verlust von DTR legt das Modem auf und setzt sich zurück.
- CTS Flusssteuerung ist für ausgehende Daten aktiviert.

- XON/XOFF Flusssteuerung ist ausgeschaltet.
- RTS Flusssteuerung ist für eingehende Daten aktiviert.
- Keine Rückmeldungen ausgeben.
- Die Echo-Funktion ist deaktiviert.

Lesen Sie die Dokumentation für das Modem, um herauszufinden welche Befehle und/oder DIP-Schalterstellungen benötigt werden.

Für ein externes 14400 U.S. Robotics® Sportster® gelten zum Beispiel die folgenden Befehle:

```
ATZ
AT&C1&D2&H1&I0&R2&W
```

Bei dieser Gelegenheit können Sie auch gleich andere Einstellungen, zum Beispiel ob Sie V42.bis und/oder MNP5 Kompression benutzen wollen, an Ihrem Modem vornehmen.

Bei einem externen 14400 U.S. Robotics® Sportster® müssen Sie auch noch einige DIP-Schalter einstellen. Die folgenden Einstellungen können verwendet werden:

- Schalter 1: OBEN – DTR normal
- Schalter 2: N/A (Rückmeldungen als Text/numerische Rückmeldungen)
- Schalter 3: OBEN – Keine Rückmeldungen ausgeben
- Schalter 4: UNTEN – Echo-Funktion aus
- Schalter 5: OBEN – Rufannahme aktiviert
- Schalter 6: OBEN – Carrier Detect normal
- Schalter 7: OBEN – Einstellungen aus dem NVRAM laden
- Schalter 8: N/A (Smart Mode/Dumb Mode)

Für Einwählverbindungen sollten die Rückmeldungen deaktiviert sein, da sonst `getty` dem Modem das Anmeldeprompt `login:` schickt und das Modem im Kommandomodus das Prompt wieder ausgibt (Echo-Funktion) oder eine Rückmeldung gibt. Das führt dann zu einer länglichen und fruchtlosen Kommunikation zwischen dem Modem und `getty`.

Die Geschwindigkeit zwischen Modem und Computer muss auf einen festen Wert eingestellt werden. Mit einem externen 14400 U.S. Robotics® Sportster® Modem setzen die folgenden Kommandos die Geschwindigkeit auf den Wert der Datenendeinrichtung fest:

```
ATZ
AT&B1&W
```

In diesem Fall muss die Geschwindigkeit der seriellen Schnittstelle des Modems der eingehenden Geschwindigkeit angepasst werden. Für ein externes 14400 U.S. Robotics® Sportster® Modem erlauben die folgenden Befehle eine Anpassung der Geschwindigkeit der seriellen Schnittstelle für Verbindungen, die keine Fehlerkorrektur verwenden:

```
ATZ
AT&B2&W
```

Verbindungen mit Fehlerkorrektur (V.42, MNP) verwenden die Geschwindigkeit der Datenendeinrichtung.

26.4.4.1. Überprüfen der Modemkonfiguration

Die meisten Modems verfügen über Kommandos, die die Konfiguration des Modems in lesbarer Form ausgeben. Auf einem externen 14400 U.S. Robotics® Sportster® zeigt `ATI5` die Einstellungen im nicht flüchtigen RAM an. Um die wirklichen Einstellungen unter Berücksichtigung der DIP-Schalter zu sehen, benutzen Sie `ATZ` gefolgt von `ATI4`.

Wenn Sie ein anderes Modem benutzen, schauen Sie bitte in der Dokumentation des Modems nach, wie Sie die Konfiguration des Modems überprüfen können.

26.4.5. Fehlersuche

Bei Problemen können Sie die Einwählverbindung anhand der folgenden Punkte überprüfen:

Schließen Sie das Modem an das FreeBSD-System an und booten Sie das System. Wenn das Modem über Statusindikatoren verfügt, überprüfen Sie, ob der DTR Indikator leuchtet, wenn das Anmeldeprompt erscheint. Dies zeigt an, dass das FreeBSD-System einen `getty` Prozess auf der entsprechenden Schnittstelle gestartet hat und das Modem auf einkommende Verbindungen wartet.

Wenn der DTR-Indikator nicht leuchtet, melden Sie sich an dem FreeBSD-System an und überprüfen mit `ps ax`, ob FreeBSD einen `getty`-Prozess auf der entsprechenden Schnittstelle gestartet hat:

```
  114 ??  I      0:00.10 /usr/libexec/getty V19200 ttyu0
  115 ??  I      0:00.10 /usr/libexec/getty V19200 ttyu1
```

Wenn das Modem noch keinen Anruf entgegengenommen hat und Sie stattdessen die folgende Zeile sehen

```
  114 d0  I      0:00.10 /usr/libexec/getty V19200 ttyu0
```

bedeutet dies, dass `getty` die Schnittstelle schon geöffnet hat und zeigt Kabelprobleme oder eine falsche Modemkonfiguration an, da `getty` die Schnittstelle erst dann öffnen kann, wenn das CD Signal (Carrier Detect) vom Modem anliegt.

Wenn Sie keine `getty`-Prozesse auf den gewünschten `ttyuN` Ports finden, untersuchen Sie `/etc/ttys` auf Fehler. Suchen Sie auch in `/var/log/messages` nach Meldungen von `init` oder `getty`. Wenn Sie dort Meldungen finden, sollten Sie noch einmal die beiden Konfigurationsdateien `/etc/ttys` und `/etc/gettytab` nach Fehlern durchsehen. Überprüfen Sie auch, ob die Gerätedateien `/dev/ttyuN` vorhanden sind.

Versuchen Sie als nächstes, sich in das System einzuwählen. Auf dem entfernten System stellen Sie bitte die folgenden Kommunikationsparameter ein: 8 Bit, keine Parität, ein Stop-Bit. Wenn kein Anmeldeprompt erscheint oder nur unleserliche Zeichen, drücken Sie mehrmals, in Abständen von ungefähr einer Sekunde, Enter. Wenn Sie immer noch nicht die `login:` Meldung sehen, schicken Sie ein `BREAK` Kommando. Wenn Sie zur Einwahl ein Highspeed-Modem benutzen, verwenden Sie eine feste Geschwindigkeit auf der seriellen Schnittstelle des Modems.

Wenn jetzt immer noch kein Anmeldeprompt erscheint, überprüfen Sie nochmals `/etc/gettytab` und stellen sicher, dass:

- der Verbindungstyp in `/etc/ttys` zu einem gültigen Eintrag in `/etc/gettytab` gehört.
- jeder der `nx=` Einträge in `gettytab` gültig ist und
- jeder `tc=` Eintrag auf einen gültigen Eintrag in `gettytab` verweist.

Wenn das Modem am FreeBSD-System auf einen eingehenden Anruf nicht antwortet, stellen Sie sicher, dass das Modem so konfiguriert ist, dass es einen Anruf beantwortet, wenn DTR anliegt. Wenn das Modem Statusindikatoren besitzt, können Sie das Anliegen von DTR anhand der Leuchten überprüfen.

Wenn Sie alles schon mehrfach überprüft haben und es immer noch noch nicht funktioniert, versuchen Sie es zu einem späteren Zeitpunkt erneut. Wenn es immer noch nicht funktioniert, können Sie eine Mail an die Mailingliste

'Fragen und Antworten zu FreeBSD' <de-bsd-questions@de.FreeBSD.org> schicken, in der Sie Ihr Modem und Ihr Problem beschreiben.

26.5. Verbindungen nach Außen

Die folgenden Ratschläge beschreiben, wie Sie mit einem Modem eine Verbindung zu einem anderen Computer herstellen. Dies können Sie nutzen, um sich auf einem entfernten Computer anzumelden.

Weiterhin ist diese Art von Verbindungen nützlich, wenn PPP mal nicht funktioniert. Wenn Sie zum Beispiel eine Datei mit FTP übertragen wollen und das über PPP gerade nicht möglich ist, melden Sie sich auf dem entfernten Rechner an und führen dort die FTP-Sitzung durch. Die Dateien können danach mit zmodem auf den lokalen Rechner übertragen werden.

26.5.1. Ein Hayes Modem benutzen

Es gibt einen eingebauten, allgemeinen Hayes Wähler in `tip`. Verwenden Sie `at=hayes` in `/etc/remote`.

Der Hayes-Treiber ist nicht schlau genug, um ein paar der erweiterten Funktionen von neueren Modems, bspw. `BUSY`, `NO DIALTONE` oder `CONNECT 115200` zu nutzen. Schalten Sie diese Nachrichten mit Hilfe von `ATX0&W` ab, wenn Sie `tip` benutzen.

Der Anwahl-Timeout von `tip` beträgt 60 Sekunden. Das Modem sollte weniger verwenden, oder `tip` denkt, dass ein Kommunikationsfehler vorliegt. Versuchen Sie es mit `ATS7=45&W`.

26.5.2. AT-Befehle benutzen

Erstellen Sie einen `direct` Eintrag in `/etc/remote`. Wenn das Modem zum Beispiel an der ersten seriellen Schnittstelle, `/dev/cuad0`, angeschlossen ist, dann fügen Sie die folgende Zeile hinzu:

```
cuad0:dv=/dev/cuad0:br#19200:pa=none
```

Verwenden Sie die höchste bps-Rate, die das Modem in der `br` Fähigkeit unterstützt. Geben Sie dann `tip cuad0` ein und Sie sind mit dem Modem verbunden.

Oder benutzen Sie `cu` als `root` mit dem folgenden Befehl:

```
# cu -llline -sspeed
```

`line` steht für die serielle Schnittstelle (`/dev/cuad0`) und *`speed`* für die Geschwindigkeit (`57600`). Wenn Sie mit dem Eingeben der AT Befehle fertig sind, beenden Sie mit `~.`.

26.5.3. Das `@` Zeichen funktioniert nicht

Das `@` Zeichen in der Telefonnummerfähigkeit sagt `tip`, dass es in `/etc/phones` nach einer Nummer suchen soll. Aber `@` ist auch ein spezielles Zeichen in den Dateien, in denen Fähigkeiten beschrieben werden, wie `/etc/remote`. Schreiben Sie es mit einem Backslash:

```
pn=\@
```

26.5.4. Wie kann ich von der Kommandozeile eine Telefonnummer wählen?

Setzen Sie einen allgemeinen Eintrag in `/etc/remote`. Zum Beispiel:

```
tip115200|Dial any phone number at 115200 bps:\
        :dv=/dev/cuad0:br#115200:at=hayes:pa=none:du:
tip57600|Dial any phone number at 57600 bps:\
```

```
        :dv=/dev/cuad0:br#57600:at=hayes:pa=none:du:
```

Folgendes sollte jetzt funktionieren:

```
# tip -115200 5551234
```

Benutzer, die cu gegenüber tip bevorzugen, können einen allgemeinen cu-Eintrag verwenden:

```
cu115200|Use cu to dial any number at 115200bps:\
        :dv=/dev/cuad1:br#57600:at=hayes:pa=none:du:
```

und benutzen zum Wählen das Kommando:

```
# cu 5551234 -s 115200
```

26.5.5. Die bps-Rate angeben

Schreiben Sie einen `tip1200`- oder einen `cu1200`-Eintrag, aber geben Sie auch die bps-Rate an, die das Modem wirklich unterstützt. Leider denkt tip(1), dass 1200 bps ein guter Standardwert ist und deswegen sucht es nach einem `tip1200`-Eintrag. Natürlich müssen Sie nicht 1200 bps benutzen.

26.5.6. Über einen Terminal-Server auf verschiedene Rechner zugreifen

Sie müssen nicht warten bis Sie verbunden sind, und jedes Mal `CONNECT` *`Rechner`* eingeben, benutzen Sie `tips` `cm`-Fähigkeit. Sie können diese Einträge in `/etc/remote` verwenden. Mit den Befehlen `tip pain` oder `tip muffin` können Sie eine Verbindungen zu den Rechnern `pain` oder `muffin` herstellen; mit `tip deep13` verbinden Sie sich mit dem Terminalserver.

```
pain|pain.deep13.com|Forrester's machine:\
        :cm=CONNECT pain\n:tc=deep13:
muffin|muffin.deep13.com|Frank's machine:\
        :cm=CONNECT muffin\n:tc=deep13:
deep13:Gizmonics Institute terminal server:\
        :dv=/dev/cuad2:br#38400:at=hayes:du:pa=none:pn=5551234:
```

26.5.7. Mehr als eine Verbindung mit `tip` benutzen

Das ist oft ein Problem, wenn eine Universität mehrere Telefonleitungen hat und viele tausend Studenten diese benutzen wollen.

Erstellen Sie einen Eintrag in `/etc/remote` und benutzen Sie `@` für die `pn`-Fähigkeit:

```
big-university:\
        :pn=\@:tc=dialout
dialout:\
        :dv=/dev/cuad3:br#9600:at=courier:du:pa=none:
```

Listen Sie dann die Telefonnummern in `/etc/phones` auf:

```
big-university 5551111
big-university 5551112
big-university 5551113
big-university 5551114
```

`tip` probiert jede der Nummern in der aufgelisteten Reihenfolge und gibt dann auf. Möchten Sie, dass `tip` beim Versuchen eine Verbindung herzustellen nicht aufgibt, lassen Sie es in einer `while`-Schleife laufen.

26.5.8. Eine Übertragung erzwingen

Ctrl+P ist das voreingestellte Zeichen, mit dem eine Übertragung erzwungen werden kann und wird benutzt, um `tip` zu sagen, dass das nächste Zeichen direkt gesendet werden soll und nicht als Fluchtzeichen interpretiert wer-

den soll. Mit Hilfe der Fluchtsequenz `~s`, mit der man Variablen setzen kann, können Sie jedes andere Zeichen als „force“-Zeichen definieren.

Geben Sie `~sforce=Zeichen` gefolgt von Enter ein. Für *`Zeichen`* können Sie ein beliebiges einzelnes Zeichen einsetzen. Wenn Sie *`Zeichen`* weglassen, ist das „force“-Zeichen „nul“, das Sie mit Ctrl+2 oder Ctrl+Leertaste eingeben können. Ein guter Wert für *`Zeichen`* ist Shift+Ctrl+6, welches nur auf wenigen Terminal Servern benutzt wird.

Sie können das „force“-Zeichen auch bestimmen, indem Sie in `$HOME/.tiprc` das Folgende einstellen:

```
force=single-char
```

26.5.9. Großbuchstaben

Dies passiert, wenn Ctrl+A eingegeben wurde, das „raise“-Zeichen von `tip`, das speziell für Leute mit defekten caps-lock Tasten eingerichtet wurde. Benutzen Sie `~s` wie oben und setzen Sie die Variable `raisechar` auf etwas, das Ihnen angemessen erscheint. Tatsächlich kann die Variable auf das gleiche Zeichen wie das „force“-Zeichen gesetzt werden, wenn diese Fähigkeiten niemals benutzt werden sollen.

Hier ist ein Muster der `.tiprc` Datei für Emacs Benutzer, die Ctrl+2 und Ctrl+A tippen müssen:

```
force=^^
raisechar=^^
```

Geben Sie für ^^ Shift+Ctrl+6 ein.

26.5.10. Dateien mit `tip` übertragen

Wenn Sie mit einem anderen UNIX® System kommunizieren, können Sie mit `~p` (put) und `~t` (take) Dateien senden und empfangen. Diese Befehle lassen `cat` und `echo` auf dem entfernten System laufen, um Dateien zu empfangen und zu senden. Die Syntax ist:

`~p` local-file [remote-file]

`~t` remote-file [local-file]

Es gibt keine Fehlerkontrolle, deshalb sollte besser ein anderes Protokoll, wie zmodem, benutzt werden.

26.5.11. zmodem mit `tip` benutzen

Um Dateien zu empfangen, starten Sie das Programm zum Senden auf dem entfernten Computer. Geben Sie dann `~C rz` ein, um die Dateien lokal zu empfangen.

Um Dateien zu senden, starten Sie das Programm zum Empfangen auf dem entfernten Computer. Geben Sie dann `~C sz Dateien` ein, um Dateien auf das entfernte System zu senden.

26.6. Einrichten der seriellen Konsole

Beigetragen von Kazutaka YOKOTA.
Based on a document by Bill Paul.

FreeBSD kann ein System mit einem Dumb-Terminal (unintelligente Datenstation) an einer seriellen Schnittstelle als Konsole booten. Diese Konfiguration ist besonders nützlich für Systemadministratoren, die FreeBSD auf Systemen ohne Tastatur oder Monitor installieren wollen, und Entwickler, die den Kernel oder Gerätetreiber debuggen.

Wie in Kapitel 13, *FreeBSDs Bootvorgang* beschrieben, besitzt FreeBSD drei Bootphasen. Der Code für die ersten beiden Bootphasen befindet sich im Bootsektor am Anfang der FreeBSD-Slice der Bootplatte. Dieser Bootblock lädt den Bootloader in Phase drei.

Um eine serielle Konsole einzurichten, muss der Bootblock, der Bootloader und der Kernel konfiguriert werden.

26.6.1. Schnelle Konfiguration der seriellen Konsole

Dieser Abschnitt bietet einen schnellen Überblick über die Einrichtung einer seriellen Konsolen. Es wird vorausgesetzt, dass die Voreinstellungen verwendet werden.

1. Verbinden Sie die serielle Konsole mit `COM1` sowie dem Kontrollterminal.

2. Um die Startmeldungen der seriellen Konsole zu sehen, geben Sie als `root` folgendes ein:

```
# echo 'console="comconsole"' >> /boot/loader.conf
```

3. Ändern Sie in `/etc/ttys` den Eintrag für `ttyu0` von `off` auf `on`. Zusätzlich sollten Sie den Wert `dialup` auf `vt100` ändern. Nur so wird auf der seriellen Konsole eine Eingabeaufforderung mit einer Passwortabfrage aktiviert.

4. Starten Sie nun das System neu, damit die serielle Konsole aktiviert wird.

Wenn Sie eine unterschiedliche Konfiguration benötigen, lesen Sie den nächsten Abschnitt für eine tiefer gehende Erklärung.

26.6.2. Konfiguration der seriellen Konsole

1. Bereiten Sie ein serielles Kabel vor.

   Sie benötigen entweder ein Nullmodemkabel oder ein serielles Standard Kabel mit einem Nullmodemkabel-Adapter. In Abschnitt 26.2.1, „Kabel und Schnittstellen“ werden serielle Kabel beschrieben.

2. Trennen Sie die Tastatur vom Computer.

   Viele PC Systeme suchen beim Power On Self Test (POST) nach einer Tastatur und geben eine Fehlermeldung aus, wenn sie keine finden. Einige Maschinen werden sich sogar weigern, ohne Tastatur zu booten.

   Wenn der Rechner trotz einer Fehlermeldung normal weiterbootet, brauchen Sie weiter nichts zu tun.

   Wenn das System ohne Tastatur nicht booten will, müssen Sie das BIOS so konfigurieren, dass es diesen Fehler ignoriert (wenn das möglich ist). Das Handbuch zum Motherboard sollte beschreiben, wie das zu bewerkstelligen ist.

Tipp

Selbst wenn Sie im BIOS „Not installed“ für die Tastatur einstellen, können Sie eine Tastatur angeschlossen haben und diese auch weiterhin benutzen, da sie mit dieser Anweisung das BIOS lediglich anweisen, nach dem Einschalten des Rechners nicht nach einer Tastatur zu suchen und den Rechner ohne entsprechende Fehlermeldung zu starten. Wenn die oben beschriebene Option nicht im BIOS vorhanden ist, halten Sie stattdessen Ausschau nach einer „Halt on Error“ Option. Sie können den gleichen Effekt wie oben erzielen, wenn Sie diese Option auf „All but Keyboard“ oder sogar „No Errors“ setzen.

Anmerkung

Wenn das System über eine PS/2® Maus verfügt, müssen Sie diese wahrscheinlich auch abziehen. Da sich die PS/2® Maus und die Tastatur einige Hardwarekomponenten tei-

len, kann das dazu führen, dass die Hardwareerkennung fälschlicherweise eine Tastatur findet, wenn eine PS/2® Maus angeschlossen ist.

3. Schließen Sie einen Dumb-Terminal an `COM1` (`sio0`) an.

   Wenn Sie keinen Dumb-Terminal besitzen, können Sie einen alten Computer mit einem Terminalemulator oder die serielle Schnittstelle eines anderen UNIX® Rechners benutzen. Sie benötigen auf jeden Fall eine freie erste serielle Schnittstelle (`COM1`). Zurzeit ist es nicht möglich, in den Bootblöcken eine andere Schnittstelle zu konfigurieren, ohne diese neu zu kompilieren. Wenn Sie `COM1` bereits für ein anderes Gerät benutzen, müssen Sie dieses Gerät temporär entfernen und einen neuen Bootblock sowie Kernel installieren, wenn FreeBSD erst einmal installiert ist.

4. Stellen Sie sicher, dass die Kernelkonfiguration die richtigen Optionen für `COM1` (`sio0`) enthält.

   Relevante Optionen sind:

   `0x10`
   : Aktiviert die Konsolenunterstützung für dieses Gerät. Zurzeit kann nur ein Gerät die Konsolenunterstützung aktiviert haben. Das erste, in der Konfigurationsdatei aufgeführte Gerät, mit dieser Option, verfügt über eine aktivierte Konsolenunterstützung. Beachten Sie, dass diese Option alleine nicht ausreicht, um die serielle Konsole zu aktivieren. Setzen Sie entweder noch die nachfolgend diskutierte Option oder verwenden Sie beim Booten, wie unten beschrieben, den Schalter `-h`.

   `0x20`
   : Das erste Gerät in der Kernelkonfigurationsdatei mit dieser Option wird, unabhängig von dem unten diskutierten Schalter `-h`, zur Konsole. Die Option `0x20` muss zusammen mit `0x10` verwendet werden.

   `0x40`
   : Reserviert dieses Gerät und sperrt es für normale Zugriffe. Sie sollten diese Option nicht auf dem Gerät setzen, das Sie als serielle Konsole verwenden wollen. Der Zweck dieser Option ist es, dieses Gerät für das Remote-Debuggen zu reservieren. Das FreeBSD Developers' Handbook enthält dazu weitere Informationen.

   Beispiel:

   ```
   device sio0 at isa? port IO_COM1 tty flags 0x10 irq 4
   ```

   Weitere Einzelheiten finden Sie in sio(4).

   Wenn diese Optionen nicht gesetzt sind, müssen Sie auf einer anderen Konsole beim Booten UserConfig starten oder den Kernel neu kompilieren.

5. Erstellen Sie `boot.config` im Rootverzeichnis der `a`-Partition des Bootlaufwerks.

   Der Code des Bootblocks entnimmt dieser Datei, wie Sie Ihr System booten möchten. Um die serielle Konsole zu aktivieren, müssen Sie hier eine oder mehrere Optionen (alle in derselben Zeile) angeben. Die folgenden Optionen stehen zur Auswahl der Konsole zur Verfügung:

   `-h`
   : Schaltet zwischen der internen und der seriellen Konsole um. Wenn Sie beispielsweise von der internen Konsole (Bildschirm) booten, weist `-h` den Bootloader und den Kernel an, die serielle Schnittstelle als Konsole zu nehmen. Wenn die Konsole normal auf der seriellen Schnittstelle liegt, wählen Sie mit `-h` den Bildschirm aus.

   `-D`
   : Schaltet zwischen Einzelkonsole und Dual-Konsole um. Die Einzelkonsole ist entweder die interne Konsole (der Bildschirm) oder die serielle Schnittstelle, je nach dem Stand von `-h`. Im Dual-Konsolen Betrieb

ist die Konsole, unabhängig von `-h`, gleichzeitig der Bildschirm und die serielle Schnittstelle. Dies trifft aber nur zu, wenn der Bootblock ausgeführt wird. Sobald der Bootloader ausgeführt wird, wird die durch `-h` gegebene Konsole die alleinige Konsole.

`-P`
Veranlasst den Bootblock nach einer Tastatur zu suchen. Wenn keine Tastatur gefunden wird, werden `-D` und `-h` automatisch gesetzt.

Anmerkung

Wegen Platzbeschränkungen in den Bootblöcken kann `-P` nur erweiterte Tastaturen erkennen. Tastaturen mit weniger als 101 Tasten und ohne F11 und F12 Tasten werden wahrscheinlich, wie vielleicht auch die Tastaturen einiger Laptops, nicht erkannt. Wenn das der Fall ist, können Sie `-P` nicht verwenden, da es leider keine Abhilfe für dieses Problem gibt.

Benutzen Sie also entweder `-P`, um die Konsole automatisch zu setzen, oder `-h`, um die serielle Konsole zu verwenden.

Weitere Optionen werden in boot(8) beschrieben.

Mit Ausnahme von `-P` werden die Optionen an den Bootloader weitergegeben. Der Bootloader untersucht dann einzig `-h` um festzustellen, welches Gerät die Konsole wird. Wenn Sie also nur `-D` angegeben haben, können Sie die serielle Schnittstelle nur als Konsole verwenden während der Bootblock ausgeführt wird. Danach wird der Bootloader, da ja `-h` fehlt, den Bildschirm zur Konsole machen.

6. Booten Sie die Maschine.

   Wenn Sie das FreeBSD-System starten, werden die Bootblöcke den Inhalt von `/boot.config` auf der Konsole ausgeben:

```
/boot.config: -P
Keyboard: no
```

   Die zweite Zeile sehen Sie nur, wenn Sie in `/boot.config` `-P` angegeben haben. Sie zeigt an, ob eine Tastatur angeschlossen ist oder nicht. Die Meldungen gehen je nach den Einstellungen in `/boot.config` auf die interne Konsole, die serielle Konsole, oder beide Konsolen.

Optionen	Meldungen erscheinen auf
keine	der internen Konsole
`-h`	der seriellen Konsole
`-D`	der seriellen und der internen Konsole
`-Dh`	der seriellen und der internen Konsole
`-P`, mit Tastatur	der internen Konsole
`-P`, ohne Tastatur	der seriellen Konsole

   Nach den oben gezeigten Meldungen gibt es eine kleine Verzögerung bevor die Bootblöcke den Bootloader laden und weitere Meldungen auf der Konsole erscheinen. Sie können die Ausführung der Bootblöcke unterbrechen, um zu überprüfen, ob auch alles richtig aufgesetzt ist, brauchen das aber unter normalen Umständen nicht zu tun.

   Drücken Sie eine Taste außer Enter um den Bootvorgang zu unterbrechen. Sie erhalten dann ein Prompt, an dem Sie weitere Eingaben tätigen können:

```
>> FreeBSD/i386 BOOT
Default: 0:ad(0,a)/boot/loader
boot:
```

Je nach Inhalt von `/boot.config` erscheint das Prompt auf der seriellen Konsole, der internen Konsole oder beiden Konsolen. Wenn die Meldung auf der richtigen Konsole erscheint, drücken Sie Enter um fortzufahren.

Wenn kein Prompt auf der seriellen Konsole erscheint, liegt ein Fehler in den Einstellungen vor. Als Abhilfe geben Sie an der momentanen Konsole `-h` ein, um den Bootblock und den Bootloader auf die serielle Konsole umzustellen. Führen Sie dann den Bootvorgang mit Enter weiter und wenn das System gebootet hat, können Sie die fehlerhaften Einstellungen korrigieren.

Während der dritten Bootphase können Sie immer noch zwischen der internen und der seriellen Konsole auswählen. Setzen Sie dazu, wie in Abschnitt 26.6.5, „Die Konsole im Bootloader ändern“ beschrieben, die entsprechenden Variablen des Bootloaders.

26.6.3. Zusammenfassung

Die folgende Tabelle bietet eine Zusammenfassung der verschiedenen Einstellungen, die in diesem Abschnitt diskutiert wurden:

Tabelle 26.5. Fall 1: Option 0x10 für `sio0`

Optionen in **`/boot.config`**	Konsole in den Bootblöcken	Konsole im Bootloader	Konsole im Kernel
keine	interne	interne	interne
`-h`	serielle	serielle	serielle
`-D`	serielle und interne	interne	interne
`-Dh`	serielle und interne	serielle	serielle
`-P`, mit Tastatur	interne	interne	interne
`-P`, ohne Tastatur	serielle und interne	serielle	serielle

Tabelle 26.6. Fall 2: Option 0x30 für `sio0`

Optionen in **`/boot.config`**	Konsole in den Bootblöcken	Konsole im Bootloader	Konsole im Kernel
keine	interne	interne	serielle
`-h`	serielle	serielle	serielle
`-D`	serielle und interne	interne	serielle
`-Dh`	serielle und interne	serielle	serielle
`-P`, mit Tastatur	interne	interne	serielle
`-P`, ohne Tastatur	serielle und interne	serielle	serielle

26.6.4. Hinweise zur seriellen Konsole

26.6.4.1. Verwenden einer höheren Geschwindigkeit

Die Vorgabewerte für die Kommunikationsparameter der seriellen Schnittstelle sind: 9600 baud, 8 Bit, keine Parität und ein Stopp-Bit. Um die Standardgeschwindigkeit zu ändern, stehen folgende Möglichkeiten zur Verfügung:

- Geben Sie die neue Konsolengeschwindigkeit mit `BOOT_COMCONSOLE_SPEED` an und kompilieren Sie die Bootblöcke neu. Ausführliche Informationen zum Bau und zur Installation von neuen Bootblöcken finden Sie im Abschnitt 26.6.4.2, „Eine andere Schnittstelle als `sio0` benutzen“ des Handbuchs.

Wenn die serielle Konsole nicht mit der Option `-h` gestartet wird, oder wenn die verwendete serielle Konsole sich von der von den Bootblöcken verwendeten unterscheidet, müsssen Sie zusätzlich die folgende Option in die Kernelkonfigurationsdatei aufnehmen und den Kernel neu bauen:

```
options CONSPEED=19200
```

- Verwenden Sie die Option `-S`, um den Kernel zu booten. Eine Beschreibung dieses Vorgangs sowie eine Auflistung der von `/boot.config` unterstützten Optionen finden Sie in boot(8).

- Aktivieren Sie die Option `comconsole_speed` in `/boot/loader.conf` .

  Diese Option setzt voraus, dass auch die Optionen `console`, `boot_serial`, sowie `boot_multicons` in `/boot/loader.conf` gesetzt sind. Im Folgenden finden Sie ein Beispiel, in dem `comconsole_speed` verwendet wird, um die Geschwindigkeit der seriellen Konsole zu ändern:

```
boot_multicons="YES"
boot_serial="YES"
comconsole_speed="115200"
console="comconsole,vidconsole"
```

26.6.4.2. Eine andere Schnittstelle als `sio0` benutzen

Wenn Sie, warum auch immer, ein anderes Gerät als `sio0` für die serielle Konsole einsetzen wollen, kompilieren Sie bitte die Bootblöcke, den Bootloader und den Kernel nach dem folgenden Verfahren neu.

1. Installieren Sie die Kernelquellen wie im Kapitel 24, *FreeBSD aktualisieren* beschrieben.

2. Setzen Sie in `/etc/make.conf` `BOOT_COMCONSOLE_PORT` auf die Adresse der Schnittstelle (0x3F8, 0x2F8, 0x3E8 oder 0x2E8), die Sie benutzen möchten. Sie können nur `sio0` bis `sio3` (`COM1` bis `COM4`) benutzen, Multiportkarten können Sie nicht als Konsole benutzen. Interrupts müssen Sie hier nicht angeben.

3. Erstellen Sie eine angepasste Kernelkonfiguration und geben Sie dort die richtigen Optionen für die Schnittstelle, die Sie benutzen möchten, an. Wenn Sie zum Beispiel `sio1` (`COM2`) zur Konsole machen wollen, geben Sie dort entweder

   ```
   device sio1 at isa? port IO_COM2 tty flags 0x10 irq 3
   ```

   oder

   ```
   device sio1 at isa? port IO_COM2 tty flags 0x30 irq 3
   ```

   an. Keine andere serielle Schnittstelle sollte als Konsole definiert werden.

4. Übersetzen und installieren Sie die Bootblöcke und den Bootloader:

   ```
   # cd /sys/boot
   # make clean
   # make
   # make install
   ```

5. Bauen und installieren Sie einen neuen Kernel.

6. Schreiben Sie die Bootblöcke mit bsdlabel(8) auf die Bootplatte und booten Sie den neuen Kernel.

26.6.4.3. DDB Debugger über die serielle Schnittstelle

Wenn Sie den Kerneldebugger über eine serielle Verbindung bedienen möchten, übersetzen Sie einen angepassten Kernel mit den folgenden Optionen. Das ist nützlich, kann aber gefährlich sein, wenn auf der Leitung falsche BREAK-Signale generiert werden.

```
options BREAK_TO_DEBUGGER
```

```
options DDB
```

26.6.4.4. Benutzung der seriellen Konsole zum Anmelden

Da Sie schon die Bootmeldungen auf der Konsole verfolgen können und den Kerneldebugger über die Konsole bedienen können, wollen Sie sich vielleicht auch an der Konsole anmelden.

Öffnen Sie `/etc/ttys` in einem Editor und suchen Sie nach den folgenden Zeilen:

```
ttyu0 "/usr/libexec/getty std.9600" unknown off secure
ttyu1 "/usr/libexec/getty std.9600" unknown off secure
ttyu2 "/usr/libexec/getty std.9600" unknown off secure
ttyu3 "/usr/libexec/getty std.9600" unknown off secure
```

`ttyu0` bis `ttyu3` entsprechen `COM1` bis `COM4`. Ändern Sie für die entsprechende Schnittstelle `off` zu `on`. Wenn Sie auch die Geschwindigkeit der seriellen Schnittstelle geändert haben, müssen Sie `std.9600` auf die momentane Geschwindigkeit anpassen.

Auch kann den Terminaltyp von `unknown` auf den tatsächlich verwendeten Terminal gesetzt werden.

Damit die Änderungen wirksam werden, müssen Sie noch `kill -HUP 1` absetzen.

26.6.5. Die Konsole im Bootloader ändern

In den vorigen Abschnitten wurde beschrieben, wie Sie die serielle Konsole durch Änderungen im Bootblock aktivieren. Dieser Abschnitt zeigt, wie Sie mit Kommandos und Umgebungsvariablen die Konsole im Bootloader definieren. Da der Bootloader die dritte Phase im Bootvorgang ist und nach den Bootblöcken ausgeführt wird, überschreiben seine Einstellungen die des Bootblocks.

26.6.5.1. Festlegen der Konsole

Mit einer einzigen Zeile in `/boot/loader.conf` können Sie den Bootloader und den Kernel anweisen, die serielle Schnittstelle zur Konsole zu machen:

```
console="comconsole"
```

Unabhängig von den Einstellungen im Bootblock legt dies die Konsole fest.

Die obige Zeile sollte die erste Zeile in `/boot/loader.conf` sein, so dass die Bootmeldungen so früh wie möglich auf der Konsole zu sehen sind.

Analog können Sie die interne Konsole verwenden:

```
console="vidconsole"
```

Wenn die Umgebungsvariable `console` nicht gesetzt ist, bestimmt der Bootloader und damit auch der Kernel, die Konsole über die `-h` Option des Bootblocks.

Die Bootkonsole kann in `/boot/loader.conf.local` oder `/boot/loader.conf` angegeben werden.

Weitere Informationen erhalten Sie in loader.conf(5).

Anmerkung

Momentan gibt es im Bootloader nichts vergleichbares zu `-P` im Bootblock. Damit kann die Konsole nicht automatisch über das Vorhandensein einer Tastatur festgelegt werden.

26.6.5.2. Eine andere Schnittstelle als `sio0` benutzen

Der Bootloader muss neu kompiliert werden, wenn eine andere Schnittstelle als `sio0` benutzt werden soll. Folgen Sie der Anleitung aus Abschnitt 26.6.4.2, „Eine andere Schnittstelle als `sio0` benutzen“.

26.6.6. Vorbehalte

Obwohl es die meisten Systeme erlauben, ohne Tastatur zu booten, gibt es nur wenige Systeme, die ohne eine Grafikkarte booten. Maschinen mit einem AMI BIOS können ohne Grafik booten, indem Sie den Grafikadapter im CMOS-Setup auf `Not installed` setzen.

Viele Maschinen unterstützen diese Option allerdings nicht. Damit diese Maschinen booten, müssen sie über eine Grafikkarte, auch wenn es nur eine alte Monochromkarte ist, verfügen. Allerdings brauchen Sie keinen Monitor an die Karte anzuschließen. Sie können natürlich auch versuchen, auf diesen Maschinen ein AMI BIOS zu installieren.

Kapitel 27. PPP

Restrukturiert, neu organisiert und aktualisiert von Jim Mock.
Übersetzt von Thomas Schwarzkopf.

27.1. Übersicht

Unter FreeBSD stehen verschiedene Möglichkeiten zur Verfügung, um Computer miteinander zu verbinden. Der Aufbau einer Netzwerk- oder Internetverbindung mit Hilfe eines Einwahlmodems – für den eigenen oder für andere Rechner – erfordert den Einsatz von PPP.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie wissen:

- Wie Sie User-PPP einrichten.
- Was zu tun ist, um PPPoE (PPP over Ethernet) einzurichten.
- Wie Sie PPPoA (PPP over ATM) einrichten.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- mit den grundlegenden Begriffen der Netzwerktechnik vertraut sein.
- die Grundlagen und den Zweck einer Einwahlverbindung sowie PPP kennen.

27.2. User-PPP

Aktualisiert und erweitert von Tom Rhodes.
Ursprünglich geschrieben von Brian Somers.
Mit Beiträgen von Nik Clayton, Dirk Frömberg und Peter Childs.

27.2.1. User-PPP

27.2.1.1. Voraussetzungen

Dieses Dokument geht davon aus, dass Sie Folgendes zur Verfügung haben:

- Einen Account bei einem Internet Service Provider (ISP), zu dem Sie mit PPP eine Verbindung aufbauen können.
- Ein Modem oder ein anderes Gerät, das, richtig konfiguriert und mit Ihrem Rechner verbunden, Ihnen die Herstellung einer Verbindung zu Ihrem ISP erlaubt.
- Die Einwahlnummer(n) Ihres ISP.
- Ihren Login-Namen und Ihr Passwort (entweder ein reguläres Login/Passwort-Paar im UNIX®-Stil oder ein PAP bzw. CHAP Login/Passwort-Paar).
- Die IP-Adresse von einem oder mehreren Nameservern. Üblicherweise werden Ihnen von Ihrem ISP zwei IP-Adressen für diesen Zweck zur Verfügung gestellt. Wenn Sie keine solche IP-Adresse von Ihrem Provider bekommen haben, können Sie das Kommando `enable dns` in der Datei `ppp.conf` verwenden, um ppp anzuweisen, den Nameserver für Sie einzutragen. Diese Funktion setzt allerdings voraus, dass Ihr ISP eine PPP-Implementierung verwendet, die das Aushandeln eines Nameservers unterstützt.

Die folgenden Informationen werden Ihnen möglicherweise von Ihrem ISP zur Verfügung gestellt, sie sind aber nicht zwingend erforderlich:

- Die Gateway IP-Adresse Ihres ISP. Als Gateway wird der Computer bezeichnet, zu dem Sie eine Verbindung aufbauen. Die IP-Adresse dieses Rechners wird als *default route* eingetragen. Wenn Sie diese Information nicht zur Verfügung haben, kann PPP so konfiguriert werden, dass der PPP-Server Ihres ISP während des Verbindungsaufbaus eine gültige Adresse übermittelt.

  ppp bezieht sich mit `HISADDR` auf diese IP-Adresse.

- Die Netzmaske, die Sie verwenden sollten. Falls Ihnen Ihr ISP keine Netzmaske vorgegeben hat, können Sie `255.255.255.255` verwenden.

- Wenn Ihnen Ihr ISP eine statische IP-Adresse zur Verfügung stellt, können Sie diese eintragen. Andernfalls lassen wir uns einfach von der Gegenstelle eine IP-Adresse zuweisen.

Falls Ihnen die erforderlichen Informationen fehlen sollten, nehmen Sie bitte Kontakt mit Ihrem ISP auf.

Anmerkung

Die Beispieldateien, die in diesem Kapitel dargestellt werden, enthalten Zeilennummern. Die Nummerierung dient lediglich einer leichteren Orientierung und sollte von Ihnen nicht in Ihre Dateien übernommen werden. Richtiges Einrücken, durch Tabulatoren und Leerzeichen, ist ebenfalls wichtig.

27.2.1.2. Automatische Konfiguration von PPP

`ppp` verwendet die Konfigurationsdateien in `/etc/ppp`. Beispiele sind in `/usr/share/examples/ppp/` zu finden.

Die Konfiguration von `ppp` erfordert, je nach Ihren besonderen Bedingungen, die Bearbeitung einiger Dateien. Was Sie in diese Dateien eintragen, hängt unter anderem davon ab, ob Ihnen Ihr ISP eine statische IP-Adresse (Sie verwenden immer dieselbe IP-Adresse, die Ihnen einmal zugeteilt wurde) oder eine dynamische IP-Adresse (Ihre IP-Adresse ändert sich bei jeder Verbindung mit dem ISP) zugewiesen hat.

27.2.1.2.1. PPP und statische IP-Adressen

Sie müssen die Konfigurationsdatei `/etc/ppp/ppp.conf` bearbeiten. Sie sollte so aussehen, wie in dem unten angegebenen Beispiel.

Anmerkung

Zeilen die mit einem `:` enden, beginnen in der ersten Spalte (am Beginn der Zeile). Alle anderen Zeilen sollten wie dargestellt durch Leerzeichen oder Tabulatoren eingerückt werden.

```
1     default:
2       set log Phase Chat LCP IPCP CCP tun command
3       ident user-ppp VERSION (built COMPILATIONDATE)
4       set device /dev/cuau0
5       set speed 115200
6       set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \
7                 \"\" AT OK-AT-OK ATE1Q0 OK \\dATDT\\T TIMEOUT 40 CONNECT"
8       set timeout 180
9       enable dns
10
11    provider:
12      set phone "(123) 456 7890"
13      set authname foo
14      set authkey bar
```

```
15      set login "TIMEOUT 10 \"\" \"\" gin:--gin: \\U word: \\P col: ppp"
16      set timeout 300
17      set ifaddr x.x.x.x y.y.y.y 255.255.255.255 0.0.0.0
18      add default HISADDR
```

Zeile 1:
Gibt den Standardeintrag an. Befehle dieses Eintrags werden automatisch ausgeführt, wenn ppp läuft.

Zeile 2:
Schaltet die Loggingparameter ein. Wenn die Verbindung zufriedenstellend funktioniert, können Sie diese Zeile verkürzen:

```
set log phase tun
```

Dies verhindert ein übermäßiges Anwachsen der Logdateien.

Zeile 3:
Gibt PPP an, wie es sich gegenüber der Gegenstelle identifizieren soll. PPP identifiziert sich gegenüber der Gegenstelle, wenn es Schwierigkeiten bei der Aushandlung und beim Aufbau der Verbindung gibt. Dabei werden Informationen bereitgestellt, die dem Administrator der Gegenstelle helfen können, die Ursache der Probleme zu finden.

Zeile 4:
Gibt das Device an, an dem das Modem angeschlossen ist. `COM1` entspricht `/dev/cuad0` und `COM2` entspricht `/dev/cuad1`.

Zeile 5:
Legt die Geschwindigkeit fest, mit der Sie die Verbindung betreiben möchten. Falls ein Wert von 115200 nicht funktioniert (was aber bei jedem einigermaßen neuen Modem funktionieren sollte), versuchen Sie es stattdessen mit 38400.

Zeilen 6 & 7:
Die Zeichenfolge für die Einwahl. User-PPP verwendet eine expect-send Syntax, ähnlich dem chat(8)-Programm. Weitere Informationen zu den Eigenschaften dieser Sprache bietet die Manual-Seite.

Beachten Sie, dass dieser Befehl aufgrund der besseren Lesbarkeit auf der nächsten Zeile weitergeht. Das kann für jeden Befehl in `ppp.conf` gelten, wenn `\` das letzte Zeichen in einer Zeile ist.

Zeile 8:
Legt den Zeitrahmen fest, innerhalb dessen eine Reaktion erfolgen muss. Der Standardwert liegt bei 180 Sekunden, so dass diese Zeile lediglich einen kosmetischen Charakter hat.

Zeile 9:
Weist PPP an, bei der Gegenstelle eine Bestätigung der lokalen Resolvereinstellungen anzufordern. Wenn Sie einen lokalen Nameserver betreiben, sollte diese Zeile auskommentiert oder gelöscht werden.

Zeile 10:
Eine leere Zeile zur besseren Lesbarkeit. Leere Zeilen werden von PPP ignoriert.

Zeile 11:
Bestimmt einen Provider, namens „provider“. Wenn Sie hier den Namen Ihres ISP einsetzen, können Sie später die Verbindung mit `load` *`ISP`* aufbauen.

Zeile 12:
Gibt die Telefonnummer des Providers an. Mehrere Telefonnummern können angegeben werden, indem Doppelpunkte (`:`) oder Pipe-Zeichen (`|`) als Trennzeichen verwendet werden. Der Unterschied zwischen diesen beiden Trennzeichen ist in ppp(8) beschrieben. Zusammenfassend: Wenn Sie die verschiedenen Nummern abwechselnd verwenden möchten, sollten Sie die Nummern durch einen Doppelpunkt trennen. Wenn Sie immer die erste Nummer verwenden möchten und die anderen nur zum Einsatz kommen sollen, wenn eine

Einwahl mit der ersten Telefonnummer nicht möglich ist, sollten Sie das Pipe-Zeichen zur Trennung verwenden. Wie im Beispiel, sollten Sie die gesamte Reihe der Telefonnummern in Anführungszeichen setzen.

Sie müssen die Telefonnummer in Anführungszeichen (") setzen, wenn Sie Leerzeichen in der Telefonnummer verwenden, ansonsten rufen Sie einen Fehler hervor, der vielleicht schwer zu finden ist.

Zeilen 13 & 14:
Gibt den Benutzernamen und das Passwort an. Wenn Sie zur Verbindung einen Login-Prompt im UNIX-Stil verwenden, bezieht sich der Befehl `set login` mit den \U und \P Variablen auf diese Werte. Wenn Sie zum Verbindungsaufbau PAP oder CHAP verwenden, werden diese Werte zum Zeitpunkt der Authentifizierung verwendet.

Zeile 15:
Wenn Sie PAP oder CHAP einsetzen, gibt es an dieser Stelle keinen Login-Prompt, weshalb Sie diese Zeile auskommentieren oder löschen sollten. Der Abschnitt Authentifizierung mit PAP und CHAP enthält hierzu weitere Einzelheiten.

Der Login-String hat die gleiche chat-ähnliche Syntax, wie der Einwahlstring. Der String in diesem Beispiel funktioniert mit einem ISP, dessen Login-Session folgendermaßen aussieht:

```
J. Random Provider
login: foo
password: bar
protocol: ppp
```

Sie müssen dieses Skript noch an Ihre eigenen Erfordernisse anpassen. Wenn Sie dieses Skript zum ersten Mal schreiben, sollten Sie sicherstellen, dass Sie „chat“-logging aktiviert haben, damit Sie überprüfen zu können, ob die Konversation zwischen Ihrem Rechner und dem Rechner des Providers wie erwartet abläuft.

Zeile 16:
Setzt einen Zeitrahmen (in Sekunden), innerhalb dessen eine Reaktion erfolgen muss. In diesem Fall, wird die Verbindung nach 300 Sekunden automatisch geschlossen, wenn keine Aktivität zu verzeichnen ist. Wenn Sie keinen Zeitrahmen festlegen wollen, nach dessen Überschreiten die Verbindung geschlossen wird, können Sie diesen Wert auf 0 setzen oder die Kommandozeilen-Option `-ddial` verwenden.

Zeile 17:
Gibt die IP-Adresse für das Interface an. Der String *x.x.x.x* sollte durch die IP-Adresse ersetzt werden, die Ihnen Ihr Provider zugeteilt hat. Der String *y.y.y.y* sollte durch die IP-Adresse ersetzt werden, die Ihr ISP als Gateway angegeben hat (das ist der Rechner, mit dem Ihr Rechner eine Verbindung aufbaut). Wenn Ihnen Ihr ISP keine Gateway Adresse zur Verfügung gestellt hat, verwenden Sie hier einfach `10.0.0.2/0` . Wenn Sie eine „erratene“ IP-Adresse verwenden müssen, sollten Sie in der Datei `/etc/ppp/ppp.linkup` einen entsprechenden Eintrag machen. Folgen Sie dazu den Anweisungen im Abschnitt PPP und dynamische IP-Adressen. Wenn diese Zeile ausgelassen wird, kann `ppp` nicht im `-auto` Modus betrieben werden.

Zeile 18:
Fügt eine Defaultroute für das Gateway Ihres Providers hinzu. Das Wort `HISADDR` wird dabei durch die in Zeile 17 angegebene Gateway Adresse ersetzt. Wichtig ist, dass diese Zeile nach Zeile 17 erscheint, da andernfalls `HISADDR` noch nicht initialisiert ist.

Wenn Sie ppp nicht im `-auto` Modus betreiben, sollte diese Zeile in die Datei `ppp.linkup` verschoben werden.

Wenn Sie eine statische IP-Adresse verwenden und ppp im `-auto` Modus läuft, ist es nicht notwendig, einen Eintrag in die Datei `ppp.linkup` hinzuzufügen. In diesem Fall hat ihre Routingtabelle bereits die richtigen Einträge, bevor Sie die Verbindung aufbauen. Sie möchten aber vielleicht einen Eintrag hinzufügen, um ein Programm aufzurufen, nachdem die Verbindung aufgebaut ist. Dies wird weiter unten am Beispiel von Sendmail erklärt.

Beispiele für Konfigurationsdateien finden Sie im Verzeichnis `/usr/share/examples/ppp/` .

27.2.1.2.2. PPP und dynamische IP-Adressen

Wenn Ihnen Ihr ISP keine statische IP-Adresse zuteilt, kann ppp so konfiguriert werden, dass die lokale und die entfernte IP-Adresse beim Verbindungsaufbau ausgehandelt werden. Dies geschieht, indem zunächst eine IP-Adresse „erraten“ wird, die von ppp, unter Verwendung des IP Configuration Protocol (IPCP) durch eine richtige ersetzt wird, wenn die Verbindung aufgebaut ist. Die Konfiguration der Datei ppp.conf entspricht derjenigen, die im Abschnitt PPP und statische IP- Adressen dargestellt wurde, jedoch mit folgender Änderung:

```
17      set ifaddr 10.0.0.1/0 10.0.0.2/0 255.255.255.255 0.0.0.0
```

Auch hier dient die Zeilennummerierung lediglich der besseren Übersichtlichkeit. Einrückungen, von mindestens einem Leerzeichen, sind allerdings erforderlich.

Zeile 17:
Die Zahl nach dem / Zeichen, gibt die Anzahl der Bits der Adresse an, auf die ppp besteht. Sie möchten vielleicht andere IP-Adressen verwenden, die oben angegebenen werden aber immer funktionieren.

Das letzte Argument (0.0.0.0) weist PPP an, den Verbindungsaufbau mit der Adresse 0.0.0.0 zu beginnen, statt 10.0.0.1 zu verwenden. Dies ist bei einigen ISPs notwendig. Verwenden Sie nicht 0.0.0.0 als erstes Argument für set ifaddr, da so verhindert wird, dass PPP im -auto Modus eine initiale Route setzt.

Wenn PPP nicht im -auto Modus läuft, müssen Sie die Datei/etc/ppp/ppp.linkup editieren. ppp.linkup kommt zum Einsatz, wenn eine Verbindung aufgebaut worden ist. Zu diesem Zeitpunkt hat ppp die Interface Adressen vergeben und es ist möglich, die Einträge in der Routingtabelle hinzuzufügen:

```
1     provider:
2      add default HISADDR
```

Zeile 1:
Beim Aufbau einer Verbindung sucht ppp in der Datei ppp.linkup nach einem Eintrag. PPP geht dabei nach folgenden Regeln vor: Suche zunächst nach der gleichen Bezeichnung, die wir auch in der Datei ppp.conf verwendet haben. Falls das nicht funktioniert, suche nach einem Eintrag der IP-Adresse unseres Gateways. Dieser Eintrag ist eine Bezeichnung im Stil von IP-Adressen, die sich aus vier Oktetts zusammensetzt. Falls immer noch kein passender Eintrag gefunden wurde, suche nach dem Eintrag MYADDR.

Zeile 2:
Diese Zeile weist ppp an, eine Defaultroute zu verwenden, die auf HISADDR zeigt. HISADDR wird nach der Aushandlung mit IPCP durch die IP-Adresse des Gateways ersetzt.

Die Dateien /usr/share/examples/ppp/ppp.conf.sample und /usr/share/examples/ppp/ppp.linkup.sample bieten detaillierte Beispiele für pmdemand Einträge.

27.2.1.2.3. Annahme eingehender Anrufe

Wenn Sie ppp auf einem Rechner, der in ein LAN eingebunden ist, so konfigurieren, dass eingehende Anrufe angenommen werden, müssen Sie entscheiden, ob Pakete an das LAN weitergeleitet werden sollen. Wenn Sie das möchten, sollten Sie an die Gegenstelle eine IP-Adresse aus Ihrem lokalen Subnetz vergeben und den Befehl enable proxy in die Datei /etc/ppp/ppp.conf einfügen. Außerdem sollte die Datei /etc/rc.conf Folgendes enthalten:

```
gateway_enable="YES"
```

27.2.1.2.4. Welches getty?

Der Abschnitt Einwählverbindungen bietet eine gute Beschreibung, wie Einwählverbindungen unter Verwendung von getty(8) genutzt werden können.

Eine Alternative zu getty ist mgetty, eine raffiniertere Version von getty, die mit Blick auf Einwählverbindungen entworfen wurde. Sie können dieses Paket über den Port comms/mgetty+sendfax installieren.

Der Vorteil von mgetty ist, dass es auf aktive Weise mit Modems *spricht*, das heißt wenn ein Port in /etc/ttys ausgeschaltet ist, wird Ihr Modem nicht auf Anrufe reagieren.

Spätere Versionen von `mgetty` (von 0.99beta aufwärts) unterstützen auch die automatische Erkennung von PPP-Streams, was Ihren Clients den skriptlosen Zugang zu Ihren Servern erlaubt.

Der Abschnitt Mgetty und AutoPPP bietet weitere Informationen zu `mgetty`.

27.2.1.2.5. PPP und Rechte

Der Befehl `ppp` muss normalerweise als `root` ausgeführt werden. Wenn Sie jedoch möchten, dass `ppp` im Server-Modus auch von einem normalen Benutzer, wie unten beschrieben, durch Aufruf von `ppp` ausgeführt werden kann, müssen Sie diesem Benutzer die Rechte erteilen, `ppp` auszuführen, indem Sie ihn in der Datei `/etc/group` der Gruppe `network` hinzufügen.

Sie werden ihm ebenfalls den Zugriff auf einen oder mehrere Abschnitte der Konfigurationsdatei geben müssen, indem Sie den `allow` Befehl verwenden:

```
allow users fred mary
```

Wenn dieser Befehl im `default` Abschnitt verwendet wird, erhalten die angegebenen Benutzer vollständigen Zugriff.

27.2.1.2.6. PPP-Shells für dynamische IP-Adressen

Erzeugen Sie eine Datei mit dem Namen `/etc/ppp/ppp-shell` , die Folgendes enthält:

```
#!/bin/sh
IDENT=`echo $0 | sed -e 's/^.*-\(.*\)$/\1/'`
CALLEDAS="$IDENT"
TTY=`tty`

if [ x$IDENT = xdialup -]; then
        IDENT=`basename $TTY`
fi

echo "PPP for $CALLEDAS on $TTY"
echo "Starting PPP for $IDENT"

exec /usr/sbin/ppp -direct $IDENT
```

Dieses Skript sollte ausführbar sein. Nun erzeugen Sie einen symbolischen Link `ppp-dialup` auf dieses Skript mit folgendem Befehl:

```
# ln -s ppp-shell /etc/ppp/ppp-dialup
```

Sie sollten dieses Skript als *Shell* für alle Benutzer von Einwählverbindungen verwenden. Dies ist ein Beispiel aus der Datei `/etc/passwd` für einen Benutzer namens `pchilds` , der PPP für Einwählverbindungen verwenden kann (Denken Sie daran, die Passwortdatei nicht direkt zu editieren, sondern dafür vipw(8) zu verwenden).

```
pchilds:*:1011:300:Peter Childs PPP:/home/ppp:/etc/ppp/ppp-dialup
```

Erstellen Sie ein Verzeichnis `/home/ppp` , das von allen Benutzern gelesen werden kann und die folgenden leeren Dateien enthält:

```
-r--r--r--   1 root     wheel           0 May 27 02:23 .hushlogin
-r--r--r--   1 root     wheel           0 May 27 02:22 .rhosts
```

Dies verhindert, dass `/etc/motd` angezeigt wird.

27.2.1.2.7. PPP-Shells für statische IP-Adressen

Erstellen Sie die Datei `ppp-shell` wie oben oben dargestellt. Erzeugen Sie nun für jeden Account mit statischer IP-Adresse einen symbolischen Link auf `ppp-shell` .

Wenn Sie beispielsweise die drei Kunden, `fred`, `sam` und `mary` haben, für die Sie CIDR-/24-Netzwerke routen, schreiben Sie Folgendes:

```
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-fred
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-sam
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-mary
```

Jeder Einwählzugang dieser Kunden sollte den oben erzeugten symbolischen Link als Shell haben (`mary`'s Shell sollte also `/etc/ppp/ppp-mary` sein).

27.2.1.2.8. Einrichten von `ppp.conf` für dynamische IP-Adressen

Die Datei `/etc/ppp/ppp.conf` sollte in etwa wie folgt aussehen:

```
default:
  set debug phase lcp chat
  set timeout 0

ttyu0:
  set ifaddr 203.14.100.1 203.14.100.20 255.255.255.255
  enable proxy

ttyu1:
  set ifaddr 203.14.100.1 203.14.100.21 255.255.255.255
  enable proxy
```

Anmerkung

Die Einrückungen sind wichtig.

Der Abschnitt `default:` wird für jede Sitzung geladen. Erstellen Sie für jede Einwählverbindung, die Sie in der Datei `/etc/ttys` ermöglicht haben, einen Eintrag, wie oben für `ttyu0:` gezeigt. Jede Verbindung sollte eine eigene IP-Adresse aus dem Pool der Adressen bekommen, die sie für diese Benutzergruppe reserviert haben.

27.2.1.2.9. Einrichten von `ppp.conf` für statische IP-Adressen

Zu dem bisher dargestellten Inhalt der Beispieldatei `/usr/share/examples/ppp/ppp.conf` sollten Sie einen Abschnitt für jeden Benutzer mit statisch zugewiesener IP-Adresse hinzufügen. Wir werden nun unser Beispiel mit den Accounts `fred`, `sam` und `mary` weiterführen.

```
fred:
  set ifaddr 203.14.100.1 203.14.101.1 255.255.255.255

sam:
  set ifaddr 203.14.100.1 203.14.102.1 255.255.255.255

mary:
  set ifaddr 203.14.100.1 203.14.103.1 255.255.255.255
```

Die Datei `/etc/ppp/ppp.linkup` sollte, falls erforderlich, ebenfalls Routinginformationen für jeden Benutzer mit statischer IP-Adresse enthalten. Die unten dargestellte Zeile würde dem Netzwerk `203.14.101.0/24` eine Route über die PPP-Verbindung des Client hinzufügen.

```
fred:
  add 203.14.101.0 netmask 255.255.255.0 HISADDR

sam:
  add 203.14.102.0 netmask 255.255.255.0 HISADDR

mary:
```

```
  add 203.14.103.0 netmask 255.255.255.0 HISADDR
```

27.2.1.2.10. `mgetty` und AutoPPP

In der Voreinstellung wird `mgetty` mit der Option `AUTO_PPP` konfiguriert und kompiliert. Dadurch kann `mgetty` die LCP Phase von PPP-Verbindungen erkennen und automatisch eine ppp-Shell starten. Da hierbei jedoch die Log-in/Passwort-Sequenz nicht durchlaufen wird, ist es notwendig, Benutzer durch PAP oder CHAP zu authentifizieren.

In diesem Abschnitt wird davon ausgegangen, dass der Benutzer den Port comms/mgetty+sendfax auf seinem System kompiliert und installiert hat.

Stellen Sie sicher, dass die Datei `/usr/local/etc/mgetty+sendfax/login.config` Folgendes enthält:

```
/AutoPPP/ -     -        /etc/ppp/ppp-pap-dialup
```

Hierdurch wird `mgetty` angewiesen, das Skript `ppp-pap-dialup` für die erkannten PPP-Verbindungen auszuführen.

Erstellen Sie nun die Datei `/etc/ppp/ppp-pap-dialup` mit folgendem Inhalt (die Datei sollte ausführbar sein):

```
#!/bin/sh
exec /usr/sbin/ppp -direct pap$IDENT
```

Erstellen Sie bitte für jede Einwählverbindung, die Sie in `/etc/ttys` ermöglicht haben, einen korrespondierenden Eintrag in der Datei `/etc/ppp/ppp.conf` . Diese Einträge können problemlos, mit den Definitionen die wir weiter oben gemacht haben, koexistieren.

```
pap:
  enable pap
  set ifaddr 203.14.100.1 203.14.100.20-203.14.100.40
  enable proxy
```

Jeder Benutzer, der sich auf diese Weise anmeldet, benötigt einen Benutzernamen und ein Passwort in der Datei `/etc/ppp/ppp.secret` . Sie haben auch die Möglichkeit, Benutzer mit Hilfe von PAP zu authentifizieren, indem Sie der Datei `/etc/passwd` folgende Option hinzufügen:

```
enable passwdauth
```

Wenn Sie bestimmten Benutzern eine statische IP-Adresse zuweisen möchten, können Sie diese als drittes Argument in der Datei `/etc/ppp/ppp.secret` angeben. In `/usr/share/examples/ppp/ppp.secret.sample` finden Sie hierfür Beispiele.

27.2.1.2.11. MS-Erweiterungen

Es ist möglich PPP so zu konfigurieren, dass bei Bedarf DNS und NetBIOS Nameserveradressen bereitgestellt werden.

Um diese Erweiterungen für die PPP Version 1.x zu aktivieren, sollte der entsprechende Abschnitt der Datei `/etc/ppp/ppp.conf` um folgende Zeilen ergänzt werden:

```
enable msext
set ns 203.14.100.1 203.14.100.2
set nbns 203.14.100.5
```

Für PPP Version 2 und höher:

```
accept dns
set dns 203.14.100.1 203.14.100.2
set nbns 203.14.100.5
```

Damit werden den Clients die primären und sekundären Nameserveradressen sowie ein NetBIOS Nameserver-Host mitgeteilt.

In Version 2 und höher verwendet PPP die Werte, die in `/etc/resolv.conf` zu finden sind, wenn die Zeile `set dns` weggelassen wird.

27.2.1.2.12. Authentifizierung durch PAP und CHAP

Einige ISPs haben ihr System so eingerichtet, dass der Authentifizierungsteil eines Verbindungsaufbaus mit Hilfe von PAP oder CHAP-Mechanismen durchgeführt wird. Wenn dies bei Ihnen der Fall sein sollte, wird Ihnen Ihr ISP bei der Verbindung keinen `login:`-Prompt präsentieren, sondern sofort mit der Aushandlung der PPP-Verbindung beginnen.

PAP ist nicht so sicher wie CHAP, doch die Sicherheit ist hierbei normalerweise kein Problem, da Passwörter, obgleich von PAP im Klartext versandt, lediglich über die serielle Verbindung verschickt werden. Es gibt für Cracker wenig Möglichkeiten zu „lauschen“.

Zurückkommend auf die Abschnitte PPP und statische IP-Adressen oder PPP und dynamische IP-Adressen müssen folgende Veränderungen vorgenommen werden:

```
13      set authname MyUserName
14      set authkey MyPassword
15      set login
```

Zeile 13:
Diese Zeile legt Ihren PAP/CHAP Benutzernamen fest. Sie müssen den richtigen Wert für *MyUserName* eingeben.

Zeile 14:
Diese Zeile legt Ihr PAP/CHAP Passwort fest. Sie müssen den richtigen Wert für *MyPassword* eingeben. Sie können eine zusätzliche Zeile, wie etwa:

```
16      accept PAP
```

oder

```
16      accept CHAP
```

verwenden, um deutlich zu machen, dass dies beabsichtigt ist, aber sowohl PAP wie auch CHAP als standardmäßig akzeptiert werden.

Zeile 15:
Ihr ISP wird normalerweise nicht von Ihnen verlangen, dass Sie sich am Server einloggen, wenn Sie PAP oder CHAP verwenden. Sie müssen deshalb den String „set login“ deaktivieren.

27.2.1.2.13. Veränderung Ihrer `ppp` Konfiguration im laufenden Betrieb

Es ist möglich, dem Programm `ppp` Befehle zu erteilen, während es im Hintergrund läuft. Dazu ist jedoch die Einrichtung eines passenden Diagnose-Ports erforderlich. Ergänzen Sie hierzu Ihre Konfigurationsdatei um folgende Zeile:

```
set server /var/run/ppp-tun%d DiagnosticPassword 0177
```

Damit wird PPP angewiesen, auf den angegebenen UNIX-Domainsocket zu hören und Clients nach dem angegebenen Passwort zu fragen, bevor der Zugang Gewährt wird. Das `%d` wird durch die Nummer des benutzten `tun`-Devices ersetzt.

Wenn ein Socket eingerichtet ist, kann das Programm pppctl(8) in Skripten verwendet werden, mit denen in das laufende Programm eingegriffen wird.

27.2.1.3. Interne NAT von PPP benutzen

PPP kann Network Address Translation (NAT) ohne Hilfe des Kernels durchführen. Wenn Sie diese Funktion benutzen wollen, fügen Sie die folgende Zeile in `/etc/ppp/ppp.conf` ein:

```
nat enable yes
```

Sie können NAT mit der Option `-nat` auf der Kommandozeile von PPP aktivieren. Weiterhin kann NAT in `/etc/rc.conf` mit der Variablen `ppp_nat` aktiviert werden. Dies ist auch die Voreinstellung.

Die nachstehende `/etc/ppp/ppp.conf` benutzt NAT für bestimmte eingehende Verbindungen:

```
nat port tcp 10.0.0.2:ftp ftp
nat port tcp 10.0.0.2:http http
```

Wenn Sie Verbindungen von außen überhaupt nicht trauen, benutzen Sie die folgende Zeile:

```
nat deny_incoming yes
```

27.2.1.4. Abschließende Systemkonfiguration

Sie haben `ppp` nun konfiguriert, aber bevor PPP eingesetzt werden kann, gibt noch einige weitere Dinge zu erledigen, die alle die Bearbeitung der Datei `/etc/rc.conf` erfordern.

Gehen Sie diese Datei von oben nach unten durch, und stellen Sie als Erstes sicher, dass die Zeile `hostname=` vorhanden ist:

```
hostname="foo.example.com"
```

Wenn Ihnen Ihr ISP eine statische IP-Adresse und einen Namen zugewiesen hat, ist es wahrscheinlich am besten, wenn Sie diesen Namen als Hostnamen verwenden.

Schauen Sie nach der Variable `network_interfaces`. Wenn Sie Ihr System so konfigurieren möchten, dass bei Bedarf eine Verbindung zu Ihrem ISP aufgebaut wird, sollten Sie das Device `tun0` zu der Liste hinzufügen oder es andernfalls entfernen.

```
network_interfaces="lo0 tun0"
ifconfig_tun0=
```

Anmerkung

Die Variable `ifconfig_tun0` sollte leer sein und eine Datei namens `/etc/start_if.tun0` sollte erstellt werden. Diese Datei sollte die nachfolgende Zeile enthalten:

```
ppp -auto mysystem
```

Dieses Skript startet Ihren ppp-Dæmon im Automatik-Modus. Es wird bei der Netzwerkkonfiguration ausgeführt. Wenn Ihr Rechner als Gateway für ein LAN fungiert, möchten Sie vielleicht auch die Option `-alias` verwenden. In der Manual-Seite sind weitere Einzelheiten hierzu zu finden.

Stellen Sie sicher, dass der Start eines Routerprogramms in `/etc/rc.conf` wie folgt deaktiviert ist:

```
router_enable="NO"
```

Es ist wichtig, dass der `routed` Dæmon nicht gestartet wird da `routed` dazu tendiert, die von `ppp` erstellten Einträge der Standardroute zu überschreiben.

Es ist außerdem sinnvoll, darauf zu achten, dass die Zeile `sendmail_flags` nicht die Option `-q` enthält, da `sendmail` sonst ab und zu die Netzwerkverbindung prüfen wird, was möglicherweise dazu führt, dass sich Ihr Rechner einwählt. Sie können hier Folgendes angeben:

```
sendmail_flags="-bd"
```

Der Nachteil dieser Lösung ist, dass Sie `sendmail` nach jedem Aufbau einer ppp-Verbindung auffordern müssen, die Mailwarteschlange zu überprüfen, indem Sie Folgendes eingeben:

```
# /usr/sbin/sendmail -q
```

Vielleicht möchten Sie den Befehl `!bg` in der Datei `ppp.linkup` verwenden, um dies zu automatisieren:

```
1     provider:
2       delete ALL
3       add 0 0 HISADDR
4       !bg sendmail -bd -q30m
```

Wenn Sie dies nicht möchten, ist es möglich, einen „dfilter“ einzusetzen, um SMTP-Verkehr zu blockieren. Weitere Einzelheiten hierzu finden Sie in den Beispieldateien.

Das Einzige, was nun noch zu tun bleibt, ist Ihren Rechner neu zu starten. Nach dem Neustart können Sie entweder:

```
# ppp
```

und danach `dial provider` eingeben, um eine PPP-Sitzung zu starten, oder Sie geben:

```
# ppp -auto provider
```

ein, um `ppp` bei Datenverkehr aus Ihrem Netzwerk heraus, automatisch eine Verbindung herstellen zu lassen (vorausgesetzt Sie haben kein `start_if.tun0` Skript erstellt).

27.2.1.5. Zusammenfassung

Die folgenden Schritte sind nötig, wenn ppp zum ersten Mal eingerichtet werden soll:

Clientseite:

1. Stellen Sie sicher, dass das `tun` Device in den Kernel eingebaut ist.
2. Vergewissern Sie sich, dass die Gerätedatei `tunN` im Verzeichnis `/dev` vorhanden ist.
3. Bearbeiten Sie die Datei `/etc/ppp/ppp.conf` . Das Beispiel `pmdemand` sollte für die meisten ISP ausreichen.
4. Wenn Sie eine dynamische IP-Adresse haben, erstellen Sie einen Eintrag in der Datei `/etc/ppp/ppp.linkup` .
5. Aktualisieren Sie die Datei `/etc/rc.conf` .
6. Erstellen Sie das Skript `start_if.tun0`, wenn Sie einen bedarfgesteuerten Einwahlprozess (*demand dialing*) benötigen.

Serverseite:

1. Stellen Sie sicher, dass das `tun` Device in den Kernel eingebaut ist.
2. Vergewissern Sie sich, dass die Gerätedatei `tunN` im Verzeichnis `/dev` vorhanden ist
3. Erstellen Sie einen Eintrag in der Datei `/etc/passwd` (verwenden Sie dazu das Programm vipw(8)).
4. Erstellen Sie ein Profil im Heimatverzeichnis des Benutzers, das `ppp -direct direct-server` o.Ä. ausführt.
5. Bearbeiten Sie die Datei `/etc/ppp/ppp.conf` . Das Beispiel `direct-server` sollte ausreichen.
6. Erzeugen Sie einen Eintrag in `/etc/ppp/ppp.linkup` .
7. Aktualisieren Sie die Datei `/etc/rc.conf` .

27.3. Probleme bei PPP-Verbindungen

Beigetragen von Tom Rhodes.

Dieser Abschnitt behandelt Probleme, die auftauchen können, wenn PPP über ein Modem verwendet wird. Sie müssen beispielsweise genau die Eingabeaufforderung des Systems kennen, in das Sie sich einwählen. Einige ISPs verwenden `ssword` andere verwenden `password`; wenn das Einwahlskript falsch ist, scheitert die Anmeldung. Üblicherweise suchen Sie nach Fehlern der PPP-Verbindung indem Sie sich manuell verbinden. Wie das genau geht, wird im Folgenden gezeigt.

27.3.1. Gerätedateien überprüfen

Wenn Sie einen eigenen Kernel verwenden, stellen Sie sicher, dass die folgende Zeile in der Kernelkonfigurationsdatei vorhanden ist:

```
device   uart
```

Das `uart`-Gerät ist bereits im `GENERIC` -Kernel vorhanden, deshalb sind in diesem Fall keine zusätzlichen Schritte vonnöten. Kontrollieren Sie die Ausgabe von `dmesg`:

```
# dmesg | grep uart
```

In der Ausgabe sollten die entsprechenden `uart`-Geräte, beispielsweise `uart1` (`COM2`), angezeigt werden. Wird ein passendes Gerät angezeigt, brauchen Sie keinen neuen Kernel zu erstellen. Wenn Ihr Modem an `uart1` angeschlossen ist (in DOS ist dieser Anschluss als `COM2` bekannt), ist `/dev/cuau1` die dazugehörende Gerätedatei.

27.3.2. Manuelle Verbindungen

Ein Verbindungsaufbau zum Internet durch manuelle Steuerung von `ppp` geht schnell, ist einfach und stellt einen guten Weg dar, eine Verbindung auf Fehler hin zu überprüfen oder einfach Informationen darüber zu sammeln, wie Ihr ISP Verbindungen handhabt. Lassen Sie uns PPP von der Kommandozeile aus starten. Beachten Sie, dass in allen Beispielen *example* der Hostname der Maschine ist, auf der PPP läuft. Sie starten `ppp`, indem Sie einfach `ppp` eingeben:

```
# ppp
```

Wir haben `ppp` nun gestartet.

```
ppp ON example> set device /dev/cuau1
```

Wir geben das Device an, an das unser Modem angeschlossen ist. In diesem Fall ist es `cuau1`.

```
ppp ON example> set speed 115200
```

Wir geben die Verbindungsgeschwindigkeit an. Im Beispiel verwenden wir 115200 kbps

```
ppp ON example> enable dns
```

Wir weisen `ppp` an, unseren Resolver zu konfigurieren und in der Datei `/etc/resolv.conf` Einträge für den Nameserver hinzuzufügen. Falls `ppp` unseren Hostnamen nicht bestimmen kann, geben wir diesen später manuell an.

```
ppp ON example> term
```

Wir wechseln in den „Terminal“-Modus, um das Modem manuell kontrollieren zu können.

```
deflink: Entering terminal mode on /dev/cuau1
type '~h' for help
```

```
at
OK
atdt123456789
```

Sie verwenden `at` zur Initialisierung Ihres Modems und dann `atdt` sowie die Nummer Ihres ISP, um den Einwählprozess zu starten.

```
CONNECT
```

Dies ist die Bestätigung, dass eine Verbindung aufgebaut wurde. Falls wir Verbindungsprobleme bekommen, die nicht mit der Hardware zusammenhängen, werden wir an dieser Stelle ansetzen müssen, um eine Lösung zu finden.

```
ISP Login:myusername
```

Hier werden Sie nach einem Benutzernamen gefragt. Geben Sie am Prompt den Namen ein, den Ihnen Ihr ISP zur Verfügung gestellt hat.

```
ISP Pass:mypassword
```

An dieser Stelle müssen Sie das Passwort angeben, das Ihnen von Ihrem ISP vorgegeben wurde. Das Passwort wird, analog dem normalen Anmeldevorgang, nicht angezeigt.

```
Shell or PPP:ppp
```

Abhängig von Ihrem ISP, kann es sein, dass dieser Prompt bei Ihnen gar nicht erscheint. Wir werden hier gefragt, ob wir eine Shell beim Provider verwenden oder `ppp` starten wollen. Weil wir eine Internetverbindung aufbauen wollen, haben wir uns in diesem Beispiel für `ppp` entschieden.

```
Ppp ON example>
```

Beachten Sie, dass sich in diesem Beispiel das erste `p` in einen Großbuchstaben verwandelt hat. Dies zeigt, dass wir erfolgreich eine Verbindung zu unserem ISP hergestellt haben.

```
PPp ON example>
```

An dieser Stelle haben wir uns erfolgreich bei unserem ISP authentifiziert und warten darauf, dass uns eine IP-Adresse zugewiesen wird.

```
PPP ON example>
```

Wir haben uns mit der Gegenstelle auf eine IP-Adresse geeinigt und den Verbindungsaufbau erfolgreich abgeschlossen

```
PPP ON example> add default HISADDR
```

Hier geben wir unsere Standardroute an. Weil zu diesem Zeitpunkt unsere einzige Verbindung zu unserer Gegenstelle besteht, müssen wir dies tun, bevor wir Kontakt zu unserer Umwelt aufnehmen können. Falls dies aufgrund bestehender Routen nicht funktionieren sollte, können Sie ein Ausrufungszeichen `!` vor `add` setzen. Sie können diese Standardroute aber auch vor dem eigentlichen Verbindungsaufbau angeben und PPP wird entsprechend eine neue Route aushandeln.

Wenn alles gut ging, sollten wir nun eine aktive Internetverbindung haben, die wir mit Ctrl+z in den Hintergrund schicken können Wenn sie feststellen, dass `PPP` wieder zu `ppp` wird, ist die Verbindung abgebrochen. Es ist gut dies zu wissen, weil dadurch der Verbindungsstatus angezeigt wird. Große `Ps` zeigen an, dass wir eine Verbindung zum ISP haben und kleine `ps` zeigen an, dass wir aus irgendeinem Grund die Verbindung verloren haben. `ppp` hat nur diese beiden Zustände.

27.3.2.1. Fehlersuche

Wenn sie einen Direktanschluss haben und keine Verbindung aufbauen können, schalten Sie die Hardware-Flusssteuerung CTS/RTS aus, indem Sie die Option `set ctsrts off` verwenden. Dies ist zumeist dann der Fall, wenn Sie mit einem PPP-fähigen Terminalserver verbunden sind. Hier bleibt PPP bei dem Versuch hängen, Daten über Ihre Nachrichtenverbindung zu schicken, weil auf ein CTS-Signal (Clear-to-Send) gewartet wird, das nie kommt. Wenn Sie diese Option jedoch gebrauchen, sollten Sie auch die Option `set accmap` verwenden, die erforderlich sein kann, um bestimmte Hardware zu kontrollieren, die auf die Übertragung bestimmter Zeichen zwischen den

Kommunikations-Endpunkten (zumeist XON/XOFF) angewiesen ist. Die Manual-Seite ppp(8) bietet mehr Informationen zu dieser Option und ihrer Verwendung.

Wenn Sie ein älteres Modem haben, benötigen Sie vielleicht die Option `set parity even`. Standardmäßig wird keine Parität vorausgesetzt, sie ist aber für die Fehlerprüfung bei älteren Modems und bei bestimmten ISPs erforderlich. Sie könnten diese Option für den ISP Compuserve benötigen.

PPP kehrt möglicherweise nicht in den Befehlsmodus zurück, was normalerweise auf einen Fehler bei der Aushandlung hinweist, wobei der ISP wartet, dass Ihre Seite den Aushandlungsprozess beginnt. Die Option `~p` erzwingt in diesem Fall den Beginn des Aushandlungsprozesses.

Wenn Sie nie einen Login-Prompt erhalten, müssen Sie statt des im Beispiel gezeigten UNIX-Stils höchst wahrscheinlich PAP oder CHAP für die Authentifizierung verwenden. Um PAP oder CHAP zu verwenden, ergänzen Sie PPP einfach um folgende Optionen, bevor Sie in den Terminalmodus wechseln:

```
ppp ON example> set authname myusername
```

Hierbei sollte *myusername* durch den Benutzernamen ersetzt werden, den Sie von Ihrem ISP bekommen haben.

```
ppp ON example> set authkey mypassword
```

mypassword sollten Sie durch das Passwort ersetzen, das Ihnen Ihr ISP gegeben hat.

Wenn die Verbindung aufgebaut wird, Sie aber keine Rechner unter ihrem Domänen-Namen erreichen können, versuchen Sie, einen Rechner mit ping(8) und seiner IP-Adresse zu erreichen. Wenn 100% der Pakete verloren gehen, ist es sehr wahrscheinlich, dass Ihnen keine Standardroute zugewiesen wurde. Überprüfen Sie, ob während des Verbindungsaufbaus die Option `add default HISADDR` gesetzt war. Wenn Sie zu einer entfernten IP-Adresse eine Verbindung aufbauen können, ist es möglich, dass die Adresse eines Nameservers nicht in die Datei `/etc/resolv.conf` eingetragen wurde. Diese Datei sollte folgendermaßen aussehen:

```
domain example.com
nameserver x.x.x.x
nameserver y.y.y.y
```

Dabei sollten *x.x.x.x* und *y.y.y.y* durch die IP-Adressen der DNS-Server Ihres ISPs ersetzt werden. Diese Information ist Ihnen bei Vertragsabschluss mitgeteilt worden. Wenn nicht, sollte ein Anruf bei Ihrem ISP Abhilfe schaffen.

Mit syslog(3) können Sie Ihre PPP-Verbindung protokollieren. Fügen Sie einfach die folgende Zeile in `/etc/syslog.conf` ein:

```
!ppp
*.*     /var/log/ppp.log
```

In den meisten Fällen existiert diese Funktionalität bereits.

27.4. PPP over Ethernet (PPPoE)

Beigetragen (durch http://node.to/freebsd/how-tos/how-to-freebsd-pppoe.html) von Jim Mock.

Dieser Abschnitt beschreibt, wie Sie PPP over Ethernet (PPPoE) einrichten.

27.4.1. Konfiguration des Kernels

Eine besondere Kernelkonfiguration ist für PPPoE nicht mehr erforderlich. Sofern die notwendige NetGraph-Unterstützung nicht in den Kernel eingebaut wurde, wird diese von ppp dynamisch geladen.

27.4.2. Einrichtung von `ppp.conf`

Dies hier ist ein Beispiel einer funktionierenden `ppp.conf`:

```
default:
  set log Phase tun command # you can add more detailed logging if you wish
  set ifaddr 10.0.0.1/0 10.0.0.2/0

name_of_service_provider:
  set device PPPoE:xl1 # replace xl1 with your Ethernet device
  set authname YOURLOGINNAME
  set authkey YOURPASSWORD
  set dial
  set login
  add default HISADDR
```

27.4.3. ppp ausführen

Als `root`, geben Sie ein:

```
# ppp -ddial name_of_service_provider
```

27.4.4. ppp beim Systemstart ausführen

Fügen Sie Folgendes in Ihre Datei `/etc/rc.conf` ein:

```
ppp_enable="YES"
ppp_mode="ddial"
ppp_nat="YES" # if you want to enable nat for your local network, otherwise NO
ppp_profile="name_of_service_provider"
```

27.4.5. Verwendung einer PPPoE-Dienstbezeichnung (service tag)

Manchmal kann es notwendig sein, eine Dienstbezeichnung (*service tag*) zu verwenden, um eine Verbindung aufzubauen. Dienstbezeichnungen werden eingesetzt, um zwischen verschiedenen PPPoE-Servern unterscheiden zu können, die einem bestehenden Netzwerk zugeteilt sind.

Die erforderlichen Dienstbezeichnungen sollten in der Dokumentation, zu finden sein, die Ihnen Ihr ISP zur Verfügung gestellt hat. Wenn Sie diese Informationen dort nicht finden, fragen Sie beim technischen Kundendienst Ihres ISP danach.

Als letzte Möglichkeit, bleibt die Methode, die von dem Programm Roaring Penguin PPPoE vorgeschlagen wird, das in der Ports-Sammlung zu finden ist. Bedenken Sie aber, dass dadurch Daten Ihres Modems gelöscht werden können, so dass es nicht mehr benutzt werden kann. Überlegen Sie also genau, ob Sie dies machen wollen. Installieren Sie einfach das Programm, das Ihnen Ihr Provider zusammen mit dem Modem geliefert hat. Gehen Sie dann in das Menü System dieses Programms. Der Name Ihres Profils, sollte in der Liste aufgeführt sein. Normalerweise ist dies *ISP*.

Der Name des Profils (*service tag*) wird im Eintrag für die PPPoE-Konfiguration in der Datei `ppp.conf` verwendet, als der Teil des Befehls `set device` (die manpage ppp(8) enthält Einzelheiten hierzu), der den Provider angibt. Dieser Eintrag sollte folgendermaßen aussehen:

```
set device PPPoE:xl1:ISP
```

Vergessen Sie nicht, statt *`xl1`* das richtige Device Ihrer Netzwerkkarte anzugeben.

Denken sie auch daran, *`ISP`* durch das Profil, das Sie oben gefunden haben zu ersetzen.

Weitere Informationen bieten:

- Cheaper Broadband with FreeBSD on DSL von Renaud Waldura.

27.4.6. PPPoE mit einem 3Com® HomeConnect® ADSL Modem Dual Link

Dieses Modem folgt nicht dem RFC 2516 (*A Method for transmitting PPP over Ethernet (PPPoE)*, verfasst von L. Mamakos, K. Lidl, J. Evarts, D. Carrel, D. Simone, und R. Wheeler). Stattdessen wurden andere Pakettyp-Codes für die Ethernet

Frames verwendet. Bitte beschweren Sie sich unter 3Com, wenn Sie der Ansicht sind, dass dieses Modem die PPPoE-Spezifikation einhalten sollte.

Um FreeBSD in die Lage zu versetzen, mit diesem Gerät zu kommunizieren, muss ein sysctl Befehl angegeben werden. Dies kann beim Systemstart automatisch geschehen, indem die Datei `/etc/sysctl.conf` angepasst wird:

```
net.graph.nonstandard_pppoe=1
```

oder, wenn der Befehl unmittelbar wirksam werden soll, durch:

```
# sysctl net.graph.nonstandard_pppoe=1
```

Da hiermit eine systemweit gültige Einstellung vorgenommen wird, ist es nicht möglich, gleichzeitig mit einem normalen PPPoE-Client oder Server und einem 3Com® HomeConnect® ADSL Modem zu kommunizieren.

27.5. PPP over ATM (PPPoA)

Nachfolgend wird beschrieben, wie PPP over ATM (PPPoA) eingerichtet wird. PPPoA ist vor allem unter europäischen DSL-Providern populär.

27.5.1. Der Einsatz von PPPoA mit dem Alcatel SpeedTouch™ USB

PPPoA-Unterstützung für dieses Gerät ist unter FreeBSD als Port verfügbar, da die Firmware unter Alcatels Lizenzvereinbarung vertrieben wird und deshalb nicht mit dem FreeBSD-Basissystem frei verteilt werden kann.

Um die Software zu installieren, verwenden Sie einfach die Ports-Sammlung. Installieren Sie den Port net/pppoa und folgen Sie den dabei angegebenen Instruktionen.

Für den ordnungsgemäßen Betrieb muss das Alcatel SpeedTouch™ USB, wie viele USB-Geräte, Firmware auf den Gastrechner laden. FreeBSD kann die Firmware automatisch laden, wenn das Gerät mit dem USB-Anschluss verbunden wird. Dazu fügen Sie als Benutzer `root` die nachstehenden Zeilen in `/etc/usbd.conf` ein:

```
device "Alcatel SpeedTouch USB"
    devname "ugen[0-9]+"
    vendor 0x06b9
    product 0x4061
    attach "/usr/local/sbin/modem_run -f /usr/local/libdata/mgmt.o"
```

Den USB-Dæmon aktivieren Sie mit der folgenden Zeile in `/etc/rc.conf` :

```
usbd_enable="YES"
```

Wenn die Verbindung beim Start von ppp aufgebaut werden soll, fügen Sie die nachstehenden Zeilen als Benutzer `root` in `/etc/rc.conf` ein:

```
ppp_enable="YES"
ppp_mode="ddial"
ppp_profile="adsl"
```

Verwenden Sie bitte diese Einstellungen zusammen mit der Beispielkonfiguration in `ppp.conf` des Ports net/pppoa.

27.5.2. Die Verwendung von mpd

Sie können mpd verwenden, um zu einer Reihe von Diensten, insbesondere PPTP-Diensten eine Verbindung herzustellen. Sie finden mpd in der Ports-Sammlung unter net/mpd. Viele ADSL Modems, wie das Alcatel SpeedTouch™ Home, sind auf einen PPTP-Tunnel zwischen dem Modem und dem Rechner angewiesen.

Zuerst müssen Sie den Port installieren, um danach mpd entsprechend Ihren Anforderungen und den Vorgaben Ihres Providers konfigurieren zu können. Der Port installiert auch einige gut dokumentierte Beispielkonfigurati-

onsdateien in PREFIX/etc/mpd/ . Beachten Sie, dass *PREFIX* hier das Verzeichnis angibt, in das Ihre Ports installiert werden. Standardmäßig ist dies das Verzeichnis /usr/local/ . Ein kompletter Leitfaden zur Konfiguration von mpd ist im HTML-Format verfügbar, sobald der Port installiert ist. Dieser ist in PREFIX/share/doc/mpd/ zu finden. Hier ist eine Beispielkonfiguration, um mit mpd eine Verbindung zu einem ADSL-Dienst aufzubauen. Die Konfiguration ist auf zwei Dateien verteilt. Zunächst die Datei mpd.conf :

Anmerkung

Dieses Beispiel für mpd.conf funktioniert nur mit mpd 4.x.

```
default:
    load adsl

adsl:
    new -i ng0 adsl adsl
    set bundle authname username ❶
    set bundle password password ❷
    set bundle disable multilink

    set link no pap acfcomp protocomp
    set link disable chap
    set link accept chap
    set link keep-alive 30 10

    set ipcp no vjcomp
    set ipcp ranges 0.0.0.0/0 0.0.0.0/0

    set iface route default
    set iface disable on-demand
    set iface enable proxy-arp
    set iface idle 0

    open
```

❶ Der Benutzername, den Sie zur Authentifizierung bei Ihrem ISP verwenden.
❷ Das Passwort, das Sie zur Authentifizierung bei Ihrem ISP verwenden.

Die Datei mpd.links enthält Informationen über die Verbindung(en), die Sie aufbauen möchten. Eine Beispieldatei mpd.links , die das vorige Beispiel ergänzt, wird unten angegeben:

```
adsl:
    set link type pptp
    set pptp mode active
    set pptp enable originate outcall
    set pptp self 10.0.0.1 ❶
    set pptp peer 10.0.0.138 ❷
```

❶ Die IP-Adresse des FreeBSD-Rechners von dem aus Sie mpd verwenden.
❷ Die IP-Adresse des ADSL-Modems. Das Alcatel SpeedTouch™ Home hat die Adresse 10.0.0.138 voreingestellt.

Ein Verbindungsaufbau kann einfach durch Eingabe des folgenden Befehls als root gestartet werden:

```
# mpd -b adsl
```

Sie können sich den Status der Verbindung durch folgenden Befehl anzeigen lassen:

```
% ifconfig ng0
ng0: flags=88d1<UP,POINTOPOINT,RUNNING,NOARP,SIMPLEX,MULTICAST> mtu 1500
```

```
     inet 216.136.204.117 --> 204.152.186.171 netmask 0xffffffff
```

Die Verwendung von mpd ist der empfehlenswerteste Weg, um mit FreeBSD eine Verbindung zu einem ADSL-Dienst aufzubauen.

27.5.3. Die Verwendung von pptpclient

Es ist außerdem möglich, mit FreeBSD eine Verbindung zu anderen PPPoA-Diensten aufzubauen. Dazu wird net/pptpclient verwendet.

Um mit net/pptpclient eine eine Verbindung zu einem DSL-Dienst aufbauen zu können, müssen Sie den entsprechenden Port bzw. das Paket installieren und die Datei `/etc/ppp/ppp.conf` bearbeiten. Sie müssen `root` sein, um diese Schritte durchführen zu können. Eine Beispieldatei für `ppp.conf` ist weiter unten angegeben. Weitere Informationen zu den Optionen von `ppp.conf` bietet die Manual-Seite ppp ppp(8):

```
adsl:
 set log phase chat lcp ipcp ccp tun command
 set timeout 0
 enable dns
 set authname username ❶
 set authkey password ❷
 set ifaddr 0 0
 add default HISADDR
```

❶ Der Benutzername für den Zugang zu den Diensten Ihres ISP.
❷ Das Passwort für Ihren Account.

Warnung

Weil Sie Ihr Passwort in der Datei `ppp.conf` in Klartext angeben müssen, sollten Sie sicherstellen, dass niemand den Inhalt dieser Datei lesen kann. Die folgende Reihe von Befehlen stellt sicher, dass die Datei nur von `root` lesbar ist. Zusätzliche Informationen bieten die Manual-Seiten chmod(1) und chown(8):

```
# chown root:wheel /etc/ppp/ppp.conf
# chmod 600 /etc/ppp/ppp.conf
```

Dies wird einen Tunnel für eine PPP-Session zu Ihrem DSL-Router öffnen. Ethernet-DSL-Modems haben eine vorkonfigurierte LAN-IP-Adresse, mit der Sie eine Verbindung aufbauen. Im Falle des Alcatel SpeedTouch™ Home handelt es sich dabei um die Adresse `10.0.0.138`. In der Dokumentation Ihres Routers sollte angegeben sein, welche Adresse Ihr Gerät verwendet. Um den Tunnel zu öffnen und eine PPP-Session zu starten, führen Sie bitte folgenden Befehl aus:

```
# pptp address adsl
```

Tipp

Vielleicht möchten Sie ein kaufmännisches Und („&“) an das Ende oben angegebenen Kommandos anfügen, da pptp sonst den Prompt nicht zurückgibt.

Ein virtuelles Tunnel-Device `tun` wird für das Zusammenspiel der Prozesse pptp und ppp geschaffen. Wenn Sie den Prompt zurückerhalten haben oder der pptp-Prozess das Vorliegen einer Verbindung bestätigt, können Sie den Tunnel folgendermaßen überprüfen:

```
% ifconfig tun0
```

```
tun0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1500
        inet 216.136.204.21 --> 204.152.186.171 netmask 0xffffff00
        Opened by PID 918
```

Wenn Sie nicht in der Lage sein sollten, eine Verbindung aufzubauen, überprüfen Sie die Konfiguration Ihres Routers, den Sie normalerweise per telnet oder mit einem Web-Browser erreichen können. Falls dennoch keine Verbindung zustande kommt, sollten Sie die Ausgabe des Befehls `pptp` und die Logdatei `/var/log/ppp.log` von ppp nach Hinweisen auf die Ursache durchsuchen.

Kapitel 28. Elektronische Post (E-Mail)

Ursprüglicher Text von Bill Lloyd.
Neugeschrieben von Jim Mock.
Übersetzt von Robert Drehmel.

28.1. Terminologie

Das Akronym *MTA* steht für *Mail Transfer Agent* was übersetzt „Mailübertragungs-Agent“ bedeutet.

Während die Bezeichnung *Server-Dämon* die Komponente eines MTA benennt, die für eingehende Verbindungen zuständig ist, wird mit dem Begriff *Mailer* öfters die Komponente des MTA bezeichnet, die E-Mails versendet.

28.2. Übersicht

„Elektronische Post“, besser bekannt als E-Mail, ist eine der am weit verbreitetsten Formen der Kommunikation heutzutage. Dieses Kapitel bietet eine grundlegende Einführung in das Betreiben eines E-Mail-Servers unter FreeBSD. Ebenfalls wird der Versand und Empfang von E-Mails unter FreeBSD behandelt. Eine umfassende Betrachtung zu diesem Thema finden Sie in den Büchern, die in Anhang B, *Bibliografie* aufgelistet sind.

Dieses Kapitel behandelt die folgenden Punkte:

- Welche Software-Komponenten beim Senden und Empfangen von elektronischer Post involviert sind.
- Wo sich grundlegende Sendmail Konfigurationsdateien in FreeBSD befinden.
- Den Unterschied zwischen entfernten und lokalen Postfächern.
- Wie man Versender von Spam daran hindern kann, E-Mail-Server illegalerweise als Weiterleitung zu verwenden.
- Wie man einen alternativen MTA installiert und konfiguriert, um Sendmail zu ersetzen.
- Wie man oft auftretende E-Mail-Server Probleme behebt.
- Wie E-Mails über einen Relay verschickt werden.
- Wie E-Mails über eine Einwahlverbindung gehandhabt werden.
- Wie SMTP-Authentifizierung einrichtet wird.
- Den Empfang und den Versand von E-Mails mithilfe von Programmen wie mutt.
- Wie E-Mails von einem entfernten Server mit POP oder IMAP abgeholt werden.
- Wie eingehende E-Mail automatisch gefiltert wird.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- Die Netzwerk-Verbindung richtig einrichten. (Kapitel 31, *Weiterführende Netzwerkthemen*).
- Die DNS-Information für einen E-Mail-Server einstellen (Kapitel 29, *Netzwerkserver*).

- Wissen, wie man zusätzliche Dritthersteller-Software installiert (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

28.3. E-Mail Komponenten

Es gibt fünf größere Komponenten die am Austausch von E-Mails beteiligt sind: der *Mail User Agent* (MUA), der *Mail Transfer Agent* (MTA), der *Mail Host*, ein entferntes oder lokales Postfach, sowie DNS. Dieser Abschnitt enthält eine Übersicht über diese Komponenten.

Mail User Agent (MUA)
: Der *Mail User Agent* (MUA) ist das Benutzerprogramm zum Verfassen, Senden und Empfangen von E-Mails. Diese Anwendung kann ein Kommandozeilenprogramm sein, wie das in FreeBSD enthaltene Programm `mail`, oder ein Programm aus der Ports-Sammlung wie beispielsweise mutt, alpine oder elm. In der Ports-Sammlung sind auch dutzende von grafischen Programmen verfügbar, darunter ClawsMail, Evolution und Thunderbird. Einige Unternehmen bieten auch ein Web-Mail-Programm an, das über einen Webbrowser verwaltet werden kann. Weitere Informationen zur Installation und Verwendung von MUAs unter FreeBSD finden Sie im Abschnitt 28.11, „E-Mail-Programme“.

Mail Transfer Agent (MTA)
: Der *Mail Transfer Agent* (MTA) ist ein E-Mail-Server Daemon, welcher für dem Empfang von eingehenden E-Mails und für den Versand von ausgehenden E-Mails verantwortlich ist. FreeBSD wird mit Sendmail als Standard-MTA ausgeliefert, aber es unterstützt auch weitere E-Mail-Server, darunter Exim, Postfix und qmail. Die Konfiguration von Sendmail wird im Abschnitt 28.4, „Sendmail-Konfigurationsdateien“ beschrieben. Wenn Sie einen anderen MTA aus der Ports-Sammlung installieren, lesen Sie die Nachrichten die nach der Installation der Anwendung ausgegeben werden, wenn Sie FreeBSD spezifische Informationen benötigen. Allgemeine Informationen zur Konfiguration finden Sie in der Regel auf der Webseite des Herstellers.

Mail Host und Postfächer
: Der *Mail Host* ist für die Zustellung und das Empfangen von E-Mails für den Rechner oder eines Netzwerks zuständig. Der Mail Host empfängt alle E-Mails für eine Domäne und speichert diese entweder im voreingestellten `mbox`-Format, oder im Maildir-Format. Diese E-Mails können lokal mit einem Benutzerprogramm MUA gelesen werden. Mithilfe von Protokollen wie POP oder IMAP können die E-Mails auch von entfernten Rechnern gelesen werden. Wenn die E-Mails direkt auf dem Mail Host gelesen werden, wird kein POP- oder IMAP-Server benötigt.

  Um auf entfernte Postfächer zuzugreifen, wird ein Zugang zu einem POP- oder IMAP-Server benötigt. Beide Protokolle ermöglichen es Benutzern, auf ein entferntes Postfach zuzugreifen. IMAP bietet gegenüber POP einige Vorteiler. Dazu zählt die Fähigkeit eine Kopie aller Nachrichten auf einem entfernten Server zu speichern, sowie gleichzeitig ablaufende Aktualisierungen. IMAP kann auch über langsame Verbindungen nützlich sein, da nicht gleich die komplette Nachricht heruntergeladen wird. Weiterhin können E-Mails auf dem Server durchsucht werden, was den Datenverkehr zwischen Clients und dem Server minimiert.

  Die Ports-Sammlung enthält einige POP- und IMAP-Server, darunter mail/qpopper, mail/imap-uw, mail/courier-imap und mail/dovecot2.

Warnung

Beachten Sie, dass sowohl POP als auch IMAP Daten, wie den Benutzernamen und das Passwort, im Klartext übertragen. Um die Übermittlung von Daten über diese Protokolle zu schützen, können Sie Sitzungen über ssh(1) (Abschnitt 14.10.8, „SSH-Tunnel“) tunneln oder SSL (Abschnitt 14.8, „OpenSSL“) verwenden.

Domain Name System (DNS)
: Das *Domain Name System* (DNS) und sein Daemon `named` spielen eine große Rolle bei der Auslieferung von E-Mails. Um E-Mails auszuliefern, fragt der MTA im DNS den Rechner ab, der E-Mails für das Zielsystem entgegennimmt. Der gleiche Vorgang läuft ab, wenn eine E-Mail von einem entfernten Server zum MTA zugestellt wird.

  Im DNS werden Rechnernamen auf IP-Adressen abgebildet. Daneben werden spezielle Informationen für das Mail-System gespeichert, die *MX-Einträge* (*MX record*) genannt werden. Der MX-Eintrag (von *Mail eXchanger*) gibt an, welche Rechner E-Mails für eine Domäne annehmen.

  Mit host(1) können die MX-Einträge für eine Domäne abgefragt werden:

```
# host -t mx FreeBSD.org
FreeBSD.org mail is handled by 10 mx1.FreeBSD.org
```

  Weitere Informationen zu DNS und dessen Konfiguration finden Sie im Abschnitt 29.7, „Domain Name System (DNS)“.

28.4. Sendmail-Konfigurationsdateien

Beigesteuert von Christopher Shumway.

Sendmail ist der standardmäßig in FreeBSD installierte MTA. Es nimmt E-Mails von E-Mail-Benutzerprogrammen (MUA) entgegen und liefert diese zu den entsprechenden Mail Hosts, die in der Konfigurationsdatei definiert sind. Sendmail kann auch Netzwerkverbindungen annehmen und E-Mails an lokale *Mailboxen*, oder an andere Programme ausliefern.

Die Konfigurationsdateien von Sendmail befinden sich in `/etc/mail` . In diesem Abschnitt werden diese Dateien im Detail beschrieben.

`/etc/mail/access`
: Diese Datenbank bestimmt, welche Rechner oder IP-Adressen Zugriff auf den lokalen Mail-Server haben und welche Art von Zugriff ihnen gestattet wird. Rechner die als `OK` aufgelistet sind, was der Standard ist, sind berechtigt E-Mails zu diesem Rechner zu schicken, solange die endgültige Zieladresse der lokale Rechner ist. Rechner die als `REJECT` aufgelistet sind, werden abgelehnt. Rechner die als `RELAY` aufgelistet sind, wird es erlaubt Post für jede Zieladresse durch diesen Mail-Server zu senden. Rechner die als `ERROR` aufgelistet sind, bekommen ihre E-Mail mit einem speziellen Fehler zurück. Wenn ein Rechner als `SKIP` aufgelistet ist, wird Sendmail die aktuelle Suche abbrechen, ohne die E-Mail zu akzeptieren oder abzulehnen. E-Mails von Rechnern die als `QUARANTAINE` aufgelistet sind, werden vorerst zurückgehalten. Dem sendenden Rechner wird ein festgelegter Text als Grund für die Quarantäne zurückgeschickt.

  Beispiele für die Verwendung dieser Optionen für IPv4- und IPv6-Adressen finden Sie in der Beispielkonfiguration `/etc/mail/access.sample` :

```
# $FreeBSD: head/de_DE.ISO8859-1/books/handbook/mail/chapter.xml 48640 2016-04-15 ↺
16:47:41Z bhd $
#
# Mail relay access control list.  Default is to reject mail unless the
# destination is local, or listed in /etc/mail/local-host-names
#
## Examples (commented out for safety)
#From:cyberspammer.com          ERROR:"550 We don't accept mail from spammers"
#From:okay.cyberspammer.com     OK
#Connect:sendmail.org           RELAY
#To:sendmail.org                RELAY
#Connect:128.32                 RELAY
#Connect:128.32.2               SKIP
#Connect:IPv6:1:2:3:4:5:6:7     RELAY
```

```
#Connect:suspicious.example.com QUARANTINE:Mail from suspicious host
#Connect:[127.0.0.3]            OK
#Connect:[IPv6:1:2:3:4:5:6:7:8] OK
```

Um die Datenbank zu konfigurieren, verwenden Sie das im Beispiel gezeigte Format, um Einträge in `/etc/mail/access` hinzuzufügen, aber setzen Sie kein Kommentarsymbol (`#`) vor die Einträge. Erstellen Sie einen Eintrag für jeden Rechner, dessen Zugriff konfiguriert werden soll. E-Mail-Versender, die mit der linken Spalte der Tabelle übereinstimmen, sind betroffen von der Aktion in der rechten Spalte.

Immer wenn diese Datei verändert wurde, muss die Datenbank aktualisiert und Sendmail neu gestartet werden:

```
# makemap hash /etc/mail/access < /etc/mail/access
# service sendmail restart
```

`/etc/mail/aliases`

Diese Datenbank enthält eine Liste der virtuellen Mailboxen, die in andere Benutzer, Dateien, Programme oder andere Aliase expandiert werden. Hier sind ein paar Beispiele, die das Dateiformat verdeutlichen:

```
root: localuser
ftp-bugs: joe,eric,paul
bit.bucket:  /dev/null
procmail: "|/usr/local/bin/procmail"
```

Der Name der Mailbox auf der linken Seite des Doppelpunkts wird mit den Zielen auf der rechten Seite ersetzt. Der erste Eintrag ersetzt die Mailbox `root` mit der Mailbox `localuser`, die dann in der Datenbank `/etc/mail/aliases` gesucht wird. Wird kein passender Eintrag gefunden, wird die Nachricht zum `localuser` geliefert. Der zweite Eintrag zeigt eine E-Mail-Verteilerliste. E-Mails an `ftp-bugs` werden zu den drei lokalen Mailboxen `joe`, `eric` und `paul` gesendet. Eine entfernte Mailbox kann auch als *`user@example.com`* angegeben werden. Der dritte Eintrag zeigt wie E-Mails in eine Datei geschrieben werden, in diesem Fall `/dev/null` . Der letzte Eintrag verdeutlicht das Senden von E-Mails an ein Programm. Hier wird die Nachricht über eine UNIX® Pipe an `/usr/local/bin/procmail` gesendet. Weitere Informationen zu dem Format dieser Datei finden Sie in aliases(5).

Wenn diese Datei geändert wird, muss `newaliases` ausgeführt werden, um die Datenbank zu aktualisieren.

`/etc/mail/sendmail.cf`

Dies ist die Hauptkonfigurations-Datei von Sendmail. Sie kontrolliert das allgemeine Verhalten von Sendmail, einschließlich allem vom Umschreiben von E-Mail Adressen bis hin zum Übertragen von Ablehnungsnachrichten an entfernte E-Mail-Server. Dementsprechend ist die Konfigurationsdatei ziemlich komplex. Glücklicherweise muss diese Datei selten für Standard E-Mail-Server geändert werden.

Die Sendmail Hauptkonfigurationsdatei kann mit m4(1) Makros erstellt werden, die Eigenschaften und Verhalten von Sendmail definieren. Einige der Details finden Sie in `/usr/src/contrib/sendmail/cf/README` .

Wenn Änderungen an dieser Datei vorgenommen werden, muss Sendmail neu gestartet werden, damit die Änderungen Wirkung zeigen.

`/etc/mail/virtusertable`

Diese Datenbank ordnet Adressen für virtuelle Domänen und Benutzern reellen Mailboxen zu. Diese Mailboxen können lokal, auf entfernten Systemen, Aliase in `/etc/mail/aliases` oder eine Datei sein. Dadurch können mehrere virtuelle Domains auf einem Rechner gehostet werden.

FreeBSD enthält eine Beispielkonfiguration in `/etc/mail/virtusertable.sample` , die das Format genauer beschreibt. Das folgende Beispiel zeigt, wie benutzerdefinierte Einträge in diesem Format erstellt werden:

```
root@example.com                root
postmaster@example.com          postmaster@noc.example.net
@example.com                    joe
```

Diese Datei wird nach dem ersten übereinstimmenden Eintrag durchsucht. Wenn eine E-Mail-Adresse mit der Adresse auf der linken Seite übereinstimmt, wird sie dem Eintrag auf der rechten Seite zugeordnet. Der erste Eintrag in diesem Beispiel ordnet eine bestimmte E-Mail-Adresse einer lokalen Mailbox zu, während der zweite Eintrag eine bestimmte E-Mail-Adresse einer entfernten Mailbox zuordnet. Zuletzt wird jede E-Mail-Adresse von `example.com`, welche nicht mit einem der vorherigen Einträge übereinstimmt, mit dem letzten Eintrag übereinstimmen und der lokalen Mailbox `joe` zugeordnet. Benutzen Sie dieses Format, wenn Sie neue Einträge in `/etc/mail/virtusertable` hinzufügen. Jedes Mal, wenn diese Datei bearbeitet wurde, muss die Datenbank aktualisiert und Sendmail neu gestartet werden:

```
# makemap hash /etc/mail/virtusertable < /etc/mail/virusertable
# service sendmail restart
```

`/etc/mail/relay-domains`
: In der standardmäßigen FreeBSD-Installation wird Sendmail nur dazu konfiguriert, E-Mails von dem Rechner, auf dem es läuft, zu senden. Wenn zum Beispiel ein POP-Server installiert ist, können Benutzer ihre E-Mails von entfernten Standorten überprüfen. Sie werden jedoch keine E-Mails von außen verschicken können. Typischerweise wird ein paar Sekunden nach dem Versuch eine E-Mail von MAILER-DAEMON mit einer 5.7 Relaying Denied Fehlermeldung versendet werden.

Die einfachste Lösung ist, wie im folgenden Beispiel gezeigt, den FQDN des Internet-Dienstanbieters und gegebenenfalls weitere Adressen in `/etc/mail/relay-domains` einzutragen:

```
your.isq.example.com
other.isp.example.net
users.isp.example.org
www.example.org
```

Nachdem diese Datei erstellt oder editiert wurde, muss Sendmail mittels `service sendmail restart` neu gestartet werden.

Ab jetzt wird jede E-Mail, die von einem in der Liste eingetragenen Rechner durch das System geschickt wird, ihr Ziel erreichen, vorausgesetzt der Benutzer hat einen Account auf dem System. Dies erlaubt es Benutzern aus der Ferne, E-Mails über das System zu versenden, ohne dem Massenversand (SPAM) die Tür zu öffnen.

28.5. Wechseln des Mailübertragungs-Agenten

Geschrieben von Andrew Boothman.
Informationen entnommen aus E-Mails geschrieben von Gregory Neil Shapiro.

FreeBSD enthält mit Sendmail bereits einen MTA, der für die ein- und ausgehenden E-Mails verantwortlich ist. Der Systemadministrator kann aber den MTA des Systems wechseln. Eine große Auswahl an alternativen MTAs ist in der Kategorie `mail` der FreeBSD Ports-Sammlung verfügbar.

Sobald ein neuer MTA installiert ist, können Sie die neue Software konfigurieren und testen, bevor Sie Sendmail ersetzen. Informationen über die Konfiguration des neu gewählten MTA finden Sie in der dazugehörigen Dokumentation.

Sobald der neue MTA wie gewünscht funktioniert, benutzen Sie die Anweisungen in diesem Abschnitt, um Sendmail zu deaktivieren und stattdessen den neuen MTA zu verwenden.

28.5.1. Sendmail deaktivieren

Warnung

Wenn der ausgehende Mail-Dienst von Sendmail deaktiviert ist, muss für den E-Mail-Versand ein alternatives System installiert werden. Andernfalls sind Systemfunktionen wie peri-

odic(8) nicht mehr in der Lage, ihre Resulate und Meldungen als E-Mail zu versenden. Aber auch viele andere Teile des Systems erwarten einen funktionalen MTA. Sind Programme auf die deaktivierten Sendmail-Binärdateien angewiesen, landen deren E-Mails ansonsten in einer inaktiven Sendmail-Warteschlange und können nicht ausgeliefert werden.

Um Sendmail komplett zu deaktivieren, müssen folgende Zeilen in `/etc/rc.conf` hinzugefügt oder editiert werden:

```
sendmail_enable="NO"
sendmail_submit_enable="NO"
sendmail_outbound_enable="NO"
sendmail_msp_queue_enable="NO"
```

Um lediglich die Funktion zum Empfang von E-Mails durch Sendmail zu deaktivieren, muss folgender Eintrag in `/etc/rc.conf` gesetzt werden:

```
sendmail_enable="NO"
```

Weitere Informationen zu den Startoptionen von Sendmail finden Sie in der Manualpage rc.sendmail(8).

28.5.2. Den voreingestellten MTA ersetzen

Wenn ein neuer MTA über die Ports-Sammlung installiert wird, werden auch die Startskripten installiert. Die Anweisungen zum starten dieser Skripte werden in den Paketnachrichten erwähnt. Bevor Sie den neuen MTA in Betrieb nehmen, stoppen Sie alle laufenden Sendmail-Prozesse. In diesem Beispiel werden alle notwendigen Dienste gestoppt und danach der Postfix Dienst gestartet:

```
# service sendmail stop
# service postfix start
```

Damit der angegebene MTA automatisch beim Hochfahren des Systems gestartet wird, fügen Sie dessen Konfigurationszeile in `/etc/rc.conf` hinzu. Dieser Eintrag startet den Postfix MTA:

```
postfix_enable="YES"
```

Da Sendmail allgegenwärtig ist und manche Anwendungen einfach davon ausgehen es bereits installiert und konfiguriert, wird einige zusätzliche Konfiguration benötigt. Überprüfen Sie `/etc/periodic.conf` und stellen Sie sicher, dass diese Werte auf `NO` gesetzt werden. Wenn die Datei nicht existiert, erstellen Sie sie mit folgenden Einträgen:

```
daily_clean_hoststat_enable="NO"
daily_status_mail_enable="NO"
daily_status_include_submit_mailq="NO"
daily_submit_queuerun="NO"
```

Viele alternative MTAs stellen ihre eigenen kompatiblen Implementierungen der Sendmail Kommandozeilen-Schnittstelle zur Verfügung, was die Verwendung als „drop-in“ Ersatz für Sendmail vereinfacht. Allerdings versuchen einige MUAs Sendmails Standard-Dateien auszuführen, anstelle der Dateien des neuen MTAs. FreeBSD verwendet `/etc/mail/mailer.conf` um die erwarteten Sendmail Dateien auf die neuen Dateien abzubilden. Weitere Informationen über diese Zuordnungen können in mailwrapper(8) gefunden werden.

In der Voreinstellung sieht `/etc/mail/mailer.conf` wie folgt aus:

```
#$FreeBSD: head/de_DE.ISO8859-1/books/handbook/mail/chapter.xml 48640 2016-04-15 ↺
16:47:41Z bhd $
#
# Execute the "real" sendmail program, named /usr/libexec/sendmail/sendmail
#
sendmail        /usr/libexec/sendmail/sendmail
send-mail       /usr/libexec/sendmail/sendmail
```

```
mailq           /usr/libexec/sendmail/sendmail
newaliases      /usr/libexec/sendmail/sendmail
hoststat        /usr/libexec/sendmail/sendmail
purgestat       /usr/libexec/sendmail/sendmail
```

Wenn eines der Kommandos auf der linken Seite ausgeführt werden soll, führt das System tatsächlich den damit verbundenen Befehl auf der rechten Seite aus. Mit diesem System lassen sich Programme, die für die Sendmail-Funktionen gestartet werden, leicht ändern.

Einige MTAs aus der Ports-Sammlung können diese Datei aktualisieren. Zum Beispiel würde Postfix die Datei wie folgt aktualisieren:

```
#
# Execute the Postfix sendmail program, named /usr/local/sbin/sendmail
#
sendmail        /usr/local/sbin/sendmail
send-mail       /usr/local/sbin/sendmail
mailq           /usr/local/sbin/sendmail
newaliases      /usr/local/sbin/sendmail
```

Falls die Installation des MTA nicht automatisch `/etc/mail/mailer.conf` aktualisiert, bearbeiten Sie diese Datei in einem Texteditor, so dass auf die neuen Dateien verwiesen wird. Dieses Beispiel zeigt auf die Dateien, die von mail/ssmtp installiert wurden:

```
sendmail        /usr/local/sbin/ssmtp
send-mail       /usr/local/sbin/ssmtp
mailq           /usr/libexec/sendmail/sendmail
newaliases      /usr/libexec/sendmail/sendmail
hoststat        /usr/libexec/sendmail/sendmail
purgestat       /usr/libexec/sendmail/sendmail
```

Sobald alles konfiguriert ist, wird empfohlen, das System neu zu starten. Ein Neustart bietet auch die Möglichkeit sicherzustellen, dass das System korrekt konfiguriert wurde, um den neuen MTA automatisch beim Hochfahren zu starten.

28.6. Fehlerbehebung

Hier finden sich ein paar häufig gestellte Fragen und ihre Antworten, die von der FAQ übernommen wurden.

F: Warum muss ich einen FQDN (fully-qualified domain name / voll ausgeschriebenen Domänennamen) für meine Rechner verwenden?

A: Vielleicht befindet sich der Rechner in einer anderen Domäne. Um beispielsweise von einem Rechner in `foo.bar.edu` einen Rechner namens `mumble` in der Domäne `foo.bar.edu` zu erreichen, geben Sie seinen voll ausgeschriebenen Domänennamen (FQDN) `mumble.bar.edu`, anstelle von `mumble` an.

Das liegt daran, dass die aktuelle Version von BIND, die mit FreeBSD ausgeliefert wird, keine Standardabkürzungen für nicht komplett angegebene Domänennamen außerhalb der lokalen Domäne unterstützt. Daher muss ein nicht-qualifizierter Rechner, wie `mumble`, entweder als `mumble.foo.bar.edu` gefunden werden, oder er wird in der root Domäne gesucht.

In älteren Versionen von BIND lief die Suche über `mumble.bar.edu` und `mumble.edu`. RFC 1535 erklärt, warum dieses Verhalten als schlechte Praxis oder sogar als Sicherheitsloch angesehen wird.

Um das zu umgehen, setzen Sie die Zeile:

```
search foo.bar.edu bar.edu
```

anstatt der vorherigen

```
domain foo.bar.edu
```

in `/etc/resolv.conf` ein. Stellen Sie jedoch sicher, dass die Suchordnung nicht die Begrenzung von „lokaler und öffentlicher Administration“, wie RFC 1535 sie nennt, überschreitet.

F: Wie kann ich einen E-Mail-Server auf einem Anwahl-PPP Rechner betreiben?

A: Sie wollen sich mit einem FreeBSD E-Mail *Gateway* im LAN verbinden. Die PPP-Verbindung ist keine Standleitung.

Ein Weg dies zu tun ist, von einem immer mit dem Internet verbundenen Server einen sekundären MX-Dienst für die Domäne zur Verfügung gestellt zu bekommen. In diesem Beispiel heißt die Domäne `example.com`, und der Internet-Dienstanbieter hat `example.net` so eingestellt, dass er für die Domäne einen sekundären MX-Dienst zur Verfügung stellt:

```
example.com.            MX      10      bigco.com.
                MX      20      example.net.
```

Nur ein Rechner sollte als Endempfänger angegeben sein. Sendmail fügen Sie `Cw example.com` zu `/etc/sendmail.cf` auf `example.com` hinzu.

Wenn der MTA des Versenders versucht die E-Mail zuzustellen, wird es versuchen das System `example.com` über die PPP-Verbindung zu erreichen. Es kommt zu einer Zeitüberschreitung, wenn das Zielsystem offline ist. Der MTA wird die E-Mail automatisch der sekundären MX-Seite des Internet-Providers `example.net` zustellen. Die sekundäre MX-Seite wird periodisch versuchen, eine Verbindung zur primären MX-Seite `example.com` aufzubauen.

Verwenden Sie etwas wie dies als Login-Skript:

```
#!/bin/sh
# Put me in /usr/local/bin/pppmyisp
( sleep 60 -; /usr/sbin/sendmail -q ) &
/usr/sbin/ppp -direct pppmyisp
```

Wenn Sie ein separates Login-Skript für einen Benutzer erstellen, benutzen Sie stattdessen `sendmail -qRexample.com` in dem oben gezeigten Skript. Das erzwingt die sofortige Verarbeitung der E-Mails in der Warteschlange für `example.com`

Eine weitere Verfeinerung der Situation kann an diesem Beispiel von FreeBSD Internet service providers entnommen werden:

```
> wir stellen einem Kunden den sekundären MX zur Verfügung.
> Der Kunde verbindet sich mit unseren Diensten mehrmals am Tag
> automatisch um die E-Mails zu seinem primären MX zu holen
> (wir wählen uns nicht bei ihm ein, wenn E-Mails für seine
> Domäne eintreffen).  Unser sendmail sendet den Inhalt der
> E-Mail-Warteschlange alle 30 Minuten.  Momentan muss er 30 Minuten
> eingewählt bleiben um sicher zu sein, dass alle seine E-Mails
> beim primären MX eingetroffen sind.
>
> Gibt es einen Befehl, der sendmail dazu bringt, alle E-Mails sofort
> zu senden? Der Benutzer hat natürlich keine root-Rechte auf
> unserer Maschine.

In der „privacy flags“ Sektion von sendmail.cf befindet sich die
Definition Opgoaway,restrictqrun

Entferne restrictqrun um nicht-root Benutzern zu erlauben, die Verarbeitung
der Nachrichten-Warteschlangen zu starten.  Möglicherweise willst du
auch die MX neu sortieren.  Wir sind der primäre MX für unsere
Kunden mit diesen Wünschen und haben definiert:
```

```
# Wenn wir der beste MX für einen Rechner sind, versuche es direkt
# anstatt einen lokalen Konfigurationsfehler zu generieren.
OwTrue

Auf diesem Weg liefern Gegenstellen direkt zu dir, ohne die Kundenverbindung
zu versuchen.  Dann sendest du zu deinem Kunden.  Das funktioniert nur
für „Rechner“, du musst also deinen Kunden dazu bringen,
ihre E-Mail Maschine „customer.com“ zu nennen, sowie
„hostname.customer.com“ im DNS.  Setze einfach einen A-Eintrag
in den DNS für „customer.com“.
```

28.7. Weiterführende Themen

Dieser Abschnitt behandelt kompliziertere Themen wie E-Mail-Konfiguration und Einrichtung von E-Mail für eine ganze Domäne.

28.7.1. Grundlegende Konfiguration

Mit der Software im Auslieferungszustand sollte es möglich sein, E-Mails an externe Rechner zu senden, vorausgesetzt `/etc/resolv.conf` ist konfiguriert, oder das Netzwerk hat Zugriff auf einen konfigurierten DNS-Server. Um E-Mails an den MTA auf dem Rechner auszuliefern, stehen zwei Möglichkeiten zur Auswahl:

- Betreiben Sie einen DNS-Server für die Domäne.
- Lassen Sie die E-Mails direkt über den FQDN des Rechners ausliefern.

Um E-Mails direkt zu einem Rechner geliefert zu bekommen, wird eine permanente statische IP-Adresse (keine dynamische IP-Adresse) benötigt. Befindet sich das System hinter einer Firewall, muss diese den SMTP-Verkehr weiterleiten. Um E-Mails direkt am Rechner zu empfangen, muss eines der folgenden Dinge konfiguriert werden:

- Vergewissern Sie sich, dass der MX-Eintrag mit der kleinsten Nummer im DNS auf die statische IP-Adresse des Rechners zeigt.
- Stellen Sie sicher, dass für den Rechner kein MX-Eintrag im DNS existiert.

Jede der erwähnten Konfigurationsmöglichkeiten erlaubt es, E-Mails direkt auf dem Rechner zu empfangen.

Versuchen Sie das:

```
# hostname
example.FreeBSD.org
# host example.FreeBSD.org
example.FreeBSD.org has address 204.216.27.XX
```

In diesem Beispiel sollte es funktionieren, E-Mails direkt an <yourlogin@example.FreeBSD.org> zu senden, vorausgesetzt dass Sendmail auf `example.FreeBSD.org` korrekt läuft.

In diesem Beispiel:

```
# host example.FreeBSD.org
example.FreeBSD.org has address 204.216.27.XX
example.FreeBSD.org mail is handled (pri=10) by devnull.FreeBSD.org
```

Hier wird jede an den Rechner `example.FreeBSD.org` gesandte E-Mail auf `hub` unter dem gleichen Benutzernamen gesammelt, anstatt diese direkt zu Ihrem Rechner zu senden.

Die obige Information wird von einem DNS-Server verwaltet. Der DNS-Eintrag, der die Information zum E-Mail-Routing enthält, ist der MX-Eintrag. Existiert kein MX-Eintrag, werden E-Mails direkt über die IP-Adresse an den Rechner geliefert.

Der MX-Eintrag für `freefall.FreeBSD.org` sah einmal so aus:

```
freefall  MX 30 mail.crl.net
freefall  MX 40 agora.rdrop.com
freefall  MX 10 freefall.FreeBSD.org
freefall  MX 20 who.cdrom.com
```

`freefall` hatte viele MX-Einträge. Die kleinste MX-Nummer definiert de Rechner, der die E-Mails direkt empfängt, wobei die anderen Rechner temporär E-Mails in Warteschlangen einreihen, falls `freefall` beschäftigt oder unerreichbar ist.

Es ist sehr sinnvoll, dass stellvertretende MX-Seiten separate Internet-Verbindungen verwenden. Ihr ISP kann diesen Dienst zur Verfügung stellen.

28.7.2. E-Mails für eine Domäne

Wird ein MTA für ein Netzwerk konfiguriert, dann sollte jede E-Mail die an einen Rechner in dieser Domäne geschickt wird, an den MTA umgeleitet werden, damit die Benutzer ihre E-Mails vom zentralen Mail-Server empfangen können.

Am einfachsten ist es, wenn Accounts mit gleichen *Benutzernamen* sowohl auf dem MTA, als auch auf dem System mit dem MUA existieren. Verwenden Sie adduser(8), um Benutzerkonten anzulegen.

Der MTA muss auf jeder Workstation im Netzwerk als der zuständige Rechner für den E-Mail-Austausch gekennzeichnet werden. Dies wird in der DNS-Konfiguration über den MX-Eintrag gesteuert:

```
example.FreeBSD.org A 204.216.27.XX  ; Workstation
   MX 10 devnull.FreeBSD.org ; Mailhost
```

Diese Einstellung wird E-Mails für die Workstations zum MTA weiterleiten, egal wo der A-Eintrag hinzeigt. Die E-Mails werden zum MX-Rechner gesendet.

Diese Einstellung muss auf dem DNS-Server konfiguriert werden. Besitzt das Netzwerk keinen eigenen DNS-Server, kontaktieren Sie Ihren ISP oder DNS-Verwalter.

Im Folgenden ist ein Beispiel für virtuelles E-Mail-Hosting. Nehmen wir an, dass für einen Kunden mit der Domäne `customer1.org`, alle E-Mails für `customer1.org` an `mail.myhost.com` gesendet werden sollen. Der entsprechende DNS-Eintrag sollte wie folgt aussehen:

```
customer1.org  MX 10 mail.myhost.com
```

Wenn für die Domäne nur E-Mails verarbeitet werden sollen, wird für `customer1.org` *kein* `A`-Eintrag benötigt. Allerdings wird ein `ping` gegen `customer1.org` nur dann funktionieren, wenn ein `A`-Eintrag existiert.

Teilen Sie dem MTA mit, für welche Domänen bzw. Hostnamen Post entgegengenommen werden soll. Die beiden folgenden Methoden funktionieren für Sendmail:

- Fügen Sie die Rechnernamen in `/etc/mail/local-host-names` hinzu, wenn `FEATURE(use_cw_file)` verwendet wird.
- Fügen Sie eine Zeile `Cwyour.host.com` in `/etc/sendmail.cf` hinzu.

28.8. Ausgehende E-Mail über einen Relay versenden

Beigetragen von Bill Moran.

In vielen Fällen möchte man E-Mail nur über einen Relay verschicken. Zum Beispiel:

- Der Rechner ist ein Arbeitsplatzrechner und benutzt Programme wie send-pr(1) über ein Relay des ISP.
- Ein Server, der E-Mails nicht selbst verarbeitet, soll alle E-Mails zu einem Relay schicken.

Obwohl jeder MTA diese Aufgabe erfüllen kann, ist es oft schwierig einen vollwertigen MTA so zu konfigurieren, dass er lediglich ausgehende E-Mails weiterleitet. Es ist übertrieben, Programme wie Sendmail und Postfix nur für diesen Zweck einzusetzen.

Weiterhin kann es sein, dass die Bestimmungen des Internetzugangs es verbieten, einen eigenen Mail-Server zu betreiben.

Um die hier beschriebenen Anforderungen zu erfüllen, installieren Sie einfach den Port mail/ssmtp:

```
# cd /usr/ports/mail/ssmtp
# make install replace clean
```

Nach der Installation kann mail/ssmtp über `/usr/local/etc/ssmtp/ssmtp.conf` konfiguriert werden:

```
root=yourrealemail@example.com
mailhub=mail.example.com
rewriteDomain=example.com
hostname=_HOSTNAME_
```

Verwenden Sie eine gültige E-Mail-Adresse für `root`. Geben Sie für `mail.example.com` den Mail-Relay des ISPs an. Einige ISPs nennen den Relay „Postausgangsserver“ oder „SMTP-Server“.

Deaktivieren Sie Sendmail, einschließlich des Services für den Postausgang. Details finden Sie in Abschnitt 28.5.1, „Sendmail deaktivieren“.

mail/ssmtp verfügt über weitere Optionen. Die Beispiele in `/usr/local/etc/ssmtp` oder die Manualpage von ssmtp enthalten weitere Informationen.

Wird ssmtp wie hier beschrieben eingerichtet, können Anwendungen E-Mails von dem lokalen Rechner verschicken. Man verstößt damit auch nicht gegen Bestimmungen des ISPs und läuft nicht Gefahr, dass der Rechner zum Versenden von Spam missbraucht wird.

28.9. E-Mail über Einwahl-Verbindungen

Wird eine feste IP-Adresse verwendet, müssen die Standardeinstellungen wahrscheinlich gar nicht geändert werden. Stellen Sie den Hostnamen auf den entsprechend zugeordneten Internetnamen ein und Sendmail übernimmt das Übrige.

Bei der Verwendung einer dynamisch zugewiesenen IP-Adresse und einer PPP-Wählverbindung mit dem Internet, hat man in der Regel ein Postfach auf dem Mailserver des ISP. In diesem Beispiel ist die Domäne des ISP `example.net`, der Benutzername ist `user`, der Rechnername ist `bsd.home` und der ISP erlaubt es, `relay.example.net` als Mail-Relayhost zu benutzen.

Um Mails aus der Mailbox des ISPs abzuholen, muss ein gesondertes Programm aus der Ports-Sammlung installiert werden. mail/fetchmail ist eine gute Wahl, weil es viele verschiedene Protokolle unterstützt. Für gewöhnlich stellt der ISP POP zur Verfügung. Falls User-PPP verwendet wird, können durch folgenden Eintrag in `/etc/ppp/ppp.linkup` E-Mails automatisch abgerufen werden, sobald eine Verbindung zum Netz aufgebaut wird:

```
MYADDR:
!bg su user -c fetchmail
```

Wird Sendmail benutzt, um E-Mails an nicht-lokale Benutzer zu versenden, konfigurieren Sie es so, dass die Warteschlange abgearbeitet wird, sobald eine Verbindung mit dem Internet besteht. Um dies zu erreichen, müssen folgende Zeilen nach dem `fetchmail` -Eintrag in `/etc/ppp/ppp.linkup` hinzugefügt werden.

```
!bg su user -c "sendmail -q"
```

In diesem Beispiel existiert auf `bsd.home` ein Benutzer `user`. Erstellen Sie auf `bsd.home` im Heimatverzeichnis von `user` die Datei `.fetchmailrc` mit folgender Zeile:

```
poll example.net protocol pop3 fetchall pass MySecret;
```

Diese Datei sollte für niemandem außer `user` lesbar sein, weil sie das Passwort `MySecret` enthält.

Um Mails mit dem richtigen `from:` -Header zu versenden, müssen Sie Sendmail so konfigurieren, dass es <user@example.net> und nicht <user@bsd.home> benutzen soll und das alle Mails über `relay.example.net` versendet werden, um eine schnellere Übertragung von Mails zu gewährleisten.

Die folgende `.mc` sollte ausreichen:

```
VERSIONID(`bsd.home.mc version 1.0')
OSTYPE(bsd4.4)dnl
FEATURE(nouucp)dnl
MAILER(local)dnl
MAILER(smtp)dnl
Cwlocalhost
Cwbsd.home
MASQUERADE_AS(`example.net')dnl
FEATURE(allmasquerade)dnl
FEATURE(masquerade_envelope)dnl
FEATURE(nocanonify)dnl
FEATURE(nodns)dnl
define(`SMART_HOST', `relay.example.net')
Dmbsd.home
define(`confDOMAIN_NAME',`bsd.home')dnl
define(`confDELIVERY_MODE',`deferred')dnl
```

Im vorherigen Abschnitt finden Sie Details dazu, wie Sie diese Datei in das Format `sendmail.cf` konvertieren können. Vergessen Sie nicht, Sendmail neu zu starten, nachdem `sendmail.cf` verändert wurde.

28.10. SMTP-Authentifizierung

Geschrieben von James Gorham.

Die Konfiguration von SMTP-Authentifizierung auf dem MTA bietet einige Vorteile. Die erforderliche Authentifizierung erhöht die Sicherheit von Sendmail und mobilen Benutzern, die auf entfernten Rechnern arbeiten. Diese Benutzer können denselben MTA verwenden, ohne jedes Mal das Benutzerprogramm neu konfigurieren zu müssen.

1. Installieren Sie security/cyrus-sasl2 aus der Ports-Sammlung. Dieser Port verfügt über einige Optionen, die während der Übersetzung festgelegt werden. Für die in diesem Abschnitt beschriebene Methode zur SMTP-Authentifizierung muss die Option `LOGIN` aktiviert werden.

2. Nach der Installation von security/cyrus-sasl2 editieren Sie `/usr/local/lib/sasl2/Sendmail.conf` . Erstellen Sie die Datei, wenn sie nicht existiert und fügen Sie die folgende Zeile hinzu:

   ```
   pwcheck_method: saslauthd
   ```

3. Als nächstes installieren Sie security/cyrus-sasl2-saslauthd, und fügen die folgende Zeile in `/etc/rc.conf` ein:

   ```
   saslauthd_enable="YES"
   ```

   Abschließend starten Sie den saslauthd-Dämon:

```
# service saslauthd start
```

Dieser Dämon agiert als Broker zwischen Sendmail und der FreeBSD-`passwd`-Datenbank. Dadurch müssen zum Versenden von E-Mails keine zusätzlichen Accounts und Passwörter angelegt werden. Die Benutzer verwenden dasselbe Passwort zum Anmelden wie zum Verschicken von E-Mails.

4. Fügen Sie danach in `/etc/make.conf` die folgenden Zeilen hinzu:

```
SENDMAIL_CFLAGS=-I/usr/local/include/sasl -DSASL
SENDMAIL_LDFLAGS=-L/usr/local/lib
SENDMAIL_LDADD=-lsasl2
```

Beim Übersetzen von Sendmail werden damit die cyrus-sasl2-Bibliotheken benutzt. Stellen Sie daher vor dem Übersetzen von Sendmail sicher, dass cyrus-sasl2 installiert ist.

5. Übersetzen Sie Sendmail mit den nachstehenden Kommandos:

```
# cd /usr/src/lib/libsmutil
# make cleandir && make obj && make
# cd /usr/src/lib/libsm
# make cleandir && make obj && make
# cd /usr/src/usr.sbin/sendmail
# make cleandir && make obj && make && make install
```

Die Übersetzung sollte keine Probleme bereiten, wenn `/usr/src` nicht umfangreich verändert wurde und die benötigten Bibliotheken installiert sind.

6. Nachdem Sendmail übersetzt und installiert wurde, editieren Sie `/etc/mail/freebsd.mc` beziehungsweise die lokale `.mc`-Datei. Viele Administratoren verwenden die Ausgabe von hostname(1), um der `.mc` einen eindeutigen Namen zu geben. Fügen Sie die folgenden Zeilen hinzu:

```
dnl set SASL options
TRUST_AUTH_MECH(`GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN')dnl
define(`confAUTH_MECHANISMS', `GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN')dnl
```

Diese Anweisungen konfigurieren die Methoden, die Sendmail zur Authentifizierung von Benutzern verwendet. Lesen Sie die Sendmail Dokumentation, wenn eine andere Methode als `pwcheck` verwendet werden soll.

7. Abschließend rufen Sie make(1) in `/etc/mail` auf. Damit wird aus der `.mc`-Datei eine neue `.cf`-Datei erzeugt. Der Name ist entweder `freebsd.cf` oder der Name der lokalen `.mc`-Datei. `make install restart` installiert die Datei nach `/etc/mail/sendmail.cf` und startet Sendmail neu. Weitere Informationen zu diesem Vorgang entnehmen Sie bitte `/etc/mail/Makefile`.

Um die Konfiguration zu testen, verwenden Sie einen MUA, um eine Testnachricht zu senden. Mail-Benutzerprogramm das Passwort für die Authentifizierung ein und versenden Sie zum Testen eine E-Mail. Zur Fehlersuche, setzen Sie den `LogLevel` von Sendmail auf `13` und untersuchen die Fehlermeldungen in `/var/log/maillog`.

Weitere Information finden Sie unter SMTP-Authentifizierung.

28.11. E-Mail-Programme

Beigetragen von Marc Silver.

Anwendungen, die E-Mails versenden und empfangen, werden als E-Mail-Programme oder Mail-User-Agents (MUA) bezeichnet. Mit der Entwicklung und Ausbreitung von E-Mail wachsen auch die E-Mail-Programme und bieten Benutzern mehr Funktionen und höhere Flexibilität. Die Kategorie `mail` der FreeBSD Ports-Sammlung enthält zahlreiche E-Mail-Programme. Dazu gehören grafische Programme, wie beispielsweise Evolution oder Balsa und Konsolen basierte Programme wie mutt oder alpine.

28.11.1. `mail`

Das standardmäßig unter FreeBSD installierte E-Mail-Programm ist mail(1). Das Programm ist konsolenorientiert und enthält alle Funktionen, die zum Versand und Empfang textbasierter E-Mails erforderlich sind. Es bietet eine begrenzte Unterstützung für Anhänge und kann auf lokale Postfächer zugreifen.

`mail` kann nicht direkt auf POP- oder IMAP-Server zugreifen. Entfernte Postfächer können aber mit einer Anwendung wie fetchmail in eine lokale `mbox` geladen werden.

Um E-Mails zu versenden oder zu empfangen, starten Sie einfach `mail` wie im nachstehenden Beispiel:

```
% mail
```

`mail` liest automatisch den Inhalt des Benutzer-Postfachs im Verzeichnis `/var/mail` . Sollte das Postfach leer sein, beendet sich `mail` mit der Nachricht, dass keine E-Mails vorhanden sind. Wenn E-Mails vorhanden sind, wird die Benutzeroberfläche gestartet und eine Liste der E-Mails angezeigt. Die E-Mails werden automatisch nummeriert wie im folgenden Beispiel gezeigt:

```
Mail version 8.1 6/6/93.  Type ? for help.
"/var/mail/marcs": 3 messages 3 new
>N  1 root@localhost        Mon Mar  8 14:05  14/510   "test"
 N  2 root@localhost        Mon Mar  8 14:05  14/509   "user account"
 N  3 root@localhost        Mon Mar  8 14:05  14/509   "sample"
```

Einzelne Nachrichten können nun durch Eingabe von t gefolgt von der Nummer der Nachricht gelesen werden. Im nachstehenden Beispiel wird die erste E-Mail gelesen:

```
& t 1
Message 1:
From root@localhost  Mon Mar  8 14:05:52 2004
X-Original-To: marcs@localhost
Delivered-To: marcs@localhost
To: marcs@localhost
Subject: test
Date: Mon,  8 Mar 2004 14:05:52 +0200 (SAST)
From: root@localhost (Charlie Root)

Das ist eine Test-Nachricht.  Antworte bitte!
```

Wie in diesem Beispiel zu sehen ist, wird die Nachricht zusammen mit dem vollständigen Nachrichtenkopf angezeigt. Um die Liste der E-Mails erneut zu sehen, drücken Sie wieder die Taste h.

Um auf eine E-Mail zu antworten, benutzen Sie entweder R oder r. R weist `mail` an, dem Versender der Nachricht zu antworten, während mit r allen Empfängern der Nachricht geantwortet wird. Den Kommandos kann die Zahl der E-Mail, auf die geantwortet werden soll, mitgegeben werden. Nachdem die Antwort E-Mail verfasst worden ist, sollte die Eingabe mit einem einzelnen Punkt (.) auf einer neuen Zeile abgeschlossen werden. Hierzu ein Beispiel:

```
& R 1
To: root@localhost
Subject: Re: test

Danke, ich habe deine E-Mail erhalten.
.
EOT
```

Neue E-Mails können mit m, gefolgt von der E-Mail-Adresse des Empfängers verschickt werden. Mehrere Empfänger werden durch Kommata (,) getrennt, angegeben. Der Betreff (*subject*) der Nachricht kann dann, gefolgt vom Inhalt der Nachricht eingegeben werden. Die Nachricht wird dann mit einem einzelnen Punkt (.) auf einer neuen Zeile abgeschlossen.

```
& mail root@localhost
Subject: Ich habe die E-Mails im Griff!
```

```
Jetzt kann ich E-Mails versenden und empfangen ... :)
.
EOT
```

Die Taste ? zeigt zu jeder Zeit einen Hilfetext an. Lesen Sie mail(1), wenn Sie weitere Hilfe zur Benutzung von `mail` erhalten möchten.

Anmerkung

mail(1) wurde nicht für den Umgang mit Anhängen entworfen und kann daher sehr schlecht mit Anhängen umgehen. Neuere MUAs gehen wesentlich besser mit Anhängen um. Benutzer, die `mail` bevorzugen, werden vielleicht den Port converters/mpack zu schätzen wissen.

28.11.2. mutt

mutt ist ein leistungsfähiges E-Mail-Programm mit vielen Funktionen, darunter:

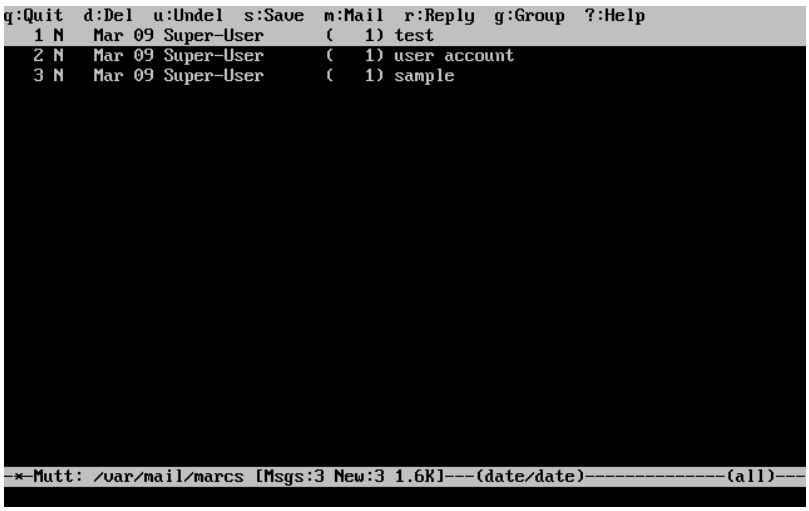
- mutt kann den Verlauf einer Diskussion (*threading*) darstellen.
- Unterstützung von PGP für das digitale signieren und verschlüsseln von E-Mail.
- MIME-Unterstützung.
- Maildir-Unterstützung.
- mutt lässt sich im höchsten Maße an lokale Bedürfnisse anpassen.

Mehr über mutt erfahren Sie auf der Seite `http://www.mutt.org`.

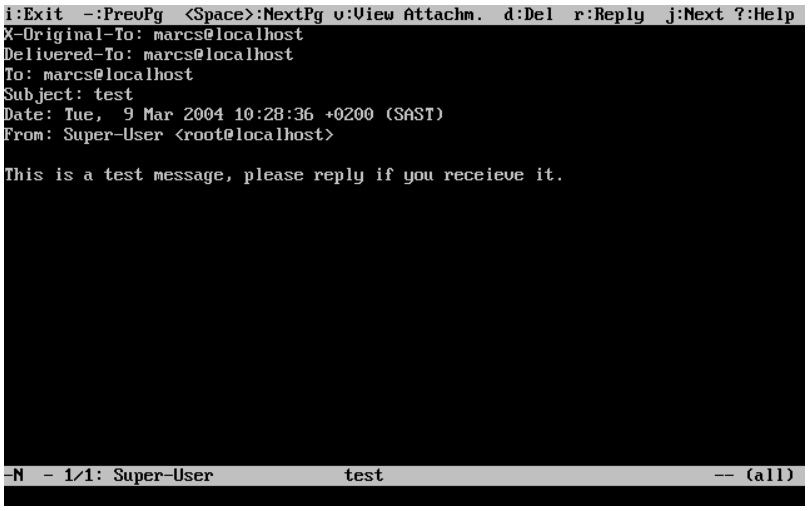
mutt kann über den Port mail/mutt installiert werden. Nachdem der Port installiert ist, kann mutt mit dem folgenden Befehl gestartet werden:

```
% mutt
```

mutt liest automatisch den Inhalt des Benutzer-Postfachs im Verzeichnis `/var/mail`. Sind keine E-Mails vorhanden, wartet mutt auf Benutzereingaben. Das folgende Beispiel zeigt, wie mutt eine Nachrichten-Liste darstellt:



Um eine E-Mail zu lesen, wählen Sie die Nachricht einfach mit den Pfeiltasten aus und drücken Enter. mutt zeigt E-Mails wie folgt an:

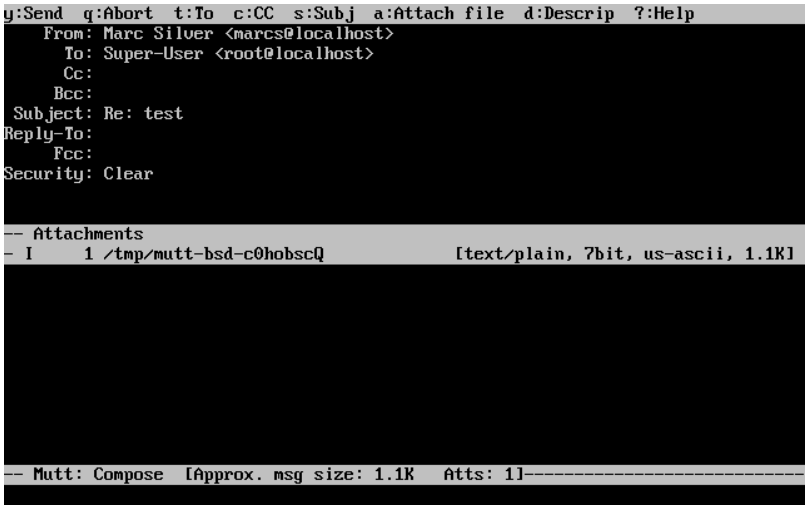


Änlich wie mail(1), kann auch mutt verwendet werden, um nur dem Absender, oder auch allen anderen Empfängern zu antworten. Um nur dem Absender der E-Mail zu antworten, drücken Sie r. Um sowohl dem Absender, als auch allen anderen Empfängern zu antworten, drücken Sie g.

Anmerkung

Zum Erstellen oder zum Beantworten von E-Mails ruft mutt den Editor vi(1) auf. Jeder Benutzer kann diese Einstellung anpassen, indem die Variable `editor` in `.muttrc` im Heimatverzeichnis gesetzt wird, oder die Umgebungsvariable `EDITOR` entsprechend angepasst wird. Weitere Informationen zur Konfiguration von mutt finden Sie unter `http://www.mutt.org/`.

Drücken Sie m, um eine neue Nachricht zu verfassen. Nachdem der Betreff (*subject*) eingegeben wurde, startet mutt den vi(1) und die Nachricht kann verfasst werden. Wenn Sie fertig sind, speichern Sie die Nachricht und verlassen den vi(1). mutt wird dann wieder aktiv und zeigt eine Zusammenfassung der zu sendenden Nachricht an. Drücken Sie y, um die E-Mail zu versenden. Der nachstehende Bildschirmabzug zeigt die Zusammenfassung der E-Mail:



mutt verfügt über eine umfangreiche Hilfestellung. Aus fast jedem Menü können Hilfeseiten mit ? aufgerufen werden. In der oberen Statuszeile werden zudem die verfügbaren Tastenkombinationen angezeigt.

28.11.3. alpine

alpine wendet sich an Anfänger bietet aber ebenfalls einige Funktionen für Profis.

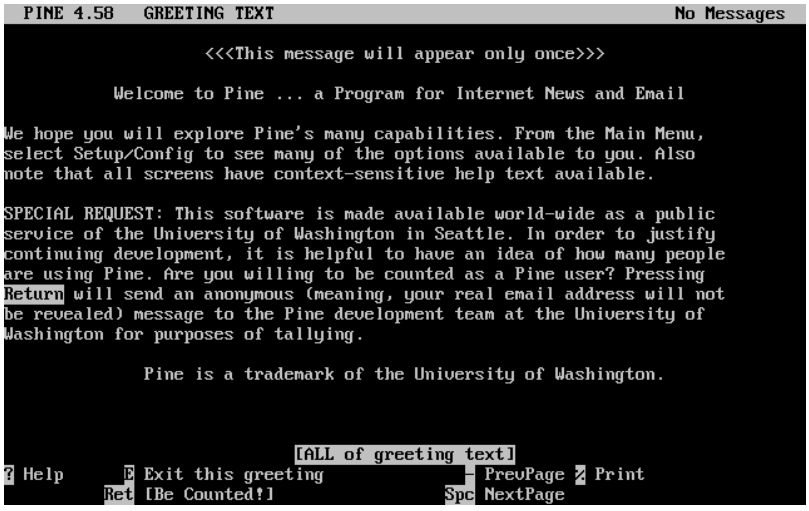
Warnung

In der Vergangenheit wurden in alpine mehrere Schwachstellen gefunden. Die Schwachstellen gestatteten entfernten Benutzern, durch das Versenden einer besonders verfassten E-Mail, Programme auf dem lokalen System laufen zu lassen. Alle *bekannten* Schwachstellen sind beseitigt worden, doch wird im Quellcode von alpine ein sehr riskanter Programmierstil verwendet, sodass der FreeBSD-Security-Officer von weiteren unbekannten Schwachstellen ausgeht. Benutzer installieren alpine auf eigene Verantwortung!

Der Port mail/alpine enthält die aktuelle Version von alpine. Nach der Installation können Sie alpine mit dem nachstehenden Kommando starten:

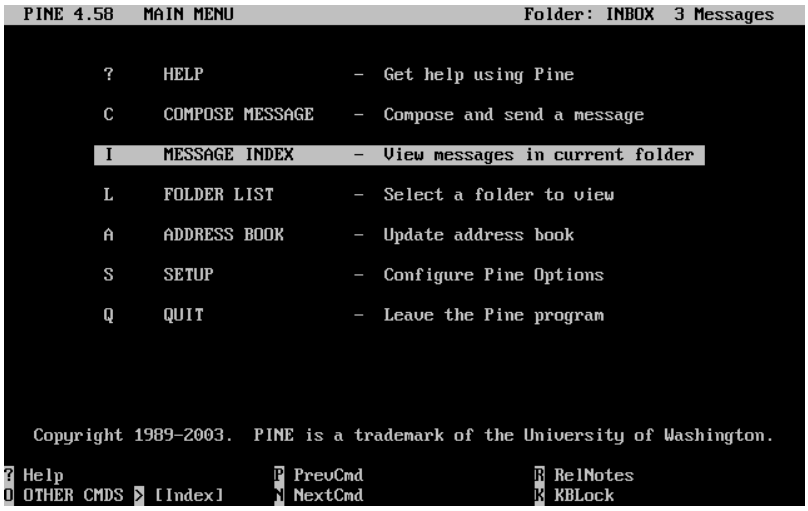
```
% alpine
```

Beim ersten Start von alpine, zeigt das Programm eine Seite mit einer kurzen Einführung an. Um die alpine-Benutzer zu zählen, bitten die Entwickler auf dieser Seite um eine anonyme E-Mail. Sie können diese anonyme E-Mail senden, indem Sie Enter drücken oder den Begrüßungsbildschirm mit der Taste E verlassen, ohne die anonyme E-Mail zu senden. Der Begrüßungsbildschirm sieht wie folgt aus:



Nach dem Begrüßungsbildschirm wird das Hauptmenü dargestellt, das sich mit den Pfeiltasten bedienen lässt. Über Tastenkombinationen können aus dem Hauptmenü neue E-Mails erstellt, Postfächer angezeigt und das Adressbuch verwaltet werden. Unterhalb des Menüs werden die Tastenkombinationen für die verfügbaren Aktionen angezeigt.

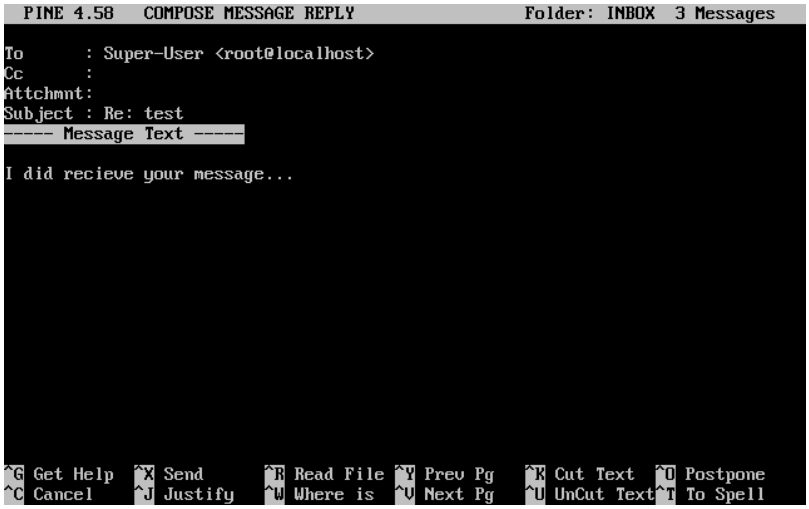
In der Voreinstellung öffnet alpine das Verzeichnis `inbox`. Die Taste I oder der Menüpunkt MESSAGE INDEX führt zu einer Nachrichten-Liste:



Die Liste zeigt die Nachrichten im Arbeitsverzeichnis. Sie können Nachrichten mit den Pfeiltasten markieren. Um eine Nachricht zu lesen, drücken Sie Enter.

Im nächsten Bildschirmabzug sehen Sie, wie alpine eine Nachricht darstellt. Die unteren Bildschirmzeilen zeigen die verfügbaren Tastenkombinationen. Mit r können Sie zum Beispiel auf die gerade angezeigte Nachricht antworten.

Zum Antworten auf eine E-Mail wird in alpine der Editor pico, der mit installiert wird, benutzt. pico ist leicht zu bedienen und gerade für Anfänger besser geeignet als vi(1) oder mail(1). Die Antwort wird mit der Tastenkombination Ctrl+X versendet. Vor dem Versand bittet alpine noch um eine Bestätigung.



Über den Menüpunkt SETUP des Hauptmenüs können Sie alpine an Ihre Bedürfnisse anpassen. Erläuterungen dazu finden Sie auf der Seite `http://www.washington.edu/pine/` .

28.12. E-Mails mit fetchmail abholen

Beigetragen von Marc Silver.

fetchmail ist ein vollwertiger IMAP- und POP-Client. Mit fetchmail können Benutzer E-Mails von entfernten IMAP- und POP-Servern in leichter zugängliche lokale Postfächer laden. fetchmail wird aus dem Port mail/fetchmail installiert. Das Programm bietet unter anderem folgende Funktionen:

- fetchmail beherrscht die Protokolle POP3, APOP, KPOP, IMAP, ETRN und ODMR.
- E-Mails können mit SMTP weiterverarbeitet werden. Dadurch ist garantiert, dass Filter, Weiterleitungen und Aliase weiterhin funktionieren.
- Das Programm kann als Dienst laufen und periodisch neue Nachrichten abrufen.
- fetchmail kann mehrere Postfächer abfragen und je nach Konfiguration die E-Mails an verschiedene lokale Benutzer zustellen.

Dieser Abschnitt erklärt einige grundlegende Funktionen von fetchmail. Das Programm benötigt eine Konfigurationsdatei `.fetchmailrc` im Heimatverzeichnis des Benutzers. In dieser Datei werden Informationen über Server wie auch Benutzerdaten und Passwörter hinterlegt. Wegen des kritischen Inhalts dieser Datei ist es ratsam, diese nur für den Benutzer lesbar zu machen:

```
% chmod 600 .fetchmailrc
```

Die folgende `.fetchmailrc` zeigt, wie das Postfach eines einzelnen Benutzers mit POP heruntergeladen wird. fetchmail wird angewiesen, eine Verbindung zu `example.com` herzustellen und sich dort als Benutzer `joesoap` mit dem Passwort XXX anzumelden. Das Beispiel setzt voraus, dass der Benutzer `joesoap` auch auf dem lokalen System existiert.

```
poll example.com protocol pop3 username "joesoap" password "XXX"
```

Im folgenden Beispiel werden mehrere POP- und IMAP-Server benutzt. Wo notwendig, werden E-Mails auf andere lokale Konten umgeleitet:

```
poll example.com proto pop3:
user "joesoap", with password "XXX", is "jsoap" here;
user "andrea", with password "XXXX";
poll example2.net proto imap:
user "john", with password "XXXXX", is "myth" here;
```

fetchmail kann als Dämon gestartet werden. Verwendet wird dazu die Kommandozeilenoption `-d` gefolgt von einer Zeitspanne in Sekunden, die angibt, wie oft die Server aus `.fetchmailrc` abgefragt werden sollen. Mit dem nachstehenden Befehl fragt fetchmail die Server alle 600 Sekunden ab:

```
% fetchmail -d 600
```

Mehr über fetchmail erfahren Sie auf der Seite `http://www.fetchmail.info/`.

28.13. E-Mails mit procmail filtern

Beigetragen von Marc Silver.

procmail ist ein mächtiges Werkzeug, mit dem sich eingehende E-Mails filtern lassen. Benutzer können Regeln für eingehende E-Mails definieren, die E-Mails zu anderen Postfächern oder anderen E-Mail-Adressen umleiten. procmail befindet sich im Port mail/procmail. procmail kann leicht in die meisten MTAs integriert werden. Lesen Sie dazu bitte die Dokumentation des verwendeten MTAs. Alternativ kann procmail in das E-Mail-System eingebunden werden, indem die nachstehende Zeile in die Datei `.forward` im Heimatverzeichnis eines Benutzers eingefügt wird:

```
"|exec /usr/local/bin/procmail || exit 75"
```

Der folgende Abschnitt zeigt einige einfache procmail-Regeln sowie eine kurze Beschreibung dessen, was sie tun. Regeln müssen in `.procmailrc` im Heimatverzeichnis des Benutzers eingefügt werden.

Den Großteil dieser Regeln finden Sie auch in procmailex(5).

Um E-Mails von <user@example.com> an die externe Adresse <goodmail@example2.com> weiterzuleiten:

```
:0
* ^From.*user@example.com
! goodmail@example2.com
```

Um E-Mails, die kürzer als 1000 Bytes sind, an <goodmail@example2.com> weiterzuleiten:

```
:0
* < 1000
! goodmail@example2.com
```

Um E-Mails, die an <alternate@example.com> geschickt werden, im Postfach `alternate` zu speichern:

```
:0
* ^TOalternate@example.com
alternate
```

Um E-Mails, die im Betreff `Spam` enthalten, nach `/dev/null` zu verschieben:

```
:0
^Subject:.*Spam
/dev/null
```

Zuletzt ein nützliches Rezept, das eingehende E-Mails von den `FreeBSD.org`-Mailinglisten in ein separates Postfach für jede Liste einsortiert:

```
:0
* ^Sender:.owner-freebsd-\/[^@]+@FreeBSD.ORG
{
 LISTNAME=${MATCH}
```

```
  :0
  * LISTNAME??^\/[^@]+
  FreeBSD-${MATCH}
}
```

Kapitel 29. Netzwerkserver

29.1. Übersicht

Dieses Kapitel beschreibt einige der häufiger verwendeten Netzwerkdienste auf UNIX®-Systemen. Dazu zählen Installation und Konfiguration sowie Test und Wartung verschiedener Netzwerkdienste. Zusätzlich sind im ganzen Kapitel Beispielkonfigurationen als Referenz enthalten.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie

- Den inetd-Daemon konfigurieren können.
- Wissen, wie das *Network File System* (NFS) eingerichtet wird.
- Einen *Network Information Server* (NIS) einrichten können, um damit Benutzerkonten im Netzwerk zu verteilen.
- Wissen, wie Sie FreeBSD einrichten, um als LDAP-Server oder -Client zu agieren.
- Rechner durch Nutzung von DHCP automatisch für ein Netzwerk konfigurieren können.
- In der Lage sein, einen *Domain Name Server* (DNS) einzurichten.
- Den Apache HTTP-Server konfigurieren können.
- Wissen, wie man einen *File Transfer Protocol* (FTP)-Server einrichtet.
- Mit Samba einen Datei- und Druckserver für Windows®-Clients konfigurieren können.
- Unter Nutzung des NTP-Protokolls Datum und Uhrzeit synchronisieren sowie einen Zeitserver installieren können.
- Wissen, wie man den Standard-Protokollierungsdienst, `syslogd`, konfiguriert, um Protokolle von anderen Hosts zu akzeptieren.
- Wissen, wie iSCSI eingerichtet wird.

Dieses Kapitel setzt folgende Grundkenntnisse voraus:

- `/etc/rc` -Skripte.
- Netzwerkterminologie
- Installation zusätzlicher Software von Drittanbietern (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

29.2. Der inetd „Super-Server“

29.2.1. Überblick

inetd(8) wird manchmal auch als „Internet Super-Server“ bezeichnet, weil er Verbindungen für viele Dienste verwaltet. Wenn eine Verbindung eintrifft, bestimmt inetd, welches Programm für die eingetroffene Verbindung zuständig ist, aktiviert den entsprechenden Prozess und reicht den Socket an ihn weiter (der Socket dient dabei als Standardein- und -ausgabe sowie zur Fehlerbehandlung). Der Einsatz des inetd-Daemons an Stelle viele einzelner Daemonen kann auf nicht komplett ausgelasteten Servern zu einer Verringerung der Systemlast führen.

inetd wird vor allem dazu verwendet, andere Daemonen zu aktivieren, einige Protokolle werden aber auch direkt verwaltet. Dazu gehören chargen, auth, sowie daytime.

Dieser Abschnitt beschreibt die Konfiguration von inetd durch Kommandozeilenoptionen sowie die Konfigurationsdatei `/etc/inetd.conf`.

29.2.2. Einstellungen

inetd wird durch das rc(8)-System initialisiert. Die Option `inetd_enable` ist in der Voreinstellung auf `NO` gesetzt. Es kann durch den Eintrag

```
inetd_enable="YES"
```

in `/etc/rc.conf` aktiviert werden. inetd wird jetzt beim Systemstart gestartet. Über den Befehl

```
# service inetd rcvar
```

können Sie die aktuelle Konfiguration abfragen.

Weitere Optionen können über die Option `inetd_flags` an inetd übergeben werden.

29.2.3. Kommandozeilenoptionen

Wie die meisten anderen Server-Daemonen lässt sich auch inetd über verschiedene Optionen steuern. Eine vollständige Liste dieser Optionen finden Sie in inetd(8).

Die verschiedenen Optionen können über die Option `inetd_flags` der Datei `/etc/rc.conf` an inetd übergeben werden. In der Voreinstellung hat diese Option den Wert `-wW -C 60`. Durch das Setzen dieser Werte wird das TCP-Wrapping für alle inetd-Dienste aktiviert. Zusätzlich kann eine einzelne IP-Adresse jeden Dienst nur maximal 60 Mal pro Minute anfordern.

Für Einsteiger ist es erfreulich, dass diese Parameter in der Regel nicht angepasst werden müssen. Da diese Parameter aber dennoch von Interesse sein können (beispielsweise, wenn Sie eine enorme Anzahl von Verbindungsanfragen erhalten), werden einige dieser einschränkenden Parameter im Folgenden näher erläutert. Eine vollständige Liste aller Optionen finden Sie in inetd(8).

-c maximum
: Legt die maximale Anzahl von parallen Aufrufen eines Dienstes fest; in der Voreinstellung gibt es keine Einschränkung. Diese Einstellung kann für jeden Dienst durch Setzen des `max-child` -Parameters festgelegt werden.

-C rate
: Legt fest, wie oft ein Dienst von einer einzelnen IP-Adresse in einer Minute aufgerufen werden kann; in der Voreinstellung gibt es keine Einschränkung. Dieser Wert kann für jeden Dienst durch Setzen des Parameters `max-connections-per-ip-per-minute` festgelegt werden.

-R rate
: Legt fest, wie oft ein Dienst in der Minute aktiviert werden kann; in der Voreinstellung sind dies 256 Aktivierungen pro Minute. Ein Wert von 0 erlaubt unbegrenzt viele Aktivierungen.

-s maximum
: Legt fest, wie oft ein Dienst in der Minute von einer einzelnen IP-Adresse aus aktiviert werden kann; in der Voreinstellung gibt es hier keine Beschränkung. Diese Einstellung kann für jeden Dienst durch die Angabe `max-child-per-ip` angepasst werden.

29.2.4. `inetd.conf`

Die Konfiguration von inetd erfolgt über `/etc/inetd.conf`.

Wenn `/etc/inetd.conf` geändert wird, kann inetd veranlasst werden, seine Konfigurationsdatei neu einzulesen.

Beispiel 29.1. Die inetd-Konfiguration neu einlesen

```
# service inetd reload
```

Jede Zeile der Konfigurationsdatei beschreibt jeweils einen Daemon. Kommentare beginnen mit einem „#“. Ein Eintrag der Datei `/etc/inetd.conf` hat folgenden Aufbau:

```
service-name
socket-type
protocol
{wait|nowait}[/max-child[/max-connections-per-ip-per-minute[/max-child-per-ip]]]
user[:group][/login-class]
server-program
server-program-arguments
```

Ein Eintrag für den IPv4 verwendenden ftpd(8)-Daemon könnte so aussehen:

```
ftp     stream  tcp     nowait  root    /usr/libexec/ftpd       ftpd -l
```

service-name
: Der Dienstname eines bestimmten Daemons. Er muss einem in `/etc/services` aufgelisteten Dienst entsprechen. In dieser Datei wird festgelegt, welchen Port inetd abhören muss. Wenn ein neuer Dienst erzeugt wird, muss er zuerst in die Datei `/etc/services` eingetragen werden.

socket-type
: Entweder `stream`, `dgram`, `raw`, oder `seqpacket`. `stream` muss für verbindungsorientierte TCP-Daemonen verwendet werden, während `dgram` das UDP-Protokoll verwaltet.

protocol
: Eines der folgenden:

Protokoll	Bedeutung
tcp, tcp4	TCP (IPv4)
udp, udp4	UDP (IPv4)
tcp6	TCP (IPv6)
udp6	UDP (IPv6)
tcp46	TCP sowohl unter IPv4 als auch unter IPv6
udp46	UDP sowohl unter IPv4 als auch unter IPv6

{wait|nowait}[/max-child[/max-connections-per-ip-per-minute[/max-child-per-ip]]]
: `wait|nowait` gibt an, ob der von inetd aktivierte Daemon seinen eigenen Socket verwalten kann oder nicht. `dgram`-Sockets müssen die Option `wait` verwenden, während Daemonen mit Stream-Sockets, die normalerweise auch aus mehreren Threads bestehen, die Option `nowait` verwenden sollten. Die Option `wait` gibt in der Regel mehrere Sockets an einen einzelnen Daemon weiter, während `nowait` für jeden neuen Socket einen Childdaemon erzeugt.

  Die maximale Anzahl an Child-Daemonen, die inetd erzeugen kann, wird durch die Option `max-child` festgelegt. Wenn ein bestimmter Daemon 10 Instanzen benötigt, sollte der Wert `/10` hinter die Option `nowait` gesetzt werden. Geben Sie hingegen den Wert `/0` an, gibt es keine Beschränkung.

  Zusätzlich zu `max-child` kann die maximale Anzahl von Verbindungen eines Rechners mit einem bestimmten Daemon durch zwei weitere Optionen beschränkt werden. Die Option `max-connections-per-ip-per-minute`

legt die maximale Anzahl von Verbindungsversuchen fest, die von einer bestimmten IP-Adresse aus unternommen werden können. Ein Wert von zehn würde die maximale Anzahl von Verbindungsversuchen einer IP-Adresse mit einem bestimmten Dienst auf zehn Versuche in der Minute beschränken. Durch die Angabe der Option `max-child-per-ip` können Sie hingegen festlegen, wie viele Child-Daemonen von einer bestimmten IP-Adresse aus gestartet werden können. Durch diese Optionen lassen sich ein absichtlicher oder unabsichtlicher Ressourcenverbrauch sowie die Auswirkungen eines `Denial of Service (DoS)`-Angriffs auf einen Rechner begrenzen.

Sie müssen hier entweder `wait` oder `nowait` angeben. Die Angabe von `max-child`, `max-connections-per-ip-per-minute` und `max-child-per-ip` ist hingegen optional.

Ein multithread-Daemon vom Streamtyp ohne die Optionen `max-child`, `max-connections-per-ip-per-minute` oder `max-child-per-ip` sieht so aus: `nowait`

Der gleiche Daemon mit einer maximal möglichen Anzahl von 10 parallelen Daemonen würde so aussehen: `nowait/10`

Wird zusätzlich die Anzahl der möglichen Verbindungen pro Minute für jede IP-Adresse auf 20 sowie die mögliche Gesamtzahl von Childdaemonen auf 10 begrenzt, so sieht der Eintrag so aus: `nowait/10/20`

All diese Optionen werden vom fingerd(8)-Daemon bereits in der Voreinstellung verwendet:

```
finger stream  tcp     nowait/3/10 nobody /usr/libexec/fingerd fingerd -s
```

Will man die maximale Anzahl von Child-Daemonen auf 100 beschränken, wobei von jeder IP-Adresse aus maximal 5 Child-Daemonen gestartet werden dürfen, verwendet man den folgenden Eintrag: `nowait/100/0/5`.

user
: Der Benutzername, unter dem der jeweilige Daemon laufen soll. Meistens laufen Daemonen als User `root`. Aus Sicherheitsgründen laufen einige Server aber auch als User `daemon`, oder als am wenigsten privilegierter User `nobody`.

server-program
: Der vollständige Pfad des Daemons, der eine Verbindung entgegennimmt. Wird der Daemon von inetd intern bereitgestellt, sollte die Option `internal` verwendet werden.

server-program-arguments
: Dieser Eintrag legt (gemeinsam mit `server-program` und beginnend mit `argv[0]`), die Argumente fest, die bei der Aktivierung an den Daemon übergeben werden. Wenn die Anweisung auf der Kommandozeile also `mydaemon -d` lautet, wäre `mydaemon -d` auch der Wert der Option `server program arguments`. Wenn es sich beim Daemon um einen internen Dienst handelt, sollte wiederum die Option `internal` verwendet werden.

29.2.5. Sicherheit

Abhängig von der bei der Installation festgelegten Konfiguration werden viele der von inetd verwalteten Dienste automatisch aktiviert! Wenn Sie einen bestimmten Daemon nicht benötigen, sollten Sie ihn deaktivieren! Dazu kommentieren Sie den jeweiligen Daemon in `/etc/inetd.conf` mit einem „#“ aus, um danach die inetd-Konfiguration neu einzulesen. Einige Daemonen, zum Beispiel fingerd, sollten generell deaktiviert werden, da sie zu viele Informationen an einen potentiellen Angreifer liefern.

Einige Daemonen sind nicht auf Sicherheit bedacht und haben lange oder nicht existierende Timeouts für Verbindungsversuche. Ein Angreifer kann dies ausnutzen, um Verbindungen zu bestimmten Daemonen aufzubauen und somit die verfügbaren Ressourcen zu verbrauchen, was zu einem *Denial of Service* (DOS) führen kann. `max-connections-per-ip-per-minute`, `max-child` und `max-child-per-ip` können verwendet werden, um solche Angriffe zu begrenzen.

TCP-Wrapping ist in der Voreinstellung aktiviert. Lesen Sie hosts_access(5), wenn Sie weitere Informationen zum Setzen von TCP-Beschränkungen für verschiedene von inetd aktivierte Daemonen benötigen.

29.2.6. Verschiedenes

Bei daytime, time, echo, discard, chargen, und auth handelt es sich um intern von inetd bereitgestellte Dienste.

Der auth-Dienst bietet Identifizierungsdienste über das Netzwerk an und ist bis zu einem bestimmten Grad konfigurierbar, während die meisten anderen Dienste nur aktiviert oder deaktiviert werden können.

Eine ausführliche Beschreibung finden Sie in inetd(8).

29.3. Network File System (NFS)

FreeBSD unterstützt das Netzwerkdateisystem NFS, das es einem Server erlaubt, Dateien und Verzeichnisse über ein Netzwerk mit Clients zu teilen. Mit NFS können Benutzer und Programme auf Daten entfernter Systeme zugreifen, und zwar so, als ob es sich um lokal gespeicherte Daten handeln würde.

Die wichtigsten Vorteile von NFS sind:

- Daten, die sonst auf jeden Client dupliziert würden, können an einem zentralen Ort aufbewahrt, und von den Clients über das Netzwerk aufgerufen werden.
- Die Heimatverzeichnisse der Benutzer werden an einem zentralen Ort gespeichert und den Benutzern über das Netzwerk zur Verfügung gestellt.
- Die Verwaltung der NFS-Exporte wird vereinfacht. Zum Beispiel gibt es dann nur noch ein Dateisystem, für das Sicherheits- oder Backup-Richtlinien festgelegt werden müssen.
- Wechselmedien können von anderen Maschinen im Netzwerk verwendet werden. Dies reduziert die Anzahl von Geräten im Netzwerk und bietet einen zentralen Ort für die Verwaltung. Oft ist es einfacher, über ein zentrales Installationsmedium Software auf mehreren Computern zu installieren.

NFS besteht aus zwei Hauptteilen: Einem Server und einem oder mehreren Clients. Der Client greift über das Netzwerk auf die Daten zu, die auf dem Server gespeichert sind. Damit dies korrekt funktioniert, müssen einige Prozesse konfiguriert und gestartet werden:

Folgende Daemonen müssen auf dem Server ausgeführt werden:

Daemon	Beschreibung
nfsd	Der NFS-Daemon. Er bearbeitet Anfragen der NFS-Clients.
mountd	Der NFS-Mount-Daemon. Er bearbeitet die Anfragen, die nfsd(8) an ihn weitergibt.
rpcbind	Der Portmapper-Daemon. Durch ihn erkennen die NFS-Clients, welchen Port der NFS-Server verwendet.

Der Einsatz von nfsiod(8) ist nicht zwingend erforderlich, kann aber die Leistung auf dem Client verbessern.

29.3.1. NFS einrichten

NFS lässt sich leicht aktivieren. Die nötigen Prozesse werden durch das Hinzufügen der folgenden Optionen in `/etc/rc.conf` bei jedem Systemstart ausgeführt:

```
rpcbind_enable="YES"
nfs_server_enable="YES"
mountd_flags="-r"
```

mountd läuft automatisch, wenn der NFS-Server aktiviert ist.

Um den Client zu aktivieren, muss folgende Option in `/etc/rc.conf` gesetzt werden:

```
nfs_client_enable="YES"
```

`/etc/exports` legt fest, welche Dateisysteme NFS exportieren soll. Jede Zeile in `/etc/exports` beschreibt ein zu exportierendes Dateisystem, Clients, die darauf Zugriff haben sowie alle Zugriffsoptionen. Es gibt viele verschiedene Optionen, allerdings werden hier nur wenige von ihnen erwähnt. Eine vollständige Liste der Optionen finden Sie in exports(5).

Die folgenden Beispiele geben Anhaltspunkte zum Exportieren von Dateisystemen, obwohl diese Einstellungen natürlich von der Arbeitsumgebung und der Netzwerkkonfiguration abhängen.

Dieses Beispiel exportiert `/cdrom` für drei Clients, *`alpha`*, *`bravo`* und *`charlie`*:

```
/cdrom -ro alpha bravo charlie
```

Die Option `-ro` kennzeichnet das exportierte Dateisystem als schreibgeschützt. Dadurch sind Clients nicht in der Lage, das exportierte Dateisystem zu verändern.

Das nächste Beispiel exportiert `/home` auf drei durch IP-Adressen bestimmte Clients. Diese Einstellung kann für Netzwerke ohne DNS-Server nützlich sein. Optional können interne Rechnernamen auch in `/etc/hosts` konfiguriert werden. Benötigen Sie hierzu weitere Informationen, lesen Sie bitte hosts(5). Die Option `-alldirs` ermöglicht es, auch Unterverzeichnisse als Mountpunkte festzulegen. Dies bedeutet aber nicht, dass alle Unterverzeichnisse eingehängt werden, vielmehr wird es dem Client ermöglicht, nur diejenigen Verzeichnisse einzuhängen, die auch benötigt werden.

```
/home  -alldirs  10.0.0.2 10.0.0.3 10.0.0.4
```

Die nächste Zeile exportiert `/a`, damit Clients von verschiedenen Domänen auf das Dateisystem zugreifen können. Die Option `-maproot=root` erlaubt es dem Benutzer `root` des Clients, als `root` auf das exportierte Dateisystem zu schreiben. Wenn diese Option nicht gesetzt ist, wird der `root`-Benutzer des Clients dem `nobody`-Konto des Servers zugeordnet und unterliegt somit den Zugriffsbeschränkungen dieses Kontos.

```
/a  -maproot=root  host.example.com box.example.org
```

Damit ein Client auf ein exportiertes Dateisystem zugreifen kann, muss er in `/etc/exports` eingetragen sein.

Jede Zeile in `/etc/exports` entspricht der Exportinformation für ein Dateisystem auf einem oder mehreren Clients. Ein entfernter Rechner kann für jedes Dateisystem nur einmal definiert werden. Nehmen wir an, dass `/usr` ein gesondertes Dateisystem ist. Dann wären folgende Zeilen in `/etc/exports` ungültig:

```
#Nicht erlaubt, wenn /usr ein einziges Dateisystem ist
/usr/src   client
/usr/ports client
```

Das Dateisystem `/usr` wird hier zweimal auf den selben Rechner (`client`) exportiert. Dies ist aber nicht zulässig. Der korrekte Eintrag sieht daher so aus:

```
/usr/src /usr/ports  client
```

Die Eigenschaften eines auf einen anderen Rechner exportierten Dateisystems müssen alle in einer Zeile stehen. Wird in einer Zeile kein Rechner festgelegt, dürfen alle Clients im Netzwerk das exportierte Dateisystem einhängen.

Eine gültige Exportliste, in der `/usr` und `/exports` lokale Dateisysteme sind, sieht so aus:

```
# Export src and ports to client01 and client02, but only
# client01 has root privileges on it
/usr/src /usr/ports -maproot=root    client01
/usr/src /usr/ports                  client02
# The client machines have root and can mount anywhere
```

```
# on /exports. Anyone in the world can mount /exports/obj read-only
/exports -alldirs -maproot=root      client01 client02
/exports/obj -ro
```

Wenn der NFS-Server startet, wird auch mountd automatisch gestartet. Allerdings liest mountd `/etc/exports` nur, wenn der Server gestartet wird. Um nachfolgende Änderungen an `/etc/exports` wirksam werden zu lassen, kann mountd angewiesen werden, die Datei neu einzulesen:

```
# service mountd reload
```

Lesen Sie bitte Abschnitt 12.4, „Dienste unter FreeBSD verwalten“ des Handbuchs für Informationen zum Einsatz der rc-Skripte.

Die NFS-Dienste können nun auf dem Server als `root` gestartet werden:

```
# service nfsd start
```

Auf dem NFS-Client:

```
# service nfsclient restart
```

Der Client ist nun in der Lage, ein entferntes Dateisystem einzuhängen. In diesen Beispielen ist der Name des Servers `server` und der Name des Clients `client`. Für Testzwecke oder zum temporären einhängen eines entfernten Dateisystems, führen Sie auf dem Rechner `client` den Befehl `mount`als `root` aus:

```
# mount server:/home /mnt
```

Die Dateien und Verzeichnisse in `/home` stehen dem Rechner `client` nun im Verzeichnis `/mnt` zur Verfügung.

Um ein entferntes Dateisystem bei jedem Systemstart automatisch einzuhängen, fügen Sie das Dateisystem in `/etc/fstab` ein:

```
server:/home /mnt nfs rw 0 0
```

fstab(5) enthält eine Beschreibung aller Optionen.

29.3.2. Dateien sperren (Locking)

Einige Anwendungen erfordern die Sperrung von Dateien, damit sie korrekt arbeiten. Um diese Sperre zu aktivieren, müssen diese Zeilen in `/etc/rc.conf` sowohl auf dem Client als auch auf dem Server hinzugefügt werden:

```
rpc_lockd_enable="YES"
rpc_statd_enable="YES"
```

Danach starten Sie die beiden Anwendungen:

```
# service lockd start
# service statd start
```

Wenn keine Dateisperren zwischen den NFS-Clients und dem NFS-Server benötigt werden, können Sie den NFS-Client durch die Übergabe der Option `-L` an mount_nfs(8) zu einer lokalen Sperrung von Dateien zwingen. Weitere Details finden Sie in mount_nfs(8).

29.3.3. Praktische Anwendungen

NFS ist in vielen Situationen nützlich. Einige Anwendungsbereiche finden Sie in der folgenden Liste:

- Mehrere Maschinen können sich ein CD-ROM-Laufwerk oder andere Medien teilen. Dies ist praktisch, um Programme von einem einzelnen Standort aus auf mehreren Rechnern zu installieren.

- In größeren Netzwerken ist es praktisch, einen zentralen NFS-Server einzurichten, auf dem die Heimatverzeichnisse der Benutzer gespeichert werden. Dadurch steht den Benutzern immer das gleiche Heimatverzeichnis zur Verfügung, unabhängig davon, an welchem Client im Netzwerk sie sich anmelden.

- Verschiedene Clients können auf ein gemeinsames Verzeichnis `/usr/ports/distfiles` zugreifen. Die gemeinsame Nutzung dieses Verzeichnisses ermöglicht einen schnellen Zugriff auf die Quelldateien, ohne sie auf jede Maschine zu kopieren zu müssen.

29.3.4. amd

amd(8) (Automatic Mounter Daemon) hängt ein entferntes Dateisystem automatisch ein, wenn auf eine Datei oder ein Verzeichnis in diesem Dateisystem zugegriffen wird. Dateisysteme, die über einen gewissen Zeitraum inaktiv sind, werden von amd automatisch abgehängt. amd ist eine Alternative zum dauerhaften Einhängen von Dateisystemen in `/etc/fstab` .

In der Voreinstellung stellt amd die Verzeichnisse `/host` und `/net` als NFS-Server bereit. Wenn auf eine Datei in diesen Verzeichnissen zugegriffen wird, sucht amd den entsprechenden Mountpunkt und hängt das Dateisystem automatisch ein. `/net` wird zum Einhängen von exportierten Dateisystemen von einer IP-Adresse verwendet, während `/host` zum Einhängen von exportierten Dateisystemen eines durch seinen Namen festgelegten Rechners dient.

Ein Zugriff auf eine Datei in `/host/foobar/usr` würde amd veranlassen, das von `foobar` exportierte Dateisystem `/usr` einzuhängen.

Beispiel 29.2. Ein exportiertes Dateisystem mit amd in den Verzeichnisbaum einhängen

`showmount -e` zeigt die exportierten Dateisysteme des NFS-Servers `foobar` an:

```
% showmount -e foobar
Exports list on foobar:
/usr                               10.10.10.0
/a                                 10.10.10.0
% cd /host/foobar/usr
```

Die Ausgabe von `showmount` zeigt `/usr` als exportiertes Dateisystem an. Wenn man in das Verzeichnis `/host/foobar/usr` wechselt, fängt amd die Anfrage ab und versucht den Rechnernamen `foobar` aufzulösen. Wenn dies gelingt, wird amd automatisch den gewünschten Export in den Verzeichnisbaum einhängen.

amd kann durch folgende Zeile in `/etc/rc.conf` automatisch gestartet werden:

```
amd_enable="YES"
```

Um amd direkt zu starten:

```
# service amd start
```

Individuelle Optionen können über die Umgebungsvariable `amd_flags` an amd übergeben werden. In der Voreinstellung ist `amd_flags` eingestellt auf:

```
amd_flags="-a /.amd_mnt -l syslog /host /etc/amd.map /net /etc/amd.map"
```

`/etc/amd.map` legt die Standardoptionen fest, mit denen exportierte Dateisysteme in den Verzeichnisbaum eingehängt werden. `/etc/amd.conf` hingegen legt einige der erweiterten Optionen von amd fest.

Weitere Informationen finden Sie in amd(8) und amd.conf(5).

29.4. Network Information System (NIS)

Das Network Information System (NIS) wurde entwickelt, um UNIX®-Systeme zentral verwalten zu können. Dazu zählen beispielsweise Solaris™, HP-UX, AIX®, Linux, NetBSD, OpenBSD und FreeBSD. NIS war ursprünglich als *Yellow Pages* bekannt, aus markenrechtlichen Gründen wurde der Name aber geändert. Dies ist der Grund, warum NIS-Kommandos mit `yp` beginnen.

Bei NIS handelt es sich um ein RPC-basiertes Client/Server-System. Eine Gruppe von Rechnern greift dabei innerhalb einer NIS-Domäne auf gemeinsame Konfigurationsdateien zu. Dies erlaubt es einem Systemadministrator, NIS-Clients mit minimalem Aufwand einzurichten, sowie Änderungen an der Systemkonfiguration von einem zentralen Ort aus durchzuführen.

FreeBSD verwendet die Version 2 des NIS-Protokolls.

29.4.1. NIS-Begriffe und -Prozesse

Tabelle 30.1 fasst die Begriffe und Anwenderprozesse zusammen, die von NIS verwendet werden:

Tabelle 29.1. NIS Begriffe

Begriff	Beschreibung
NIS-Domänenname	NIS-Masterserver und Clients benutzen einen gemeinsamen NIS-Domänennamen. In der Regel hat dieser Name nichts mit DNS zu tun.
rpcbind(8)	Dieser Dienst aktiviert RPC und muss gestartet sein, damit ein NIS-Server oder -Client ausgeführt werden kann.
ypbind(8)	Dieser Dienst „bindet“ einen NIS-Client an seinen NIS-Server. Der Client bezieht den NIS-Domänennamen vom System und stellt über das RPC-Protokoll eine Verbindung zum NIS-Server her. ypbind ist der zentrale Bestandteil der Client-Server-Kommunikation in einer NIS-Umgebung. Wird der Dienst auf einem Client beendet, ist dieser nicht mehr in der Lage, auf den NIS-Server zuzugreifen.
ypserv(8)	Dies ist der Prozess für den NIS-Server. Wenn dieser Dienst nicht mehr läuft, kann der Server nicht mehr auf NIS-Anforderungen reagieren. Wenn ein Slaveserver existiert, kann dieser als Ersatz fungieren. Einige NIS-Systeme (allerdings nicht das von FreeBSD) versuchen allerdings erst gar nicht, sich mit einem anderen Server zu verbinden, wenn der Masterserver nicht mehr reagiert. Die einzige Lösung besteht darin, den Serverprozess oder den ypbind-Prozess auf dem Client neu zu starten.
rpc.yppasswdd(8)	Dieser Prozess läuft nur auf dem NIS-Masterserver. Es handelt sich um einen Daemonprozess, der es NIS-Clients ermöglicht, ihre NIS-Passwörter zu ändern. Wenn dieser Daemon nicht läuft, müssen sich die Benutzer am NIS-Masterserver anmelden und ihre Passwörter dort ändern.

29.4.2. Arten von NIS-Rechnern

- NIS-Masterserver

  Dieser Server dient als zentraler Speicherort für Rechnerkonfigurationen. Zudem verwaltet er die maßgebliche Kopie, der von den NIS-Clients gemeinsam verwendeten Dateien. `passwd`, `group`, sowie verschiedene andere von den Clients verwendete Dateien existieren auf dem Masterserver. Obwohl ein Rechner auch für mehrere NIS-Domänen als Masterserver fungieren kann, wird diese Art von Konfiguration nicht behandelt, da sich dieser Abschnitt auf eine relativ kleine NIS-Umgebung konzentriert.

- NIS-Slaveserver

NIS-Slaveserver verwalten Kopien der Daten des NIS-Masterservers um Redundanz zu bieten. Zudem entlasten Slaveserver den Masterserver: NIS-Clients verbinden sich immer mit dem NIS-Server, welcher zuerst reagiert. Dieser Server kann auch ein Slaveserver sein.

- NIS-Clients

  NIS-Clients identifizieren sich gegenüber dem NIS-Server während der Anmeldung.

Mit NIS können Informationen aus verschiedenen Dateien von mehreren Rechnern gemeinsam verwendet werden. `master.passwd`, `group`, und `hosts` werden oft gemeinsam über NIS verwendet. Immer, wenn ein Prozess auf einem Client auf Informationen zugreifen will, die normalerweise in lokalen Dateien vorhanden wären, wird stattdessen eine Anfrage an den NIS-Server gestellt, an den der Client gebunden ist.

29.4.3. Planung

Dieser Abschnitt beschreibt eine einfache NIS-Umgebung, welche aus 15 FreeBSD-Maschinen besteht, für die keine zentrale Verwaltung existiert. Jeder Rechner hat also eine eigene Version von `/etc/passwd` und `/etc/master.passwd`. Diese Dateien werden manuell synchron gehalten; wird ein neuer Benutzer angelegt, so muss dies auf allen fünfzehn Rechnern manuell erledigt werden.

In Zukunft soll die Konfiguration wie folgt aussehen:

Rechnername	IP-Adresse	Rechneraufgabe
`ellington`	`10.0.0.2`	NIS-Master
`coltrane`	`10.0.0.3`	NIS-Slave
`basie`	`10.0.0.4`	Workstation der Fakultät
`bird`	`10.0.0.5`	Clientrechner
`cli[1-11]`	`10.0.0.[6-17]`	Verschiedene andere Clients

Wenn erstmalig ein NIS-Schema eingerichtet wird, sollte es im Voraus sorgfältig geplant werden. Unabhängig von der Größe des Netzwerks müssen einige Entscheidungen im Rahmen des Planungsprozesses getroffen werden.

29.4.3.1. Einen NIS-Domänennamen wählen

Wenn ein Client Informationen anfordert, ist in dieser Anforderung der Name der NIS-Domäne enthalten. Dadurch weiß jeder Server im Netzwerk, auf welche Anforderung er antworten muss. Stellen Sie sich den NIS-Domänennamen als einen Namen einer Gruppe von Rechnern vor.

Manchmal wird der Name der Internetdomäne auch für die NIS-Domäne verwendet. Dies ist allerdings nicht empfehlenswert, da es bei der Behebung von Problemen verwirrend sein kann. Der Name der NIS-Domäne sollte innerhalb des Netzwerks eindeutig sein. Hilfreich ist es, wenn der Name die Gruppe der in ihr zusammengefassten Rechner beschreibt. Die Kunstabteilung von Acme Inc. hätte daher vielleicht die NIS-Domäne „acme-art“. Für dieses Beispiel wird der Name `test-domain` verwendet.

Es gibt jedoch auch Betriebssysteme, die als NIS-Domänennamen den Namen der Internetdomäne verwenden. Wenn dies für einen oder mehrere Rechner des Netzwerks zutrifft, *muss* der Name der Internetdomäne als NIS-Domänennamen verwendet werden.

29.4.3.2. Anforderungen an den Server

Bei der Wahl des NIS-Servers müssen einige Dinge beachtet werden. Da die NIS-Clients auf die Verfügbarkeit des Servers angewiesen sind, sollten Sie einen Rechner wählen, der nicht regelmäßig neu gestartet werden muss. Der NIS-Server sollte idealerweise ein alleinstehender Rechner sein, dessen einzige Aufgabe es ist, als NIS-Server zu dienen. Wenn das Netzwerk nicht zu stark ausgelastet ist, ist es auch möglich, den NIS-Server als weiteren Dienst

auf einem anderen Rechner laufen zu lassen. Wenn jedoch ein NIS-Server ausfällt, wirkt sich dies negativ auf *alle* NIS-Clients aus.

29.4.4. Einen NIS-Masterserver konfigurieren

Die verbindlichen Kopien aller NIS-Dateien befinden sich auf dem Masterserver. Die Datenbanken, in denen die Informationen gespeichert sind, bezeichnet man als NIS-Maps. Unter FreeBSD werden diese Maps unter `/var/yp/[domainname]` gespeichert, wobei `[domainname]` der Name der NIS-Domäne ist. Da ein NIS-Server mehrere Domänen verwalten kann, können auch mehrere Verzeichnisse vorhanden sein. Jede Domäne verfügt über ein eigenes Verzeichnis sowie einen eigenen, von anderen Domänen unabhängigen Satz von NIS-Maps.

NIS-Master- und Slaveserver verwenden ypserv(8), um NIS-Anfragen zu bearbeiten. Dieser Daemon ist für eingehende Anfragen der NIS-Clients verantwortlich. Er ermittelt aus der angeforderten Domäne und Map einen Pfad zur entsprechenden Datenbank, und sendet die angeforderten Daten von der Datenbank zum Client.

Abhängig von den Anforderungen ist die Einrichtung eines NIS-Masterservers relativ einfach, da NIS von FreeBSD bereits in der Standardkonfiguration unterstützt wird. Es kann durch folgende Zeilen in `/etc/rc.conf` aktiviert werden:

1. ```
   nisdomainname="test-domain"
   ```

   Diese Zeile setzt den NIS-Domänennamen auf `test-domain` .

2. ```
   nis_server_enable="YES"
   ```

   Dadurch werden die NIS-Serverprozesse beim Systemstart automatisch ausgeführt.

3. ```
   nis_yppasswdd_enable="YES"
   ```

   Durch diese Zeile wird der rpc.yppasswdd(8)-Daemon aktiviert, der die Änderung von NIS-Passwörtern von einem Client aus ermöglicht.

Wird ypserv in einer Multi-Serverdomäne verwendet, in der NIS-Server gleichzeitig als NIS-Clients arbeiten, ist es eine gute Idee, diese Server zu zwingen, sich an sich selbst zu binden. Damit wird verhindert, dass Bindeanforderungen gesendet werden und sich die Server gegenseitig binden. Sonst könnten seltsame Fehler auftreten, wenn ein Server ausfällt, auf den andere Server angewiesen sind. Letztlich werden alle Clients einen Timeout melden, und versuchen, sich an andere Server zu binden. Die dadurch entstehende Verzögerung kann beträchtlich sein. Außerdem kann der Fehler erneut auftreten, da sich die Server wiederum aneinander binden könnten.

Server, die auch als Client arbeiten, können durch das Hinzufügen der folgenden Zeilen in `/etc/rc.conf` zu gezwungen werden, sich an einen bestimmten Server zu binden:

```
nis_client_enable="YES" # run client stuff as well
nis_client_flags="-S NIS domain,server"
```

Nachdem die Parameter konfiguriert wurden, muss noch `/etc/netstart` ausgeführt werden, um alles entsprechend den Vorgaben in `/etc/rc.conf` einzurichten. Bevor die NIS-Maps einrichtet werden können, muss der ypserv(8)-Daemon manuell gestartet werden:

```
# service ypserv start
```

29.4.4.1. Die NIS-Maps initialisieren

NIS-Maps Sie werden am NIS-Masterserver aus den Konfigurationsdateien unter `/etc` erzeugt. Einzige Ausnahme: `/etc/master.passwd`. Dies verhindert, dass die Passwörter für `root`- oder andere Administratorkonten an alle Server in der NIS-Domäne verteilt werden. Deshalb werden die primären Passwort-Dateien konfiguriert, bevor die NIS-Maps initialisiert werden:

```
# cp /etc/master.passwd /var/yp/master.passwd
# cd /var/yp
# vi master.passwd
```
```

Es ist ratsam, alle Einträge für Systemkonten sowie Benutzerkonten, die nicht an die NIS-Clients weitergegeben werden sollen, wie beispielsweise `root` und weitere administrative Konten, zu entfernen.

Anmerkung

Stellen Sie sicher, dass `/var/yp/master.passwd` weder von der Gruppe noch von der Welt gelesen werden kann, indem Sie Zugriffsmodus auf `600` einstellen.

Nun können die NIS-Maps initialisiert werden. FreeBSD verwendet dafür das Skript ypinit(8). Geben Sie `-m` und den NIS-Domänennamen an, wenn Sie NIS-Maps für den Masterserver erzeugen:

```
ellington# ypinit -m test-domain
Server Type: MASTER Domain: test-domain
Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.
Do you want this procedure to quit on non-fatal errors? [y/n: n] n
Ok, please remember to go back and redo manually whatever fails.
If not, something might not work.
At this point, we have to construct a list of this domains YP servers.
rod.darktech.org is already known as master server.
Please continue to add any slave servers, one per line. When you are
done with the list, type a <control D>.
master server   :  ellington
next host to add:  coltrane
next host to add:  ^D
The current list of NIS servers looks like this:
ellington
coltrane
Is this correct?  [y/n: y] y

[..output from map generation..-]

NIS Map update completed.
ellington has been setup as an YP master server without any errors.
```

Dadurch erzeugt `ypinit /var/yp/Makefile` aus `/var/yp/Makefile.dist`. Diese Datei geht in der Voreinstellung davon aus, dass in einer NIS-Umgebung mit nur einem Server gearbeitet wird und dass alle Clients unter FreeBSD laufen. Da `test-domain` aber auch über einen Slaveserver verfügt, muss `/var/yp/Makefile` entsprechend angepasst werden, sodass es mit einem Kommentar (`#`) beginnt:

```
NOPUSH = "True"
```

29.4.4.2. Neue Benutzer hinzufügen

Jedes Mal, wenn ein neuer Benutzer angelegt wird, muss er am NIS-Masterserver hinzugefügt und die NIS-Maps anschließend neu erzeugt werden. Wird dieser Punkt vergessen, kann sich der neue Benutzer *nur* am NIS-Masterserver anmelden. Um beispielsweise den neuen Benutzer `jsmith` zur Domäne `test-domain` hinzufügen wollen, müssen folgende Kommandos auf dem Masterserver ausgeführt werden:

```
# pw useradd jsmith
# cd /var/yp
# make test-domain
```

Statt `pw useradd jsmith` kann auch `adduser jsmith` verwendet werden.

29.4.5. Einen NIS-Slaveserver einrichten

Um einen NIS-Slaveserver einzurichten, melden Sie sich am Slaveserver an und bearbeiten Sie `/etc/rc.conf` analog zum Masterserver. Erzeugen Sie aber keine NIS-Maps, da diese bereits auf dem Server vorhanden sind. Wenn

`ypinit` auf dem Slaveserver ausgeführt wird, benutzen Sie `-s` (Slave) statt `-m` (Master). Diese Option benötigt den Namen des NIS-Masterservers und den Domänennamen, wie in diesem Beispiel zu sehen:

```
coltrane# ypinit -s ellington test-domain

Server Type: SLAVE Domain: test-domain Master: ellington

Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.

Do you want this procedure to quit on non-fatal errors? [y/n: n]  n

Ok, please remember to go back and redo manually whatever fails.
If not, something might not work.
There will be no further questions. The remainder of the procedure
should take a few minutes, to copy the databases from ellington.
Transferring netgroup...
ypxfr: Exiting: Map successfully transferred
Transferring netgroup.byuser...
ypxfr: Exiting: Map successfully transferred
Transferring netgroup.byhost...
ypxfr: Exiting: Map successfully transferred
Transferring master.passwd.byuid...
ypxfr: Exiting: Map successfully transferred
Transferring passwd.byuid...
ypxfr: Exiting: Map successfully transferred
Transferring passwd.byname...
ypxfr: Exiting: Map successfully transferred
Transferring group.bygid...
ypxfr: Exiting: Map successfully transferred
Transferring group.byname...
ypxfr: Exiting: Map successfully transferred
Transferring services.byname...
ypxfr: Exiting: Map successfully transferred
Transferring rpc.bynumber...
ypxfr: Exiting: Map successfully transferred
Transferring rpc.byname...
ypxfr: Exiting: Map successfully transferred
Transferring protocols.byname...
ypxfr: Exiting: Map successfully transferred
Transferring master.passwd.byname...
ypxfr: Exiting: Map successfully transferred
Transferring networks.byname...
ypxfr: Exiting: Map successfully transferred
Transferring networks.byaddr...
ypxfr: Exiting: Map successfully transferred
Transferring netid.byname...
ypxfr: Exiting: Map successfully transferred
Transferring hosts.byaddr...
ypxfr: Exiting: Map successfully transferred
Transferring protocols.bynumber...
ypxfr: Exiting: Map successfully transferred
Transferring ypservers...
ypxfr: Exiting: Map successfully transferred
Transferring hosts.byname...
ypxfr: Exiting: Map successfully transferred

coltrane has been setup as an YP slave server without any errors.
Remember to update map ypservers on ellington.
```

Hierbei wird auf dem Slaveserver ein Verzeichnis namens `/var/yp/test-domain` erstellt, welches Kopien der NIS-Masterserver-Maps enthält. Durch hinzufügen der folgenden Zeilen in `/etc/crontab` wird der Slaveserver angewiesen, seine Maps mit den Maps des Masterservers zu synchronisieren:

```
20      *       *       *       *       root   /usr/libexec/ypxfr passwd.byname
21      *       *       *       *       root   /usr/libexec/ypxfr passwd.byuid
```

Diese Einträge sind nicht zwingend notwendig, da der Masterserver automatisch versucht, alle Änderungen seiner NIS-Maps an seine Slaveserver weiterzugeben. Da Passwortinformationen aber auch für nur vom Slaveserver abhängige Systeme vital sind, ist es eine gute Idee, diese Aktualisierungen zu erzwingen. Besonders wichtig ist dies in stark ausgelasteten Netzen, in denen Map-Aktualisierungen unvollständig sein könnten.

Um die Konfiguration abzuschließen, führen Sie `/etc/netstart` auf dem Slaveserver aus, um die NIS-Dienste erneut zu starten.

29.4.6. Einen NIS-Client einrichten

Ein NIS-Client `bindet` sich unter Verwendung von `ypbind` an einen NIS-Server. Dieser Daemon sendet RPC-Anfragen auf dem lokalen Netzwerk. Diese Anfragen legen den Namen der Domäne fest, die auf dem Client konfiguriert ist. Wenn der Server der entsprechenden Domäne eine solche Anforderung erhält, schickt er eine Antwort an `ypbind`, das wiederum die Adresse des Servers speichert. Wenn mehrere Server verfügbar sind, verwendet der Client die erste erhaltene Adresse und richtet alle Anfragen an genau diesen Server. `ypbind` „pingt“ den Server gelegentlich an, um sicherzustellen, dass der Server funktioniert. Antwortet der Server innerhalb eines bestimmten Zeitraums nicht (Timeout), markiert `ypbind` die Domäne als ungebunden und beginnt erneut, RPCs über das Netzwerk zu verteilen, um einen anderen Server zu finden.

Einen FreeBSD-Rechner als NIS-Client einrichten:

1. Fügen Sie folgende Zeilen in `/etc/rc.conf` ein, um den NIS-Domänennamen festzulegen, und um ypbind(8) bei der Initialisierung des Netzwerks zu starten:

   ```
   nisdomainname="test-domain"
   nis_client_enable="YES"
   ```

2. Um alle Passworteinträge des NIS-Servers zu importieren, löschen Sie alle Benutzerkonten in `/etc/master.passwd` mit `vipw`. Denken Sie daran, zumindest ein lokales Benutzerkonto zu behalten. Dieses Konto sollte außerdem Mitglied der Gruppe `wheel` sein. Wenn es mit NIS Probleme gibt, können Sie diesen Zugang verwenden, um sich als Superuser anzumelden und das Problem zu beheben. Bevor Sie die Änderungen speichern, fügen Sie folgende Zeile am Ende der Datei hinzu:

   ```
   +::::::::::
   ```

   Diese Zeile legt für alle gültigen Benutzerkonten der NIS-Server-Maps einen Zugang an. Es gibt verschiedene Wege, den NIS-Client durch Änderung dieser Zeile zu konfigurieren. Eine Methode wird in Abschnitt 29.4.8, „Netzgruppen verwenden“ beschrieben. Weitere detaillierte Informationen finden Sie im Buch `Managing NFS and NIS` vom O'Reilly Verlag.

3. Um alle möglichen Gruppeneinträge vom NIS-Server zu importieren, fügen Sie folgende Zeile in `/etc/group` ein:

   ```
   +:*::
   ```

Um den NIS-Client direkt zu starten, führen Sie als Superuser die folgenden Befehle aus:

```
# /etc/netstart
# service ypbind start
```

Danach sollte bei der Eingabe von `ypcat passwd` auf dem Client die `passwd-Map` des NIS-Servers angezeigt werden.

29.4.7. Sicherheit unter NIS

Da RPC ein Broadcast-basierter Dienst ist, kann jedes System innerhalb der Domäne mittels ypbind den Inhalt der NIS-Maps abrufen. Um nicht autorisierte Transaktionen zu verhindern, unterstützt ypserv(8) eine Funktion namens „securenets“, durch die der Zugriff auf bestimmte Rechner beschränkt werden kann. In der Voreinstellung sind diese Informationen in `/var/yp/securenets` gespeichert, es sei denn, ypserv(8) wurde mit der Option `-p` und einem alternativen Pfad gestartet. Diese Datei enthält Einträge, die aus einer Netzwerkadresse und einer Netzmaske bestehen. Kommentarzeilen beginnen mit „#“. `/var/yp/securnets` könnte beispielsweise so aussehen:

```
# allow connections from local host -- mandatory
127.0.0.1     255.255.255.255
# allow connections from any host
# on the 192.168.128.0 network
192.168.128.0 255.255.255.0
# allow connections from any host
# between 10.0.0.0 to 10.0.15.255
# this includes the machines in the testlab
10.0.0.0      255.255.240.0
```

Wenn ypserv(8) eine Anforderung von einer zu diesen Regeln passenden Adresse erhält, wird die Anforderung bearbeitet. Gibt es keine passende Regel, wird die Anforderung ignoriert und eine Warnmeldung aufgezeichnet. Wenn `securenets` nicht existiert, erlaubt `ypserv` Verbindungen von jedem Rechner.

Abschnitt 14.6, „TCP-Wrapper“ beschreibt eine alternative Methode zur Zugriffskontrolle. Obwohl beide Methoden einige Sicherheit gewähren, sind sie anfällig für „IP-Spoofing“-Angriffe. Der NIS-Verkehr sollte daher von einer Firewall blockiert werden.

Server, die `securenets` verwenden, können Schwierigkeiten bei der Anmeldung von NIS-Clients haben, die ein veraltetes TCP/IP-Subsystem besitzen. Einige dieser TCP/IP-Subsysteme setzen alle Rechnerbits auf Null, wenn sie einen `Broadcast` durchführen oder können die Subnetzmaske nicht auslesen, wenn sie die Broadcast-Adresse berechnen. Einige Probleme können durch Änderungen der Clientkonfiguration behoben werden. Andere hingegen lassen sich nur durch das Entfernen des betreffenden Rechners aus dem Netzwerk oder den Verzicht auf `securenets` umgehen.

Die Verwendung der TCP-Wrapper verlangsamt die Reaktion des NIS-Servers. Diese zusätzliche Reaktionszeit kann in Clientprogrammen zu Timeouts führen. Dies vor allem in Netzwerken, die stark ausgelastet sind, oder nur über langsame NIS-Server verfügen. Wenn ein oder mehrere Clients dieses Problem aufweisen, sollten Sie die betreffenden Clients in NIS-Slaveserver umwandeln, und diese an sich selbst binden.

29.4.7.1. Bestimmte Benutzer an der Anmeldung hindern

In diesem Beispiel gibt es innerhalb der NIS-Domäne den Rechner `basie`, der nur für Mitarbeiter der Fakultät bestimmt ist. Die `passwd` Datenbank des NIS-Masterservers enthält Benutzerkonten sowohl für Fakultätsmitarbeiter als auch für Studenten. Dieser Abschnitt beschreibt, wie Sie den Mitarbeitern der Fakultät die Anmeldung am System ermöglichen, während den Studenten die Anmeldung verweigert wird.

Es gibt eine Möglichkeit, bestimmte Benutzer an der Anmeldung an einem bestimmten Rechner zu hindern, selbst wenn diese in der NIS-Datenbank vorhanden sind. Dazu kann mit `vipw` der Eintrag *`-Benutzername`* und die richtige Anzahl von Doppelpunkten an das Ende von `/etc/master.passwd` gesetzt werden, wobei *`Benutzername`* der zu blockierende Benutzername ist. Die Zeile mit dem geblockten Benutzer muss dabei vor der `+` Zeile, für zugelassene Benutzer stehen. In diesem Beispiel wird die Anmeldung für den Benutzer `bill` am Rechner `basie` blockiert:

```
basie# cat /etc/master.passwd
root:[password]:0:0::0:0:The super-user:/root:/bin/csh
toor:[password]:0:0::0:0:The other super-user:/root:/bin/sh
daemon:*:1:1::0:0:Owner of many system processes:/root:/sbin/nologin
operator:*:2:5::0:0:System &:/:/sbin/nologin
bin:*:3:7::0:0:Binaries Commands and Source,,,:/:/sbin/nologin
tty:*:4:65533::0:0:Tty Sandbox:/:/sbin/nologin
kmem:*:5:65533::0:0:KMem Sandbox:/:/sbin/nologin
games:*:7:13::0:0:Games pseudo-user:/usr/games:/sbin/nologin
news:*:8:8::0:0:News Subsystem:/:/sbin/nologin
man:*:9:9::0:0:Mister Man Pages:/usr/share/man:/sbin/nologin
bind:*:53:53::0:0:Bind Sandbox:/:/sbin/nologin
uucp:*:66:66::0:0:UUCP pseudo-user:/var/spool/uucppublic:/usr/libexec/uucp/uucico
xten:*:67:67::0:0:X-10 daemon:/usr/local/xten:/sbin/nologin
pop:*:68:6::0:0:Post Office Owner:/nonexistent:/sbin/nologin
nobody:*:65534:65534::0:0:Unprivileged user:/nonexistent:/sbin/nologin
-bill:::::::::
+:::::::::
```

```
basie#
```

29.4.8. Netzgruppen verwenden

Bestimmten Benutzern die Anmeldung an einzelnen Systemen zu verweigern, kann in großen Netzwerken schnell unübersichtlich werden. Dadurch verlieren Sie den Hauptvorteil von NIS: die *zentrale* Verwaltung.

Netzgruppen wurden entwickelt, um große, komplexe Netzwerke mit Hunderten Benutzern und Rechnern zu verwalten. Ihre Aufgabe ist vergleichbar mit UNIX® Gruppen. Die Hauptunterschiede sind das Fehlen einer numerischen ID sowie die Möglichkeit, Netzgruppen zu definieren, die sowohl Benutzer als auch andere Netzgruppen enthalten.

Um das Beispiel in diesem Kapitel fortzuführen, wird die NIS-Domäne um zusätzliche Benutzer und Rechner erweitert:

Tabelle 29.2. Zusätzliche Benutzer

Benutzername(n)	Beschreibung
`alpha`, `beta`	Mitarbeiter der IT-Abteilung
`charlie`, `delta`	Lehrlinge der IT-Abteilung
`echo`, `foxtrott`, `golf`, ...	Mitarbeiter
`able`, `baker`, ...	Praktikanten

Tabelle 29.3. Zusätzliche Rechner

Rechnername(n)	Beschreibung
`war`, `death`, `famine`, `pollution`	Nur Mitarbeiter der IT-Abteilung dürfen sich an diesen Rechnern anmelden.
`pride`, `greed`, `envy`, `wrath`, `lust`, `sloth`	Nur Mitarbeiter und Lehrlinge der IT-Abteilung dürfen sich auf diesen Rechnern anmelden.
`one`, `two`, `three`, `four`, ...	Gewöhnliche Arbeitsrechner für Mitarbeiter.
`trashcan`	Ein sehr alter Rechner ohne kritische Daten. Sogar Praktikanten dürfen diesen Rechner verwenden.

Bei der Verwendung von Netzgruppen wird jeder Benutzer einer oder mehreren Netzgruppen zugewiesen und die Anmeldung wird dann für die Netzgruppe erlaubt oder verwehrt. Wenn ein neuer Rechner hinzugefügt wird, müssen die Zugangsbeschränkungen nur für die Netzgruppen festgelegt werden. Wird ein neuer Benutzer angelegt, muss er einer oder mehreren Netzgruppen zugewiesen werden. Wenn die Einrichtung von NIS sorgfältig geplant wurde, muss nur noch eine zentrale Konfigurationsdatei bearbeitet werden, um den Zugriff auf bestimmte Rechner zu erlauben oder zu verbieten.

Dieses Beispiel erstellt vier Netzgruppen: IT-Mitarbeiter, IT-Lehrlinge, normale Mitarbeiter sowie Praktikanten:

```
IT_EMP  (,alpha,test-domain)    (,beta,test-domain)
IT_APP  (,charlie,test-domain)  (,delta,test-domain)
USERS   (,echo,test-domain)     (,foxtrott,test-domain) \
        (,golf,test-domain)
INTERNS (,able,test-domain)     (,baker,test-domain)
```

Jede Zeile konfiguriert eine Netzgruppe. Die erste Spalte der Zeile bezeichnet den Namen der Netzgruppe. Die Einträge in den Klammern stehen entweder für eine Gruppe von einem oder mehreren Benutzern, oder für den Namen einer weiteren Netzgruppe. Wenn ein Benutzer angegeben wird, haben die drei Felder in der Klammer folgende Bedeutung:

1. Der Name des Rechner(s), auf dem die weiteren Felder für den Benutzer gültig sind. Wird kein Rechnername festgelegt, ist der Eintrag auf allen Rechnern gültig.

2. Der Name des Benutzerkontos, der zu dieser Netzgruppe gehört.

3. Die NIS-Domäne für das Benutzerkonto. Benutzerkonten können von anderen NIS-Domänen in eine Netzgruppe importiert werden.

Wenn eine Gruppe mehrere Benutzer enthält, müssen diese durch Leerzeichen getrennt werden. Darüber hinaus kann jedes Feld Wildcards enthalten. Weitere Einzelheiten finden Sie in netgroup(5).

Netzgruppennamen sollten nicht länger als 8 Zeichen sein. Es wird zwischen Groß- und Kleinschreibung unterschieden. Die Verwendung von Großbuchstaben für Netzgruppennamen ermöglicht eine leichte Unterscheidung zwischen Benutzern, Rechnern und Netzgruppen.

Einige NIS-Clients (dies gilt nicht für FreeBSD) können keine Netzgruppen mit mehr als 15 Einträgen verwalten. Diese Grenze kann umgangen werden, indem mehrere Subnetzgruppen mit weniger als fünfzehn Benutzern angelegt werden und diese Subnetzgruppen wiederum in einer Netzgruppe zusammengefasst wird, wie in diesem Beispiel zu sehen:

```
BIGGRP1  (,joe1,domain)  (,joe2,domain)  (,joe3,domain) [...-]
BIGGRP2  (,joe16,domain)  (,joe17,domain) [...-]
BIGGRP3  (,joe31,domain)  (,joe32,domain)
BIGGROUP  BIGGRP1 BIGGRP2 BIGGRP3
```

Wiederholen Sie diesen Vorgang, wenn mehr als 225 (15*15) Benutzer in einer einzigen Netzgruppe existieren.

Die neue NIS-Map aktivieren und verteilen:

```
ellington# cd /var/yp
ellington# make
```

Dadurch werden die NIS-Maps `netgroup`, `netgroup.byhost` und `netgroup.byuser` erzeugt. Prüfen Sie die Verfügbarkeit der neuen NIS-Maps mit ypcat(1):

```
ellington% ypcat -k netgroup
ellington% ypcat -k netgroup.byhost
ellington% ypcat -k netgroup.byuser
```

Die Ausgabe des ersten Befehls gibt den Inhalt von `/var/yp/netgroup` wieder. Der zweite Befehl erzeugt nur dann eine Ausgabe, wenn rechnerspezifische Netzgruppen erzeugt wurden. Der dritte Befehl gibt die Netzgruppen nach Benutzern sortiert aus.

Wenn Sie einen Client einrichten, verwenden Sie vipw(8) um den Namen der Netzgruppe anzugeben. Ersetzen Sie beispielsweise auf dem Server namens `war` die folgende Zeile:

```
+::::::::::
```

durch

```
+@IT_EMP:::::::::
```

ersetzt werden.

Diese Zeile legt fest, dass nur noch Benutzer der Netzgruppe `IT_EMP` in die Passwortdatenbank dieses Systems importiert werden. Nur diese Benutzer dürfen sich an diesem Server anmelden.

Diese Konfiguration gilt auch für die ~-Funktion der Shell und für alle Routinen, die auf Benutzernamen und numerische Benutzer-IDs zugreifen. Oder anders formuliert, `cd ~user` ist nicht möglich, `ls -l` zeigt die numerische Benutzer-ID statt dem Benutzernamen und `find . -user joe -print` erzeugt die Fehlermeldung No such user. Um dieses Problem zu beheben, müssen alle Benutzereinträge importiert werden, ohne ihnen jedoch zu erlauben, sich am Server anzumelden. Dies kann durch das Hinzufügen einer zusätzlichen Zeile erreicht werden:

```
+:::::::::/sbin/nologin
```

Diese Zeile weist den Client an, alle Einträge zu importieren, aber die Shell in diesen Einträgen durch `/sbin/nologin` zu ersetzen.

Stellen Sie sicher, dass die zusätzliche Zeile *nach* der Zeile `+@IT_EMP:::::::::` eingetragen ist. Andernfalls haben alle via NIS importierten Benutzerkonten `/sbin/nologin` als Loginshell und niemand wird sich mehr am System anmelden können.

Um die weniger wichtigen Server zu konfigurieren, ersetzen Sie den alten Eintrag `+:::::::::` auf den Servern mit diesen Zeilen:

```
+@IT_EMP:::::::::
+@IT_APP:::::::::
+:::::::::/sbin/nologin
```

Die entsprechenden Zeilen für Arbeitsplätze lauten:

```
+@IT_EMP:::::::::
+@USERS:::::::::
+:::::::::/sbin/nologin
```

NIS ist in der Lage, Netzgruppen aus anderen Netzgruppen zu bilden. Dies kann nützlich sein, wenn sich die Firmenpolitik ändert. Eine Möglichkeit ist die Erzeugung rollenbasierter Netzgruppen. Sie könnten eine Netzgruppe `BIGSRV` erzeugen, um den Zugang zu den wichtigsten Servern zu beschränken, eine weitere Gruppe `SMALLSRV` für die weniger wichtigen Server und eine dritte Netzgruppe `USERBOX` für die Arbeitsplatzrechner. Jede dieser Netzgruppen enthält die Netzgruppen, die sich auf diesen Rechnern anmelden dürfen. Die Einträge der Netzgruppen in der NIS-Map sollten ähnlich den folgenden aussehen:

```
BIGSRV    IT_EMP  IT_APP
SMALLSRV  IT_EMP  IT_APP  ITINTERN
USERBOX   IT_EMP  ITINTERN USERS
```

Diese Methode funktioniert besonders gut, wenn Rechner in Gruppen mit identischen Beschränkungen eingeteilt werden können. Unglücklicherweise ist dies die Ausnahme und nicht die Regel. Meistens wird die Möglichkeit zur rechnerspezischen Zugangsbeschränkung benötigt.

Rechnerspezifische Netzgruppen sind eine weitere Möglichkeit, um mit den oben beschriebenen Änderungen umzugehen. In diesem Szenario enthält `/etc/master.passwd` auf jedem Rechner zwei mit „+“ beginnende Zeilen. Die erste Zeile legt die Netzgruppe mit den Benutzern fest, die sich auf diesem Rechner anmelden dürfen. Die zweite Zeile weist allen anderen Benutzern `/sbin/nologin` als Shell zu. Verwenden Sie auch hier (analog zu den Netzgruppen) Großbuchstaben für die Rechnernamen:

```
+@BOXNAME:::::::::
+:::::::::/sbin/nologin
```

Sobald dies für alle Rechner erledigt ist, müssen die lokalen Versionen von `/etc/master.passwd` nie mehr verändert werden. Alle weiteren Änderungen geschehen über die NIS-Maps. Nachfolgend ein Beispiel für eine mögliche Netzgruppen-Map:

```
# Define groups of users first
IT_EMP    (,alpha,test-domain)    (,beta,test-domain)
IT_APP    (,charlie,test-domain)  (,delta,test-domain)
DEPT1     (,echo,test-domain)     (,foxtrott,test-domain)
DEPT2     (,golf,test-domain)     (,hotel,test-domain)
DEPT3     (,india,test-domain)    (,juliet,test-domain)
ITINTERN  (,kilo,test-domain)     (,lima,test-domain)
D_INTERNS (,able,test-domain)     (,baker,test-domain)
#
# Now, define some groups based on roles
USERS     DEPT1   DEPT2     DEPT3
BIGSRV    IT_EMP  IT_APP
```

```
SMALLSRV  IT_EMP  IT_APP    ITINTERN
USERBOX   IT_EMP  ITINTERN  USERS
#
# And a groups for a special tasks
# Allow echo and golf to access our anti-virus-machine
SECURITY  IT_EMP  (,echo,test-domain)  (,golf,test-domain)
#
# machine-based netgroups
# Our main servers
WAR       BIGSRV
FAMINE    BIGSRV
# User india needs access to this server
POLLUTION  BIGSRV  (,india,test-domain)
#
# This one is really important and needs more access restrictions
DEATH     IT_EMP
#
# The anti-virus-machine mentioned above
ONE       SECURITY
#
# Restrict a machine to a single user
TWO       (,hotel,test-domain)
# [...more groups to follow]
```

Es ist nicht immer ratsam, rechnerbasierte Netzgruppen zu verwenden. Wenn Dutzende oder Hunderte identische Rechner eingerichtet werden müssen, sollten rollenbasierte Netzgruppen verwendet werden, um die Größe der NIS-Maps in Grenzen zu halten.

29.4.9. Passwortformate

Alle Rechner innerhalb der NIS-Domäne müssen für die Verschlüsselung von Passwörtern das gleiche Format benutzen. Wenn Benutzer Schwierigkeiten bei der Authentifizierung auf einem NIS-Client haben, liegt dies möglicherweise an einem anderen Passwort-Format. In einem heterogenen Netzwerk muss das verwendete Format von allen Betriebssystemen unterstützt werden, wobei DES der kleinste gemeinsame Standard ist.

Welches Format die Server und Clients verwenden, steht in `/etc/login.conf`:

```
default:\
 :passwd_format=des:\
 :copyright=/etc/COPYRIGHT:\
 [weitere Einträge]
```

In diesem Beispiel verwendet das System das Format DES. Weitere mögliche Werte sind unter anderem `blf` und `md5` (mit Blowfish und MD5 verschlüsselte Passwörter).

Wird auf einem Rechner das Format entsprechend der NIS-Domäne geändert, muss anschließend die Login-Capability Datenbank neu erstellt werden:

```
# cap_mkdb /etc/login.conf
```

Anmerkung

Das Format der schon bestehenden Passwörter wird erst aktualisiert, wenn ein Benutzer sein Passwort ändert, *nachdem* die Datenbank neu erstellt wurde.

29.5. Lightweight Access Directory Protocol (LDAP)

Das Lightweight Directory Access Protocol (LDAP) ist ein Protokoll der Anwendungsschicht und wird verwendet, um Objekte mithilfe eines verteilten Verzeichnisdienstes abzurufen, zu verändern und zu authentifizieren. Betrachten Sie es als ein Telefonbuch, das homogene Informationen in mehreren hierarchischen Ebenen speichert. Es wird häufig in Netzwerken genutzt, in denen Benutzer unter Verwendung eines einzigen Kontos auf diverse interne Informationen zugreifen müssen. Beispielsweise kann E-Mail-Authentifizierung, Abfrage von Kontaktinformationen und Website-Authentifizierung über ein einzelnes Benutzerkonto aus der Datenbank des LDAP-Servers erfolgen.

Dieser Abschnitt behandelt nicht die Geschichte oder Details der Implementierung des Protokolls. Diese Abschnitte wurden verfasst, um einen LDAP-Server und/oder -Client sowohl schnell, als auch sicher zu konfigurieren. Jedoch erfordert jede Informationsbasis sorgfältige Planung, und dies ist keine Ausnahme.

Bei der Planung sollte bestimmt werden, welche Art von Informationen gespeichert werden, für was diese Informationen verwendet werden, wer Zugriff auf die Daten bekommen soll, und wie diese Daten vor neugierigen Blicken geschützt werden können.

29.5.1. LDAP Terminologie und Struktur

Da bei der Konfiguration leicht Verwechselungen entstehen, werden zuvor einige Aspekte von LDAP erklärt. Zunächst bestehen alle Verzeichniseinträge aus einer Gruppe von *Attributen*. Jede Attributgruppe enthält einen Namen, also einen eindeutigen Bezeichner, der als DN oder *distinguished name* bekannt ist. Dieser setzt sich normalerweise aus mehreren anderen Attributen, wie dem RDN zusammen. RDN, oder *relative distinguished name*, ist ein geläufiger Begriff für ein Attribut. Wie bei Verzeichnissen gibt es auch hier absolute und relative Pfade. Betrachten Sie DN als absoluten Pfad und RDN als relativen Pfad.

Beispielsweise könnte ein Eintrag wie folgt aussehen:

```
% ldapsearch -xb "uid=trhodes,ou=users,o=example.com"
```

```
# extended LDIF
#
# LDAPv3
# base <uid=trhodes,ou=users,o=example.com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#


# trhodes, users, example.com
dn: uid=trhodes,ou=users,o=example.com
mail: trhodes@example.com
cn: Tom Rhodes
uid: trhodes
telephoneNumber: (xxx) xxx-xxxx

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries:1
```

Obwohl die Bedeutung der einzelnen Attribute in diesem Beispiel offensichtlich ist, sollte das Attribut cn genauer betrachtet werden. Dies ist der zuvor beschriebene RDN. Darüber hinaus gibt es eine eindeutige Benutzer-ID. Es ist gängige Praxis, in den Einträgen einheitliche uid oder uuids zu haben, um zukünftige Migrationen zu erleichtern.

29.5.2. Konfiguration eines LDAP-Servers

Um FreeBSD so zu konfigurieren, dass es als LDAP-Server fungiert, muss der Port OpenLDAP installiert werden. Dies kann unter Verwendung von `pkg_add` , oder durch Installation von net/openldap24-server erreicht werden. Der Bau des Ports wird empfohlen, da der Administrator zu diesem Zeitpunkt eine Menge Optionen aktivieren und

deaktivieren kann. In den meisten Fällen werden die Standardwerte ausreichend sein. Sollte jedoch Unterstützung für SQL nötig sein, ist dies der richtige Zeitpunkt zur Aktivierung.

Von nun an werden ein paar Verzeichnisse benötigt. Ein Verzeichnis für Daten, sowie ein Verzeichnis zum Speichern der Zertifikate. Erstellen Sie beide Verzeichnisse mit den folgenden Befehlen:

```
# mkdir /var/db/openldap-data
```

```
# mkdir /usr/local/etc/openldap/private
```

Kopieren Sie die Konfigurationsdatei der Datenbank:

```
# cp /usr/local/etc/openldap/DB_CONFIG.example /var/db/openldap-data/DB_CONFIG
```

Im nächsten Schritt werden die SSL-Zertifikate konfiguriert. Das Erstellen von Zertifikaten wird zwar in OpenSSL beschrieben, aber hier ist eine Zertifizierungsstelle erforderlich, weshalb eine andere Methode verwendet wird. Es wird empfohlen, diesen Teil genau zu überprüfen, um sicherzustellen, dass bei der Erstellung des Zertifikats die richtigen Informationen eingetragen werden.

Die folgenden Befehle müssen im Verzeichnis `/usr/local/etc/openldap/private` ausgeführt werden. Dies ist wichtig, da die Dateiberechtigungen restriktiv gesetzt werden und Benutzer keinen direkten Zugriff auf diese Daten haben sollten. Geben Sie folgende Befehle ein, um die Zertifikate zu erstellen:

```
# openssl req -days 365 -nodes -new -x509 -keyout ca.key -out ../ca.crt
```

Diese Einträge sind frei wählbar, *mit Ausnahme* von *Common Name*. Hier muss etwas anderes als der Hostname des Systems eingetragen werden, ansonsten würde das System versuchen, den eigenen Hostnamen zu überprüfen. Wenn wie in diesem Beispiel ein selbstsigniertes Zertifikat verwendet wird, stellen Sie dem Hostnamen einfach das Präfix CA für die Zertifizierungsstelle voran.

Die nächste Aufgabe besteht darin, einen Zertifikatsregistrierungsanforderung (CSR) sowie einen privaten Schlüssel zu erstellen. Dazu geben Sie die folgenden Befehle ein:

```
# openssl req -days 365 -nodes -new -keyout server.key -out server.csr
```

Stellen Sie hierbei sicher, dass der *common name* richtig eingetragen wird. Anschließend muss der Schlüssel signiert werden:

```
# openssl x509 -req -days 365 -in server.csr -out ../server.crt -CA ../ca.crt -CAkey ca.key
 -CAcreateserial
```

Der letzte Schritt für die Erstellung der Zertifikate besteht darin, die Client-Zertifikate zu erstellen und zu signieren:

```
# openssl req -days 365 -nodes -new -keyout client.key -out client.csr
```

```
# openssl x509 -req -days 3650 -in client.csr -out ../client.crt -CAkey ca.key
```

Achten Sie wieder auf das Attribut *common name*. Dies sorgt häufig für Verwirrung bei der erstmaligen Konfiguration von LDAP. Stellen Sie außerdem sicher, dass bei diesem Verfahren acht (8) neue Dateien erzeugt worden sind. Der nächste Schritt besteht darin, `/usr/local/etc/openldap/slapd.conf` zu editieren und folgende Optionen hinzuzufügen:

```
TLSCipherSuite HIGH:MEDIUM:+SSLv3
TLSCertificateFile /usr/local/etc/openldap/server.crt
TLSCertificateKeyFile /usr/local/etc/openldap/private/server.key
TLSCACertificateFile /usr/local/etc/openldap/ca.crt
```

Editieren Sie zusätzlich `/usr/local/etc/openldap/ldap.conf` und fügen die folgenden Zeilen hinzu:

```
TLS_CACERT /usr/local/etc/openldap/ca.crt
TLS_CIPHER_SUITE HIGH:MEDIUM:+SSLv3
```

Setzen Sie für `BASE` die gewünschten Werte ein und kommentieren Sie die Optionen `URI`, `SIZELIMIT` und `TIMELIMIT` aus. Setzen Sie bei `URI` `ldap://` und `ldaps://` ein.

Die daraus resultierende Datei sollte der hier gezeigten ähnlich sehen:

```
BASE    dc=example,dc=com
URI     ldap:// ldaps://

SIZELIMIT       12
TIMELIMIT       15
#DEREF          never

TLS_CACERT /usr/local/etc/openldap/ca.crt$
TLS_CIPHER_SUITE HIGH:MEDIUM:+SSLv3
```

Anschließend sollte das Standardpasswort für den Server geändert werden. Das folgende Kommando schreibt die Ausgabe in `slapd.conf`:

```
# slappasswd -h "{SHA}" >> /usr/local/etc/openldap/slapd.conf
```

Dieser Befehl wird nach einem Passwort fragen und, wenn der Prozess nicht fehlschlägt, ein Passwort-Hash an das Ende von `slapd.conf` hinzufügen. `slappasswd` versteht verschiedene Hash-Formate. Weitere Informationen hierzu finden Sie in der Manualpage.

Bearbeiten Sie `/usr/local/etc/openldap/slapd.conf` und fügen folgende Zeilen hinzu:

```
password-hash {sha}
allow bind_v2
```

Das Suffix in dieser Datei muss `BASE` aus der vorherigen Konfiguration entsprechen. Zudem sollte die Option `rootdn` ebenfalls gesetzt werden. Ein guter Vorschlag ist beispielsweise `cn=Manager`. Bevor die Datei gespeichert wird, setzen Sie die Passwortausgabe von `slappasswd` hinter die Option `rootpw`. Das Endergebnis sollte in etwa wie folgt aussehen:

```
TLSCipherSuite HIGH:MEDIUM:+SSLv3
TLSCertificateFile /usr/local/etc/openldap/server.crt
TLSCertificateKeyFile /usr/local/etc/openldap/private/server.key
TLSCACertificateFile /usr/local/etc/openldap/ca.crt
rootpw {SHA}W6ph5Mm5Pz8GgiULbPgzG37mj9g=
```

Aktivieren Sie abschließend OpenLDAP in `/etc/rc.conf`. Zu diesem Zeitpunkt kann es sinnvoll sein, eine URI sowie die Gruppe und den Benutzer einzurichten. Editieren Sie dazu `/etc/rc.conf` und fügen folgende Zeilen hinzu:

```
slapd_enable="YES"
slapd_flags="-4 -h ldaps:///"
```

An dieser Stelle sollte der Server bereit sein, gestartet und getestet zu werden. Führen Sie dazu folgenden Befehl aus:

```
# service slapd start
```

Wurde alles richtig konfiguriert, sollte eine Suche im Verzeichnis, wie in diesem Beispiel, eine erfolgreiche Verbindung mit einer Antwort liefern:

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL

# search result
search: 3
result: 32 No such object
```

```
# numResponses: 1
```

Wenn der Dienst wie im Beispiel oben antwortet, kann das Verzeichnis mit dem Befehl `ldapadd` bestückt werden. In diesem Beispiel gibt es eine Datei mit einer Liste von Benutzern, die diesem Verzeichnis hinzugefügt werden. Erstellen Sie zunächst eine Datei mit folgendem Datensatz für den Import:

```
dn: dc=example,dc=com
objectclass: dcObject
objectclass: organization
o: Example
dc: Example

dn: cn=Manager,dc=example,dc=com
objectclass: organizationalRole
cn: Manager
```

Anmerkung

Zur Fehlersuche, stoppen Sie den `slapd`-Dienst mit dem `service`-Befehl. Starten Sie anschließend die Anwendung mit *Debugging*-Optionen:

```
# /usr/local/libexec/slapd -d -1
```

Angenommen das die Importdatei `import.ldif` heißt, geben Sie folgenden Befehl ein, um die Datendatei zu importieren:

```
# ldapadd -Z -D "cn=Manager,dc=example,dc=com" -W -f    import.ldif
```

Es wird wieder eine Aufforderung zur Passworteingabe geben und die Ausgabe sollte wie folgt aussehen:

```
Enter LDAP Password:
adding new entry "dc=example,dc=com"

adding new entry "cn=Manager,dc=example,dc=com"
```

Stellen Sie mit einer Suche auf dem Server sicher, dass die Daten importiert wurden. Nutzen Sie dazu `ldapsearch`. In diesem Fall sollte die Ausgabe wie folgt aussehen:

```
% ldapsearch -Z
```

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> (default) with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# example.com
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
o: Example
dc: Example

# Manager, example.com
dn: cn=Manager,dc=example,dc=com
objectClass: organizationalRole
cn: Manager
```

```
# search result
search: 3
result: 0 Success

# numResponses: 3
# numEntries: 2
```

Es ist natürlich sinnvoll, sich über die Struktur der LDAP-Verzeichnisse in den hier erwähnten Manualpages zu informieren. An dieser Stelle sollte der Server konfiguriert sein und ordnungsgemäß funktionieren.

29.6. Dynamic Host Configuration Protocol (DHCP)

Das Dynamic Host Configuration Protocol (DHCP) ermöglicht es einem System, sich mit einem Netzwerk zu verbinden und die für die Kommunikation mit diesem Netzwerk nötigen Informationen zu beziehen. FreeBSD verwendet den von OpenBSD stammenden `dhclient`, um die Adressinformationen zu beziehen. FreeBSD installiert keinen DHCP-Server, aber es stehen einige Server in der FreeBSD Ports-Sammlung zu Verfügung. Das DHCP-Protokoll wird vollständig im RFC 2131 beschrieben. Eine weitere, lehrreiche Informationsquelle existiert unter `isc.org/downloads/dhcp/`.

In diesem Abschnitt wird beschrieben, wie der integrierte DHCP-Client verwendet wird. Anschließend wird erklärt, wie ein DHCP-Server zu installieren und konfigurieren ist.

29.6.1. Einen DHCP-Client konfigurieren

Die Unterstützung für den DHCP-Client ist im Installationsprogramm von FreeBSD enthalten, sodass ein neu installiertes System automatisch die Adressinformationen des Netzwerks vom DHCP-Server erhält.

`dhclient` beginnt von einem Clientrechner aus über den UDP-Port 68 Konfigurationsinformationen anzufordern. Der Server antwortet auf dem UDP-Port 67, indem er dem Client eine IP-Adresse zuweist und ihm weitere relevante Informationen über das Netzwerk, wie Netzmasken, Router und DNS-Server mitteilt. Diese Informationen werden als *DHCP-Lease* bezeichnet und sind nur für bestimmte Zeit, die vom Administrator des DHCP-Servers vorgegeben wird, gültig. Dadurch fallen verwaiste IP-Adressen, deren Clients nicht mehr mit dem Netzwerk verbunden sind, automatisch an den Server zurück.

DHCP-Clients können sehr viele Informationen von einem DHCP-Server erhalten. Eine ausführliche Liste finden Sie in dhcp-options(5).

Das Gerät `bpf` ist im `GENERIC` -Kernel bereits enthalten. Für die Nutzung von DHCP muss also kein angepasster Kernel erzeugt werden. In einer angepassten Kernelkonfigurationsdatei muss das Gerät enthalten sein, damit DHCP ordnungsgemäß funktioniert.

Anmerkung

Diejenigen, die besonders um die Sicherheit besorgt sind, sollten wissen, dass `bpf` auch zur Ausführung von Paketsniffern erforderlich ist (obwohl diese dennoch als `root` ausgeführt werden müssen). `bpf` *muss* vorhanden sein, damit DHCP funktioniert. Sehr sicherheitsbewusste Personen, sollten `bpf` aus dem Kernel entfernen, wenn DHCP nicht verwendet wird.

Standardmässig läuft die DHCP-Konfiguration bei FreeBSD im Hintergrund oder auch *asynchron*. Andere Startskripte laufen weiter, während DHCP fertig abgearbeitet wird, was den Systemstart beschleunigt.

DHCP im Hintergrund funktioniert gut, wenn der DHCP-Server schnell auf Anfragen antwortet und der DHCP-Konfigurationsprozess ebenso schnell abläuft. Jedoch kann DHCP eine lange Zeit benötigen, um auf manchen Systemen fertig zu werden. Falls Netzwerkdienste versuchen, vor DHCP zum Ende zu kommen, werden diese fehlschlagen.

Durch die Verwendung von DHCP im *asynchronen* Modus wird das Problem verhindert, so dass die Startskripte pausiert werden, bis die DHCP-Konfiguration abgeschlossen ist.

Um sich zu einem DHCP-Server im Hintergrund zu verbinden, während andere Startskripte fortfahren (asynchroner Modus), benutzen Sie den „`DHCP`“-Wert in `/etc/rc.conf` :

```
ifconfig_fxp0="DHCP"
```

Um den Start zu pausieren, damit DHCP vorher abgeschlossen werden kann, benutzen Sie den synchronen Modus mit dem Eintrag „`SYNCDHCP`“:

```
ifconfig_fxp0="SYNCDHCP"
```

Anmerkung

Ersetzen Sie *`fxp0`*, das in diesen Beispielen verwendet wurde, durch den Namen Ihrer Netzwerkschnittstelle, so wie es in Abschnitt 12.5, „Einrichten von Netzwerkkarten“ beschrieben ist.

Wenn `dhclient` an einem anderen Ort installiert wurde, oder zusätzliche Flags an `dhclient` übergeben werden müssen, fügen Sie folgende (entsprechend angepasste) Zeilen ein:

```
dhclient_program="/sbin/dhclient"
dhclient_flags=""
```

Der DHCP-Client verwendet die folgenden Dateien:

- `/etc/dhclient.conf`

  `dhclient` benötigt die Konfigurationsdatei `/etc/dhclient.conf`. Diese Datei enthält normalerweise nur Kommentare, da die Vorgabewerte zumeist ausreichend sind. Die Konfigurationsdatei wird in dhclient.conf(5) beschrieben.

- `/sbin/dhclient`

  Weitere Informationen über `dhclient` finden Sie in dhclient(8).

- `/sbin/dhclient-script`

  Bei `dhclient-script` handelt es sich um das FreeBSD-spezifische Konfigurationsskript des DHCP-Clients. Es wird in dhclient-script(8) beschrieben und kann meist unverändert übernommen werden.

- `/var/db/dhclient.leases.` *`interface`*

  Der DHCP-Client verfügt über eine Datenbank, die alle derzeit gültigen Leases enthält und als Logdatei erzeugt wird. Weitere Informationen finden Sie in dhclient(8), dhcp-options(5) und dhclient.conf(5).

29.6.2. Einen DHCP-Server installieren und einrichten

Dieser Abschnitt beschreibt die Einrichtung eines FreeBSD-Systems als DHCP-Server. Dazu wird die DHCP-Implementation von ISC (Internet Systems Consortium) verwendet.

Der DHCP-Server dhcpd ist als Teil des Ports net/isc-dhcp42-server verfügbar. Dieser Port enthält den ISC DHCP-Server und die Dokumentation.

Der Server ist nicht im Basissystem von FreeBSD enthalten, daher muss der Port net/isc-dhcp42-server installiert. Lesen Sie Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*, wenn Sie weitere Informationen zur Ports-Sammlung benötigen.

Stellen Sie sicher, dass bpf(4) im Kernel kompiliert ist. Dazu fügen Sie die Zeile `device bpf` in die Kernelkonfigurationsdatei ein und erzeugen einen neuen Kernel. Die Kernelkonfiguration wird in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben.

Das Gerät `bpf` ist im `GENERIC` -Kernel bereits enthalten. Für die Nutzung von DHCP muss also kein angepasster Kernel erzeugt werden.

Anmerkung

Diejenigen, die um die Sicherheit besorgt sind, sollten wissen, dass `bpf` auch zur Ausführung von Paketsniffern erforderlich ist (obwohl diese dennoch als `root` ausgeführt werden müssen). `bpf` *muss* vorhanden sein, damit DHCP funktioniert. Sehr sicherheitsbewusste Personen, sollten `bpf` aus dem Kernel entfernen, wenn DHCP nicht verwendet wird.

Der Port net/isc-dhcp42-server installiert eine Beispiel-Konfigurationsdatei. Kopieren Sie `/usr/local/etc/dhcpd.conf.example` nach `/usr/local/etc/dhcpd.conf` und nehmen Sie die Änderungen an der neuen Datei vor.

29.6.2.1. Den DHCP-Server einrichten

`dhcpd.conf` besteht aus Festlegungen zu Subnetzen und Rechnern und lässt sich am besten an einem Beispiel erklären:

```
option domain-name "example.com";❶
option domain-name-servers 192.168.4.100;❷
option subnet-mask 255.255.255.0;❸

default-lease-time 3600;❹
max-lease-time 86400;❺
ddns-update-style none;❻

subnet 192.168.4.0 netmask 255.255.255.0 {
  range 192.168.4.129 192.168.4.254;❼
  option routers 192.168.4.1;❽
}

host mailhost {
  hardware ethernet 02:03:04:05:06:07;❾
  fixed-address mailhost.example.com;❿
}
```

❶ Diese Option beschreibt die Domäne, die den Clients als Standardsuchdomäne zugewiesen wird. Weitere Informationen finden Sie in resolv.conf(5).

❷ Diese Option legt eine, durch Kommata getrennte Liste von DNS-Servern fest, die von den Clients verwendet werden sollen.

❸ Die den Clients zugewiesene Netzmaske.

❹ Ein Client kann eine Lease einer bestimmten Dauer anfordern. Geschieht dies nicht, weist der Server eine Lease mit einer vorgegebenen Ablaufdauer (in Sekunden) zu.

❺ Die maximale Zeitdauer, für die der Server Konfigurationsinformationen vergibt. Sollte ein Client eine längere Zeitspanne anfordern, wird dennoch nur der Wert `max-lease-time` in Sekunden zugewiesen.

❻ Diese Option legt fest, ob der DHCP-Server eine DNS-Aktualisierung versuchen soll, wenn Konfigurationsdateien vergeben oder zurückgezogen werden. In der ISC-Implementation *muss* diese Option gesetzt sein.

❼ Dadurch werden die IP-Adressen festgelegt, die den Clients zugewiesen werden können. IP-Adressen zwischen diesen Grenzen sowie die einschließenden Adressen werden den Clients zugewiesen.

❽ Legt das Standard-Gateway fest, das den Clients zugewiesen wird.

❾ Die (Hardware-)MAC-Adresse eines Rechners (durch die der DHCP-Server den Client erkennt, der eine Anforderung an ihn stellt).

⓾ Einem Rechner soll immer die gleiche IP-Adresse zugewiesen werden. Beachten Sie, dass hier auch ein Rechnername gültig ist, da der DHCP-Server den Rechnernamen auflöst, bevor er die Konfigurationsinformationen zuweist.

Nachdem `dhcpd.conf` fertig konfiguriert ist, sollten Sie den DHCP-Server aktivieren, indem Sie folgende Zeilen in `/etc/rc.conf` aufnehmen:

```
dhcpd_enable="YES"
dhcpd_ifaces="dc0"
```

Dabei müssen Sie den Geräteeintrag `dc0` durch die Gerätedatei (mehrere Gerätedateien müssen durch Leerzeichen getrennt werden) ersetzen, die der DHCP-Server auf Anfragen von DHCP-Clients hin überwachen soll.

Danach können Sie den Server durch Eingabe des folgenden Befehls starten:

```
# service isc-dhcpd start
```

Künftige Änderungen an der Konfiguration des Servers erfordern, dass ein `SIGTERM`-Signal anstelle von `SIGHUP` an dhcpd gesendet wird. Es ist jedoch definitiv einfacher, den Dienst mit service(8) neu zu starten.

29.6.2.2. Dateien

- `/usr/local/sbin/dhcpd`

  dhcpd ist statisch gelinkt und befindet sich in `/usr/local/sbin` . Weitere Informationen zu dhcpd finden Sie in dhcpd(8).

- `/usr/local/etc/dhcpd.conf`

  dhcpd benötigt die Konfigurationsdatei `/usr/local/etc/dhcpd.conf` , damit der Server den Clients seine Dienste anbieten kann. Diese Datei muss alle Informationen enthalten, die an die Clients weitergegeben werden soll. Außerdem sind hier Informationen zur Konfiguration des Servers enthalten. Die mit dem Port installierte dhcpd.conf(5) beschreibt diese Konfigurationsdatei.

- `/var/db/dhcpd.leases`

  Der DHCP-Server hat eine Datenbank, die alle vergebenen Leases enthält. Diese wird als Logdatei erzeugt. Die mit dem Port installierte dhcpd.leases(5) enthält eine ausführliche Beschreibung.

- `/usr/local/sbin/dhcrelay`

  dhcrelay wird in komplexen Umgebungen verwendet, in denen ein DHCP-Server eine Anfrage eines Clients an einen DHCP-Server in einem separaten Netzwerk weiterleitet. Wenn Sie diese Funktion benötigen, müssen Sie den Port net/isc-dhcp42-relay installieren. Weitere Informationen zu diesem Thema finden Sie in dhcrelay(8).

29.7. Domain Name System (DNS)

29.7.1. Überblick

DNS ist das für die Umwandlung von Rechnernamen in IP-Adressen zuständige Protokoll. FreeBSD verwendet dazu BIND (Berkeley Internet Name Domain), die am häufigsten verwendete Implementierung von DNS). Eine Anfrage nach `www.FreeBSD.org` gibt die IP-Adresse des FreeBSD-Webservers, eine Anfrage nach `ftp.FreeBSD.org` die IP-Adresse des entsprechenden FTP-Servers zurück. Der umgekehrte Weg ist ebenso möglich, eine IP-Adresse kann also auch in ihren Rechnernamen aufgelöst werden. Um eine DNS-Abfrage durchzuführen, muss auf dem jeweiligen Rechner kein Nameserver installiert sein.

FreeBSD verwendet derzeit in der Voreinstellung BIND9 als DNS-Serversoftware. Unsere Installation bietet Ihnen eine erhöhte Sicherheit, ein neues Dateisystemlayout sowie eine automatisierte chroot(8)-Konfiguration.

Im Internet wird DNS durch ein komplexes System von autoritativen Root-Nameservern, Top Level Domain-Servern (TLD) sowie anderen kleineren Nameservern verwaltet, die individuelle Rechnerinformationen speichern und untereinander abgleichen.

Derzeit wird BIND vom Internet Systems Consortium (`https://www.isc.org/`) verwaltet.

29.7.2. Begriffsbestimmungen

Um dieses Dokument besser verstehen zu können, müssen einige DNS-spezifische Begriffe genauer definiert werden.

Begriff	Bedeutung
Forward-DNS	Rechnernamen in IP-Adressen umwandeln.
Origin (Ursprung)	Die in einer bestimmten Zonendatei beschriebene Domäne.
named, BIND	Gebräuchliche Namen für das unter FreeBSD verwendete BIND-Nameserverpaket.
Resolver	Ein Systemprozess, durch den ein Rechner Zoneninformationen von einem Nameserver anfordert.
Reverse-DNS	die Umwandlung von IP-Adressen in Rechnernamen
Root-Zone	Der Beginn der Internet-Zonenhierarchie. Alle Zonen befinden sich innerhalb der Root-Zone. Dies ist analog zu einem Dateisystem, in dem sich alle Dateien und Verzeichnisse innerhalb des Wurzelverzeichnisses befinden.
Zone	Eine individuelle Domäne, Unterdomäne, oder ein Teil von DNS, der von der gleichen Autorität verwaltet wird.

Es folgen nun einige Zonenbeispiele:

- Innerhalb der Dokumentation wird die Root-Zone in der Regel mit `.` bezeichnet.
- `org.` ist eine Top level Domain (TLD) innerhalb der Root-Zone.
- `example.org.` ist eine Zone innerhalb der `org.`-TLD.
- `1.168.192.in-addr.arpa.` ist die Zone mit allen IP-Adressen des `192.168.1.*` -IP-Bereichs.

Wie man an diesen Beispielen erkennen kann, befindet sich der spezifischere Teil eines Rechnernamens auf der linken Seite der Adresse. `example.org.` beschreibt einen Rechner also genauer als `org.`, während `org.` genauer als die Root-Zone ist. Jeder Teil des Rechnernamens hat Ähnlichkeiten mit einem Dateisystem, in dem etwa `/dev` dem Wurzelverzeichnis untergeordnet ist.

29.7.3. Gründe für die Verwendung eines Nameservers

Es gibt zwei Arten von Nameservern: Autoritative Nameserver sowie zwischenspeichernde (cachende, auch bekannt als auflösende) Nameserver.

Ein autoritativer Nameserver ist notwendig, wenn

- Sie anderen verbindliche DNS-Auskünfte erteilen wollen.
- eine Domain, beispielsweise `example.org`, registriert wird, und den zu dieser Domain gehörenden Rechnern IP-Adressen zugewiesen werden müssen.
- ein IP-Adressblock reverse-DNS-Einträge benötigt, um IP-Adressen in Rechnernamen auflösen zu können.
- ein Backup-Nameserver (auch Slaveserver genannt) oder ein zweiter Nameserver auf Anfragen antworten soll.

Ein cachender Nameserver ist notwendig, weil

- ein lokaler DNS-Server Daten zwischenspeichern und daher schneller auf Anfragen reagieren kann als ein entfernter Server.

Wird nach `www.FreeBSD.org` gesucht, leitet der Resolver diese Anfrage an den Nameserver des ISPs weiter und nimmt danach das Ergebnis der Abfrage entgegen. Existiert ein lokaler, zwischenspeichernder DNS-Server, muss dieser die Anfrage nur einmal nach außen weitergeben. Für alle weiteren Anfragen ist dies nicht mehr nötig, da diese Information nun lokal gespeichert ist.

29.7.4. Wie funktioniert DNS?

Unter FreeBSD wird der BIND-Daemon als named bezeichnet.

Datei	Beschreibung
named	Der BIND-Daemon.
rndc(8)	Das Steuerprogramm für named.
`/etc/namedb`	Das Verzeichnis, in dem sich die Zoneninformationen für BIND befinden.
`/etc/namedb/named.conf`	Die Konfigurationsdatei für named.

Je nachdem, wie eine Zone auf dem Server konfiguriert wurde, finden sich die zur Zone gehörendenden Dateien in den Unterverzeichnissen `master`, `slave`, oder `dynamic` des Verzeichnisses `/etc/namedb` . Diese Dateien enthalten die DNS-Informationen, die der Nameserver für die Beantwortung von Anfragen benötigt.

29.7.5. BIND starten

Da BIND automatisch installiert wird, ist die Konfiguration relativ einfach.

In der Voreinstellung wird ein in einer chroot(8)-Umgebung betriebener named-Server zur einfachen Namensauflösung eingerichtet, der nur im lokalen IPv4-Loopback-Adressbereich (127.0.0.1) lauscht. Um den Server manuell zu starten, verwenden Sie den folgenden Befehl:

```
# service named onestart
```

Um den named-Daemon beim Systemstart automatisch zu starten, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
named_enable="YES"
```

`/etc/namedb/named.conf` bietet zahlreiche Konfigurationsoptionen, die in diesem Dokument nicht alle beschrieben werden können. Weitere Startoptionen von named unter FreeBSD finden Sie in den `named_*` -Flags in `/etc/defaults/rc.conf` sowie in rc.conf(5). Zusätzliche Informationen finden Sie im Abschnitt 12.4, „Dienste unter FreeBSD verwalten“ des Handbuchs.

29.7.6. Konfigurationsdateien

Die Konfigurationsdateien von named finden sich unter `/etc/namedb` und müssen in der Regel an Ihre Bedürfnisse angepasst werden. Es sei denn, Sie benötigen nur einen einfachen Resolver. Ein Großteil der Konfigurationsarbeiten erfolgt dabei in diesem Verzeichnis.

29.7.6.1. `/etc/namedb/named.conf`

```
// $FreeBSD$
//
// Refer to the named.conf(5) and named(8) man pages, and the documentation
// in /usr/share/doc/bind9 for more details.
//
// If you are going to set up an authoritative server, make sure you
// understand the hairy details of how DNS works.  Even with
// simple mistakes, you can break connectivity for affected parties,
// or cause huge amounts of useless Internet traffic.
```

```
options {
// All file and path names are relative to the chroot directory,
// if any, and should be fully qualified.
    directory "/etc/namedb/working";
 pid-file "/var/run/named/pid";
 dump-file "/var/dump/named_dump.db";
 statistics-file "/var/stats/named.stats";

// If named is being used only as a local resolver, this is a safe default.
// For named to be accessible to the network, comment this option, specify
// the proper IP address, or delete this option.
 listen-on { 127.0.0.1; };

// If you have IPv6 enabled on this system, uncomment this option for
// use as a local resolver.  To give access to the network, specify
// an IPv6 address, or the keyword "any".
// listen-on-v6 { ::1; };

// These zones are already covered by the empty zones listed below.
// If you remove the related empty zones below, comment these lines out.
 disable-empty-zone "255.255.255.255.IN-ADDR.ARPA";
 disable-empty-zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ↄ
IP6.ARPA";
 disable-empty-zone "1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.ↄ
IP6.ARPA";

// If you've got a DNS server around at your upstream provider, enter
// its IP address here, and enable the line below.  This will make you
// benefit from its cache, thus reduce overall DNS traffic in the Internet.
/*
 forwarders {
  127.0.0.1;
 };
*/

// If the 'forwarders' clause is not empty the default is to 'forward first'
// which will fall back to sending a query from your local server if the name
// servers in 'forwarders' do not have the answer.  Alternatively you can
// force your name server to never initiate queries of its own by enabling the
// following line:
// forward only;

// If you wish to have forwarding configured automatically based on
// the entries in /etc/resolv.conf, uncomment the following line and
// set named_auto_forward=yes in /etc/rc.conf.  You can also enable
// named_auto_forward_only (the effect of which is described above).
// include "/etc/namedb/auto_forward.conf";
```

Um vom Cache Ihres Internetproviders zu profitieren, können hier `forwarders` aktiviert werden. Normalerweise sucht ein Nameserver das Internet rekursiv ab, bis er die gesuchte Antwort findet. Durch diese Option wird stets der Nameserver Ihres Internetproviders zuerst abgefragt, um von dessen Cache zu profitieren. Wenn es sich um einen schnellen, viel benutzten Nameserver handelt, kann dies zu einer Geschwindigkeitssteigerung führen.

Warnung

`127.0.0.1` funktioniert hier *nicht*. Ändern Sie diese Adresse in einen Nameserver des Einwahlproviders.

```
 /*
    Modern versions of BIND use a random UDP port for each outgoing
```

```
    query by default in order to dramatically reduce the possibility
    of cache poisoning.  All users are strongly encouraged to utilize
    this feature, and to configure their firewalls to accommodate it.

    AS A LAST RESORT in order to get around a restrictive firewall
    policy you can try enabling the option below.  Use of this option
    will significantly reduce your ability to withstand cache poisoning
    attacks, and should be avoided if at all possible.

    Replace NNNNN in the example with a number between 49160 and 65530.
 */
 // query-source address * port NNNNN;
};

// If you enable a local name server, don't forget to enter 127.0.0.1
// first in your /etc/resolv.conf so this server will be queried.
// Also, make sure to enable it in /etc/rc.conf.

// The traditional root hints mechanism. Use this, OR the slave zones below.
zone "." { type hint; file "/etc/namedb/named.root"; };

/* Slaving the following zones from the root name servers has some
 significant advantages:
 1. Faster local resolution for your users
 2. No spurious traffic will be sent from your network to the roots
 3. Greater resilience to any potential root server failure/DDoS

 On the other hand, this method requires more monitoring than the
 hints file to be sure that an unexpected failure mode has not
 incapacitated your server.  Name servers that are serving a lot
 of clients will benefit more from this approach than individual
 hosts.  Use with caution.

 To use this mechanism, uncomment the entries below, and comment
 the hint zone above.

 As documented at http://dns.icann.org/services/axfr/ these zones:
 "." (the root), ARPA, IN-ADDR.ARPA, IP6.ARPA, and ROOT-SERVERS.NET
 are availble for AXFR from these servers on IPv4 and IPv6:
 xfr.lax.dns.icann.org, xfr.cjr.dns.icann.org
*/
/*
zone "." {
 type slave;
 file "/etc/namedb/slave/root.slave";
 masters {
  192.5.5.241; // F.ROOT-SERVERS.NET.
 };
 notify no;
};
zone "arpa" {
 type slave;
 file "/etc/namedb/slave/arpa.slave";
 masters {
  192.5.5.241; // F.ROOT-SERVERS.NET.
 };
 notify no;
};
*/

/* Serving the following zones locally will prevent any queries
 for these zones leaving your network and going to the root
 name servers.  This has two significant advantages:
 1. Faster local resolution for your users
 2. No spurious traffic will be sent from your network to the roots
*/
```

```
// RFCs 1912 and 5735 (and BCP 32 for localhost)
zone "localhost"        { type master; file "/etc/namedb/master/localhost-forward.db"; };
zone "127.in-addr.arpa" { type master; file "/etc/namedb/master/localhost-reverse.db"; };
zone "255.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// RFC 1912-style zone for IPv6 localhost address
zone "0.ip6.arpa"       { type master; file "/etc/namedb/master/localhost-reverse.db"; };

// "This" Network (RFCs 1912 and 5735)
zone "0.in-addr.arpa"   { type master; file "/etc/namedb/master/empty.db"; };

// Private Use Networks (RFCs 1918 and 5735)
zone "10.in-addr.arpa"     { type master; file "/etc/namedb/master/empty.db"; };
zone "16.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "17.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "18.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "19.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "20.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "21.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "22.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "23.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "24.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "25.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "26.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "27.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "28.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "29.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "30.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "31.172.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "168.192.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// Link-local/APIPA (RFCs 3927 and 5735)
zone "254.169.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IETF protocol assignments (RFCs 5735 and 5736)
zone "0.0.192.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// TEST-NET-[1-3] for Documentation (RFCs 5735 and 5737)
zone "2.0.192.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "100.51.198.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "113.0.203.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Range for Documentation (RFC 3849)
zone "8.b.d.0.1.0.0.2.ip6.arpa" { type master; file "/etc/namedb/master/empty.db"; };


// Domain Names for Documentation and Testing (BCP 32)
zone "test" { type master; file "/etc/namedb/master/empty.db"; };
zone "example" { type master; file "/etc/namedb/master/empty.db"; };
zone "invalid" { type master; file "/etc/namedb/master/empty.db"; };
zone "example.com" { type master; file "/etc/namedb/master/empty.db"; };
zone "example.net" { type master; file "/etc/namedb/master/empty.db"; };
zone "example.org" { type master; file "/etc/namedb/master/empty.db"; };


// Router Benchmark Testing (RFCs 2544 and 5735)
zone "18.198.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "19.198.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IANA Reserved - Old Class E Space (RFC 5735)
zone "240.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "241.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "242.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "243.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "244.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "245.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
```

```
zone "246.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "247.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "248.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "249.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "250.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "251.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "252.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "253.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };
zone "254.in-addr.arpa" { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Unassigned Addresses (RFC 4291)
zone "1.ip6.arpa"       { type master; file "/etc/namedb/master/empty.db"; };
zone "3.ip6.arpa"       { type master; file "/etc/namedb/master/empty.db"; };
zone "4.ip6.arpa"       { type master; file "/etc/namedb/master/empty.db"; };
zone "5.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "6.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "7.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "8.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "9.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "a.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "b.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "c.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "d.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "e.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "0.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "1.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "2.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "3.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "4.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "5.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "6.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "7.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "8.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "9.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "a.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "b.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "0.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "1.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "2.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "3.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "4.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "5.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "6.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "7.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 ULA (RFC 4193)
zone "c.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "d.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Link Local (RFC 4291)
zone "8.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "9.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "a.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "b.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };

// IPv6 Deprecated Site-Local Addresses (RFC 3879)
zone "c.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "d.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "e.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };
zone "f.e.f.ip6.arpa"  { type master; file "/etc/namedb/master/empty.db"; };

// IP6.INT is Deprecated (RFC 4159)
zone "ip6.int"   { type master; file "/etc/namedb/master/empty.db"; };

// NB: Do not use the IP addresses below, they are faked, and only
// serve demonstration/documentation purposes!
```

```
//
// Example slave zone config entries.  It can be convenient to become
// a slave at least for the zone your own domain is in.  Ask
// your network administrator for the IP address of the responsible
// master name server.
//
// Do not forget to include the reverse lookup zone!
// This is named after the first bytes of the IP address, in reverse
// order, with ".IN-ADDR.ARPA" appended, or ".IP6.ARPA" for IPv6.
//
// Before starting to set up a master zone, make sure you fully
// understand how DNS and BIND work.  There are sometimes
// non-obvious pitfalls.  Setting up a slave zone is usually simpler.
//
// NB: Don't blindly enable the examples below. :-)  Use actual names
// and addresses instead.

/* An example dynamic zone
key "exampleorgkey" {
 algorithm hmac-md5;
 secret "sf87HJqjkqh8ac87a02lla==";
};
zone "example.org" {
 type master;
 allow-update {
  key "exampleorgkey";
 };
 file "/etc/named/dynamic/example.org";
};
*/

/* Example of a slave reverse zone
zone "1.168.192.in-addr.arpa" {
 type slave;
 file "/etc/namedb/slave/1.168.192.in-addr.arpa";
 masters {
  192.168.1.1;
 };
};
*/
```

Hierbei handelt es sich um Slave-Einträge für eine Reverse- und Forward-DNS-Zone, die in der Datei `named.conf` definiert sind.

Für jede neue Zone muss ein zusätzlicher Eintrag in `named.conf` erstellt werden.

Ein einfacher Eintrag für eine Zone `example.org` könnte beispielsweise so aussehen:

```
zone "example.org" {
        type master;
        file "master/example.org";
};
```

Die Option `type` legt fest, dass es sich um eine Master-Zone handelt, deren Zoneninformationen sich in der Datei `/etc/namedb/master/example.org` befinden. Diese Datei wird durch die Option `file` festgelegt.

```
zone "example.org" {
        type slave;
        file "slave/example.org";
};
```

Hier handelt es sich um einen Slaveserver, der seine Informationen vom Masterserver der betreffenden Zone bezieht und diese in der angegebenen Datei speichert. Wenn der Masterserver nicht erreichbar ist, verfügt der Slaveserver über die transferierten Zoneninformationen und kann diese an andere Rechner weitergeben.

29.7.6.2. Zonendateien

Die in der Datei `/etc/namedb/master/example.org` definierte Zonendatei für `example.org` könnte etwa so aussehen:

```
$TTL 3600        -; 1 hour default TTL
example.org.    IN      SOA      ns1.example.org. admin.example.org. (
                                2006051501      -; Serial
                                10800           -; Refresh
                                3600            -; Retry
                                604800          -; Expire
                                300             -; Negative Response TTL
                        )

; DNS Servers
                IN      NS      ns1.example.org.
                IN      NS      ns2.example.org.

; MX Records
                IN      MX 10   mx.example.org.
                IN      MX 20   mail.example.org.

                IN      A       192.168.1.1

; Machine Names
localhost       IN      A       127.0.0.1
ns1             IN      A       192.168.1.2
ns2             IN      A       192.168.1.3
mx              IN      A       192.168.1.4
mail            IN      A       192.168.1.5

; Aliases
www             IN      CNAME   example.org.
```

Beachten Sie, dass jeder mit einem „.“ endende Rechnername ein exakter Rechnername ist, während sich alles ohne einen abschließenden „.“ relativ auf den Ursprung bezieht. `ns1` steht daher beispielsweise für `ns1.example.org.`.

Eine Zonendatei hat folgenden Aufbau:

```
recordname      IN recordtype   value
```

Die am häufigsten verwendeten DNS-Einträge sind:

SOA
: Start der Zonenautorität

NS
: Ein autoritativer Nameserver

A
: Eine Rechneradresse

CNAME
: Der kanonische Name eines Alias

MX
: Mail Exchanger

PTR
: Ein (bei Reverse-DNS verwendeter) Domain Name Pointer

```
example.org. IN SOA ns1.example.org. admin.example.org. (
```

```
                        2006051501      -; Serial
                        10800           -; Refresh after 3 hours
                        3600            -; Retry after 1 hour
                        604800          -; Expire after 1 week
                        300 )           -; Negative Response TTL
```

`example.org.`
: Der Name der Domäne und damit der Ursprung dieser Zonendatei.

`ns1.example.org.`
: Der primäre/autoritative Nameserver dieser Zone.

`admin.example.org.`
: Die für diese Zone verantwortliche Person. Das Zeichen „@“ wird dabei ersetzt (<admin@example.org> wird also zu `admin.example.org`).

`2006051501`
: Die Seriennummer der Datei. Sie muss stets inkrementiert werden, wenn die Zonendatei geändert wird. Viele Administratoren bevorzugen ein `JJJJMMTTRR`-Format, um die Seriennummer festzulegen. `2006051501` steht also für den 15.05.2006, die beiden letzten Stellen für die erste Modifikation der Zonendatei an diesem Tag. Die Seriennummer ist von großer Bedeutung, da Slaveserver daran eine aktualisierte Zonendatei erkennen können.

```
    IN NS           ns1.example.org.
```

Ein NS-Eintrag. Jeder Nameserver, der für eine Zone verantwortlich ist, muss über einen solchen Eintrag verfügen.

```
    localhost       IN      A       127.0.0.1
    ns1             IN      A       192.168.1.2
    ns2             IN      A       192.168.1.3
    mx              IN      A       192.168.1.4
    mail            IN      A       192.168.1.5
```

Der Eintrag `A` bezieht sich auf Rechnernamen. `ns1.example.org` würde also zu `192.168.1.2` aufgelöst werden.

```
    IN      A       192.168.1.1
```

Diese Zeile weist die IP-Adresse `192.168.1.1` dem aktuellen Ursprung, in unserem Fall also `example.org`, zu.

```
    www             IN CNAME        @
```

Der Eintrag für den kanonischen Namen wird dazu verwendet, Aliase für einen Rechner zu vergeben. Im Beispiel ist `www` ein Alias für den „Master“-Rechner, dessen Name dem Domainnamen `example.org` (oder `192.168.1.1`) entspricht. CNAMEs können daher niemals gleichzeitig mit einem anderen Eintrag für denselben Hostname eingerichtet werden.

```
    IN MX   10      mail.example.org.
```

Der MX-Eintrag legt fest, welcher Mailserver für eintreffende Mails der Zone verantwortlich ist. `mail.example.org` ist der Rechnername des Mailservers, der eine Priorität von 10 hat.

Es können auch mehrere Mailserver mit verschiedener Priorität (10, 20, ...) vorhanden sein. Ein Mailserver, der eine Mail an `example.org` verschicken will, verwendet zuerst den MX mit der höchsten Priorität (das heißt den mit der niedrigsten Prioritätsnummer), danach den mit der nächsthöheren Priorität. Und dies solange, bis die E-Mail zugestellt werden kann.

Für (bei Reverse-DNS verwendete) `in-addr.arpa`-Zonendateien wird das gleiche Format verwendet. Der einzige Unterschied besteht in der Verwendung der Option PTR an Stelle der Optionen A und CNAME.

```
$TTL 3600

1.168.192.in-addr.arpa. IN SOA ns1.example.org. admin.example.org. (
                        2006051501      -; Serial
                        10800           -; Refresh
                        3600            -; Retry
                        604800          -; Expire
                        300 )           -; Negative Response TTL

        IN      NS      ns1.example.org.
        IN      NS      ns2.example.org.

1       IN      PTR     example.org.
2       IN      PTR     ns1.example.org.
3       IN      PTR     ns2.example.org.
4       IN      PTR     mx.example.org.
5       IN      PTR     mail.example.org.
```

Durch diese Datei werden den Rechnernamen der fiktiven Domäne IP-Adressen zugewiesen.

Beachten Sie bitte, dass es sich bei allen Namen auf der rechten Seite eines PTR-Eintrags um absolute (*fully qualified*) Domainnamen handeln muss, die mit „.“ enden.

29.7.7. Zwischenspeichernde (cachende) Nameserver

Ein cachender Nameserver hat primär die Aufgabe, rekursive Abfragen aufzulösen. Er stellt lediglich eigene Anfragen und speichert deren Ergebnisse ab.

29.7.8. DNSSEC

Domain Name System Security Extensions, oder kurz DNSSEC, ist eine Sammlung von Spezifikationen, um auflösende Nameserver von gefälschten DNS-Daten, wie beispielsweise vorgetäuschte DNS-Einträge, zu schützen. Durch die Verwendung von digitalen Signaturen kann ein Resolver die Integrität des Eintrages überprüfen. Wichtig dabei ist, dass DNSSEC nur die Integrität über digital signierte Resource Records (RRe) bereitstellt. Weder wird die Vertraulichkeit noch der Schutz vor falschen Annahmen des Endbenutzers sichergestellt. Dies bedeutet, dass es Leute nicht davor schützen kann, zu `example.net` anstatt zu `example.com` zu gelangen. Das einzige, was DNSSEC tut, ist die Authentifizierung, dass die Daten während der Übertragung nicht verändert wurden. Die Sicherheit von DNS ist ein wichtiger Schritt in der generellen Absicherung des Internets. Für weitere, tiefergehende Details über die Funktionsweise von DNSSEC sind die dazugehörigen RFCs ein guter Einstieg in die Thematik. Sehen Sie sich dazu die Liste in Abschnitt 29.7.10, „Weitere Informationsquellen“ an.

Der folgende Abschnitt wird zeigen, wie man DNSSEC für einen autoritativen DNS-Server und einen rekursiven (oder cachenden) DNS-Server, der jeweils BIND 9 verwenden, einrichten kann. Obwohl alle Versionen von BIND 9 DNSSEC unterstützen, ist es notwendig, mindestens die Version 9.6.2 zu verwenden, um in der Lage zu sein, die signierten Root-Zonen zu benutzen, wenn DNS-Abfragen geprüft werden. Der Grund dafür ist, dass früheren Versionen die Algorithmen fehlen, um die Überprüfung des Root-Zonenschlüssels zu aktivieren. Es wird dringend empfohlen, die letzte Version von BIND 9.7 oder höher einzusetzen, um von den Vorteilen der automatischen Schlüsselaktualisierung des Root-Zonenschlüssels Gebrauch zu machen, genauso wie andere Eigenschaften, um automatisch Zonen signieren zu lassen und Signaturen aktuell zu halten. Unterschiede zwischen den Versionen 9.6.2 und 9.7 und höher werden an den betreffenden Stellen angesprochen.

29.7.8.1. Rekursive DNS-Server Konfiguration

Die Aktivierung der DNSSEC-Überprüfung von Anfragen, die von einem rekursiven DNS-Server stammen, benötigt ein paar Änderungen in der `named.conf`. Bevor man jedoch diese Änderungen durchführt, muss der Root-Zonenschlüssel oder Vertrauensanker erworben werden. Momentan ist der Root-Zonenschlüssel nicht in einem Datei-

format verfügbar, dass von BIND benutzt werden kann, so dass dieser manuell in das richtige Format konvertiert werden muss. Der Schlüssel selbst kann durch Abfrage an die Root-Zone erhalten werden, indem man dazu dig verwendet. Durch Aufruf von

```
% dig +multi +noall +answer DNSKEY . > root.dnskey
```

wird der Schlüssel in `root.dnskey` abgelegt. Der Inhalt sollte so ähnlich wie folgt aussehen:

```
. 93910 IN DNSKEY 257 3 8 (
 AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQ
 bSEW0O8gcCjFFVQUTf6v58fLjwBd0YI0EzrAcQqBGCzh
 /RStIoO8g0NfnfL2MTJRkxoXbfDaUeVPQuYEhg37NZWA
 JQ9VnMVDxP/VHL496M/QZxkjf5/Efucp2gaDX6RS6CXp
 oY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3
 LQpzW5hOA2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGIcGO
 Yl7OyQdXfZ57relSQageu+ipAdTTJ25AsRTAoub8ONGc
 LmqrAmRLKBP1dfwhYB4N7knNnulqQxA+Uk1ihz0=
   ) -; key id = 19036
. 93910 IN DNSKEY 256 3 8 (
 AwEAAcaGQEA+OJmOzfzVfoYN249JId7gx+OZMbxy69Hf
 UyuGBbRN0+HuTOpBxxBCkNOL+EJB9qJxt+0FEY6ZUVjE
 g58sRr4ZQ6Iu6b1xTBKgc193zUARk4mmQ/PPGxn7Cn5V
 EGJ/1h6dNaiXuRHwR+7oWh7DnzkIJChcTqlFrXDW3tjt
) -; key id = 34525
```

Seien Sie nicht alarmiert, wenn der von Ihnen bezogene Schlüssel anders als in diesem Beispiel aussieht. Diese könnten sich in der Zwischenzeit geändert haben. In dieser Ausgabe sind eigentlich zwei Schlüssel enthalten. Der erste Schüssel mit dem Wert 257 nach dem DNSKEY-Eintrag ist derjenige, der benötigt wird. Der Wert zeigt an, dass es sich um einen sicheren Einstiegspunkt (SEP), gemein auch als Schlüsselsignierungsschlüssel (KSK) bekannt, handelt. Der zweite Schüssel mit dem Wert 256 ist der untergeordnete Schlüssel, im allgemeinen auch als Zonen-Signaturschlüssel (ZSK) bezeichnet. Weitere Schlüsselarten werden später in Abschnitt 29.7.8.2, „Autoritative DNS-Server Konfiguration“ erläutert.

Nun muss der Schlüssel verifiziert und so formatiert werden, dass BIND diesen verwenden kann. Um den Schlüssel zu verifizieren, erzeugen Sie einen DS RR-Satz. Erstellen Sie eine Datei, welche die RRs enthält, mittels

```
% dnssec-dsfromkey -f root-dnskey . > root.ds
```

Diese Einträge verwenden SHA-1 sowie SHA-256 und sollten ähnlich zu folgendem Beispiel aussehen, in dem der längere, SHA-256, benutzt wird.

```
. IN DS 19036 8 1 B256BD09DC8DD59F0E0F0D8541B8328DD986DF6E
. IN DS 19036 8 2 49AAC11D7B6F6446702E54A1607371607A1A41855200FD2CE1CDDE32F24E8FB5
```

Der SHA-256 RR kann nun mit dem Abriss in https://data.iana.org/root-anchors/root-anchors.xml verglichen werden. Um absolut sicher zu sein, dass der Schlüssel nicht zusammen mit den XML-Daten verändert wurde, kann die Datei mittels der PGP Signatur in https://data.iana.org/root-anchors/root-anchors.asc überprüft werden.

Als nächstes muss der Schlüssel in das passende Format gebracht werden. Dies unterscheidet sich ein bisschen von den BIND Versionen 9.6.2 und 9.7 und höhere. In Version 9.7 wurde die Ünterstützung zur automatischen Verfolgung und notwendigen Aktualisierung von Änderungen am Schlüssel eingebaut. Dies wird durch den Einsatz von `managed-keys` erreicht, wie in dem Beispiel unten gezeigt ist. Wenn die ältere Version eingesetzt wird, kann der Schlüssel durch eine `trusted-keys` -Anweisung eingebaut werden und die Aktualisierung muss händisch erfolgen. In BIND 9.6.2 sollte das Format folgendermassen aussehen:

```
trusted-keys {
 "." 257 3 8
 "AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQbSEW0O8gcCjF
 FVQUTf6v58fLjwBd0YI0EzrAcQqBGCzh/RStIoO8g0NfnfL2MTJRkxoX
 bfDaUeVPQuYEhg37NZWAJQ9VnMVDxP/VHL496M/QZxkjf5/Efucp2gaD
 X6RS6CXpoY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3LQpz
```

```
 W5hOA2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGIcGOYl7OyQdXfZ57relS
 Qageu+ipAdTTJ25AsRTAoub8ONGcLmqrAmRLKBP1dfwhYB4N7knNnulq
 QxA+Uk1ihz0=";
};
```

In 9.7 wird das Format stattdessen wie folgt aussehen:

```
managed-keys {
 "." initial-key 257 3 8
 "AwEAAagAIKlVZrpC6Ia7gEzahOR+9W29euxhJhVVLOyQbSEW0O8gcCjF
   FVQUTf6v58fLjwBd0YI0EzrAcQqBGCzh/RStIoO8g0NfnfL2MTJRkxoX
   bfDaUeVPQuYEhg37NZWAJQ9VnMVDxP/VHL496M/QZxkjf5/Efucp2gaD
   X6RS6CXpoY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3LQpz
   W5hOA2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGIcGOYl7OyQdXfZ57relS
   Qageu+ipAdTTJ25AsRTAoub8ONGcLmqrAmRLKBP1dfwhYB4N7knNnulq
   QxA+Uk1ihz0=";
};
```

Der Root-Schlüssel kann nun zu `named.conf` hinzugefügt werden, entweder direkt oder durch Inkludierung der Datei, die den Schlüssel enthält. Nachdem diese Schritte absolviert sind, muss BIND konfiguriert werden, um DNSSEC-Validierung für Anfragen durchzuführen, indem `named.conf` bearbeitet und die folgende `options`-Direktive hinzugefügt wird:

```
dnssec-enable yes;
dnssec-validation yes;
```

Um zu prüfen, dass es tatsächlich funktioniert, benutzen Sie dig, um eine Anfrage zu einer signierten Zone durch den Resolver, der gerade konfiguriert wurde, zu stellen. Eine erfolgreiche Antwort wird den `AD`-Eintrag aufweisen, um anzudeuten, dass die Daten authentisiert sind. Eine Anfrage wie

```
% dig @resolver +dnssec se ds
```

sollte den DS RR für die `.se`-Zone zurückgeben. In dem Abschnitt `flags:` sollte der `AD`-Eintrag gesetzt sein, wie im folgenden zu sehen ist:

```
...
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1
...
```

Der Resolver ist nun in der Lage, Anfragen ans DNS zu authentisieren.

29.7.8.2. Autoritative DNS-Server Konfiguration

Um einen autoritativen Nameserver dazu zu bringen, als eine DNSSEC-signierte Zone zu fungieren, ist ein wenig mehr Aufwand nötig. Eine Zone ist durch kryptographische Schlüssel signiert, die erzeugt werden müssen. Es ist möglich, nur einen Schlüssel dazu zu verwenden. Die vorgeschlagene Methode ist jedoch, einen starken, gut geschützten Schlüsselsignierungsschlüssel (KSK) einzusetzen, der nicht oft gewechselt wird und einen Zonensignierungsschlüssel (ZSK), der öfter ausgewechselt wird. Informationen zu vorgeschlagenen Einsatzarten können in RFC 4641: DNSSEC Operational Practices nachgelesen werden. Einsatzszenarien, welche die Root-Zone betreffen, finden Sie in DNSSEC Practice Statement for the Root Zone KSK operator sowie DNSSEC Practice Statement for the Root Zone ZSK operator. Der KSK wird dazu verwendet, um eine Kette von Autorität für die Daten, die diese Validierung benötigen, zu erschaffen und wird als solche auch als sicherer Einstiegspunkt (SEP)-Schlüssel bezeichnet. Ein Nachrichtenabriss dieses Schlüssels, der auch Delegation Signer (DS)-Eintrag genannt wird, muss in der Elternzone veröffentlicht werden, um die Vertrauenskette herzustellen. Wie dies erreicht wird, hängt von dem Besitzer der Elternzone ab. Der ZSK wird verwendet, um die Zone zu signieren und muss nur dort öffentlich zugänglich gemacht werden.

Um DNSSEC für die `example.com`-Zone, welche in den vorherigen Beispielen verwendet wird, zu aktivieren, muss als erster Schritt dnssec-keygen benutzt werden, um das KSK und ZSK Schlüsselpaar zu generieren. Dieses Schlüsselpaar kann unterschiedliche kryptographische Algorithmen nutzen. Es wird empfohlen, RSA/SHA256 für die

Schlüssel zu nutzen. Eine Schlüssellänge von 2048 Bits sollte genügen. Um den KSK für `example.com` zu generieren, geben Sie

```
% dnssec-keygen -f KSK -a RSASHA256 -b 2048 -n ZONE example.com
```

ein und um den ZSK zu erzeugen, setzen Sie folgenden Befehl ab:

```
% dnssec-keygen -a RSASHA256 -b 2048 -n ZONE example.com
```

dnssec-keygen gibt zwei Dateien aus, den öffentlichen und den privaten Schlüssel und zwar in Dateinamen, die ähnlich lauten wie `Kexample.com.+005+nnnnn.key` (öffentlich) und `Kexample.com.+005+nnnnn.private` (privat). Der `nnnnn`-Teil des Dateinamens ist eine fünfstellige Schlüsselkennung. Passen Sie genau auf, welche Kennung zu welchem Schlüssel gehört. Das ist besonders wichtig, wenn mehrere Schlüssel in einer Zone vorliegen. Es ist auch möglich, die Schlüssel umzubenennen. Für jede KSK-Datei tun Sie folgendes:

```
% mv Kexample.com.+005+nnnnn.key Kexample.com.+005+nnnnn.KSK.key
% mv Kexample.com.+005+nnnnn.private Kexample.com.+005+nnnnn.KSK.private
```

Für die ZSK-Dateien ersetzen Sie `KSK` für `ZSK` wenn nötig. Die Dateien können nun in der Zonendatei inkludiert werden, indem die `$include` Anweisung verwendet wird. Es sollte folgendermassen aussehen:

```
$include Kexample.com.+005+nnnnn.KSK.key    -; KSK
$include Kexample.com.+005+nnnnn.ZSK.key    -; ZSK
```

Schliesslich signieren Sie die Zone und weisen BIND an, die signierte Zonendatei zu benutzen. Um eine Zone zu signieren, wird dnssec-signzone eingesetzt. Der Befehl, um eine Zone `example.com` zu signieren, die in `example.com.db` liegt, sollte wie folgt aussehen:

```
% dnssec-signzone -o example.com -k Kexample.com.+005+nnnnn.KSK example.com.db Kexample.↺
com.+005+nnnnn.ZSK.key
```

Der Schlüssel, welcher mit dem Argument `-k` übergeben wird, ist der KSK und die andere Schlüsseldatei ist der ZSK, welcher für die Signatur benutzt werden soll. Es ist möglich, mehr als einen KSK und ZSK anzugeben, was das Ergebnis zur Folge hat, dass die Zone mit allen übergebenen Schlüsseln signiert wird. Dies kann dann benötigt werden, um Zonendaten mit mehr als einem Algorithmus zur Signierung zu verwenden. Die Ausgabe von dnssec-signzone ist eine Zonendatei mit allen signierten RRs. Diese Ausgabe wird in einer Datei mit der Endung `.signed` abgelegt, wie beispielsweise `example.com.db.signed`. Die DS-Einträge werden ebenfalls in eine separate Datei `dsset-example.com` geschrieben. Um diese signierte Zone zu verwenden, ändern Sie die Zonendirektive in `named.conf`, so dass `example.com.db.signed` benutzt wird. Standardmässig sind die Signaturen nur 30 Tage gültig, was bedeutet, dass die Zone in etwa 15 Tagen erneut signiert werden muss, um sicher zu stellen, dass Resolver keine Einträge mit veralteten Signaturen zwischenspeichern. Es ist möglich, ein Skript und einen cron-Job zu schreiben, um dies zu erledigen. Lesen Sie dazu die relevanten Anleitungen, um Details zu erfahren.

Stellen Sie sicher, dass die privaten Schlüssel vertraulich bleiben, genau wie mit allen anderen kryptographischen Schlüsseln auch. Wenn ein Schlüssel geändert wird, ist es gute Praxis den neuen Schlüssel in die Zone zu inkludieren, noch während der alte Schlüssel noch zum signieren eingesetzt wird, um dann auf den neuen Schlüssel zum signieren zu wechseln. Nachdem diese Schritte erfolgt sind, kann der alte Schlüssel aus der Zone entfernt werden. Wenn das nicht geschieht, können DNS-Daten für einige Zeit nicht verfügbar sein, bis der neue Schlüssel durch die DNS-Hierarchie propagiert wurde. Für weitere Informationen bezüglich Schlüsselübergabe und andere DNSSEC-Einsatzszenarien lesen Sie RFC 4641: DNSSEC Operational practices.

29.7.8.3. Automatisierung mittels BIND 9.7 oder höher

Beginnend mit der Version 9.7 von BIND wurde eine neue Eigenschaft vorgestellt, die *Smart Signing* genannt wird. Diese zielt darauf ab, das Schlüsselmanagement und den Signierungsprozess einfacher zu gestalten und zu automatisieren. Durch ablegen der Schlüssel in ein Verzeichnis, genannt *key repository* und die Verwendung der neuen Option `auto-dnssec` , ist es möglich eine dynamische Zone zu erzeugen, welche dann erneut signiert wird, wenn dazu der Bedarf besteht. Um diese Zone zu aktualisieren, benutzen Sie nsupdate mit der neuen Option `-l`. Es hat also rndc die Fähigkeit gewonnen, Zonen mit Schlüsseln im Key Repository zu verwenden, indem die Option `sign`

eingesetzt wird. Um BIND anzuweisen, diese automatische Signierung und Zonenaktualisierung für `example.com` zu nutzen, fügen Sie die folgenden Zeilen zur `named.conf` hinzu:

```
zone example.com {
 type master;
 key-directory "/etc/named/keys";
 update-policy local;
 auto-dnssec maintain;
 file "/etc/named/dynamic/example.com.zone";
};
```

Nachdem diese Änderungen durchgeführt wurden, erzeugen Sie die Schlüssel für die Zone wie in Abschnitt 29.7.8.2, „Autoritative DNS-Server Konfiguration“ beschrieben wird, legen diese Schlüssel im Key Repository ab, dass als Argument `key-directory` in der Zonenkonfiguration steht und die Zone wird automatisch signiert. Aktualisierungen für eine Zone, die auf diese Art und Weise konfiguriert wurde, muss mittels nsupdate erfolgen, dass sich um die erneute Signierung der Zone mit den hinzugefügten Daten kümmern wird. Für weitere Details, lesen Sie Abschnitt 29.7.10, „Weitere Informationsquellen“ und die Dokumentation von BIND.

29.7.9. Sicherheit

Obwohl BIND die am meisten verwendete Implementierung von DNS darstellt, werden dennoch manchmal neue Sicherheitsprobleme entdeckt.

Zwar startet FreeBSD named automatisch in einer chroot(8)-Umgebung, es gibt aber noch weitere Sicherheitsmechanismen, mit denen Sie potentielle DNS-Serviceattacken erschweren können.

Es ist daher eine gute Idee, die Sicherheitshinweise von CERT zu lesen sowie die Mailingliste FreeBSD security notifications zu abonnieren, um sich über Sicherheitsprobleme im Zusammenhang mit dem Internet und FreeBSD zu informieren.

Tipp

Tritt ein Problem auf, kann es nie schaden, die Quellen zu aktualisieren und named neu zu kompilieren.

29.7.10. Weitere Informationsquellen

Hilfeseiten zu BIND/named: rndc(8) named(8) named.conf(5) nsupdate(1) dnssec-signzone(8) dnssec-keygen(8)

- Offizielle ISC-Seite zu BIND
- Offizielles Forum zu ISC- BIND
- O'Reilly DNS and BIND 5th Edition
- Root DNSSEC
- DNSSEC Vertrauensanker-Publikation für die Root-Zone
- RFC1034 - Domain Names - Concepts and Facilities
- RFC1035 - Domain Names - Implementation and Specification
- RFC4033 - DNS Security Introduction and Requirements
- RFC4034 - Resource Records for the DNS Security Extensions
- RFC4035 - Protocol Modifications for the DNS Security Extensions

- RFC4641 - DNSSEC Operational Practices
- RFC 5011 - Automated Updates of DNS Security (DNSSEC) Trust Anchors

29.8. Der Apache HTTP-Server

Einige der weltgrößten Internetauftritte laufen unter FreeBSD. Die Mehrzahl der Webserver im Internet nutzt den Apache HTTP-Server. Die Installationspakete für den Apache sollten auf Ihrem Installationsmedium vorhanden sein. Wenn Sie den Apache noch nicht installiert haben, können Sie dies jederzeit über den Port www/apache22 nachholen.

Nachdem der Apache erfolgreich installiert wurde, muss er noch konfiguriert werden.

Anmerkung

Dieser Abschnitt beschreibt die Version 2.2.X des Apache HTTP-Servers, da diese Version unter FreeBSD am häufigsten verwendet wird. Weiterführende Informationen zu Apache 2.X finden Sie auf `http://httpd.apache.org/` .

29.8.1. Konfiguration

Der Apache HTTP-Server wird unter FreeBSD primär über die Datei `/usr/local/etc/apache22/httpd.conf` konfiguriert. Bei dieser Datei handelt es sich um eine typische UNIX®-Konfigurationsdatei, in der Kommentarzeilen mit einem #-Zeichen beginnen. Eine komplette Beschreibung aller Optionen würde den Rahmen dieses Handbuchs sprengen, daher beschreiben wir hier nur die am häufigsten verwendeten Optionen.

`ServerRoot "/usr/local"`
: Legt das Standardwurzelverzeichnis für die Apache-Installation fest. Binärdateien werden in die Verzeichnisse `bin` und `sbin` unterhalb des Serverwurzelverzeichnisses installiert, während sich Konfigurationsdateien im Verzeichnis `etc/apache` befinden.

`ServerAdmin you@your.address`
: Die E-Mail-Adresse, an die Mitteilungen über Serverprobleme geschickt werden sollen. Diese Adresse erscheint auf vom Server erzeugten Seiten, beispielsweise auf Fehlerseiten.

`ServerName www.example.com`
: `ServerName` erlaubt dem Administrator, einen anderen Rechnernamen festzulegen, den der Server an die Clients sendet, wenn sich dieser vom tatsächlichen Rechnernamen unterscheidet (sie könnten etwa `www` statt des richtigen Rechnernamens verwenden).

`DocumentRoot "/usr/local/www/apache22/data"`
: `DocumentRoot`: Das Verzeichnis, in dem die Dokumente abgelegt sind. In der Voreinstellung befinden sich alle Seiten in diesem Verzeichnis, durch symbolische Links oder Aliase lassen sich aber auch andere Orte festlegen.

Es ist empfehlenswert, eine Sicherungskopie der Apache-Konfigurationsdatei anzulegen, bevor Änderungen durchgeführt werden. Wenn die Konfiguration von Apache abgeschlossen ist, speichern Sie die Datei und überprüfen Sie die Konfiguration mit apachectl(8). Geben Sie dazu `apachectl configtest` ein. Dieser Befehl sollte `Syntax OK` zurückgeben.

29.8.2. Den Apache betreiben

Der `www/apache22` Port installiert ein rc(8) Skript, welches zum starten, stoppen und neustarten von Apache benutzt werden kann. Das Skript befindet sich in `/usr/local/etc/rc.d/` .

Um den Apache beim Systemstart zu starten, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
apache22_enable="YES"
```

Wenn Sie während des Systemstarts weitere Parameter an den Apache übergeben wollen, können Sie diese durch eine zusätzliche Zeile in `rc.conf` angeben:

```
apache22_flags=""
```

Die Konfiguration von Apache kann bei nachfolgenden Änderungen an der Konfigurationsdatei bei laufendem `httpd`, auf Fehler überprüft werden. Dies kann durch das rc(8)-Skript direkt , oder über das Dienstprogramm service(8) geschehen, indem Sie eines der folgenden Kommandos ausführen:

```
# service apache22
configtest
```

Anmerkung

Es ist wichitg zu beachten, dass `configtest` kein rc(8)-Standard ist, und somit nicht zwingend mit anderen rc(8)-Startskripten funktioniert.

Wenn der Apache keine Fehler in der Konfiguration meldet, kann der Apache `httpd` mithilfe von service(8) gestartet werden:

```
# /usr/local/etc/rc.d/apache22
start
```

```
# service apache22
start
```

Sie können den `httpd`-Dienst testen, indem Sie `http://` *`localhost`* in Ihren Browser eingeben, wobei Sie *`localhost`* durch den vollqualifizierten Domainnamen der Machine ersetzen, auf dem der `httpd` läuft. Die Standard Webseite, die angezeigt wird, ist `/usr/local/www/apache22 /data/index.html` .

29.8.3. Virtual Hosting

Der Apache unterstützt zwei Formen des *Virtual Hostings*. Die erste Möglichkeit bezeichnet man als namenbasiertes virtuelles Hosting. Dabei wird der HTTP/1.1-Header der Clients dazu verwendet, den Rechnernamen zu bestimmen. Dadurch wird es möglich, mehrere Domains unter der gleichen IP-Adresse zu betreiben.

Damit der Apache namenbasierte virtuelle Domains verwalten kann, fügen Sie die folgende Zeile in `httpd.conf` ein:

```
NameVirtualHost *
```

Wenn der Webserver `www.domain.tld` heißt und die virtuelle Domain `www.someotherdomain.tld` einrichtet werden soll, ergänzen Sie `httpd.conf` um folgende Einträge:

```
<VirtualHost *>
ServerName www.domain.tld
DocumentRoot /www/domain.tld
</VirtualHost>

<VirtualHost *>
ServerName www.someotherdomain.tld
DocumentRoot /www/someotherdomain.tld
</VirtualHost>
```

Ersetzen Sie dabei die Adressen sowie den Pfad zu den Dokumenten durch Ihre eigenen Einstellungen.

Ausführliche Informationen zum Einrichten von virtuellen Domains finden Sie in der offiziellen Apache-Dokumentation unter `http://httpd.apache.org/docs/vhosts/` .

29.8.4. Häufig verwendete Apache-Module

Es gibt viele verschiedene Apache-Module, die den Server um zusätzliche Funktionen erweitern. Die FreeBSD Ports-Sammlung ermöglicht es Ihnen, den Apache gemeinsam mit einigen der beliebtesten Zusatzmodule zu installieren.

29.8.4.1. mod_ssl

Das Modul mod_ssl verwendet die OpenSSL-Bibliothek, um, unter Nutzung der Protokolle Secure Sockets Layer (SSL v2/v3) sowie Transport Layer Security (TLS v1) starke Verschlüsselung zu ermöglichen. Durch dieses Modul können Sie ein signiertes Zertifikat von einer Zertifizierungsstelle anfordern, damit Sie einen sicheren Webserver unter FreeBSD betreiben können.

Das Modul mod_ssl wird standardmäßig kompiliert, kann aber auch noch nachträglich durch die Angabe von `-DWITH_SSL` zur Kompilierzeit aktiviert werden.

29.8.4.2. Skriptsprachen

Für die wichtigsten Skriptsprachen existieren Module, die es erlauben, Apache-Module nahezu vollständig in einer Skriptsprache zu programmieren. Derartige Module dienen oft dazu, einen Sprach-Interpreter in den Webserver einzubetten. Dadurch wird ein zusätzlicher externer Interpreter überflüssig, was die Startzeit von dynamischen Internetseiten deutlich verringert.

29.8.5. Dynamische Webseiten

In den vergangenen Jahren haben immer mehr Unternehmen das Internet als Mittel für die Steigerung ihrer Einnahmen sowie für die Erhöhung ihrer Reichweite entdeckt. Dadurch stieg auch die Nachfrage nach interaktiven Internetinhalten. Neben einigen Unternehmen, darunter Microsoft®, die dafür proprietäre Produkte entwickelt haben, hat auch die Open Source Community auf diesen Umstand reagiert und unter anderem mit Django, Ruby on Rails, mod_perl2, und mod_php Möglichkeiten zur Generierung dynamischer Internetseiten geschaffen.

29.8.5.1. Django

Bei *Django* handelt es sich um ein unter der BSD-Lizenz verfügbares Framework zur schnellen Erstellung von mächtigen Internet-Applikationen. Es beinhaltet einen objekt-relationalen Mapper (wodurch Datentypen als Phyton-Objekte entwickelt werden können) sowie eine API für den dynamischen Datenbankzugriff auf diese Objekte, ohne dass Entwickler jemals SQL-Code schreiben müssen. Zusätzlich existiert ein umfangreiches Template-System, wodurch die Programmlogik von der HTML-Präsentation getrennt werden kann.

Django setzt das Modul mod_python, den Apache-Webserver sowie eine SQL-Datenbank voraus. Der FreeBSD-Port wird alle Abhängigkeiten mit sinnvollen Optionen konfigurieren und installieren.

Beispiel 29.3. Django mit Apache2, mod_python3, und PostgreSQL installieren

```
# cd /usr/ports/www/py-django; make all install clean -DWITH_MOD_PYTHON3 -
DWITH_POSTGRESQL
```

Nachdem Django und die abhängigen Pakete installiert sind, benötigt die Anwendung ein Projektverzeichnis und die Apache-Konfiguration, um den eingebetteten Python-Interpreter zu nutzen. Dieser wird spezifische URLs der Seite aufrufen.

Beispiel 29.4. Apache-Konfiguration für Django/mod_python

Sie müssen `httpd.conf` anpassen, damit Apache Anfragen für bestimmte URLs an die Internet-Applikation übergibt:

```
<Location "/">
    SetHandler python-program
    PythonPath "['/dir/to/the/django/packages/'] + sys.path"
    PythonHandler django.core.handlers.modpython
    SetEnv DJANGO_SETTINGS_MODULE mysite.settings
    PythonAutoReload On
    PythonDebug On
</Location>
```

29.8.5.2. Ruby on Rails

Bei *Ruby on Rails* handelt es sich um ein weiteres, als Open Source verfügbares Webframework. Es bietet einen kompletten Entwicklungsstack und erlaubt es Webentwicklern, umfangreiche und mächtige Applikationen in kurzer Zeit zu programmieren. Das Framework kann über die Ports-Sammlung installiert werden.

```
# cd /usr/ports/www/rubygem-rails; make all install clean
```

29.8.5.3. mod_perl2

Die Kombination Apache/Perl vereinigt die Vorteile der Programmiersprache Perl und des Apache HTTP-Servers. Durch das Modul mod_perl2 ist es möglich, vollständig in Perl geschriebene Apache-Module zu erzeugen. Da der Perl-Interpreter in den Server eingebettet wird, müssen Sie weder einen externen Interpreter noch Perl zusätzlich aufrufen.

mod_perl2 ist über den Port `www/mod_perl2` erhältlich.

29.8.5.4. mod_php

Bei PHP>, dem „Hypertext Preprocessor“, handelt es sich um eine vielseitig verwendbare Skriptsprache, die besonders für die Web-Entwicklung geeignet ist. PHP kann in HTML eingebettet werden und ähnelt von der Syntax her Sprachen wie C, Java™ und Perl. Das Hauptanliegen von PHP ist es, Web-Entwicklern die rasche Erstellung von dynamisch erzeugten Internetseiten zu ermöglichen.

Damit Ihr System PHP5 unterstützt, müssen Sie als Erstes den Apache Webserver über den Port lang/php5 installieren.

Wenn Sie den Port lang/php5 das erste Mal installieren, werden die verfügbaren Optionen (`OPTIONS`) automatisch angezeigt. Erscheint das Konfigurationsmenü bei Ihnen nicht, so liegt dies daran, dass Sie den Port lang/php5 schon einmal auf Ihrem System installiert hatten. Es ist aber jederzeit möglich, dieses Menü aus dem Ports-Verzeichnis heraus über folgenden Befehl erneut aufzurufen:

```
# make config
```

In diesem Konfigurationsmenü müssen Sie die Option `APACHE` auswählen, damit mod_php5 als ein vom Apache-Webserver ladbares Modul gebaut wird.

Anmerkung

Viele Seiten verwenden nach wie vor (beispielsweise wegen der benötigten Kompatibilität zu bereits vorhandenen Web-Applikationen) PHP4. Ist dies bei Ihnen der Fall, so müssen Sie statt

mod_php5 mod_php4 über den Port lang/php4 installieren. Der Port lang/php4 unterstützt viele der Konfigurations- und Laufzeitoptionen von lang/php5.

Dieser Port installiert und konfiguriert die Module, die für die Unterstützung von dynamischen PHP-Anwendungen benötigt werden. Stellen Sie danach sicher, dass Ihre `/usr/local/etc/apache22/httpd.conf` die folgenden Abschnitte enthält:

```
LoadModule php5_module        libexec/apache/libphp5.so
```

```
AddModule mod_php5.c
    <IfModule mod_php5.c>
        DirectoryIndex index.php index.html
    </IfModule>
    <IfModule mod_php5.c>
        AddType application/x-httpd-php .php
        AddType application/x-httpd-php-source .phps
    </IfModule>
```

Nachdem dies erledigt ist, rufen Sie `apachectl` auf, um das PHP-Modul zu laden:

```
# apachectl graceful
```

Bei künftigen Upgrades von PHP wird `make config` nicht mehr benötigt, da die von Ihnen ursprünglich ausgewählten Optionen (`OPTIONS`) vom FreeBSD-Ports-Framework automatisch gespeichert werden.

Die PHP-Unterstützung von FreeBSD ist stark modular aufgebaut, daher verfügt eine Basisinstallation nur über wenige Funktionen. Eine Erweiterung um zusätzliche Funktionen ist allerdings sehr einfach über den Port lang/php5-extensions möglich. Der Port bietet Ihnen ein Auswahlmenü, über das Sie verschiedene PHP-Erweiterungen installieren können. Alternativ können Sie einzelne Erweiterungen aber weiterhin direkt über den jeweiligen Port installieren.

Um beispielsweise die Unterstützung des Datenbankservers MySQL in PHP5 zu aktivieren, installieren Sie den Port `databases/php5-mysql`.

Nachdem Sie eine Erweiterung installiert haben, müssen Sie den Apache-Server neu starten, damit die Erweiterung auch erkannt wird:

```
# apachectl graceful
```

Ab nun wird MySQL von PHP unterstützt.

29.9. File Transfer Protocol (FTP)

Das File Transfer Protocol (FTP) ermöglicht auf einfache Art und Weise den Dateiaustausch mit einem FTP-Server. Der FTP-Server ftpd ist bei FreeBSD bereits im Basisystem enthalten. Daher sind Konfiguration und Betrieb eines FTP-Servers unter FreeBSD relativ einfach.

29.9.1. Konfiguration

Der wichtigste Punkt ist hier die Entscheidung darüber, welche Benutzer auf Ihren FTP-Server zugreifen dürfen. Ein FreeBSD-System verfügt über diverse Systembenutzerkonten, um einzelnen Daemonen den Zugriff auf das System zu ermöglichen. Anonyme Benutzer sollten sich allerdings nicht über diese Benutzerkonten anmelden dürfen. Die Datei `/etc/ftpusers` enthält alle Benutzer, die vom FTP-Zugriff ausgeschlossen sind. In der Voreinstellung gilt dies auch die gerade erwähnten Systembenutzerkonten. Sie können über diese Datei weitere Benutzer vom FTP-Zugriff ausschließen.

In einigen Fällen kann es wünschenswert sein, den Zugang für manche Benutzer einzuschränken, ohne dabei FTP komplett zu verbieten. Dazu passen Sie `/etc/ftpchroot` entsprechend an. Diese Datei enthält Benutzer und Gruppen sowie die für sie geltenden FTP-Einschränkungen und wird in ftpchroot(5) ausführlich beschrieben.

Um anonymen FTP-Zugriff auf dem Server zu aktivieren, muss ein Benutzer `ftp` auf dem FreeBSD-System angelegt werden. Danach können sich Benutzer mit dem Benutzernamen `ftp` oder `anonymous` am FTP-Server anmelden. Das Passwort ist dabei beliebig (allerdings wird dazu in der Regel eine E-Mail-Adresse verwendet). Meldet sich ein anonymer Benutzer an, aktiviert der FTP-Server chroot(2), um den Zugriff auf das Heimatverzeichnis des Benutzers `ftp` zu beschränken.

Es gibt zwei Textdateien, deren Inhalt Sie bei der Anmeldung an Ihrem FTP-Server anzeigen lassen können. Der Inhalt von `/etc/ftpwelcome` wird angezeigt, bevor der Login-Prompt erscheint. Nach einer erfolgreichen Anmeldung wird der Inhalt von `/etc/ftpmotd` angezeigt. Beachten Sie aber, dass es dabei um einen Pfad relativ zur Umgebung des anzumeldenden Benutzers handelt. Bei einer anonymen Anmeldung würde also die Datei `~ftp/etc/ftpmotd` angezeigt.

Nachdem Sie den FTP-Server konfiguriert haben, müssen Sie Ihn in `/etc/inetd.conf` aktivieren. Dazu müssen Sie lediglich das Kommentarsymbol „#“ am Beginn der bereits vorhandenen ftpd-Zeile entfernen:

```
ftp stream tcp nowait root /usr/libexec/ftpd ftpd -l
```

Nachdem Sie diese Änderung durchgeführt haben, müssen Sie, wie in Beispiel 29.1, „Die inetd-Konfiguration neu einlesen“ beschrieben, die inetd-Konfiguration neu einlesen. Lesen Sie bitte Abschnitt Abschnitt 29.2.2, „Einstellungen“ des Handbuchs für weitere Informationen zur Aktivierung von inetd auf dem System.

Alternativ können Sie auch nur den ftpd-Server starten. In diesem Fall ist es ausreichend, die entsprechende Variable in der Datei `/etc/rc.conf` zu setzen:

```
ftpd_enable="YES"
```

Nachdem Sie diese Variable gesetzt haben, wird künftig beim Systemstart nur der FTP-Server gestartet. Alternativ können Sie den Server auch manuell starten, indem Sie als Benutzer `root` den folgenden Befehl ausführen:

```
# service ftpd start
```

Danach können Sie sich am FTP-Server anmelden:

```
% ftp localhost
```

29.9.2. Wartung

Der ftpd-Daemon verwendet syslog(3), um Protokolldateien zu erstellen. In der Voreinstellung werden alle FTP betreffenden Nachrichten in die Datei `/var/log/xferlog` geschrieben. Dies lässt sich aber durch das Einfügen der folgenden Zeile in `/etc/syslog.conf` ändern:

```
ftp.info      /var/log/xferlog
```

Beachten Sie, dass mit dem Betrieb eines anonymen FTP-Servers verschiedene Sicherheitsrisiken verbunden sind. Problematisch ist hier vor allem die Erlaubnis zum anonymen Upload von Dateien. Dadurch könnte der Server zur Verbreitung von illegaler oder nicht lizensierter Software oder noch Schlimmeren missbraucht werden. Wenn anonyme FTP-Uploads dennoch erforderlich sind, sollten Sie die Zugriffsrechte so setzen, dass solche Dateien erst nach Zustimmung eines Administrators von anderen Benutzern heruntergeladen werden können.

29.10. Datei- und Druckserver für Microsoft® Windows®-Clients (Samba)

Samba ist ein beliebtes Open Source-Softwarepaket, das es Ihnen ermöglicht, einen Datei- und Druckserver für Microsoft® Windows®-Clients einzurichten. Clients können sich dadurch mit einem FreeBSD-System verbinden und dessen Speicherplatz oder dessen Drucker verwenden. Dies genauso, als wenn es sich um lokale Drucker oder Festplatten handeln würde.

Samba sollte als Softwarepaket auf den FreeBSD-Installationsmedien vorhanden sein. Wenn Samba noch nicht installiert ist, können Sie dies über den Port oder das Paket net/samba34 nachholen.

29.10.1. Konfiguration

Die Standardkonfigurationsdatei von Samba heißt `/usr/local/share/examples/samba34/smb.conf.default` . Diese Datei muss nach `/usr/local/etc/smb.conf` kopiert und angepasst werden, bevor Samba verwendet werden kann.

Die Datei `smb.conf` enthält Laufzeitinformationen für Samba, beispielsweise Druckerdefinitionen oder *file system shares*, also Bereiche des Dateisystems, die mit Windows®-Clients geteilt werden sollen. Die Konfiguration der Datei `smb.conf` erfolgt webbasiert über das im Samba-Paket enthaltene Programm swat.

29.10.1.1. Das Samba Web Administration Tool (SWAT) verwenden

Das *Samba Web Administration Tool* (SWAT) wird als Daemon von inetd aktiviert. Daher müssen Sie inetd, wie in Abschnitt 29.2, „Der inetd „Super-Server““ beschrieben, aktivieren und die folgende Zeile in `/etc/inetd.conf` entfernen, bevor Sie swat zur Konfiguration von Samba verwenden können:

```
swat    stream  tcp     nowait/400      root    /usr/local/sbin/swat    swat
```

Wie bereits in Beispiel 29.1, „Die inetd-Konfiguration neu einlesen“ beschrieben, müssen Sie die inetd-Konfiguration neu einlesen, nachdem Sie diese Änderung durchgeführt haben.

Nachdem swat in der Datei `inetd.conf` aktiviert wurde, rufen Sie im Internetbrowser die Adresse `http://localhost:901` auf. Bei der ersten Anmeldung muss das `root`-Benutzerkonto verwendet werden.

Nachdem erfolgreicher Anmeldung an der Hauptkonfigurationseite von Samba steht die Systemdokumentation zur Verfügung, und durch einen Klick auf die Globals-Karteikarte kann mit der Konfiguration begonnen werden. Die Einstellungen, die Sie hier vornehmen können, entsprechen denen des Abschnitts `[global]` von `/usr/local/etc/smb.conf` .

29.10.1.2. Globale Einstellungen

Unabhängig davon, ob swat verwendet, oder `/usr/local/etc/smb.conf` direkt editiert wird, sollten zuerst folgende Richtlinien angepasst werden:

`workgroup`
: Der NT-Domänenname oder der Arbeitsgruppenname der Rechner, die auf den Server Zugriff haben sollen.

`netbios name`
: Legt den NetBIOS-Namen fest, unter dem der Samba-Server bekannt ist. In der Regel handelt es sich dabei um den ersten Teil des DNS-Namens des Servers.

`server string`
: Legt die Beschreibung fest, die angezeigt werden soll, wenn mit `net view` oder über andere Netzwerkprogramme Informationen über den Server angefordert werden.

29.10.1.3. Samba absichern

Zwei der wichtigsten Einstellungen in `/usr/local/etc/smb.conf` betreffen das zu verwendende Sicherheitsmodell sowie das Backend-Passwortformat für die Benutzer der Samba-Clients. Folgende Optionen sind dafür verantwortlich:

`security`
: Die häufigsten Optionen sind `security = share` und `security = user`. Wenn die Clients Benutzernamen verwenden, die den Benutzernamen auf dem FreeBSD-Rechner entsprechen, dann sollten Sie die Einstellung *user level* verwenden. Dies ist die Standardeinstellung. Allerdings ist es dazu erforderlich, dass sich die Clients auf dem Rechner anmelden, bevor sie auf gemeinsame Ressourcen zugreifen können.

  In der Einstellung *share level* müssen sich Clients nicht unter Verwendung eines gültigen Logins auf dem Rechner anmelden, bevor sie auf gemeinsame Ressourcen zugreifen können. In früheren Samba-Versionen war dies die Standardeinstellung.

`passdb backend`
: Samba erlaubt verschiedene Backend-Authentifizierungsmodelle. Clients können sich durch LDAP, NIS+, eine SQL-Datenbank oder eine Passwortdatei authentifizieren. In der Voreinstellung wird `smbpasswd` verwendet. Diese Methode wird im folgenden Abschnitt näher beschrieben.

Wenn Sie `smbpasswd` verwenden, müssen Sie die Datei `/usr/local/etc/samba/smbpasswd` erzeugen, damit Samba in der Lage ist, Clients zu authentifizieren. Um den Zugriff auf UNIX®-Benutzerkonten von einem Windows®-Client aus zu ermöglichen, verwenden Sie den folgenden Befehl:

```
# smbpasswd -a username
```

Anmerkung

Als Backend wird inzwischen `tdbsam` empfohlen. Mit dem folgenden Befehl legen Sie neue Benutzerkonten an:

```
# pdbedit -a -u username
```

Ausführliche Informationen zur Konfiguration von Samba finden Sie im Official Samba HOWTO. Mit den hier skizzierten Grundlagen, sollten Sie in der Lage sein, Samba zu starten. Zusätzlich zu den Informationen hier, sollte weitere Dokumentation hinzugezogen werden.

29.10.2. Samba starten

Der Port net/samba34 legt ein neues Startskript an, mit dem Samba gesteuert (also etwa gestartet oder beendet) werden kann. Um dieses Skript zu aktivieren, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
samba_enable="YES"
```

Alternativ können Sie auch die folgenden beiden Einträge verwenden:

```
nmbd_enable="YES"
```

```
smbd_enable="YES"
```

Anmerkung

Durch diese Einträge wird Samba beim Systemstart automatisch aktiviert.

Danach können Sie Samba jederzeit durch folgenden Befehl starten:

```
# service samba start
Starting SAMBA: removing stale tdbs :
```

```
Starting nmbd.
Starting smbd.
```

Weitere Informationen zu den rc-Startskripten finden Sie im Abschnitt 12.4, „Dienste unter FreeBSD verwalten“ des Handbuchs.

Samba verwendet drei Daemonen. Beachten Sie, dass sowohl nmbd als auch smbd durch das Skript `samba` gestartet werden. Wurde *winbind name resolution services* in `smb.conf` aktiviert, wird zusätzlich der winbindd-Daemon gestartet.

Samba kann jederzeit durch folgenden Befehl beendet werden:

```
# service samba stop
```

Samba ist ein komplexes Softwarepaket mit umfassenden Funktionen, die eine weitreichende Integration von Microsoft® Windows®-Netzwerken ermöglichen. Für eine Beschreibung dieser Zusatzfunktionen sollten Sie sich auf `http://www.samba.org` umsehen.

29.11. Die Uhrzeit mit NTP synchronisieren

Da die interne Uhrzeit eines Computers nie ganz exakt ist, wurde mit dem *Network Time Protocol* (NTP) eine Möglichkeit geschaffen, die exakte Uhrzeit zu ermitteln und festzulegen.

Viele Internetdienste sind von einer exakten Uhrzeit abhängig. Ein Webserver könnte beispielsweise die Anforderung erhalten, eine Datei zu versenden, wenn sich diese in einer bestimmten Zeitspanne geändert hat. In einem lokalen Netzwerk ist es unbedingt notwendig, dass Rechner, die Dateien von einem gemeinsamen Dateiserver beziehen, ihre Uhrzeit synchronisieren, damit die Zeitstempel der Dateien konstistent bleiben. Dienste wie cron(8) führen Befehle zu einem bestimmten Zeitpunkt aus. Ist die Uhrzeit nicht korrekt, kann dies zu Problemen führen.

FreeBSD verwendet den ntpd(8)- NTP-Server, um die genaue Uhrzeit von anderen NTP-Servern abzufragen, die eigene Systemzeit zu setzen, oder um diese anderen Rechnern anzubieten.

29.11.1. Einen passenden NTP-Server auswählen

Um die Uhrzeit zu synchronisieren, müssen ein oder mehrere NTP-Server definiert werden. Der Netzwerkadministrator oder Internetprovider hat vielleicht schon einen NTP-Server eingerichtet. Lesen Sie deren Dokumentation, um dies zu überprüfen. Es gibt im Internet eine Liste mit frei zugänglichen NTP-Servern, aus der man einen in der Nähe gelegenen Server auswählen kann. Beachten Sie aber auf jeden Fall die Nutzungsbedingungen des entsprechenden Servers, und fragen Sie um Erlaubnis, wenn dies nötig ist.

Die Auswahl von mehreren NTP-Servern kann sinnvoll sein, wenn ein Server ausfällt oder falsche Zeiten liefert. ntpd(8) verwendet die Antworten anderer Server, um zuverlässige Server zu bestimmen, die dann bevorzugt abgefragt werden.

29.11.2. NTP unter FreeBSD einrichten

29.11.2.1. NTP aktivieren

Wenn Sie die Uhrzeit nur beim Systemstart synchronisieren wollen, benutzen Sie ntpdate(8). Für Desktoprechner, die regelmäßig neu gestartet werden und keine ständige Synchronisation benötigen, ist dies akzeptabel. In allen anderen Fällen sollten Sie jedoch ntpd(8) verwenden.

Die Ausführung von ntpdate(8) während des Systemstarts ist aber auch für Rechner, die ntpd(8) verwenden, sinnvoll. ntpd(8) passt die Systemzeit nur bei größeren Abweichungen an, während ntpdate(8) die Zeit immer synchronisiert, egal wie groß die Differenz zwischen Systemzeit und korrekter Zeit ist.

Um ntpdate(8) beim Systemstart zu aktivieren, fügen Sie den Eintrag `ntpdate_enable="YES"` in `/etc/rc.conf` ein. Außerdem müssen alle Server, mit denen Sie sich synchronisieren wollen, sowie alle an ntpdate(8) zu übergebenden Optionen in `ntpdate_flags` angeben werden.

29.11.2.2. NTP einrichten

Die Konfiguration von NTP erfolgt über die Datei `/etc/ntp.conf`, und wird in der Hilfeseite ntp.conf(5) beschrieben. Dazu ein einfaches Beispiel:

```
server ntplocal.example.com prefer
server timeserver.example.org
server ntp2a.example.net

driftfile /var/db/ntp.drift
```

Die Option `server` legt die zu verwendenden Server fest, wobei jeder Server in einer eigenen Zeile steht. Wenn ein Server mit der Option `prefer` versehen ist, wie dies hier bei `ntplocal.example.com` der Fall ist, wird dieser Server bevorzugt verwendet. Eine Antwort von einem bevorzugten Server wird nur dann verworfen, wenn sie signifikant von denen anderer Server abweicht, ansonsten wird sie ohne Abfrage weiterer Server verwendet. Die Option `prefer` wird gewöhnlich nur für sehr zuverlässige und genaue Server verwendet, die über eine spezielle Hardware zur Zeitüberwachung verfügen.

Die Option `driftfile` legt fest, in welcher Datei die Abweichungen der Systemuhr protokolliert werden. ntpd(8) verwendet diese Datei, um die Systemzeit automatisch anzupassen, selbst wenn kurzzeitig kein NTP-Server zur Synchronisation verfügbar ist.

Weiterhin legt die Option `driftfile` fest, wo Informationen über frühere Antworten des verwendeten NTP-Servers gespeichert werden sollen. Diese Datei enthält NTP-interne Informationen, sie sollte daher von anderen Prozessen nicht verändert werden.

29.11.2.3. Den Zugang zu Ihrem NTP-Server beschränken

In der Voreinstellung ist der NTP-Server für alle Rechner im Internet erreichbar. Die Option `restrict` in `/etc/ntp.conf` steuert, welche Rechner auf den Server zugreifen können.

Wenn Sie alle Rechner vom Zugriff auf den NTP-Server ausschließen wollen, fügen Sie folgende Zeile in `/etc/ntp.conf` ein:

```
restrict default ignore
```

Anmerkung

Dieser Eintrag verhindert auch den Zugriff des Servers auf alle angegebenen Server in der lokalen Konfiguration. Besteht die Notwendigkeit, den NTP-Server mit einem externen NTP-Server zu synchronisieren, muss dieser Server dezidiert zugelassen werden. Weitere Informationen finden Sie in ntp.conf(5).

Wenn Sie nur Rechnern innerhalb des Netzwerks die Synchronisation mit dem NTP-Server erlauben, gleichzeitig aber verhindern wollen, dass diese den NTP-Server konfigurieren oder als Server für andere Rechner dienen können, fügen Sie folgende Zeile ein:

```
restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap
```

Bei `192.168.1.0` handelt es sich um einen Rechner innerhalb des Netzwerks. `255.255.255.0` ist die Netzmaske des Netzwerks.

`/etc/ntp.conf` kann verschiedene `restrict` -Optionen enthalten. Weitere Details finden Sie im Abschnitt `Access Control Support` von ntp.conf(5).

29.11.3. Den NTP-Server starten

Damit der NTP-Server beim Systemstart automatisch gestartet wird, fügen Sie den Eintrag `ntpd_enable="YES"` in `/etc/rc.conf` ein. Um weitere Argumente an ntpd(8) zu übergeben , passen Sie die Option `ntpd_flags` in der Datei `/etc/rc.conf` entsprechend an.

Um den NTP-Server ohne einen Systemneustart zu starten, rufen Sie `ntpd` mit den unter `ntpd_flags` in `/etc/rc.conf` festgelegten Parametern auf. Hierzu ein Beispiel:

```
# ntpd -p /var/run/ntpd.pid
```

29.11.4. ntpd mit einer Einwahlverbindung verwenden

ntpd(8) benötigt keine ständige Internetverbindung. Wenn Sie sich ins Internet einwählen, ist es sinnvoll, zu verhindern, dass NTP-Verkehr eine Verbindung aufbauen oder aufrechterhalten kann. Benutzer von PPP können dies in den `filter`-Direktiven von `/etc/ppp/ppp.conf` festlegen. Ein Beispiel:

```
set filter dial 0 deny udp src eq 123
# Prevent NTP traffic from initiating dial out
set filter dial 1 permit 0 0
set filter alive 0 deny udp src eq 123
# Prevent incoming NTP traffic from keeping the connection open
set filter alive 1 deny udp dst eq 123
# Prevent outgoing NTP traffic from keeping the connection open
set filter alive 2 permit 0/0 0/0
```

Weitere Informationen finden Sie im Abschnitt `PACKET FILTERING` von ppp(8) sowie in den Beispielen unter `/usr/share/examples/ppp/`.

Anmerkung

Einige Internetprovider blockieren Ports mit niedrigen Nummern. In solchen Fällen funktioniert NTP leider nicht, da Antworten eines NTP-Servers den Rechner nicht erreichen werden.

29.11.5. Weitere Informationen

Weiterführende Dokumentation (im HTML-Format) zum NTP-Server finden Sie unter `/usr/share/doc/ntp/` .

29.12. Protokollierung von anderen Hosts mit `syslogd`

Die Interaktion mit Systemprotokollen ist ein wichtiger Aspekt, sowohl was Sicherheit als auch Systemadministration anbelangt. Überwachen der Protokolldateien von mehreren Hosts kann sehr unhandlich werden, wenn diese Hosts über mittlere oder grosse Netze verteilt sind oder wenn sie Teile von unterschiedlichen Netzwerken sind. In diesen Fällen macht die Konfiguration der Protokollierung von anderen Hosts diesen Prozess wesentlich komfortabler.

Die zentralisierte Protokollierung auf einen bestimmten Protokollierungshost kann manche der administrativen Belastungen der Protokolldateiadministration reduzieren. Protokolldateiaggregation, -zusammenführung und -rotation kann an einer zentralen Stelle mit den FreeBSD-eigenen Werkzeugen wie syslogd(8) und newsyslog(8) konfiguriert werden. In der folgenden Beispielkonfiguration sammelt Host `A`, genannt `logserv.example.com`, Protokollinformationen für das lokale Netzwerk. Host `B`, genannt `logclient.example.com` wird seine Protokollinformationen an den Server weiterleiten. In realen Konfigurationen benötigen beide Hosts passende Vorwärts- und Umkehr-Einträge im DNS oder in `/etc/hosts` . Andernfalls werden die Daten vom Server abgelehnt.

29.12.1. Konfiguration des Protokollierungs-Servers

Protokollierungs-Server sind Maschinen, die konfiguriert sind, Protokollinformationen von anderen Hosts zu akzeptieren. In den meisten Fällen wird dies zur Vereinfachung der Konfiguration eingesetzt, in anderen Fällen ist es einfach nur ein Schritt in eine bessere Verwaltung. Was auch immer die Gründe sind, ein paar Anforderungen müssen vorher erfüllt sein.

Ein richtig konfigurierter Protokollierungs-Server muss minimal die folgenden Anforderungen erfüllen:

- Das Regelwerk der Firewall muss UDP auf Port 514 sowohl auf Client- als auch auf Serverseite erlauben;
- `syslogd` wurde so konfiguriert, dass es Nachrichten von anderen Clientrechnern akzeptiert;
- Der `syslogd`-Server und alle Clientrechner müssen gültige Einträge für sowohl Vorwärts- als auch Umkehr-DNS besitzen, oder in `/etc/hosts` korrekt eingetragen sein.

Um den Protokollierungs-Server zu konfigurieren, muss der Client in `/etc/syslog.conf` eingetragen sein und der Verbindungsweg der Protokollierung muss spezifiziert sein:

```
+logclient.example.com
*.*     /var/log/logclient.log
```

Anmerkung

Weitere Informationen zu den verschiedenen unterstützten und verfügbaren *Verbindungswegen* finden Sie in syslog.conf(5).

Einmal hinzugefügt, werden alle Nachrichten über den `Verbindungsweg` in die zuvor angegebene Datei, `/var/log/logclient.log` protokolliert.

Der Server benötigt ausserdem die folgenden Zeilen in der `/etc/rc.conf` :

```
syslogd_enable="YES"
syslogd_flags="-a logclient.example.com -v -v"
```

Die erste Option aktiviert den `syslogd`-Dienst während des Systemstarts und die zweite Option erlaubt es, Daten von dem spezifizierten Client auf diesem Server zu akzeptieren. Die Verwendung von `-v -v` im letzten Teil erhöht die Anzahl von Protokollnachrichten. Dies ist sehr hilfreich für die Feineinstellung der Verbindungspfade, da Administratoren auf diese Weise erkennen, welche Arten von Nachrichten unter welchen Einstellungen protokolliert werden.

Mehrere `-a`-Optionen können angegeben werden, um die Protokollierung von mehreren Clients zu erlauben. IP-Adressen und ganze Netzblöcke können ebenfalls spezifiziert werden. Eine vollständige Liste der Optionen finden Sie in syslog(3).

Zum Schluss muss noch die Protokolldatei erstellt werden. Auf welche Weise dies geschieht ist nicht wichtig, aber in den meisten Fällen funktioniert touch(1) grossartig, wie hier dargestellt:

```
# touch /var/log/logclient.log
```

Zu diesem Zeitpunkt sollte der `syslogd`-Dienst neu gestartet und überprüft werden:

```
# service syslogd restart
# pgrep syslog
```

Wenn eine PID zurückgegeben wird, wurde der Server erfolgreich neu gestartet und die Clientkonfiguration kann beginnen. Wenn der Server nicht neu gestartet wurde, suchen Sie im `/var/log/messages` -Protokoll nach eventuellen Fehlermeldungen.

29.12.2. Konfiguration des Protokollierungs-Clients

Ein Protokollierungs-Client ist eine Maschine, die Protokollinformationen an einen Protokollierungs-Server sendet, zusätzlich zu ihren lokalen Kopien.

Ähnlich wie Protokollierungs-Server müssen Clients auch ein paar minimale Anforderungen erfüllen:

- syslogd(8) muss so konfiguriert sein, dass es Nachrichten eines bestimmten Typs an einen Protokollierungs-Server schickt, welcher diese akzeptieren muss;
- Die Firewall muss UDP-Pakete durch Port 514 erlauben;
- Sowohl Vorwärts- als auch Umkehr-DNS muss konfiguriert sein oder es müssen passende Einträge in `/etc/hosts` vorhanden sein.

Die Clientkonfiguration ist ein bisschen entspannter, verglichen mit der des Servers. Der Clientrechner muss ebenfalls die folgenden Einträge in der `/etc/rc.conf` besitzen:

```
syslogd_enable="YES"
syslogd_flags="-s -v -v"
```

Wie zuvor aktivieren diese Einträge den `syslogd`-Dienst während des Systemstarts und erhöhen die Anzahl der Protokollnachrichten. Die Option `-s` verhindert, dass dieser Client Protokolle von anderen Hosts akzeptiert.

Verbindungspfade beschreiben den Systemteil, für den eine Nachricht generiert wird. Beispielsweise sind ftp und ipfw beides Verbindungspfade. Wenn Protokollnachrichten für diese beiden Dienste generiert werden, sind diese beiden Werkzeuge normalerweise in jeder Protokollnachricht enthalten. Verbindungspfade sind mit einer Priorität oder Stufe verbunden, die dazu verwendet wird, zu markieren, wie wichtig eine Nachricht im Protokoll ist. Die Häftigste ist `warning` und `info`. Eine vollständig Liste der verfügbaren Verbindungspfade und Prioritäten finden Sie in syslog(3).

Der Protokollierungs-Server muss in der `/etc/syslog.conf` des Clients eingetragen sein. In diesem Beispiel wird das `@`-Symbol benutzt, um Protokolldaten an einen anderen Server zu senden. Der Eintrag sieht wie folgt aus:

```
*.*             @logserv.example.com
```

Einmal hinzugefügt, muss `syslogd` neu gestartet werden, damit diese Änderungen wirksam werden:

```
# service syslogd restart
```

Um zu testen, ob Protokollnachrichten über das Netzwerk gesendet werden, kann logger(1) auf dem Client benutzt werden, um eine Nachricht an `syslogd` zu schicken:

```
# logger "Test message from logclient "
```

Diese Nachricht sollte jetzt sowohl in `/var/log/messages` auf dem Client, als auch in `/var/log/logclient.log` auf dem Server vorhanden sein.

29.12.3. Fehlerbehebung beim Protokollierungs-Server

In bestimmten Fällen ist die Fehlerbehebung notwendig, wenn Nachrichten nicht auf dem Protokollierungs-Server empfangen werden. Es gibt mehrere Gründe dafür, jedoch treten am häufigsten Probleme bei der Netzwerkverbindung und beim DNS auf. Um diese Fälle zu überprüfen, stellen Sie sicher, dass beide Hosts in der Lage sind, sich gegenseitig über den Hostnamen zu erreichen, der in `/etc/rc.conf` angegeben ist. Wenn das funktioniert, ist möglicherweise eine Änderung der `syslogd_flags`-Option in `/etc/rc.conf` notwendig.

Im folgenden Beispiel ist `/var/log/logclient.log` leer und die `/var/log/messages` -Dateien enthalten keine Gründe für den Fehler. Um die Fehlerausgabe zu erhöhen, ändern Sie die `syslogd_flags`-Option so, dass diese wie in dem folgenden Beispiel aussieht und initiieren Sie dann einen Neustart:

```
syslogd_flags="-d -a logclien.example.com -v -v"
```

```
# service syslogd restart
```

Fehlerausgabedaten ähnlich der Folgenden werden sofort nach dem Neustart auf dem Bildschirm erscheinen:

```
logmsg: pri 56, flags 4, from logserv.example.com, msg syslogd: restart
syslogd: restarted
logmsg: pri 6, flags 4, from logserv.example.com, msg syslogd: kernel boot file is /boot/
kernel/kernel
Logging to FILE /var/log/messages
syslogd: kernel boot file is /boot/kernel/kernel
cvthname(192.168.1.10)
validate: dgram from IP 192.168.1.10, port 514, name logclient.example.com;
rejected in rule 0 due to name mismatch.
```

Es scheint klar zu sein, dass die Nachrichten aufgrund eines fehlerhaften Namens abgewiesen werden. Nach genauer Untersuchung der Konfiguration, kommt ein Tippfehler in der folgenden Zeile der `/etc/rc.conf` als Fehler in Betracht:

```
syslogd_flags="-d -a logclien.example.com -v -v"
```

Die Zeile sollte `logclient` und nicht `logclien` enthalten. Nachdem die entsprechenden Veränderungen gemacht wurden, ist ein Neustart fällig, mit den entsprechenden Ergebnissen:

```
# service syslogd restart
logmsg: pri 56, flags 4, from logserv.example.com, msg syslogd: restart
syslogd: restarted
logmsg: pri 6, flags 4, from logserv.example.com, msg syslogd: kernel boot file is /boot/
kernel/kernel
syslogd: kernel boot file is /boot/kernel/kernel
logmsg: pri 166, flags 17, from logserv.example.com,
msg Dec 10 20:55:02 <syslog.err> logserv.example.com syslogd: exiting on signal 2
cvthname(192.168.1.10)
validate: dgram from IP 192.168.1.10, port 514, name logclient.example.com;
accepted in rule 0.
logmsg: pri 15, flags 0, from logclient.example.com, msg Dec 11 02:01:28 trhodes: Test ↺
message 2
Logging to FILE /var/log/logclient.log
Logging to FILE /var/log/messages
```

Zu diesem Zeitpunkt werden die Nachrichten korrekt empfangen und in die richtige Datei geschrieben.

29.12.4. Sicherheitsbedenken

Wie mit jedem Netzwerkdienst, müssen Sicherheitsanforderungen in Betracht gezogen werden, bevor diese Konfiguration umgesetzt wird. Manchmal enthalten Protokolldateien sensitive Daten über aktivierte Dienste auf dem lokalen Rechner, Benutzerkonten und Konfigurationsdaten. Daten, die vom Client an den Server geschickt werden, sind weder verschlüsselt noch mit einem Passwort geschützt. Wenn ein Bedarf für Verschlüsselung besteht, ist es möglich, security/stunnel zu verwenden, welches die Daten über einen verschlüsselten Tunnel versendet.

Lokale Sicherheit ist ebenfalls ein Thema. Protokolldateien sind während der Verwendung oder nach ihrer Rotation nicht verschlüsselt. Lokale Benutzer versuchen vielleicht, auf diese Dateien zuzugreifen, um zusätzliche Einsichten in die Systemkonfiguration zu erlangen. In diesen Fällen ist es absolut notwendig, die richtigen Berechtigungen auf diesen Dateien zu setzen. Das newsyslog(8)-Werkzeug unterstützt das Setzen von Berechtigungen auf gerade erstellte oder rotierte Protokolldateien. Protokolldateien mit Zugriffsmodus `600` sollten verhindern, dass lokale Benutzer darin herumschnüffeln.

29.13. iSCSI Initiator und Target Konfiguration

iSCSI bietet die Möglichkeit, Speicherkapazitäten über ein Netzwerk zu teilen. Im Gegensatz zu NFS, das auf Dateisystemebene arbeitet, funktioniert iSCSI auf Blockgerätebene.

In der iSCSI-Terminologie wird das System, das den Speicherplatz zur Verfügung stellt, als *Target* bezeichnet. Der Speicherplatz selbst kann aus einer physischen Festplatte bestehen, oder auch aus einem Bereich, der mehrere Festplatten, oder nur Teile einer Festplatte, repräsentiert. Wenn beispielsweise die Festplatte(n) mit ZFS formatiert ist, kann ein zvol erstellt werden, welches dann als iSCSI-Speicher verwendet werden kann.

Die Clients, die auf den iSCSI-Speicher zugreifen, werden *Initiator* genannt. Ihnen steht der verfügbare Speicher als rohe, nicht formatierte Festplatte, die auch als LUN bezeichnet wird, zur Verfügung. Die Gerätedateien für die Festplatten erscheinen in `/dev/` und müssen separat formatiert und eingehangen werden.

Seit 10.0-RELEASE enthält FreeBSD einen nativen, kernelbasierten iSCSI *Target* und *Initiator*. Dieser Abschnitt beschreibt, wie ein FreeBSD-System als Target oder Initiator konfiguriert wird.

29.13.1. Ein iSCSI-Target konfigurieren

Anmerkung

Ein natives iSCSI-Target wird seit FreeBSD 10.0-RELEASE unterstützt. Um iSCSI mit älteren Versionen zu benutzen, installieren Sie ein Target aus der Ports-Sammlung, beispielsweise net/istgt. Dieses Kapitel beschreibt nur das native Target.

Um ein iSCSI-Target zu konfigurieren, erstellen Sie die Konfigurationsdatei `/etc/ctl.conf` und fügen Sie eine Zeile in `/etc/rc.conf` hinzu, um sicherzustellen, dass ctld(8) automatisch beim Booten gestartet wird. Starten Sie dann den Daemon.

Das folgende Beispiel zeigt eine einfache `/etc/ctl.conf` . Eine vollständige Beschreibung dieser Datei und der verfügbaren Optionen finden Sie in ctl.conf(5).

```
portal-group pg0 {
 discovery-auth-group no-authentication
 listen 0.0.0.0
 listen [::]
}

target iqn.2012-06.com.example:target0 {
 auth-group no-authentication
 portal-group pg0

 lun 0 {
  path /data/target0-0
  size 4G
 }
}
```

Der erste Eintrag definiert die Portalgruppe `pg0`. Portalgruppen legen fest, auf welchen Netzwerk-Adressen der ctld(8)-Daemon Verbindungen entgegennehmen wird. Der Eintrag `discovery-auth-group no-authentication` zeigt an, dass jeder Initiator iSCSI-Targets suchen darf, ohne sich authentifizieren zu müssen. Die dritte und vierte Zeilen konfigurieren ctld(8) so, dass er auf allen IPv4- (`listen 0.0.0.0`) und IPv6-Adressen (`listen [::]`) auf dem Standard-Port 3260 lauscht.

Es ist nicht zwingend notwendig eine Portalgruppe zu definieren, da es bereits eine integrierte Portalgruppe namens `default` gibt. In diesem Fall ist der Unterschied zwischen `default` und `pg0` der, dass bei `default` eine Authentifizierung nötig ist, während bei `pg0` die Suche nach Targets immer erlaubt ist.

Der zweite Eintrag definiert ein einzelnes Target. Ein Target hat zwei mögliche Bedeutungen: eine Maschine die iSCSI bereitstellt, oder eine Gruppe von LUNs. Dieses Beispiel verwendet die letztere Bedeutung, wobei `iqn.2012-06.com.example:target0` der Name des Targets ist. Dieser Name ist nur für Testzwecke geeignet. Für den tatsächlichen Gebrauch ändern Sie `com.example` auf einen echten, rückwärts geschriebenen Domainnamen.

`2012-06` steht für das Jahr und den Monat, an dem die Domain erworben wurde. `target0` darf einen beliebigen Wert haben und in der Konfigurationsdatei darf eine beliebige Anzahl von Targets definiert werden.

Der Eintrag `auth-group no-authentication` erlaubt es allen Initiatoren sich mit dem angegebenen Target zu verbinden und `portal-group pg0` macht das Target über die Portalgruppe `pg0` erreichbar.

Die nächste Sektion definiert die LUN. Jede LUN wird dem Initiator als separate Platte präsentiert. Für jedes Target können mehrere LUNs definiert werden. Jede LUN wird über eine Nummer identifiziert, wobei LUN 0 verpflichtend ist. Die Zeile mit dem Pfad `path /data/target0-0` definiert den absoluten Pfad zu der Datei oder des zvols für die LUN. Der Pfad muss vorhanden sein, bevor ctld(8) gestartet wird. Die zweite Zeile ist optional und gibt die Größe der LUN an. Als nächstes fügen Sie folgende Zeile in `/etc/rc.conf` ein, um ctld(8) automatisch beim Booten zu starten:

```
ctld_enable="YES"
```

Um ctld(8) jetzt zu starten, geben Sie dieses Kommando ein:

```
# service ctld start
```

Der ctld(8)-Daemon liest beim Start `/etc/ctl.conf` . Wenn diese Datei nach dem Starten des Daemons bearbeitet wird, verwenden Sie folgenden Befehl, damit die Änderungen sofort wirksam werden:

```
# service ctld reload
```

29.13.1.1. Authentifizierung

Die vorherigen Beispiele sind grundsätzlich unsicher, da keine Authentifizierung verwendet wird und jedermann vollen Zugriff auf alle Targets hat. Um für den Zugriff auf die Targets einen Benutzernamen und ein Passwort vorauszusetzen, ändern Sie die Konfigurationsdatei wie folgt:

```
auth-group ag0 {
 chap username1 secretsecret
 chap username2 anothersecret
}

portal-group pg0 {
 discovery-auth-group no-authentication
 listen 0.0.0.0
 listen [::]
}

target iqn.2012-06.com.example:target0 {
 auth-group ag0
 portal-group pg0
 lun 0 {
  path /data/target0-0
  size 4G
 }
}
```

Die Sektion `auth-group` definiert die Benutzernamen und Passwörter. Um sich mit `iqn.2012-06.com.example:target0` zu verbinden, muss ein Initiator zuerst einen Benutzernamen und ein Passwort angeben. Eine Suche nach Targets wird jedoch immer noch ohne Authentifizierung gestattet. Um eine Authentifizierung zu erfordern, setzen Sie `discovery-auth-group` auf eine definierte `auth-group` anstelle von `no-autentication`.

In der Regel wird für jeden Initiator ein einzelnes Target exportiert. In diesem Beispiel wird der Benutzername und das Passwort direkt im Target-Eintrag festgelegt:

```
target iqn.2012-06.com.example:target0 {
 portal-group pg0
 chap username1 secretsecret

 lun 0 {
```

```
  path /data/target0-0
  size 4G
 }
}
```

29.13.2. Einen iSCSI-Initiator konfigurieren

Anmerkung

Der in dieser Sektion beschriebene iSCSI-Initiator wird seit FreeBSD 10.0-RELEASE unterstützt. Lesen Sie iscontrol(8), wenn Sie den iSCSI-Initiator mit älteren Versionen benutzen möchten.

Um den Initiator zu verwenden, muss zunächst ein iSCSI-Daemon gestartet sein. Der Daemon des Initiators benötigt keine Konfigurationsdatei. Um den Daemon automatisch beim Booten zu starten, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
iscsid_enable="YES"
```

Um iscsid(8) jetzt zu starten, geben Sie dieses Kommando ein:

```
# service iscsid start
```

Die Verbindung mit einem Target kann mit, oder ohne eine Konfigurationsdatei `/etc/iscsi.conf` durchgeführt werden. Dieser Abschnitt beschreibt beide Möglichkeiten.

29.13.2.1. Verbindung zu einem Target herstellen - ohne Konfigurationsdatei

Um einen Initiator mit einem Target zu verbinden, geben Sie die IP-Adresse des Portals und den Namen des Ziels an:

```
# iscsictl -A -p 10.10.10.10 -t iqn.2012-06.com.example:target0
```

Um zu überprüfen, ob die Verbindung gelungen ist, rufen Sie `iscsictl` ohne Argumente auf. Die Ausgabe sollte in etwa wie folgt aussehen:

```
Target name                                  Target portal   State
iqn.2012-06.com.example:target0              10.10.10.10     Connected: da0
```

In diesem Beispiel wurde die iSCSI-Sitzung mit der LUN `/dev/da0` erfolgreich hergestellt. Wenn das Target `iqn.2012-06.com.example:target0` mehr als nur eine LUN exportiert, werden mehrere Gerätedateien in der Ausgabe angezeigt:

```
Connected: da0 da1 da2.
```

Alle Fehler werden auf die Ausgabe und in die Systemprotokolle geschrieben. Diese Meldung deutet beispielsweise darauf hin, dass der iscsid(8)-Daemon nicht ausgeführt wird:

```
Target name                                  Target portal   State
iqn.2012-06.com.example:target0              10.10.10.10     Waiting for iscsid(8)
```

Die folgende Meldung deutet auf ein Netzwerkproblem hin, zum Beispiel eine falsche IP-Adresse oder einen falschen Port:

```
Target name                                  Target portal   State
iqn.2012-06.com.example:target0              10.10.10.11     Connection refused
```

Diese Meldung bedeutet, dass der Name des Targets falsch angegeben wurde:

```
Target name                                  Target portal   State
```

```
iqn.2012-06.com.example:target0                  10.10.10.10     Not found
```

Diese Meldung bedeutet, dass das Target eine Authentifizierung erfordert:

```
Target name                                      Target portal   State
iqn.2012-06.com.example:target0                  10.10.10.10     Authentication failed
```

Verwenden Sie diese Syntax, um einen CHAP-Benutzernamen und ein Passwort anzugeben:

```
# iscsictl -A -p 10.10.10.10 -t iqn.2012-06.com.example:target0 -u user -s secretsecret
```

29.13.2.2. Verbindung mit einem Target herstellen - mit Konfigurationsdatei

Wenn Sie für die Verbindung eine Konfigurationsdatei verwenden möchten, erstellen Sie `/etc/iscsi.conf` mit etwa folgendem Inhalt:

```
t0 {
 TargetAddress   = 10.10.10.10
 TargetName      = iqn.2012-06.com.example:target0
 AuthMethod      = CHAP
 chapIName       = user
 chapSecret      = secretsecret
}
```

`t0` gibt den Namen der Sektion in der Konfigurationsdatei an. Diser Name wird vom Initiator benutzt, um zu bestimmen, welche Konfiguration verwendet werden soll. Die anderen Einträge legen die Parameter fest, die während der Verbindung verwendet werden. `TargetAddress` und `TargetName` müssen angegeben werden, die restlichen sind optional. In diesen Beispiel wird der CHAP-Benuztername und das Passwort angegeben.

Um sich mit einem bestimmten Target zu verbinden, geben Sie dessen Namen an:

```
# iscsictl -An t0
```

Um sich stattdessen mit allen definierten Targets aus der Konfigurationsdatei zu verbinden, verwenden Sie:

```
# iscsictl -Aa
```

Damit sich der Initiator automatisch mit allen Targets aus `/etc/iscsi.conf` verbindet, fügen Sie folgendes in `/etc/rc.conf` hinzu:

```
iscsictl_enable="YES"
iscsictl_flags="-Aa"
```

Kapitel 30. Firewalls

Beigetragen von Joseph J. Barbish.
Nach SGML konvertiert und aktualisiert von Brad Davis.
Übersetzt von Michael Bunzel, Johann Kois und Benjamin Lukas.

30.1. Einführung

Firewalls ermöglichen es, den ein- und ausgehenden Netzwerkverkehr Ihres Systems zu filtern. Dazu verwendet eine Firewall eine oder mehrere Gruppen von „Regeln“, um ankommende Netzwerkpakete zu untersuchen und entweder durchzulassen oder zu blockieren. Die Regeln einer Firewall untersuchen charakteristische Eigenschaften von Datenpaketen, darunter den Protokolltyp, die Quell- und Zieladresse sowie den Quell- und Zielport.

Firewalls können die Sicherheit eines Rechners oder eines Netzwerks erhöhen, indem sie folgende Aufgaben übernehmen:

- Den Schutz der Anwendungen, Dienste und Rechner Ihres internen Netzwerks vor unerwünschtem Datenverkehr aus dem Internet.
- Die Beschränkung des Zugriffs von Rechnern des internen Netzwerk auf Rechner oder Dienste des externen Internets.
- Den Einsatz von Network Address Translation (NAT), die es Ihnen durch die Verwendung von privaten IP-Adressen ermöglicht, eine einzige gemeinsame Internetverbindung für mehrere Rechner zu nutzen (entweder über eine einzige Adresse oder über eine Gruppe von jeweils automatisch zugewiesenen öffentlichen IP-Adressen).

Nachdem Sie dieses Kapitel gelesen haben, werden Sie:

- Wissen, wie man korrekte Paketfilterregeln erstellt.
- Die Unterschiede zwischen den in FreeBSD eingebauten Firewalls kennen.
- Wissen, wie man die PF-Firewall von OpenBSD konfiguriert und einsetzt.
- IPFILTER konfigurieren und einsetzen können.
- Wissen, wie man IPFW konfiguriert und einsetzt.

Bevor Sie dieses Kapitel lesen, sollten Sie:

- Die grundlegenden Konzepte von FreeBSD und dem Internet verstehen.

30.2. Firewallkonzepte

Es gibt zwei grundlegende Arten, Regelgruppen für Firewalls zu erstellen: „einschließend“ (*inclusive firewall*) sowie „auschließend“ (*exclusive Firewall*). Eine auschließende Firewall lässt jeden Datenverkehr durch, der nicht durch eine Regel ausgeschlossen wurde. Eine einschließende Firewall macht das genaue Gegenteil. Sie lässt Datenverkehr nur dann durch, wenn er einer der definierten Regeln entspricht.

Eine inclusive Firewall bietet eine wesentlich bessere Kontrolle des ausgehenden Verkehrs, macht sie zur besseren Wahl für Systeme, die Services für das Internet anbieten. Sie kontrolliert auch den Verkehr vom Internet zu ihrem privaten Netzwerk. Jeder Verkehr, der keiner Regel entspricht wird geblockt und geloggt. Inclusive Firewalls sind generell sicherer als exclusive Firewalls, da sie das Risiko, dass unerwünschter Verkehr hindurch geht, drastisch reduzieren.

Anmerkung

Wenn nicht anders vermerkt, verwenden alle Konfigurationen und Beispielregelsets dieses Kapitels inclusive Firewalls.

Die Sicherheit einer Firewall kann durch den Einsatz einer „zustandsabhängigen Firewall“ (*stateful firewall*) weiter erhöht werden. Dieser Typ einer Firewall überwacht alle durch die Firewall gehenden offenen Verbindungen und erlaubt nur schon bestehenden Verkehr oder Datenverkehr, der eine neue Verbindung öffnet. Der Nachteil einer zustandsabhängigen Firewall ist allerdings, dass sie anfällig für Denial of Service (DoS) -Attacken ist, wenn sehr schnell sehr viele neue Verbindungen erstellt werden. Bei den meisten Firewalls können Sie eine Kombination aus zustandsabhängigem und nicht zustandsabhängigem Verhalten verwenden, um eine für Ihre Bedürfnisse optimale Firewall einzurichten.

30.3. Firewallpakete

Das Basissystem von FreeBSD enthält bereits drei Firewallpakete: *IPFILTER* (auch als IPF bekannt), *IPFIREWALL* (auch als IPFW bezeichnet) sowie das von OpenBSD übernommene *PacketFilter* (das auch als PF bezeichnet wird). Zusätzlich verfügt FreeBSD über zwei eingebaute Pakete für das sogenannte *traffic shaping* (dabei handelt es sich die Steuerung des Bandbreitenverbrauchs): altq(4) sowie dummynet(4). Dummynet steht traditionell in enger Verbindung mit IPFW, während ALTQ gemeinsam mit PF eingesetzt wird. Traffic Shaping für IPFILTER ist derzeit mit IPFILTER für NAT sowie Filterung und mit IPFW und dummynet(4) *oder* durch die Kombination von PF mit ALTQ möglich. Gemeinsam ist allen Firewallpaketen (IPF, IPFW sowie PF), dass sie Regeln einsetzen, um den Transfer von Datenpaketen auf und von Ihrem System zu regeln. Unterschiedlich sind aber die Art und Weise, wie dies realisiert wird. Auch die für diese Regeln verwendete Syntax ist unterschiedlich.

FreeBSD überlässt es dem Anwender, das Firewallsystem zu wählen, dass seinen Anforderungen und Vorlieben am Besten entspricht. Keines der im Basissystem enthaltenen Firewallpakete wird dabei als „das beste“ angesehen.

IPFILTER hat etwa den Vorteil, dass dessen zustandsabhängige Regeln relativ einfach in einer NAT-Umgebung implementiert werden können. Außerdem verfügt es über einen eigenen FTP-Proxy, der die Erstellung von sicheren Regeln für ausgehende FTP-Verbindungen vereinfacht.

Da alle Firewalls auf der Untersuchung der Werte ausgewählter Kontrollfelder von Datenpaketen basieren, ist es für die Erstellung von Firewallregeln notwendig, die Funktionsweise von TCP/IP zu verstehen. Außerdem muss man dazu wissen, was die Werte der einzelnen Kontrollfelder bedeuten und wie diese während einer Verbindung eingesetzt werden. Eine gute Erklärung dieser Thematik finden Sie unter `http://www.ipprimer.com/overview.cfm` .

30.4. Paket Filter (PF) von OpenBSD und ALTQ

Revised and updated by John Ferrell.

Im Juli 2003 wurde PF, die Standard-Firewall von OpenBSD, nach FreeBSD portiert und in die FreeBSD-Ports-Sammlung aufgenommen. 2004 war PF in FreeBSD 5.3 Teil des Basissystems. Bei PF handelt es sich um eine komplette, vollausgestattete Firewall, die optional auch ALTQ (Alternatives Queuing) unterstützt. ALTQ bietet Ihnen *Quality of Service* (QoS)-Bandbreitenformung.

Das OpenBSD-Projekt leistet bereits hervorragende Dokumentationsarbeit mit der PF FAQ. Aus diesem Grund konzentriert sich dieser Handbuchabschnitt nur auf diejenigen Besonderheiten von PF, die FreeBSD betreffen, sowie ein paar allgemeine Informationen hinsichtlich der Verwendung. Genauere Informationen zum Einsatz erhalten Sie in der PF FAQ.

Weitere Informationen zu PF für FreeBSD finden Sie unter `http://pf4freebsd.love2party.net/` .

30.4.1. Verwendung der PF-Kernelmodule

Um die PF Kernel Module zu laden, fügen Sie folgende Zeile in ihre `/etc/rc.conf` ein:

```
pf_enable="YES"
```

Danach starten Sie das Startup Script um die Module zu laden:

```
# /etc/rc.d/pf start
```

Das PF Modul wird nicht geladen, falls es die Ruleset Konfigurationsdatei nicht findet. Standardmässig befindet sich diese Datei in `/etc/pf.conf` . Falls das PF Ruleset sich an einem anderen Platz befindet, können Sie das durch Hinzufügen einer Zeile ähnlich der folgenden, in ihrer `/etc/rc.conf` ändern:

```
pf_rules="/path/to/pf.conf "
```

Anmerkung

Ein Beispiel für die Datei `pf.conf` finden Sie im Verzeichnis `/usr/share/examples/pf/` .

Das PF-Modul kann auch manuell über die Kommandozeile geladen werden:

```
# kldload pf.ko
```

Protokollierungsfunktionen für PF werden durch das Modul `pflog.ko` zur Verfügung gestellt und können durch folgenden Eintrag in der `/etc/rc.conf` aktiviert werden:

```
pflog_enable="YES"
```

Danach starten Sie das Startup Script, um das Modul zu laden:

```
# /etc/rc.d/pflog start
```

Falls Sie noch weitere Features für PF benötigen, müssen Sie diese in den Kernel einbauen.

30.4.2. PF Kernel-Optionen

Es ist nicht zwingend nötig, dass Sie PF-Unterstützung in den FreeBSD-Kernel kompilieren. Sie werden dies tun müssen, um eine von PFs fortgeschritteneren Eigenschaften nutzen zu können, die nicht als Kernelmodul verfügbar ist. Genauer handelt es sich dabei um pfsync(4), ein Pseudo-Gerät, welches bestimmte Änderungen der PF-Zustandstabelle offenlegt. Es kann mit carp(4) kombiniert werden, um ausfallsichere Firewalls mit PF zu realisieren. Weitere Informationen zu CARP erhalten Sie in Abschnitt 31.10, „Common Address Redundancy Protocol (CARP)“ des Handbuchs.

Die Kernelkonfigurationsoptionen von PF befinden sich in `/usr/src/sys/conf/NOTES` und sind im Folgenden wiedergegeben:

```
device pf
device pflog
device pfsync
```

Die Option `device pf` aktiviert die Unterstützung für die „Packet Filter“-Firewall (pf(4)).

Die Option `device pflog` aktiviert das optionale pflog(4)-Pseudonetzwerkgerät, das zum Protokollieren des Datenverkehrs über einen bpf(4)-Deskriptor dient. pflogd(8) ist in der Lage, diese Protokolldateien auf Ihre Platte zu speichern.

Die Option `device pfsync` aktiviert das optionale pfsync(4)-Pseudonetzwerkgerät für die Überwachung von „Statusänderungen“.

30.4.3. Verfügbare rc.conf-Optionen

Die folgenden rc.conf(5)-Einträge konfigurieren PF und pflog(4) beim Systemstart:

```
pf_enable="YES"                 # PF aktivieren (Modul, wenn nötig, aktivieren)
pf_rules="/etc/pf.conf"         # Datei mit Regeldefinitionen für pf
pf_flags=""                     # zusätzliche Parameter für den Start von pfctl
pflog_enable="YES"              # starte pflogd(8)
pflog_logfile="/var/log/pflog"  # wo soll pflogd die Protokolldatei speichern
pflog_flags=""                  # zusätzliche Parameter für den Start von pflogd
```

Wenn Sie ein lokales Netzwerk hinter dieser Firewall betreiben und Pakete für dessen Rechner weiterleiten oder NAT verwenden wollen, benötigen Sie zusätzlich die folgende Option:

```
gateway_enable="YES"            # LAN Gateway aktivieren
```

30.4.4. Filterregeln erstellen

PF liest seine konfigurierten Regeln aus pf.conf(5) (standardmässig `/etc/pf.conf`) und modifiziert, verwirft oder lässt Pakete passieren anhand der Regeln oder Definitionen, die in dieser Datei gespeichert sind. FreeBSD enthält dazu nach der Installation mehrere Beispieldateien, die in `/usr/share/examples/pf/` abgelegt sind. Für eine ausführliche Behandlung des PF-Regelwerks lesen Sie bitte die PF FAQ.

Warnung

Beim Lesen der PF FAQ wollten Sie darauf achten, dass verschiedene Versionen von FreeBSD auch unterschiedliche Versionen von PF enthalten. FreeBSD 8.*X* (und älter) FreeBSD-Versionen benutzen PF aus OpenBSD 4.1. FreeBSD 9.*X* (und neuer) benutzen hingegen PF aus OpenBSD 4.5.

Die FreeBSD packet filter mailing list ist eine erste Anlaufstelle für Fragen zur Konfiguration und dem Einsatz der PF Firewall. Vergessen Sie nicht, vorher die Mailinglistenarchive zu durchsuchen, bevor Sie dort eine Frage stellen!

30.4.5. Arbeiten mit PF

Benutzen Sie pfctl(8), um PF zu steuern. Unten finden Sie ein paar nützliche Befehle (lesen Sie auch die Manualpage zu pfctl(8), um alle verfügbaren Optionen nachzuschlagen):

Befehl	Zweck
`pfctl -e`	PF aktivieren
`pfctl -d`	PF deaktivieren
`pfctl -F all -f /etc/pf.conf`	Alle Filterregeln zurücksetzen (NAT, Filter, Zustand, Tabelle, etc.) und erneut aus der Datei `/etc/pf.conf` auslesen
`pfctl -s [ rules | nat | states ]`	Bericht über die Filterregeln, NAT-Regeln, oder Zustandstabellen
`pfctl -vnf /etc/pf.conf`	überprüft `/etc/pf.conf` auf Fehler, lädt aber das Regelwerk nicht neu

30.4.6. ALTQ aktivieren

ALTQ muss vor der Verwendung in den FreeBSD-Kernel kompiliert werden. Beachten Sie, dass ALTQ nicht von allen verfügbaren Netzwerkkartentreibern unterstützt wird. Sehen Sie daher zuerst in altq(4) nach, ob Ihre Netzwerkkarte diese Funktion unter Ihrer FreeBSD-Version unterstützt.

Die folgenden Kerneloptionen aktivieren ALTQ sowie alle Zusatzfunktionen:

```
options         ALTQ
options         ALTQ_CBQ        # Class Bases Queuing (CBQ)
options         ALTQ_RED        # Random Early Detection (RED)
options         ALTQ_RIO        # RED In/Out
options         ALTQ_HFSC       # Hierarchical Packet Scheduler (HFSC)
options         ALTQ_PRIQ       # Priority Queuing (PRIQ)
options         ALTQ_NOPCC      # Wird von SMP benötigt
```

`options ALTQ` aktiviert das ALTQ-Framework.

`options ALTQ_CBQ` aktiviert das *Class Based Queuing* (CBQ). CBQ erlaubt es, die Bandbreite einer Verbindung in verschiedene Klassen oder Warteschlangen zu unterteilen, um die Priorität von Datenpaketen basierend auf Filterregeln zu ändern.

`options ALTQ_RED` aktiviert *Random Early Detection* (RED). RED wird zur Vermeidung einer Netzwerkverstopfung verwendet. Dazu ermittelt RED die Größe der Warteschlange und vergleicht diesen Wert mit den minimalen und maximalen Grenzwerten der Warteschlange. Ist die Warteschlange größer als das erlaubte Maximum, werden alle neuen Pakete verworfen. Getreu seinem Namen verwirft RED Pakete unterschiedlicher Verbindungen nach dem Zufallsprinzip.

`options ALTQ_RIO` aktiviert *Random Early Detection In and Out*.

`options ALTQ_HFSC` aktiviert den *Hierarchical Fair Service Curve*-Paketplaner. Weitere Informationen zu HFSC finden Sie unter `http://www-2.cs.cmu.edu/~hzhang/HFSC/main.html` .

`options ALTQ_PRIQ` aktiviert *Priority Queuing* (PRIQ). PRIQ lässt Verkehr einer Warteschlange mit höherer Priorität zuerst durch.

`options ALTQ_NOPCC` aktiviert die SMP Unterstützung von ALTQ. Diese Option ist nur auf SMP-System erforderlich.

30.5. Die IPFILTER-Firewall (IPF)

Geschrieben wurde IPFILTER von Darren Reed. IPFILTER ist vom Betriebssystem unabhängig: Es ist eine Open Source Anwendung, die auf die Betriebssysteme FreeBSD, NetBSD, OpenBSD, SunOS™, HP/UX und Solaris™ portiert wurde. IPFILTER wird aktiv betreut und gepflegt. Es werden regelmäßig neue Versionen herausgegeben.

IPFILTER basiert auf einer kernelseitigen Firewall und einem NAT Mechanismus, der durch Anwenderprogramme betreut und gesteuert werden kann. Die Regeln der Firewall werden mit dem Programm ipf(8) gesetzt oder gelöscht. Für die Manipulation der NAT Regeln verwendet man ipnat(1). Mit ipfstat(8) werden Laufzeitstatistiken der kernelseitigen Anteile von IPFILTER aufgelistet. Und mit dem Programm ipmon(8) kann man die Aktionen von IPFILTER in die Protokolldateien des Systems speichern.

IPF funktionierte ursprünglich mit einer Regel-Prozess-Logik à la „die letzte Regel, die passt, entscheidet“ und verwendete ausschließlich Regeln ohne feste Zustände. Inzwischen wurde die Regel-Prozess-Logik drastisch modernisiert: Es gibt eine `quick` und eine zustandsorientierte `keep-state` Option. Die offizielle Dokumentation beinhaltet leider nur die veralteten Parameter zur Regelerstellung - die neuen Funktionen werden nur als Zusatzoptionen aufgelistet, was ihre Vorteile beim Erstellen einer weit überlegenen und viel sichereren Firewall völlig untergräbt.

Die Anweisungen in diesem Kapitel basieren darauf, Regeln mit den Optionen `quick` und `keep-state` zu erstellen. Mit diesem Grundwissen wird man einen kompletten einschließenden Regelsatz erstellen können.

Für eine ausführliche Erläuterung der alten Methode zur Regelverarbeitung schauen Sie bitte auf `http://www.obfuscation.org/ipf/ipf-howto.html#TOC_1` oder `http://coombs.anu.edu.au/~avalon/ip-filter.html` .

Antworten auf häufige Fragen finden Sie unter `http://www.phildev.net/ipf/index.html` .

Und ein durchsuchbares Archiv der Mailingliste zu IPFILTER gibt es unter `http://marc.theaims-group.com/?l=ipfilter`.

30.5.1. Aktivieren von IPF

FreeBSD enthält IPF in der Standardversion als ladbares Kernelmodul. Dieses Modul wird vom System automatisch geladen, wenn in der `rc.conf` der Eintrag `ipfilter_enable="YES"` angelegt wird. In dieser ursprünglichen Konfiguration ist die Protokollierung aktiv und die Option `default pass all` ("Pakete passieren lassen") als Standard gesetzt. Um die `block all` ("alles Blockieren") Option zu setzen, muss man nicht gleich einen neuen Kernel bauen - es reicht, `block all` als letzte Position des Regelsatzes aufzulisten.

30.5.2. Kernel-Optionen

Es ist nicht unbedingt notwendig, IPF durch die folgenden Optionen direkt in der Kernel einzubinden. Diese Möglichkeit der Verwendung von IPF wird hier mehr als Hintergrundwissen angeboten. Man sollte nur wissen, dass dadurch nicht mehr das Kernelmodul geladen wird - und dementsprechend auch nicht mehr entladen werden kann.

Die Beschreibung der einzelnen Optionen von IPF für die Verwendung in der Kernelkonfiguration finden Sie auch in der Datei `/usr/src/sys/conf/NOTES` .

```
options IPFILTER
options IPFILTER_LOG
options IPFILTER_DEFAULT_BLOCK
```

`options IPFILTER` aktiviert die Verwendung der „IPFILTER“ Firewall.

`options IPFILTER_LOG` aktiviert den Logging-Mechanismus. Das bedeutet, dass jedes Paket geloggt wird, auf das eine Regel passt, die das Schlüsselwort `log` enthält. Dazu wird der Pseudo—Device `ipl` verwendet.

`options IPFILTER_DEFAULT_BLOCK` ändert das Verhalten der Firewall dahingehend, dass jedes Paket, dass nicht explizit von einer `pass` Regel Zugang erhält, abgewiesen, bzw. geblockt, wird.

Diese Einstellungen werden erst aktiv, wenn der Kernel, in den sie eingebunden wurden, kompiliert, installiert und gebootet wurde.

30.5.3. Optionen in rc.conf

Um IPF während des Bootvorgangs einzubinden, braucht man lediglich die folgenden Zeilen der Datei `/etc/rc.conf` anzufügen:

```
ipfilter_enable="YES"             # Startet IPF
ipfilter_rules="/etc/ipf.rules"   # liest den Regelsatz aus einer Datei
ipmon_enable="YES"                # Startet das IP-Monitor Log
ipmon_flags="-Ds"                 # D = Als Da:mon starten
                                  # s = Protokollierung via syslog
                                  # v = Protokollierung von tcp window, ack, seq
                                  # n = Namen statt IP & port ausgeben
```

Falls sich hinter der Firewall ein lokales Netzwerk befindet, das den reservierten privaten Adressbereich verwendet, müssen die folgenden Zeilen zur Aktivierung von NAT ebenfalls in `/etc/rc.conf` eingetragen werden:

```
gateway_enable="YES"              # Aktivierung des LAN-Gateways
ipnat_enable="YES"                # Startet die ipnat Funktion
ipnat_rules="/etc/ipnat.rules"    # Liest die ipnat-Regeldefinitionen aus einer Datei
```

30.5.4. Der Befehl ipf

Mit dem Befehl ipf(8) liest man die Datei, die den Regelsatz enthält ein. Mit dem folgenden Befehl können Sie Ihre eigenen, für Ihr System maßgeschneiderten Regeln einlesen und so in einem Schritt alle Regeln der laufenden Firewall ersetzen:

```
# ipf -Fa -f /etc/ipf.rules
```

`-Fa` bedeutet, dass alle intern gespeicherten Tabellen mit Regeln gelöscht werden.

`-f` gibt die Datei an, aus der die neuen Regeln gelesen werden sollen.

Mit diesen beiden Optionen erhalten Sie die Möglichkeit, Änderungen an der Datei mit Ihrem Regelsatz vorzunehmen und gleich die Firewall mit den neuen Regeln zu bestücken, ohne den Rechner neu starten zu müssen. Da dieser Vorgang beliebig wiederholt werden kann, ist es ein sehr bequemer Weg, neue Regeln einzuarbeiten und zu testen.

Um mehr über diese und weitere Optionen von ipf(8) zu erfahren, konsultieren Sie bitte die Manpage.

ipf(8) erwartet, dass es sich bei der Datei mit dem Regelsatz um eine Standard-Textdatei handelt. Eine Datei, die ein Skript oder Variablen enthält, wird nicht verarbeitet.

Es gibt allerdings doch einen Weg, IPF Regeln mit Hilfe von Skripten und Variablen zu erstellen. Weitere Informationen dazu finden Sie unter Abschnitt 30.5.9, „Die Erstellung eines Regelsatzes mit Variablen“.

30.5.5. IPFSTAT

Das normale Verhalten von ipfstat(8) ist, die Zusammenfassung der angefallenen Statistiken, die als Resultat der Anwendung von nutzerspezifischen Regeln auf ein- und ausgehende Pakete seit dem letzten Start der Firewall oder seit dem letzten Zurücksetzen der Zähler auf Null durch das Kommando `ipf -Z` angesammelt wurden, abzurufen und anzuzeigen.

Für weiterführende Informationen schauen Sie bitte auf die Manpage von ipfstat(8)!

Die Ausgabe von ipfstat(8), wenn keine Parameter übergeben wurden, sieht etwa so aus:

```
input packets: blocked 99286 passed 1255609 nomatch 14686 counted 0
output packets: blocked 4200 passed 1284345 nomatch 14687 counted 0
input packets logged: blocked 99286 passed 0
output packets logged: blocked 0 passed 0
packets logged: input 0 output 0
log failures: input 3898 output 0
fragment state(in): kept 0 lost 0
fragment state(out): kept 0 lost 0
packet state(in): kept 169364 lost 0
packet state(out): kept 431395 lost 0
ICMP replies: 0 TCP RSTs sent: 0
Result cache hits(in): 1215208 (out): 1098963
IN Pullups succeeded: 2 failed: 0
OUT Pullups succeeded: 0 failed: 0
Fastroute successes: 0 failures: 0
TCP cksum fails(in): 0 (out): 0
Packet log flags set: (0)
```

Wenn die Option `-i` für „eingehend“ oder `-o` für „ausgehend“ übergeben wird, liefert das Kommando eine entsprechende Liste von Filter-Regeln, die gerade installiert sind und vom Kernel verwendet werden.

`ipfstat -in` zeigt alle aktive Regeln für eingehende Verbindungen zusammen mit ihren Nummern.

`ipfstat -on` erledigt dasselbe für die ausgehenden Verbindungen.

Die Ausgabe sieht in etwa folgendermaßen aus:

```
@1 pass out on xl0 from any to any
```

```
@2 block out on dc0 from any to any
@3 pass out quick on dc0 proto tcp/udp from any to any keep state
```

`ipfstat -ih` zeigt die Tabelle der aktiven Regeln für eingehende Verbindungen zusammen mit der Anzahl, wie oft jeder einzelnen Regel entsprochen wurde.

`ipfstat -oh` zeigt das Gleiche für die ausgehenden Verbindungen.

Hier wird die Ausgabe so oder so ähnlich aussehen:

```
2451423 pass out on xl0 from any to any
354727 block out on dc0 from any to any
430918 pass out quick on dc0 proto tcp/udp from any to any keep state
```

Einer der wichtigsten Funktionen von `ipfstat` wird über die Option `-t` bereitgestellt. Mit ihr wird eine Statustabelle vergleichbar der Prozess-Tabelle von top(1) ausgegeben. Mit dieser Funktion erhalten Sie im Falle eines Angriffs die Möglichkeit, die angreifenden Pakete zu identifizieren, abzufangen und auszuwerten. Weitere Unteroptionen eröffnen, die IP-Adresse, den Port oder das Protokoll, geteilt nach Herkunft und Ziel, auszuwählen und dann in Echtzeit zu beobachten. Lesen Sie dazu bitte auch die Manpage von ipfstat(8).

30.5.6. IPMON

Damit der Befehl `ipmon` korrekt arbeiten kann, muss die Option `IPFILTER_LOG` in die Kernelkonfiguration eingearbeitet werden. Das Kommando selbst arbeitet in zwei verschiedenen Modi. Für den nativen Modus startet man `ipmon` auf der Kommandozeile ohne die Option `-D`.

Der Hintergrundmodus (`daemon mode`) dient der Erstellung eines stetigen Systemprotokolls, so dass Einträge vergangener Ereignisse inspiziert werden können. So sollen FreeBSD und IPFILTER entsprechend ihrer Konfiguration zusammen arbeiten. FreeBSD kann mit einem eingebauten Mechanismus Systemprotokolle turnusmäßig abspeichern. Aus diesem Grund sollte man besser syslogd(8) verwenden anstatt die Protokollinformationen in eine Datei zu schreiben, wie es als Standard vorgesehen ist. In der Standard-`rc.conf`-Datei (im Ordner `/etc/defaults/`) wird dem Eintrag `ipmon_flags` die Option `-Ds` übergeben:

```
ipmon_flags="-Ds" # D = Als Da:mon starten
# s = Protokollierung via syslog
# v = Protokollierung von tcp window, ack, seq
# n = Namen statt IP & port ausgeben
```

Die Vorteile des Protokollierens liegen auf der Hand: Sie versetzen den Administrator in die Lage, nach einem Vorfall Informationen abzurufen, etwa welche Pakete aussortiert wurden, welche Adressen diese Pakete gesendet haben oder wohin sie gesendet werden sollten. Alles in allem erhält er ein sehr gutes Werkzeug zum Aufspüren von Angreifern.

Jedoch, auch wenn die Protokollierung aktiviert ist, wird IPF keine einzige Regel zum Protokollieren von alleine entwerfen und umsetzen. Der Administrator der Firewall entscheidet, welche Regeln in seinem Regelsatz mitgeschrieben werden sollen und er muss dementsprechend das Schlüsselwort `log` in dieser Regel angeben. Normalerweise werden nur Treffer auf abweisende Regeln protokolliert.

Es ist üblich, als letzte Regel eine alles blockierende Regel mit dem Schlüsselwort `log` in den Regelsatz einzutragen. Dadurch erkennt man alle Pakete, die keiner Regel im Regelsatz entsprachen.

30.5.7. IPMON Logging

Syslogd verwendet seine eigene Methode zum Sortieren der gesammtelten Protokolldaten - spezielle Gruppierungen namens „facility“ und „level“. IPMON verwendet im `daemon`-Modus als „facility“ den Wert `security`. Die folgenden „level“ können für eine genauere Trennung der Protokolldaten verwendet werden:

```
LOG_INFO - Alle zu protokollierenden Pakete
LOG_NOTICE - Protokollierte Pakete, die passieren durften
LOG_WARNING - Protokollierte Pakete, die blockiert wurden
```

```
LOG_ERR - Protokollierte Pakete, deren Headerdaten nicht komplett vorlagen
```

Damit IPFILTER angewiesen werden kann, alle Protokolldaten in die Datei `/var/log/ipfilter.log` zu schreiben, muss diese erst erstellt werden. Folgendes Kommando übernimmt diese Aufgabe:

```
# touch /var/log/ipfilter.log
```

Die Funktionen von syslogd(8) werden durch Definition in der Datei `/etc/syslog.conf` gesteuert. In dieser Datei kann sehr weitläfig eingestellt werden, wie syslog mit den Systemnachrichten umgehen soll, die ihm von Anwendungen wie IPF übergeben werden.

Fügen Sie folgende Definition in die Datei `/etc/syslog.conf` ein, um die Protokollierung für IPF via `syslog` zu aktivieren:

```
security.* /var/log/ipfilter.log
```

`security.*` bedeutet, dass alle Nachrichten der Klasse `security.*` am angegebenen Ort (hier eine Datei) geschrieben werden sollen.

Um Änderungen an der Datei `/etc/syslog.conf` zu aktivieren müssen Sie den Rechner neu starten, oder den Befehl

```
# /etc/rc.d/syslogd reload
```

ausführen.

Vergessen Sie nicht, `/etc/newsyslog.conf` anzupassen, damit die neuen Protokolldateien, die eben konfiguriert wurden, auch in den Rotationsturnus eingefügt werden!

30.5.8. Die Formatierung der Logdatei

Nachrichten, die durch `ipmon` erzeugt werden, bestehen aus durch Leerstellen getrennten Datenfeldern. Folgende Felder sind in allen Nachrichten enthalten:

1. Das Datum der Paketerstellung.
2. Die Uhrzeit der Paketerstellung in der Form `HH:MM:SS.F`, mit Stunden, Minuten, Sekunden und Sekundenbruchteilen, wobei letztere mehrere Stellen lang sein können.
3. Der Name der Schnittstelle, die das Paket verarbeitet hat, bspw. `dc0`.
4. Die Gruppe und die Nummer der angewandten Regel, bspw. `@0:17`.
5. Die ausgeführte Aktion: p für `passed` (zugelassen), b für blockiert, S für `short packet` (unvollständiger Header), n für `no match` (gar keine Regel wurde berührt) und L für Log-Regel. Die Reihe, in der die Flags angezeigt werden ist: S, p, b, n, L. Ein groß geschriebenes P oder B bedeutet, dass das Paket aufgrund einer globalen Einstellung protokolliert wurde und nicht wegen einer einzelnen Regel.
6. Die Adressen. Diese bestehen aus drei Feldern: Der Quelladresse mit Port (getrennt durch ein Komma), dem Symbol „->“ und der Zieladresse. Also bspw. `209.53.15.22,80 -> 198.64.221.18,1722` .
7. `PR` gefolgt vom Namen eines Netzwerk-Protokolls oder dessen Nummer. Bspw. `PR tcp`.
8. `len` gefolgt von der Länge des Headers und der Gesamtlänge des Paketes, beispielsweise `len 20 40`.

Wenn es sich um ein TCP-Paket handelt, wird ein weiteres Feld, beginnend mit einem Querstrich und gefolgt von Buchstaben, die den gesetzten Flags entsprechen, angezeigt. Lesen Sie bitte die Manpage ipmon(8) für eine Liste der Buchstaben und deren Bedeutungen.

Falls das Paket ein ICMP-Paket ist, werden zwei Felder am Ende hinzugefügt - das erstere ist immer „ICMP“, das zweite enthält die ICMP-Nachricht und den Nachrichtentyp, getrennt durch einen Schrägstrich. `ICMP 3/3` steht beispielsweise für „Port nicht erreichbar“.

30.5.9. Die Erstellung eines Regelsatzes mit Variablen

Erfahrenere IPF Anwender erstellen sich eine Datei, die die Regeln enthält und gestalten diese als ein Skript, in dem Variablen verwendet werden. Der wichtigste Vorteil besteht darin, dass man lediglich den Wert der Variablen anpassen muss und diese, sobald das Skript gestartet wird, durch die entsprechenden Werte ersetzt und die Regeln entsprechend formuliert werden. In Skripten kann man so häufig verwendete Werte einfach als Variable in mehreren Regeln zuweisen. Am folgenden Beispiel soll das verdeutlicht werden.

Die Syntax dieses Skriptes ist kompatibel mit den Shells sh(1), csh(1) und tcsh(1).

Variablen beginnen mit einem Dollar-Zeichen: `$Variablenname`. Im Beispiel unten steht `$oif` für die Variable, in der der Name der Schnittstelle abgelegt wird, über die der Verkehr nach außen erfolgt.

In Variablenzuweisungen fehlt das beginnende $-Zeichen. Alleine der Name der Variable wird angegeben, gefolgt von einem Gleichheitszeichen, und dem Wert, der der Variablen zugewiesen werden soll. Dieser muss in doppelten Anführungszeichen (`""`) stehen. Also folgt eine Zuweisung dem Schema `Variablenname = "Wert"`.

```
############ Start of IPF rules script ######################

oif="dc0"           # Name der Internet-Schnittstelle
odns="192.0.2.11"   # IP des DNS-Servers unseres ISPs
myip="192.0.2.7"    # die statische IP, die uns der ISP zugeteilt hat
ks="keep state"
fks="flags S keep state"

# Sie haben die Wahl, aus diesem Skript eine eigene
# /etc/ipf.rules erstellen zu lassen oder es einfach
# direkt als Skript laufen zu lassen.
#
# Entfernen Sie dazu das eine Kommentarzeichen
# und kommentieren Sie die andere Zeile aus!
#
# 1) Diese Zeile verwenden Sie zur Erstellung von /etc/ipf.rules
#cat > /etc/ipf.rules << EOF
#
# 2) Diese Zeile, wenn Sie direkt mit dem Skript arbeiten wollen
/sbin/ipf -Fa -f - << EOF

# Erlaubnis ausgehenden Verkehrs an den Nameserver des ISPs
pass out quick on $oif proto tcp from any to $odns port = 53 $fks
pass out quick on $oif proto udp from any to $odns port = 53 $ks

# Erlaubnis ausgehenden unsicheren www-Verkehrs
pass out quick on $oif proto tcp from $myip to any port = 80 $fks

# Erlaubnis ausgehenden sicheren www-Verkehrs https via TLS SSL
pass out quick on $oif proto tcp from $myip to any port = 443 $fks
EOF
################## End of IPF rules script ######################
```

Das ist schon alles. Die Regeln selbst sind im Beispiel nicht so wichtig - achten Sie auf die Anwendung der Variablenzuweisung am Anfang und die Verwendung der Variablen im Skript. Falls das obige Beispiel in einer Datei namens `/etc/ipf.rules.script` gespeichert wurde, können die Regeln mit folgenden Kommando neu geladen werden:

```
# sh /etc/ipf.rules.script
```

Es gibt ein Problem mit Regelsatz-Dateien, die Variablen verwenden: IPF kann mit Variablen nichts anfangen - und kann derartige Skripte nicht direkt einlesen.

Unser kleines Skript kann daher nur auf eine der beiden folgenden Weisen verwendet werden:

- Entfernen Sie das Kommentarzeichen der Zeile, die mit `cat` beginnt. Kommentieren Sie die Zeile aus, die mit `/sbin/ipf` beginnt. Schreiben Sie die Zeile `ipfilter_enable="YES"` in die Datei `/etc/rc.conf` und rufen Sie dann das Skript auf, um `/etc/ipf.rules` zu erstellen oder zu erneuern.

- Deaktivieren Sie IPFILTER in den Systemstart-Skripten, indem Sie die Zeile `ipfilter_enable="NO"` in die Datei `/etc/rc.conf` eintragen (was auch der Standard-Einstellung entspricht).

  Fügen Sie ein Skript ähnlich dem folgenden in Ihr Verzeichnis `/usr/local/etc/rc.d/` . Es sinnvoll, dem Skript einen offensichtlichen Namen zu geben, wie etwa `ipf.loadrules.sh`. Die Endung `.sh` ist dabei verbindlich.

```
#!/bin/sh
sh /etc/ipf.rules.script
```

  Die Zugriffsrechte für die Datei, die das Skript enthält, müssen für den Eigentümer `root` auf Lesen, Schreiben und Ausführen gesetzt werden.

```
# chmod 700 /usr/local/etc/rc.d/ipf.loadrules.sh
```

Wenn nun Ihr System startet, werden Ihre IPF-Regeln geladen.

30.5.10. IPF Regelsätze

Ein Regelsatz ist eine Gruppe von IPF-Regeln, die anhand der Werte eines Netzwerkpaketes entscheiden, ob dieses Paket durchgelassen oder blockiert wird. Der Austausch von Paketen erfolgt immer zweiseitig in Form einer sogenannten Session. Der Regelsatz der Firewall verarbeitet sowohl die eingehenden Pakete aus dem öffentlichen Internet als auch die Pakete, die vom System als Antwort auf die Ersteren gesendet werden. Jeder Dienst, der via TCP/IP arbeitet, zum Beispiel `telnet`, `www` oder `mail`, ist vordefiniert durch sein Protokoll und seinen privilegierten Port, an dem er auf Anfragen wartet und reagieren kann. Pakete, die gezielt einen Dienst ansprechen sollen, werden von einem unprivilegierten Port des Senders an einen konkreten privilegierten Port des Zielsystems geschickt. Alle genannten Parameter (Ports, Adressen usw.) können als Auswahlkriterien zum erstellen von Regeln eingesetzt werden, die Dienste erlauben oder blockieren.

IPF wurde ursprünglich mit einer Regel-Prozess-Logik geschrieben, die ausschließlich statusfreie Regeln zuließ und nach dem Prinzip „die letzte Regel, die passt, entscheidet“ arbeitete. Mit der Zeit erhielt IPF eine `quick` Option sowie `keep-state` Option für die Anwendung von zustandsorientierten Regeln, was die Regel-Prozess-Logik signifikant modernisierte.

Die Anweisungen in diesem Kapitel basieren auf der Verwendung von Regeln, die diese beiden neuen Optionen verarbeiten. Dies ist das Framework zur Entwicklung eines Firewallregelsatzes.

Warnung

Wenn Sie mit einer Firewall arbeiten, seien Sie *sehr vorsichtig*. Durch wenige Einstellungen können Sie sich aus Ihrem System *aussperren*. Wenn Sie auf der sicheren Seite sein wollen, führen Sie die Firewall-Konfiguration direkt am entsprechenden Gerät aus und nicht über eine Netzwerkverbindung wie bspw. ssh.

30.5.11. IPF Regel-Syntax

Die Syntax zur Erstellung der Regeln, die hier vorgestellt wird, ist dahingehend vereinfacht worden, dass sie ausschliesslich auf den modernen Regelkontext, mit statusbehafteten Regeln und einer „die erste Regel, die passt, gewinnt“-Logik, zurückgreift. Um alles über die veraltete Syntax zu erfahren, lesen Sie bitte die Man-Page von ipf(8).

Ein #-Zeichen markiert den Beginn eines Kommentars. Es darf nach nach einer Regel stehen oder als erstes Zeichen einer Zeile. Leere Zeilen werden von der Regel-Prozess-Logik ignoriert.

Regeln enthalten Schlüsselwörter. Diese Schlüsselwörter müssen in einer bestimmten Reihenfolge von links nach rechts in einer Zeile erscheinen. Als solche identifizierte Schlüsselwörter werden fett wiedergegeben. Einige

Schlüsselwörter haben Unteroptionen, die wiederum selbst Schlüsselwörter sein und ebenfalls weiter Unteroptionen einschließen können.

ACTION IN-OUT OPTIONS SELECTION STATEFUL PROTO SRC_ADDR,DST_ADDR OBJECT PORT_NUM TCP_FLAG STATEFUL

ACTION = block | pass

IN-OUT = in | out

OPTIONS = log | quick | on interface-name

SELECTION = proto value | source/destination IP | port = number | flags flag-value

PROTO = tcp/udp | udp | tcp | icmp

SRC_ADD,DST_ADDR = all | from object to object

OBJECT = IP address | any

PORT_NUM = port number

TCP_FLAG = S

STATEFUL = keep state

30.5.11.1. ACTION

Die „ACTION“ bestimmt, was mit dem Paket passieren soll, wenn der Rest der Regel zutrifft. Dieser Teil muss für jede Regel angegeben werden.

Das Schlüsselwort `block` gibt an, dass das Paket verfallen soll, wenn die Auswahlparameter zutreffen.

Das Schlüsselwort `pass` gibt an, dass das Paket durch die Firewall durchgelassen werden soll, wenn die Auswahlparameter zutreffen.

30.5.11.2. IN-OUT

Ebenfalls verbindlich ist die Angabe, welchen Teil der Verbindung, Ein- oder Ausgang, die Regel beeinflussen soll. Das nächste Schlüsselwort muss daher entweder `in`, für eingehend, oder `out`, für ausgehend, lauten - oder die Regel wird aufgrund eines Syntaxfehlers nicht umgesetzt.

`in` bedeutet, dass diese Regel auf eingehende Pakete angewendet wird, die gerade an der dem öffentlichen Internet zugewandten Schnittstelle empfangen wurden.

`out` bedeutet, das diese Regel auf ausgehende Pakete angewendet wird, also Pakete die gerade gesendet werden und deren Zieladresse im öffentlichen Internet liegt.

30.5.11.3. OPTIONS

Anmerkung

Die Optionen müssen in der hier aufgeführten Reihenfolge verwendet werden.

`log` bestimmt, dass die Kopfdaten des Paketes an die Systemschnittstelle ipl(4) geschrieben werden sollen. Genaueres dazu weiter unten im Abschnitt LOGGING.

`quick` bestimmt, dass, *wenn* die Auswahlkriterien der Regel auf das Paket zutreffen, keine weiteren Regeln ausgewertet werden. So vermeidet man das Abarbeiten des gesamten Regelsatzes. Diese Option ist eine verbindliche Vorraussetzung der modernen Regel-Prozess-Logik.

`on` bestimmt den Namen der Schnittstelle, der als Auswahlkriterium hinzugefügt werden soll. Die Namen aller verfügbaren Schnittstellen werden durch den Befehl ifconfig(8) angezeigt. wenn man diese Option verwendet, passt die Regeln nur auf Pakete, die durch diese Schnittstelle empfangen (`in`) oder gesendet (`out`) wurden. Für die modernisierte Regel-Prozess-Logik ist die Verwendung dieser Option verbindlich.

Wenn ein Paket protokolliert wird, werden die Kopfdaten in die Pseudo-Schnittstelle ipl(4) geschrieben. Folgende Parameter können zusätzlich übergeben werden, müssen dazu aber direkt nach dem Schlüsselwort `log` und in gleicher Reihenfolge stehen:

`body` bestimmt, dass die ersten 128 Bytes des Paketinhaltes zusätzlich zu den Kopfdaten protokolliert werden.

`first` trifft nur zu, wenn das Schlüsselwort `log` zusammen mit `keep-state` verwendet wird. Es bestimmt, dass nur das auslösende Paket protokolliert wird und nicht jedes weitere Paket, dass von der gespeicherten Status-Regel betroffen ist.

30.5.11.4. SELECTION

Die Schlüsselwörter, die in diesem Abschnitt vorgestellt werden, dienen zur Beschreibung von Attributen, anhand derer geprüft und entschieden wird, ob eine Regel zutrifft oder nicht. Es gibt ein Schlüsselwort, und das hat mehrere Optionen, von denen eine ausgewählt werden muss. Die folgenden allgemeinen Attribute können beliebig zum Erstellen einer Regel verwendet werden, allerdings nur in der vorgestellten Reihenfolge:

30.5.11.5. PROTO

`proto` ist das Schlüsselwort für das im Paket angewendete Protokoll. Als Option ein Protokoll aus Auswahlkriterium übergeben. Diese Option ist verbindlich, wenn man die moderne Regel-Prozess-Logik verwendet.

`tcp/udp | udp | tcp | icmp` oder irgendein Protokollname, der in der Datei `/etc/protocols` zu finden ist, kann übergeben werden. Außerdem kann das Schlüsselwort `tcp/udp` verwendet werden, wenn sowohl TCP als auch UDP von der Regel betroffen sein sollen. Dieses Schlüsselwort wurde eingeführt, um Duplikate sonst identischer Regeln zu vermeiden.

30.5.11.6. SRC_ADDR/DST_ADDR

Das Schlüsselwort `all` ist ein Synonym für „from any to any“ ohne weitere Auswahlkriterien.

`from src to dst` : Die Schlüsselwörter `from` und `to` dienen zur Angabe von Quelle und Ziel in Form von IP-Adressen oder -Bereichen. Innerhalb einer Regel muss immer beides angegeben werden. Statt einer Adresse kann auch das Schlüsselwort `any` übergeben werden, das für jede beliebige IP-Adresse steht. Zum Beispiel: `from any to any` oder `from 0.0.0.0/0 to any` oder `from any to 0.0.0.0/0` oder `from 0.0.0.0 to any` oder `from any to 0.0.0.0` bedeuten alle das Gleiche.

IP-Bereiche können nur in der CIDR-Notation angegeben werden. Der Port net-mgmt/ipcalc hilft Ihnen bei der Berechnung der richtigen Angaben. Weiterführende Informationen zu CIDR finden Sie auf der Webseite von `ipcalc`.

30.5.11.7. PORT

Wenn ein Port als Auswahlkriterium übergeben wurde, bei Quelle und/oder Ziel, wird er nur bei TCP und UDP Paketen verwendet. Angegeben werden kann entweder die Portnummer oder der Dienstname aus `/etc/services`. Die Verwendung der Portoption mit dem `to`-Objekt ist verbindlich für die Verwendung der modernisierten Regel-Prozess-Logik. Ein Beispiel für die Filterung Paketen von allen Quell-IPs mit beliebiger Portnummer auf beliebige Ziel-IPs mit der Portnummer 80 (dem `www`-Port): `from any to any port = 80` .

Einfache Portvergleiche können auf verschiedenen Wegen erfolgen. Mehrere Vergleichsoperatoren stehen dafür zur Verfügung. Genauso können Bereiche angegeben werden.

port "=" | "!=" | "<" | ">" | "<=" | ">=" | "eq" | "ne" | "lt" | "gt" | "le" | "ge".

Um einen Bereich anzugeben: port "<>" | "><"

Warnung

Genau wie die Trefferspezifikation für Quelle und Ziel sind auch die beiden folgenden Parameter obligatorisch bei der Verwendung der modernen Regel-Prozess-Logik.

30.5.11.8. TCP_FLAG

Flags spielen nur beim Filtern von TCP eine Rolle. Die Buchstaben entsprechen jeweils einem möglichen Flag, dass in den Kopfdaten der TCP-Pakete geprueft werden soll.

Die moderne Regel-Prozess-Logik verwendet den Parameter `flags S` um eine Anfrage zum Start einer TCP-Session zu identifizieren.

30.5.11.9. STATEFUL

`keep state` zeigt bei einer Passage-Regel an, dass für alle Pakete, die die Selektion erfolgreich durchlaufen, `Stateful Filtering` eingerichtet werden soll.

Anmerkung

Diese Option ist obligatorisch für die Verwendung der modernen Prozess-Regel-Logik.

30.5.12. Stateful Filtering

Stateful filtering treats traffic as a bi-directional exchange of packets comprising a session conversation. When activated, keep-state dynamically generates internal rules for each anticipated packet being exchanged during the bi-directional session conversation. It has sufficient matching capabilities to determine if the session conversation between the originating sender and the destination are following the valid procedure of bi-directional packet exchange. Any packets that do not properly fit the session conversation template are automatically rejected as impostors.

Keep state will also allow ICMP packets related to a TCP or UDP session through. So if you get ICMP type 3 code 4 in response to some web surfing allowed out by a keep state rule, they will be automatically allowed in. Any packet that IPF can be certain is part of an active session, even if it is a different protocol, will be let in.

What happens is:

Packets destined to go out through the interface connected to the public Internet are first checked against the dynamic state table. If the packet matches the next expected packet comprising an active session conversation, then it exits the firewall and the state of the session conversation flow is updated in the dynamic state table. Packets that do not belong to an already active session, are simply checked against the outbound ruleset.

Packets coming in from the interface connected to the public Internet are first checked against the dynamic state table. If the packet matches the next expected packet comprising an active session conversation, then it exits the firewall and the state of the session conversation flow is updated in the dynamic state table. Packets that do not belong to an already active session, are simply checked against the inbound ruleset.

When the conversation completes it is removed from the dynamic state table.

Stateful filtering allows you to focus on blocking/passing new sessions. If the new session is passed, all its subsequent packets will be allowed through automatically and any impostors automatically rejected. If a new session is blocked, none of its subsequent packets will be allowed through. Stateful filtering has technically advanced

matching abilities capable of defending against the flood of different attack methods currently employed by attackers.

30.5.13. Inclusive Ruleset Example

The following ruleset is an example of how to code a very secure inclusive type of firewall. An inclusive firewall only allows services matching `pass` rules through, and blocks all others by default. Firewalls intended to protect other machines, also called „network firewalls“, should have at least two interfaces, which are generally configured to trust one side (the LAN) and not the other (the public Internet). Alternatively, a firewall might be configured to protect only the system it is running on—this is called a „host based firewall“, and is particularly appropriate for servers on an untrusted network.

All UNIX® flavored systems including FreeBSD are designed to use interface `lo0` and IP address `127.0.0.1` for internal communication within the operating system. The firewall rules must contain rules to allow free unmolested movement of these special internally used packets.

The interface which faces the public Internet is the one to place the rules that authorize and control access of the outbound and inbound connections. This can be your user PPP `tun0` interface or your NIC that is connected to your DSL or cable modem.

In cases where one or more NICs are cabled to private network segments, those interfaces may require rules to allow packets originating from those LAN interfaces transit to each other and/or to the outside (Internet).

The rules should be organized into three major sections: first trusted interfaces, then the public interface outbound, and last the public untrusted interface inbound.

The rules in each of the public interface sections should have the most frequently matched rules placed before less commonly matched rules, with the last rule in the section blocking and logging all packets on that interface and direction.

The Outbound section in the following ruleset only contains `pass` rules which contain selection values that uniquely identify the service that is authorized for public Internet access. All the rules have the `quick`, `on`, `proto`, `port`, and `keep state` options set. The `proto tcp` rules have the `flag` option included to identify the session start request as the triggering packet to activate the stateful facility.

The Inbound section has all the blocking of undesirable packets first, for two different reasons. The first is that malicious packets may be partial matches for legitimate traffic. These packets have to be discarded rather than allowed in, based on their partial matches against `allow` rules. The second reason is that known and uninteresting rejects may be blocked silently, rather than being caught and logged by the last rules in the section. The final rule in each section, blocks and logs all packets and can be used to create the legal evidence needed to prosecute the people who are attacking your system.

Another thing that should be taken care of, is to ensure there is no response returned for any of the undesirable traffic. Invalid packets should just get dropped and vanish. This way the attacker has no knowledge if his packets have reached your system. The less the attackers can learn about your system, the more time they must invest before actually doing something bad. Rules that include a `log first` option, will only log the event the first time they are triggered. This option is included in the sample `nmap OS fingerprint` rule. The security/nmap utility is commonly used by attackers who attempt to identify the operating system of your server.

Any time there are logged messages on a rule with the `log first` option, an `ipfstat -hio` command should be executed to evaluate how many times the rule has actually matched. Large number of matches usually indicate that the system is being flooded (i.e.: under attack).

The `/etc/services` file may be used to lookup unknown port numbers. Alternatively, visit `http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers` and do a port number lookup to find the purpose of a particular port number.

Check out this link for port numbers used by Trojans `http://www.sans.org/security-resources/idfaq/oddports.php`.

The following ruleset creates a complete and very secure `inclusive` type of firewall ruleset that has been tested on production systems. It can be easily modified for your own system. Just comment out any `pass` rules for services that should not be authorized.

To avoid logging unwanted messages, just add a `block` rule in the inbound section.

The `dc0` interface name has to be changed in every rule to the real interface name of the NIC card that connects your system to the public Internet. For user PPP it would be `tun0`.

Add the following statements to `/etc/ipf.rules`:

```
########################################################################
# No restrictions on Inside LAN Interface for private network
# Not needed unless you have LAN
########################################################################

#pass out quick on xl0 all
#pass in quick on xl0 all

########################################################################
# No restrictions on Loopback Interface
########################################################################
pass in quick on lo0 all
pass out quick on lo0 all

########################################################################
# Interface facing Public Internet (Outbound Section)
# Match session start requests originating from behind the
# firewall on the private network
# or from this gateway server destined for the public Internet.
########################################################################

# Allow out access to my ISP's Domain name server.
# xxx must be the IP address of your ISP's DNS.
# Dup these lines if your ISP has more than one DNS server
# Get the IP addresses from /etc/resolv.conf file
pass out quick on dc0 proto tcp from any to xxx port = 53 flags S keep state
pass out quick on dc0 proto udp from any to xxx port = 53 keep state

# Allow out access to my ISP's DHCP server for cable or DSL networks.
# This rule is not needed for 'user ppp' type connection to the
# public Internet, so you can delete this whole group.
# Use the following rule and check log for IP address.
# Then put IP address in commented out rule & delete first rule
pass out log quick on dc0 proto udp from any to any port = 67 keep state
#pass out quick on dc0 proto udp from any to z.z.z.z port = 67 keep state


# Allow out non-secure standard www function
pass out quick on dc0 proto tcp from any to any port = 80 flags S keep state

# Allow out secure www function https over TLS SSL
pass out quick on dc0 proto tcp from any to any port = 443 flags S keep state

# Allow out send & get email function
pass out quick on dc0 proto tcp from any to any port = 110 flags S keep state
pass out quick on dc0 proto tcp from any to any port = 25 flags S keep state

# Allow out Time
pass out quick on dc0 proto tcp from any to any port = 37 flags S keep state

# Allow out nntp news
pass out quick on dc0 proto tcp from any to any port = 119 flags S keep state

# Allow out gateway & LAN users' non-secure FTP ( both passive & active modes)
# This function uses the IPNAT built in FTP proxy function coded in
```

```
# the nat rules file to make this single rule function correctly.
# If you want to use the pkg_add command to install application packages
# on your gateway system you need this rule.
pass out quick on dc0 proto tcp from any to any port = 21 flags S keep state

# Allow out ssh/sftp/scp (telnet/rlogin/FTP replacements)
# This function is using SSH (secure shell)
pass out quick on dc0 proto tcp from any to any port = 22 flags S keep state

# Allow out insecure Telnet
pass out quick on dc0 proto tcp from any to any port = 23 flags S keep state

# Allow out FreeBSD CVSup
pass out quick on dc0 proto tcp from any to any port = 5999 flags S keep state

# Allow out ping to public Internet
pass out quick on dc0 proto icmp from any to any icmp-type 8 keep state

# Allow out whois from LAN to public Internet
pass out quick on dc0 proto tcp from any to any port = 43 flags S keep state

# Block and log only the first occurrence of everything
# else that's trying to get out.
# This rule implements the default block
block out log first quick on dc0 all

################################################################
# Interface facing Public Internet (Inbound Section)
# Match packets originating from the public Internet
# destined for this gateway server or the private network.
################################################################

# Block all inbound traffic from non-routable or reserved address spaces
block in quick on dc0 from 192.168.0.0/16 to any    #RFC 1918 private IP
block in quick on dc0 from 172.16.0.0/12 to any     #RFC 1918 private IP
block in quick on dc0 from 10.0.0.0/8 to any        #RFC 1918 private IP
block in quick on dc0 from 127.0.0.0/8 to any       #loopback
block in quick on dc0 from 0.0.0.0/8 to any         #loopback
block in quick on dc0 from 169.254.0.0/16 to any    #DHCP auto-config
block in quick on dc0 from 192.0.2.0/24 to any      #reserved for docs
block in quick on dc0 from 204.152.64.0/23 to any   #Sun cluster interconnect
block in quick on dc0 from 224.0.0.0/3 to any       #Class D & E multicast

##### Block a bunch of different nasty things. ############
# That I do not want to see in the log

# Block frags
block in quick on dc0 all with frags

# Block short tcp packets
block in quick on dc0 proto tcp all with short

# block source routed packets
block in quick on dc0 all with opt lsrr
block in quick on dc0 all with opt ssrr

# Block nmap OS fingerprint attempts
# Log first occurrence of these so I can get their IP address
block in log first quick on dc0 proto tcp from any to any flags FUP

# Block anything with special options
block in quick on dc0 all with ipopts

# Block public pings
block in quick on dc0 proto icmp all icmp-type 8
```

```
# Block ident
block in quick on dc0 proto tcp from any to any port = 113

# Block all Netbios service. 137=name, 138=datagram, 139=session
# Netbios is MS/Windows sharing services.
# Block MS/Windows hosts2 name server requests 81
block in log first quick on dc0 proto tcp/udp from any to any port = 137
block in log first quick on dc0 proto tcp/udp from any to any port = 138
block in log first quick on dc0 proto tcp/udp from any to any port = 139
block in log first quick on dc0 proto tcp/udp from any to any port = 81

# Allow traffic in from ISP's DHCP server. This rule must contain
# the IP address of your ISP's DHCP server as it's the only
# authorized source to send this packet type. Only necessary for
# cable or DSL configurations. This rule is not needed for
# 'user ppp' type connection to the public Internet.
# This is the same IP address you captured and
# used in the outbound section.
pass in quick on dc0 proto udp from z.z.z.z to any port = 68 keep state

# Allow in standard www function because I have apache server
pass in quick on dc0 proto tcp from any to any port = 80 flags S keep state

# Allow in non-secure Telnet session from public Internet
# labeled non-secure because ID/PW passed over public Internet as clear text.
# Delete this sample group if you do not have telnet server enabled.
#pass in quick on dc0 proto tcp from any to any port = 23 flags S keep state

# Allow in secure FTP, Telnet, and SCP from public Internet
# This function is using SSH (secure shell)
pass in quick on dc0 proto tcp from any to any port = 22 flags S keep state

# Block and log only first occurrence of all remaining traffic
# coming into the firewall. The logging of only the first
# occurrence avoids filling up disk with Denial of Service logs.
# This rule implements the default block.
block in log first quick on dc0 all
################### End of rules file ####################################
```

30.5.14. NAT

NAT stands for *Network Address Translation*. To those familiar with Linux®, this concept is called IP Masquerading; NAT and IP Masquerading are the same thing. One of the many things the IPF NAT function enables is the ability to have a private Local Area Network (LAN) behind the firewall sharing a single ISP assigned IP address on the public Internet.

You may ask why would someone want to do this. ISPs normally assign a dynamic IP address to their non-commercial users. Dynamic means that the IP address can be different each time you dial in and log on to your ISP, or for cable and DSL modem users, when the modem is power cycled. This dynamic IP address is used to identify your system to the public Internet.

Now lets say you have five PCs at home and each one needs Internet access. You would have to pay your ISP for an individual Internet account for each PC and have five phone lines.

With NAT only a single account is needed with your ISP. The other four PCs may then be cabled to a switch and the switch to the NIC in your FreeBSD system which is going to service your LAN as a gateway. NAT will automatically translate the private LAN IP address for each separate PC on the LAN to the single public IP address as it exits the firewall bound for the public Internet. It also does the reverse translation for returning packets.

There is a special range of IP addresses reserved for NATed private LANs. According to RFC 1918, the following IP ranges may be used for private nets which will never be routed directly to the public Internet:

Start IP `10.0.0.0`	-	Ending IP `10.255.255.255`

Start IP `172.16.0.0`	-	Ending IP `172.31.255.255`
Start IP `192.168.0.0`	-	Ending IP `192.168.255.255`

30.5.15. IPNAT

NAT rules are loaded by using the `ipnat` command. Typically the NAT rules are stored in `/etc/ipnat.rules`. See ipnat(1) for details.

When changing the NAT rules after NAT has been started, make your changes to the file containing the NAT rules, then run the `ipnat` command with the `-CF` flags to delete the internal in use NAT rules and flush the contents of the translation table of all active entries.

To reload the NAT rules issue a command like this:

```
# ipnat -CF -f /etc/ipnat.rules
```

To display some statistics about your NAT, use this command:

```
# ipnat -s
```

To list the NAT table's current mappings, use this command:

```
# ipnat -l
```

To turn verbose mode on, and display information relating to rule processing and active rules/table entries:

```
# ipnat -v
```

30.5.16. IPNAT Rules

NAT rules are very flexible and can accomplish many different things to fit the needs of commercial and home users.

The rule syntax presented here has been simplified to what is most commonly used in a non-commercial environment. For a complete rule syntax description see the ipnat(5) manual page.

The syntax for a NAT rule looks something like this:

```
map IF LAN_IP_RANGE  -> PUBLIC_ADDRESS
```

The keyword `map` starts the rule.

Replace *`IF`* with the external interface.

The *`LAN_IP_RANGE`* is what your internal clients use for IP Addressing, usually this is something like `192.168.1.0/24` .

The *`PUBLIC_ADDRESS`* can either be the external IP address or the special keyword `0/32`, which means to use the IP address assigned to *`IF`*.

30.5.17. How NAT works

A packet arrives at the firewall from the LAN with a public destination. It passes through the outbound filter rules, NAT gets its turn at the packet and applies its rules top down, first matching rule wins. NAT tests each of its rules against the packet's interface name and source IP address. When a packet's interface name matches a NAT rule then the source IP address (i.e.: private LAN IP address) of the packet is checked to see if it falls within the IP address range specified to the left of the arrow symbol on the NAT rule. On a match the packet has its source IP address rewritten with the public IP address obtained by the `0/32` keyword. NAT posts an entry in its internal NAT table so when the packet returns from the public Internet it can be mapped back to its original private IP address and then passed to the filter rules for processing.

30.5.18. Enabling IPNAT

To enable IPNAT add these statements to `/etc/rc.conf` .

To enable your machine to route traffic between interfaces:

```
gateway_enable="YES"
```

To start IPNAT automatically each time:

```
ipnat_enable="YES"
```

To specify where to load the IPNAT rules from:

```
ipnat_rules="/etc/ipnat.rules"
```

30.5.19. NAT for a very large LAN

For networks that have large numbers of PC's on the LAN or networks with more than a single LAN, the process of funneling all those private IP addresses into a single public IP address becomes a resource problem that may cause problems with the same port numbers being used many times across many NATed LAN PC's, causing collisions. There are two ways to relieve this resource problem.

30.5.19.1. Assigning Ports to Use

A normal NAT rule would look like:

```
map dc0 192.168.1.0/24 -> 0/32
```

In the above rule the packet's source port is unchanged as the packet passes through IPNAT. By adding the `portmap` keyword, IPNAT can be directed to only use source ports in the specified range. For example the following rule will tell IPNAT to modify the source port to be within the range shown:

```
map dc0 192.168.1.0/24 -> 0/32 portmap tcp/udp 20000:60000
```

Additionally we can make things even easier by using the `auto` keyword to tell IPNAT to determine by itself which ports are available to use:

```
map dc0 192.168.1.0/24 -> 0/32 portmap tcp/udp auto
```

30.5.19.2. Using a Pool of Public Addresses

In very large LANs there comes a point where there are just too many LAN addresses to fit into a single public address. If a block of public IP addresses is available, these addresses can be used as a „pool“, and IPNAT may pick one of the public IP addresses as packet-addresses are mapped on their way out.

For example, instead of mapping all packets through a single public IP address, as in:

```
map dc0 192.168.1.0/24 -> 204.134.75.1
```

A range of public IP addresses can be specified either with a netmask:

```
map dc0 192.168.1.0/24 -> 204.134.75.0/255.255.255.0
```

or using CIDR notation:

```
map dc0 192.168.1.0/24 -> 204.134.75.0/24
```

30.5.20. Port Redirection

A very common practice is to have a web server, email server, database server and DNS server each segregated to a different PC on the LAN. In this case the traffic from these servers still have to be NATed, but there has to be some way to direct the inbound traffic to the correct LAN PCs. IPNAT has the redirection facilities of NAT to solve this

problem. For example, assuming a web server operating on LAN address `10.0.10.25` and using a single public IP address of `20.20.20.5` the rule would be coded as follows:

```
rdr dc0 20.20.20.5/32 port 80 -> 10.0.10.25 port 80
```

or:

```
rdr dc0 0.0.0.0/0 port 80 -> 10.0.10.25 port 80
```

or for a LAN DNS Server on LAN address of `10.0.10.33` that needs to receive public DNS requests:

```
rdr dc0 20.20.20.5/32 port 53 -> 10.0.10.33 port 53 udp
```

30.5.21. FTP and NAT

FTP is a dinosaur left over from the time before the Internet as it is known today, when research universities were leased lined together and FTP was used to share files among research Scientists. This was a time when data security was not a consideration. Over the years the FTP protocol became buried into the backbone of the emerging Internet and its username and password being sent in clear text was never changed to address new security concerns. FTP has two flavors, it can run in active mode or passive mode. The difference is in how the data channel is acquired. Passive mode is more secure as the data channel is acquired by the ordinal ftp session requester. For a real good explanation of FTP and the different modes see `http://www.slacksite.com/other/ftp.html` .

30.5.21.1. IPNAT Rules

IPNAT has a special built in FTP proxy option which can be specified on the NAT map rule. It can monitor all outbound packet traffic for FTP active or passive start session requests and dynamically create temporary filter rules containing only the port number really in use for the data channel. This eliminates the security risk FTP normally exposes the firewall to from having large ranges of high order port numbers open.

This rule will handle all the traffic for the internal LAN:

```
map dc0 10.0.10.0/29 -> 0/32 proxy port 21 ftp/tcp
```

This rule handles the FTP traffic from the gateway:

```
map dc0 0.0.0.0/0 -> 0/32 proxy port 21 ftp/tcp
```

This rule handles all non-FTP traffic from the internal LAN:

```
map dc0 10.0.10.0/29 -> 0/32
```

The FTP map rule goes before our regular map rule. All packets are tested against the first rule from the top. Matches on interface name, then private LAN source IP address, and then is it a FTP packet. If all that matches then the special FTP proxy creates temp filter rules to let the FTP session packets pass in and out, in addition to also NATing the FTP packets. All LAN packets that are not FTP do not match the first rule and fall through to the third rule and are tested, matching on interface and source IP, then are NATed.

30.5.21.2. IPNAT FTP Filter Rules

Only one filter rule is needed for FTP if the NAT FTP proxy is used.

Without the FTP Proxy, the following three rules will be needed:

```
# Allow out LAN PC client FTP to public Internet
# Active and passive modes
pass out quick on rl0 proto tcp from any to any port = 21 flags S keep state

# Allow out passive mode data channel high order port numbers
pass out quick on rl0 proto tcp from any to any port > 1024 flags S keep state

# Active mode let data channel in from FTP server
pass in quick on rl0 proto tcp from any to any port = 20 flags S keep state
```

30.6. IPFW

Die *IPFIREWALL* (IPFW) ist eine vom FreeBSD Project gesponserte Software-Firewall. Sie wurde und wird freiwillig von Mitgliedern des FreeBSD Projects geschrieben und gewartet. Mit zustandslosen Regeln und einer Grammatik für Regeln implementiert sie eine sogenannte „Einfache Zustandsgesteuerte Logik“.

Die Standardinstallation von IPFW enthält einen beispielhaften Regelsatz (`/etc/rc.firewall` und `/etc/rc.firewall6`). Dieser ist eher einfach gehalten; es ist nicht zu erwarten, dass dieser ohne Modifikationen angewandt werden kann. Dieses Beispiel nutzt keine zustandsorientierte Filterung, von der allerdings die meisten Installationen profitieren sollten. Deshalb wird sich dieser Abschnitt auch nicht auf diese Beispiele stützen.

Die zustandslose IPFW Regel-Syntax ist durch ihre technisch ausgefeilten Selektions-Fähigkeiten, die über das Niveau der gebrächlichen Firewall-Installationsprogramme weit hinausgehen, sehr mächtig. IPFW richtet sich an professionelle oder technisch versierte Nutzer mit weitergehenden Anforderungen an die Paket-Auswahl. Um die Ausdrucksstärke der IPFW zu nutzen, ist sehr detailliertes Wissen über die Art und Weise, wie verschiedene Protokolle ihre jeweilige Paket-Header-Information erzeugen und nutzen, erforderlich. Im Rahmen dieses Abschnitts ist es nicht möglich, auf alle diese Punkte detailliert einzugehen.

IPFW besteht aus sieben Komponenten: Hauptbestandteil ist der Kernel Firewall Filter, ein Regel-Prozessor mit integrierter Paket-Buchführung. Außerdem enthalten ist eine Komponente zur Protokollierung der Aktivitäten der Firewall (also ein Logfunktion). Weiters besteht die IPFW aus einer Regel zum Umleiten des Datenverkehrs (`divert`), die auch Network Address Translation (NAT) unterstützt. Die restlichen Bestandteile dienen verschiedenen fortgeschrittenen Zwecken. Der *Traffic Shaper* dummynet(4) gestattet es beispielsweise, den Datenverkehr zu lenken, während die `fwd`-Regel zum Weiterleiten von Datenpaketen dient. Komplettiert wird IPFW durch Funktionen zum Überbrücken von Netzwerkgrenzen (*Bridge*-Funktion) sowie *ipstealth*, das es gestattet, bridging-Funktionen durchzuführen, ohne dabei das TTL-Feld im IP-Paket zu erhöhen. IPFW unterstützt IPv4 und IPv6.

30.6.1. IPFW aktivieren

IPFW ist in der FreeBSD-Installation standardmäßig als ein zur Laufzeit ladbares Kernelmodul enthalten, das vom System automatisch geladen wird, wenn in der Datei `rc.conf` die Option `firewall_enable="YES"` gesetzt wird. Es ist daher nicht notwendig, IPFW statisch in den Kernel zu kompilieren.

Während des Systemstart wird bei gesetzter Option `firewall_enable="YES"` (in der Datei `rc.conf`) folgende Nachricht ausgegeben:

```
ipfw2 initialized, divert disabled, rule-based forwarding disabled, default to deny, ↺
logging disabled
```

Das Kernelmodul hat eine Protokollierungsfunktion. Um diese zu aktivieren und einen Schwellwert für die Protokollierung zu definieren, ist es erforderlich, folgende Ausdrücke der `/etc/sysctl.conf` hinzuzufügen:

```
net.inet.ip.fw.verbose=1
net.inet.ip.fw.verbose_limit=5
```

30.6.2. Kerneloptionen

Es ist für die Aktivierung von IPFW nicht zwingend erforderlich, die folgenden Optionen in den Kernel zu kompilieren. Es wird hier lediglich als Hintergrundinformation aufgeführt.

```
options IPFIREWALL
```

Diese Option aktiviert IPFW als Bestandteil des Kernels.

```
options IPFIREWALL_VERBOSE
```

Diese Option aktiviert die Funktion, alle Pakete, die durch IPFW verarbeitet werden und bei denen das Schlüsselwort `log` gesetzt ist, zu protokollieren.

```
options IPFIREWALL_VERBOSE_LIMIT=5
```

Diese Option limitiert die Anzahl der durch syslogd(8) protokollierten Pakete auf das angegebene Maximum. Sie wird in feindlichen Umgebungen verwandt, in denen die Protokollierung der Firewall-Aktivität erwünscht ist. Dadurch wird ein möglicher Denial-of-Service-Angriff durch Überflutung von syslogd(8) verhindert.

```
options IPFIREWALL_DEFAULT_TO_ACCEPT
```

Diese Option erlaubt allen Paketen, die Firewall zu passieren. Diese Einstellung kann beispielsweise bei der ersten Konfiguration der Firewall hilfreich sein.

```
options IPDIVERT
```

Dies aktiviert die Nutzung der NAT-Funktionalität.

Anmerkung

Die Firewall wird alle eingehenden oder ausgehenden Pakete blockieren, wenn entweder die Kernel-Option `IPFIREWALL_DEFAULT_TO_ACCEPT` fehlt oder aber keine Regel, die die betreffenden Verbindungen explizit gestattet, existiert. Dies enstpricht im Wesentlichen der Einstellung „default to deny“

30.6.3. Optionen in `/etc/rc.conf`

Der Eintrag

```
firewall_enable="YES"
```

aktiviert die Firewall während des Systemstarts.

Die Auswahl einer für FreeBSD verfügbaren Firewall erfolgt durch einen entsprechenden Eintrag in der Datei `/etc/rc.firewall`, durch den der Firewalltyp festgelegt wird.

```
firewall_type="open"
```

Konkret sind folgende Einträge erlaubt:

- `open` — gestattet jeglichen Datenverkehr
- `client` — schützt nur die jeweilige Maschine (Client/Mandant)
- `simple` — schützt das gesamte Netzwerk
- `closed` — unterbindet jeglichen IP-Datenverkehr mit Ausnahme des Verkehrs über die Loopback-Schnittstelle.
- `UNKNOWN` — deaktiviert das Laden von Firewallregeln
- `filename` — absoluter Pfad zu einer Datei, in der die Firewallregeln definiert sind

Angepasste Regeln für ipfw(8) können auf zwei verschiedene Arten geladen werden. Einerseits kann man durch die Variable `firewall_type` den absoluten Pfad der Datei angeben, welche die *Firewallregeln* (ohne weitere Optionen) für ipfw(8) enthält. Ein einfaches Beispiel für einen Regelsatz, der jeglichen eingehenden und ausgehenden Datenverkehr blockiert, könnte beispielsweise so aussehen:

```
add deny in add deny out
```

Andererseits ist es möglich, den Wert der `firewall_type`-Variable mit dem absoluten Pfad einer Datei zu belegen, die (als ausführbares Skript) die ipfw(8)-Kommandos enthält, die beim Booten ausgeführt werden sollen. Ein gültiges Skript (das die gleiche Funktion hat wie die Zeile im letzten Beispiel) könnte beispielsweise so aussehen:

```
#!/bin/sh

ipfw -q flush

ipfw add deny in
ipfw add deny out
```

Anmerkung

Wenn die Variable `firewall_type` entweder auf `client` oder `simple` gesetzt ist, sollten die Standardregeln in der Datei `/etc/rc.firewall` geprüft und an die Konfiguration der gegebenen Maschine angepasst werden. Beachten Sie dabei bitte, dass die Beispiele dieses Kapitels davon ausgehen, dass das `firewall_script` auf `/etc/ipfw.rules` gesetzt ist.

Das Logging wird durch folgenden Eintrag aktiviert:

```
firewall_logging="YES"
```

Warnung

Die Variable `firewall_logging` definiert lediglich die sysctl-Variable als `net.inet.ip.fw.verbose = 1` (lesen Sie dazu bitte auch den Abschnitt Abschnitt 30.6.1, „IPFW aktivieren“ des Handbuchs). Es gibt keine `rc.conf`-Variable, mit der man Protokollierungsschwellen setzen könnte. Dies kann lediglich über sysctl(8) geschehen, wobei Sie in der Datei `/etc/sysctl.conf` nur Werte > 1 angeben sollten:

```
net.inet.ip.fw.verbose_limit=5
```

Sollte Ihre Maschinen als Gateway fungieren (also mittels natd(8) *Network Address Translation* (NAT) durchführen), finden Sie weitere Optionen in `/etc/rc.conf` .

30.6.4. Der Befehl IPFW

Mit ipfw(8) ist es möglich, im laufenden Betrieb einzelne Regeln hinzuzufügen oder zu entfernen. Problematisch ist allerdings, dass diese Änderungen verloren gehen, wenn das System neu gestartet wird. Daher ist es empfehlenswert, eigene Regeln in einer Datei zu definieren und diese zu laden, um die Regeln der Firewall im laufenden Betrieb anzupassen.

ipfw(8) ist jedoch hilfreich, um die Regeln der laufenden Firewall in der Konsole auszugeben. IPFW erzeugt dynamisch einen Zähler, der jedes Paket, auf das eine Regel zutrifft, zählt. Dadurch wird es möglich, die Funktion einer Regel zu überprüfen.

Eine sequentielle Liste aller Regeln erhalten Sie mit:

```
# ipfw list
```

Eine Liste aller Regeln inklusive des letzten Treffers erhalten Sie durch den folgenden Befehl:

```
# ipfw -t list
```

Um eine Liste aller Regeln inklusive der Anzahl der Pakete, die von einer Regel gefiltert wurden, zu erhalten, geben Sie den folgenden Befehl ein:

```
# ipfw -a list
```

Eine Liste, die zusätzlich allen dynamischen Regeln enthält, erhalten Sie mit:

```
# ipfw -d list
```

Um diese Liste um alle „abgelaufenen“ Regeln zu erweitern, ädern Sie diesen Befehl wie folgt ab:

```
# ipfw -d -e list
```

Alle Zähler auf Null zurücksetzen:

```
# ipfw zero
```

Es ist auch möglich, einen spezifischen Zähler auszuwählen und zurückzusetzen:

```
# ipfw zero NUM
```

30.6.5. IPFW-Regeln

Ein Regelwerk ist eine Menge von IPFW-Regeln, die in Abhängigkeit von bestimmten Paketeigenschaften Pakete entweder passieren lassen oder abweisen. Der zustandshafte bidirektionale Transfer von Paketen zwischen Rechnern wird als Sitzung bezeichnet. Das Regelwerk der Firewall verarbeitet sowohl ankommende Pakete (aus dem öffentlichen Internet) als auch Pakete, deren Ursprung in einer Antwort des Systems auf empfangene Pakete liegt. Jeder TCP/IP-Dienst (wie telnet, www, mail) ist durch sein Protokoll und durch den priveligierten (eingehenden) Port definiert. An einen spezifischen Dienst adressierte Pakete kommen von einer Quelladresse und einem unprivilegierten (high order) Port. Sie adressieren den spezifischen Port des Dienstes an der Zieladresse. Alle weiter oben aufgeführten Parameter (also Ports und Adressen) können als Selektionskriterium zur Erzeugung von Regeln genutzt werden, die ein Passieren der Firewall für oder ein Blockieren von Diensten bewirken.

Wenn ein Paket die Firewall „betritt“, also von der Firewall geprüft und verarbeitet wird, wird die erste Regel des Regelwerkes auf das Paket angewandt. Auf diese Weise wird in aufsteigender Reihenfolge der Regelnummer mit allen weiteren Regeln verfahren. Falls die Selektionsparameter einer Regel auf ein Paket zutreffen, wird das Aktionsfeld der Regel ausgeführt und die Prüfung des Pakets beendet, nachfolgende Regeln werden also nicht mehr geprüft. Diese Suchmethode wird als „erster Treffer gewinnt“ bezeichnet. Falls keine Regel auf das betreffende Paket zutrifft, wird die obligatorische IPFW-Rückfallregel (also Regel 65535) angewendet und das Paket wird ohne Rückantwort verworfen.

Anmerkung

Die Prüfung der Regeln wird nach Treffern von mit `count`, `skipto` und `tee` parametrisierten Regeln ungeachtet des „erster Treffer gewinnt“-Prinzips weiter fortgeführt.

Die Anweisungen basieren auf der Nutzung von Regeln mit den zustandsgesteuerten Optionen `keep`, `state`, `limit`, `in` und `out`. Diese bilden die Basis für die Spezifikation von Firewallregeln.

Warnung

Bei der Arbeit mit Firewallregeln ist Vorsicht geboten. Es ist sehr einfach, sich selbst auszuschließen.

30.6.5.1. Syntax der Firewallregeln

Mit der in diesem Abschnitt dargestellten Syntax der Regeln kann ein Standardregelsatz für eine „einschließende“ Firewall erstellt werden. Für eine vollständige Beschreibung der Regelsyntax lesen Sie bitte die Manualpage ipfw(8)

Regelausdrücke werden „von links nach rechts“ ausgewertet. Schlüsselwörter werden in fetter Schrift dargestellt. Manche Schlüsselworte beinhalten Unteroptionen, die wiederum selbst aus Schlüsselworten samt Optionen bestehen können.

Kommentare sind mit einen führenden Doppelkreuz (`#`) ausgezeichnet. Sie können am Ende einer Regel oder in einzelnen, separaten Zeilen stehen. Leerzeilen werden ignoriert.

`CMD RULE_NUMBER ACTION LOGGING SELECTION STATEFUL`

30.6.5.1.1. CMD

Jede neue Regel benötigt das Präfix `add`, um die Regel der internen Tabelle hinzuzufügen.

30.6.5.1.2. RULE_NUMBER

Zu jeder Regel gehört eine Regelnummer zwischen 1 und 65535.

30.6.5.1.3. ACTION

Eine Regel kann mit einer der vier folgenden Aktionen verbunden sein, die ausgeführt werden, wenn ein Paket den Selektionskriterien der Regel entspricht.

`allow | accept | pass | permit`

Alle diese Aktionen bewirken das Gleiche: Pakete, die den Selektionskriterien der Regel entsprechen, verlassen den Regelprüfungsabschnitt der Firewall und die Regelprüfung wird beendet.

`check-state`

Diese Aktion prüft das Paket gegen die Regeln aus den dynamischen Regeltabellen. Trifft ein Selektionskriterium zu, wird die zur dynamischen Regel gehörende Aktion ausgeführt. Anderenfalls wird gegen die nächste Regel geprüft. Die `check-state` -Regel selbst hat kein Selektionskriterium. Sollte eine `check-state` -Regel im Regelwerk fehlen, wird gegen die erste `keep-state` - oder `limit` -Regel in den dynamischen Regeln geprüft.

`deny | drop`

Beide Schlüsselworte bewirken dieselbe Aktion: Ein Paket, dass die Selektionskriterien der Regel erfüllt, wird verworfen und die Regelprüfung wird beendet.

30.6.5.1.4. Protokollierung

`log` oder *`logamount`*

Erfüllt ein Paket die Selektionskriterien mit dem Schlüsselwort `log`, wird dies von syslogd(8) mit der Annotation SECURITY protokolliert. Dies erfolgt allerdings nur, wenn die Anzahl der protokollierten Pakete der betreffenden Regel die im `logamount`-Parameter definierte Schwelle nicht übersteigt. Ist der Parameter `logamount` nicht definiert, wird diese Grenze aus der `sysctl` -Variable `net.inet.ip.fw.verbose_limit` ermittelt. Ist einer dieser beiden Werte auf „Null“ gesetzt, wird unbegrenzt protokolliert. Wurde hingegen ein definierter Schwellenwert erreicht, wird die Protokollierung deaktiviert. Um sie zu reaktivieren, können Sie entweder den Protokoll- oder den Paketzähler rücksetzen (und zwar über den Befehl `ipfw reset log`).

Anmerkung

Die Protokollierung findet statt, nachdem alle Paketselektionskriterien geprüft und bevor die daraus folgende, endgültige Aktion (`accept` oder `deny`) auf das Paket ausgeführt wird. Die Entscheidung, welche Regel protokolliert werden soll, bleibt Ihnen überlassen.

30.6.5.1.5. Selektion

Die in diesem Abschnitt beschriebenen Schlüsselwörter beschreiben die Attribute eines Pakets, durch die bestimmt wird, ob eine Regel auf ein Paket zutrifft. Die folgenden Attribute dienen der Bestimmung des Protokolls und müssen in der angegebenen Reihenfolge verwendet werden.

udp | tcp | icmp

Weitere in `/etc/protocols` angegebene Protokolle werden ebenfalls erkannt und können daher verwendet werden, um das Protokoll zu definieren, gegen das Pakete geprüft werden. Die Angabe des Protokolls ist verpflichtend.

from src to dst

Die Schlüsselwörter `from` und `to` beziehen sich auf IP-Adressen und definieren sowohl Ursprungs- als auch Zieladresse einer Datenverbindung. Firewallregeln müssen Parameter für den Ursprung *und* das Ziel enthalten. Das Schlüsselwort `any` steht für beliebige IP-Adressen. Bei `me` handelt es sich um ein spezielles Schlüsselwort, das alle IP-Adressen beschreibt, die einer bestimmten Netzwerkschnittstelle Ihres Systems (auf dem die Firewall läuft) zugeordnet sind. Beispiele hierfür sind `from me to any`, `from any to me`, `from 0.0.0.0/0 to any`, `from any to 0.0.0.0/0`, `from 0.0.0.0 to any`, `from any to 0.0.0.0` oder `from me to 0.0.0.0`. IP-Adressen werden entweder in CIDR-Notation oder durch Punkte getrennt mit Suffixen (`192.168.2.101/24`) für die Netzmaske oder als einzelne numerische, durch Punkte getrennte Adressen (`192.168.2.101`) angegeben. Die dafür notwendigen Berechnungen erleichtert der Port net-mgmt/ipcalc. Weiterführende Informationen finden sich auf `http://jodies.de/ipcalc`.

port number

Bei der Verarbeitung von Protokollen wie TCP oder UDP, die Portnummern verwenden, muss die Portnummer des betreffenden Dienstes angegeben werden. Anstelle der Portnummer kann auch der in der Datei `/etc/services` definierte Name des Dienstes angegeben werden.

in | out

Diese Schlüsselwörter beziehen sich auf die Richtung des Datenverkehrs. Jede Regel *muss* eines dieser beiden Schlüsselwörter enthalten.

via IF

Eine Regel mit dem Schlüsselwort `via IF` betrifft nur Pakete, die über die angegebene Schnittstellte geroutet werden (ersetzen Sie `IF` durch den Namen Ihrer Netzwerkschnittstelle). Die Angabe des Schlüsselwortes `via` bewirkt, dass die Netzwerkschnittstelle in die Regelprüfung aufgenommen wird.

setup

Dieses obligatorische Schlüsselwort bezeichnet die Anforderung des Sitzungsstarts für TCP-Pakete.

keep-state

Dieses obligatorische Schlüsselwort bewirkt, dass die Firewall eine dynamische Regel erzeugt, die bidirektionalen Datenverkehr zwischen Ursprungs- und Zieladresse sowie Ursprungs- und Zielport prüft, der das gleiche Protokoll verwendet.

limit {src-addr | src-port | dst-addr | dst-port}

Wird das Schlüsselwort `limit` verwendet, sind nur `N` durch diese Regel definierte Verbindungen erlaubt. Es können dabei ein oder mehrere Ursprungs- und Zieladressen sowie ein oder mehrere Ports angegeben werden. Die Schlüsselwörter `limit` und `keep-state` können nicht in derselben Regel verwendet werden. Die Option `limit` bewirkt dieselbe Zustandsteuerung wie die Option `keep-state`, erweitert diese jedoch um eigene Regeln.

30.6.5.2. Optionen für zustandsgesteuerte Regeln

Eine zustandsgesteuerte Filterung behandelt Datenverkehr als einen bidirektionalen Austausch von Datenpaketen (die eine sogenannte Konversation innerhalb einer Sitzung darstellen). Sie ist in der Lage, zu bestimmen, ob die Konversation von originärem Sender und Empfänger gültigen Prozeduren des bidirektionalen Pakettausches entspricht. Pakete, die dem Muster von Konversationen in Sitzungen nicht folgen, werden automatisch als „Betrüger“ abgelehnt.

Die `check-state`-Option wird verwendet, wo genau innerhalb des IPFW-Regelwerks die Prüfung dynamischer Regeln stattfinden soll. Erfüllt ein Datenpaket die Selektionskriterien der Regel, verlässt das Paket die Firewall. Gleichzeitig wird eine neue dynamische Regel erzeugt, die für das nächste Paket der bidirektionalen Konversation in der Sitzung vorgesehen ist. Falls ein Paket die (dyanmische) Regel nicht erfüllt, wird es gegen die nächste Regel im Regelwerk geprüft.

Dynamische Regeln sind für einem sogenannten *SYN-flood*-Angriff anfällig, bei dem eine riesige Anzahl „schwebender“ dynamischer Regelprüfungungsinstanzen erzeugt wird. Um einem solchen Angriff zu begegnen, wurde in FreeBSD die neue Option `limit` geschaffen. Diese Option begrenzt die Anzahl der gleichzeitig möglichen Sitzungen und/oder Konversationen. Es handelt sich dabei um einen Zähler, der die Anzahl von Instanzen dynamischer Regelprüfungen in Abhängigkeit von einer eindeutigen Urspungs- und Quelladresskombination zählt. Übersteigt der Zähler den durch `limit` definierten Schwellenwert, wird das Paket verworfen.

30.6.5.3. Protokollierung von Firewall-Nachrichten

Die Vorteile einer Protokollierung sind offensichtlich. Sie ermöglicht nach Aktivierung von Regeln zu untersuchen, welche Pakete verworfen wurden, von wo diese stammen und für welche Systeme sie bestimmt waren. Diese Informationen sind sehr nützlich bei der Erkennung eventueller Angriffe sowie bei deren Abwehr.

IPFW protokolliert nur jene Regeln, für die ein Administrator dies explizit aktiviert. Ein Aktivieren der Protolllfunktion führt also nicht dazu, dass automatisch alle Regeln protokolliert werden. Vielmehr entscheidet der Administrator der Firewall, welche Regeln protokolliert werden sollen. Dazu wird die Option `log` für diese Regeln aktiviert. Im Regelfall werden nur `deny`-Regeln protokolliert, beispielsweise die `deny`-Regel für eintreffende ICMP-Nachrichten. Üblicherweise wird die „ipfw default deny everything“-Regel doppelt angelegt. Einmal mit und einmal ohne aktivierte Option `log`. Dadurch erhält man eine Auflistung aller Pakete, auf die keine Regel zutraf.

Protokollierung ist allerdings ein zweischneidiges Schwert, bei mangelnder Vorsicht wird man mit einer enormen Flut von Protokollierungsdaten förmlich *überschwemmt* und belastet zusätzlich die Festplatte des Systems durch rasch wachsende Protokolldateien. DoS-Angriffe, die auf diese Art und Weise Festplatten an die Kapazitätsgrenze treiben, gehören zu den ältesten Angriffen überhaupt. Außerdem werden Protokollnachrichten nicht nur an syslogd(8) geschickt, sondern auch auf einem root-Terminal angezeigt.

Die Kerneloption `IPFIREWALL_VERBOSE_LIMIT=5` begrenzt die Anzahl gleicher Nachrichten an syslogd(8) für eine gegebene Regel auf fünf Nachrichten. Ist diese Option im Kernel aktiviert, wird nach Erreichen der festgelegten Anzahl die Protokollierung einer (sich unmittelbar hintereinander wiederholenden) Nachricht auf den angegebenen Schwellenwert begrenzt, da beispielsweise die Speicherung von 200 gleichen Protokollnachrichten durch syslogd(8) sinnlos ist. Daher werden durch diesen nur füf derartige Nachrichten protokolliert. Alle weiteren derartigen Nachrichten werden nur gezählt und deren Gesamtzahl wird schließlich von syslogd(8) durch folgenden Ausdruck ausgegeben:

```
last message repeated 45 times
```

Alle protokollierten Nachrichten für Datenpakete werden in der Voreinstellung in die Datei `/var/log/security` (die in der Datei `/etc/syslog.conf` definiert wird), geschrieben.

30.6.5.4. Skripte zur Regeldefinition erstellen

Die meisten fortgeschrittenen IPFW-Nutzer erzeugen eine Datei, die die Regeln für die Firewall enthält, um diese als Skript ausführen zu können. Der Hauptvorteil einer derartigen Konfiguration ist es, dass dadurch mehrere

Regeln gleichzeitig geändert und (re-)aktiviert werden können, ohne dass dazu das System neu gestartet werden muss. Dies ist auch beim Testen von Regeländerungen sehr hilfreich. Weil es sich bei der Datei, in der die Regeln gespeichert sind, um ein Skript handelt, ist es auch möglich, häufig verwendete Werte/Befehle durch Aliase zu ersetzen und diese so in mehreren Regeln zu nutzen. Diese Funktion wird im folgenden Beispiel näher vorgestellt.

Die Syntax des folgenden Skripts entspricht der Syntax von sh(1), csh(1) sowie tcsh(1). Felder, die symbolisch substituiert werden, haben das Präfix $ (das Dollarzeichen). Symbolische Felder haben dieses $-Praefix nicht. Der Wert, mit dem das symbolische Feld belegt wird, muss in „doppelten Anführungszeichen“ eingeschlossen sein.

Beginnen Sie Ihre Regeldatei wie folgt:

```
############### start of example ipfw rules script #############
#
ipfw -q -f flush       # Delete all rules
# Set defaults
oif="tun0"             # out interface
odns="192.0.2.11"      # ISP's DNS server IP address
cmd="ipfw -q add "     # build rule prefix
ks="keep-state"        # just too lazy to key this each time
$cmd 00500 check-state
$cmd 00502 deny all from any to any frag
$cmd 00501 deny tcp from any to any established
$cmd 00600 allow tcp from any to any 80 out via $oif setup $ks
$cmd 00610 allow tcp from any to $odns 53 out via $oif setup $ks
$cmd 00611 allow udp from any to $odns 53 out via $oif $ks
################### End of example ipfw rules script ############
```

Die Regeln in diesem Beispiel sind nicht wichtig. Wichtig ist es, zu zeigen, wie die symbolische Substitution innerhalb der Regeln verwendet wird.

Wurde dieses Beispiel in der Datei `/etc/ipfw.rules` gespeichert, so können alle Regeln durch die Ausführung des folgenden Befehls neu geladen werden:

```
# sh /etc/ipfw.rules
```

Statt `/etc/ipfw.rules` können Sie auch einen beliebigen anderen Namen und/oder Speicherort verwenden.

Alternativ könnten Sie die einzelnen Befehle dieses Skripts auch manuell starten:

```
# ipfw -q -f flush
# ipfw -q add check-state
# ipfw -q add deny all from any to any frag
# ipfw -q add deny tcp from any to any established
# ipfw -q add allow tcp from any to any 80 out via tun0 setup keep-state
# ipfw -q add allow tcp from any to 192.0.2.11 53 out via tun0 setup keep-state
# ipfw -q add 00611 allow udp from any to 192.0.2.11 53 out via tun0 keep-state
```

30.6.5.5. Zustandsgesteuertes Regelwerk

Das folgende Regelwerk (ohne NAT-Funktionalität) ist ein Beispiel dafür, wie man eine sehr sichere „einschließende“ Firewall aufsetzen kann. Eine einschließende Firewall erlaubt es nur Diensten, für die explizite Regeln existieren, die Firewall zu passieren. Alle anderen Dienste und Pakete werden hingegen blockiert. Firewalls, die ganze Netzwerksegmente schützen sollen, benötigen mindestens zwei Netzwerkschnittstellen, für die jeweils eigene Regeln definiert werden müssen, damit die Firewall ordnungsgemäß funktioniert.

Alle unixoiden Betriebssysteme (aber auch solche, die Konzepte aus UNIX® implementieren), darunter auch FreeBSD, verwenden die Schnittstelle `lo0` mit der IP-Adresse `127.0.0.1` zur internen Kommunikation mit dem Betriebssystem. Die Firewall muss so eingestellt sein, dass sie den Datenverkehr dieser speziellen (und nur intern genutzten) Pakete ungehindert durchlässt.

Die Regeln, die den Zugriff auf eingehene und ausgehende Verbindungen regeln, autorisieren und kontrollieren, müssen mit der für die Verbindung zum öffentlichen Internet verantwortlichen Schnittstelle assoziiert werden.

Bei dieser Schnittstelle kann es sich beispielsweise um PPP/`tun0` oder die Netzwerkkarte handelt, über, die mit Ihrem DSL- oder Kabelmodem verbunden ist.

Falls mehr als eine Netzwerkkarte mit einem privaten Netzwerk (hinter der Firewall) verbunden sind, müssen die Firewallregeln für alle diese Schnittstellen entstammenden Datenpakete freien und ungehinderten Datenverkehr erlauben.

Es ist sinnvoll, die Regeln in drei Abschnitte aufzuteilen. Der erste Abschnitt enthält die freien, von der Firewall nicht zu überwachenden Netzwerkschnittstellen. Danach folgen die öffentlichen, für den ausgehenden Verkehr verantwortlichen Schnittstellen. Zuletzt kommen dann die Schnittstellen, die für den eingehenden Datenverkehr verantwortlich sind.

Innerhalb der einzelnen Abschnitte ist es sinnvoll, die am häufigsten verwendeten Regeln vor den seltener verwendeten Regel zu platzieren. Jeder Abschnitt sollte mit einer letzten Regel (die alle Pakete, auf die keine Regel zutraf, verwirft) abgeschlossen werden.

Der Abschnitt für den ausgehenden Datenverkehr des folgenden Beispiels enthät nur `allow`)-Regeln, in denen der Dienst, dem der Zugriff auf das öffentliche Internet gewährt wird, eindeutig definiert ist. Alle Regeln verwenden die Optionen `proto`, `port`, `in/out`, `via` sowie `keep state` kodiert. Die Regeln mit `proto tcp` verwenden zusätzlich die Option `setup`, damit die initiale, eine Sitzung beginnende Anfrage identifiziert werden kann, damit die die Zustandsttabelle gefüllt werden kann.

Der Abschnitt für den eingehenden Datenverkehr beginnt mit allen Regeln, die zur Blockierung unerwünschten Datenverkehrs benötigt werden. Für diese Vorgehensweise gibt es zwei Gründe: Zum einen könnten bösartige Pakete legtitimen Datenverker so sehr ähneln, dass sie die Bedingungen von `allow`-Regeln erfüllen und daher die Firewall passieren dürfen. Daher sollten derartige Pakete direkt verworden werden. Zum anderen sollten unerwünschte Pakete mit bekannten (und somit uninteressanten Mustern) sofort ohne Rückmeldung blockiert werden, anstatt erst von der letzten, generischen Regel blockiert (und, was noch wichtiger ist, auch noch protokolliert). Die letzte Regel jedes Abschnittes blockiert und protokolliert; sie kann daher dazu verwendet werden, vor Gericht haltbare Beweise zu erhalten, damit sie gegen Personen vorgehen können, die versuchen, Ihre Systeme anzugreifen.

Achten Sie darauf, dass Sie keine Netwerkantworten für geblockte Pakete senden. Diese müssen ohne Rückmeldung verworfen werden, damit ein Angreifer keine Informationen darüber erhält, ob seine Datenpakete Ihr System erreicht hat. Je weniger Information ein Angreifer über Ihr System erhält, desto sicherer ist Ihr System. Datenpakete an Ports, die nicht bekannten Diensten zugeordnet werden können, können über die Datei `/etc/services` identifiziert werden. Alternativ kann eine Anfrage an `http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers` Klarheit über die Aufgabe/Funktion einer bestimmten Portnummer bringen. Auf der Seite `http://www.sans.org/security-resources/idfaq/oddports.php` kann man Information über bekannte Trojaner und von diesen verwendete Portnummern erhalten.

30.6.5.6. Ein Beispiel für einschließende Regeln

Das folgende Regelwerk (ohne NAT-Funktionalität) beschreibt ein vollständiges, einschließendes Regelwerk. Dieses Regelwerk kann direkt auf Ihren eigenen Systemen eingesetzt werden, wenn alle `pass`-Regeln für von Ihnen nicht benötigten Dienste auskommentiert werden. Falls Sie keine Protokollierung benötigen, können Sie diese im Abschnitt für den eingehenden Datenverkehr durch eine `deny` deaktivieren. Die im Beispiel verwendete Netzwerkschnittstelle `dc0` müssen Sie durch die auf Ihrem System für ausgehenden Datenverkehr vorgesehenen Netzwerkschnittstelle ersetzen. Im Falle von benutzergesteuertem PPPs wäre dies beispielsweise `tun0`.

Alle Regeln folgen einem bestimmten Muster.

- Alle Ausdrücke, die eine Anfrage zum Beginn einer zustandsgesteuerten darstellen, beinhalten den Ausdruck `keep-state`.
- Alle Dienste aus dem öffentlichen Internet beinhalten die Option `limit`, um gegebenenfalls *flooding* zu unterbinden.
- Alle Regeln bezeichnen die Richtung durch der Ausdrücke `in` oder `out`.

- Alle Regeln legen die verwendete Netzwerkschnittstelle die Ausdrücke `via` und *`interface-name`* fest.

Die folgenden Regeln werden in der Datei `/etc/ipfw.rules` definiert.

```
################ Start of IPFW rules file ###############################
# Flush out the list before we begin.
ipfw -q -f flush

# Set rules command prefix
cmd="ipfw -q add"
pif="dc0"     # public interface name of NIC
              # facing the public Internet

#########################################################################
# No restrictions on Inside LAN Interface for private network
# Not needed unless you have LAN.
# Change xl0 to your LAN NIC interface name
#########################################################################
#$cmd 00005 allow all from any to any via xl0

#########################################################################
# No restrictions on Loopback Interface
#########################################################################
$cmd 00010 allow all from any to any via lo0

#########################################################################
# Allow the packet through if it has previous been added to the
# the "dynamic" rules table by a allow keep-state statement.
#########################################################################
$cmd 00015 check-state

#########################################################################
# Interface facing Public Internet (Outbound Section)
# Interrogate session start requests originating from behind the
# firewall on the private network or from this gateway server
# destined for the public Internet.
#########################################################################

# Allow out access to my ISP's Domain name server.
# x.x.x.x must be the IP address of your ISP.s DNS
# Dup these lines if your ISP has more than one DNS server
# Get the IP addresses from /etc/resolv.conf file
$cmd 00110 allow tcp from any to x.x.x.x 53 out via $pif setup keep-state
$cmd 00111 allow udp from any to x.x.x.x 53 out via $pif keep-state

# Allow out access to my ISP's DHCP server for cable/DSL configurations.
# This rule is not needed for .user ppp. connection to the public Internet.
# so you can delete this whole group.
# Use the following rule and check log for IP address.
# Then put IP address in commented out rule & delete first rule
$cmd 00120 allow log udp from any to any 67 out via $pif keep-state
#$cmd 00120 allow udp from any to x.x.x.x 67 out via $pif keep-state

# Allow out non-secure standard www function
$cmd 00200 allow tcp from any to any 80 out via $pif setup keep-state

# Allow out secure www function https over TLS SSL
$cmd 00220 allow tcp from any to any 443 out via $pif setup keep-state

# Allow out send & get email function
$cmd 00230 allow tcp from any to any 25 out via $pif setup keep-state
$cmd 00231 allow tcp from any to any 110 out via $pif setup keep-state

# Allow out FBSD (make install & CVSUP) functions
# Basically give user root "GOD" privileges.
$cmd 00240 allow tcp from me to any out via $pif setup keep-state uid root
```

```
# Allow out ping
$cmd 00250 allow icmp from any to any out via $pif keep-state

# Allow out Time
$cmd 00260 allow tcp from any to any 37 out via $pif setup keep-state

# Allow out nntp news (i.e. news groups)
$cmd 00270 allow tcp from any to any 119 out via $pif setup keep-state

# Allow out secure FTP, Telnet, and SCP
# This function is using SSH (secure shell)
$cmd 00280 allow tcp from any to any 22 out via $pif setup keep-state

# Allow out whois
$cmd 00290 allow tcp from any to any 43 out via $pif setup keep-state

# deny and log everything else that.s trying to get out.
# This rule enforces the block all by default logic.
$cmd 00299 deny log all from any to any out via $pif

################################################################
# Interface facing Public Internet (Inbound Section)
# Check packets originating from the public Internet
# destined for this gateway server or the private network.
################################################################

# Deny all inbound traffic from non-routable reserved address spaces
$cmd 00300 deny all from 192.168.0.0/16 to any in via $pif  #RFC 1918 private IP
$cmd 00301 deny all from 172.16.0.0/12 to any in via $pif     #RFC 1918 private IP
$cmd 00302 deny all from 10.0.0.0/8 to any in via $pif          #RFC 1918 private IP
$cmd 00303 deny all from 127.0.0.0/8 to any in via $pif        #loopback
$cmd 00304 deny all from 0.0.0.0/8 to any in via $pif            #loopback
$cmd 00305 deny all from 169.254.0.0/16 to any in via $pif   #DHCP auto-config
$cmd 00306 deny all from 192.0.2.0/24 to any in via $pif       #reserved for docs
$cmd 00307 deny all from 204.152.64.0/23 to any in via $pif  #Sun cluster interconnect
$cmd 00308 deny all from 224.0.0.0/3 to any in via $pif         #Class D & E multicast

# Deny public pings
$cmd 00310 deny icmp from any to any in via $pif

# Deny ident
$cmd 00315 deny tcp from any to any 113 in via $pif

# Deny all Netbios service. 137=name, 138=datagram, 139=session
# Netbios is MS/Windows sharing services.
# Block MS/Windows hosts2 name server requests 81
$cmd 00320 deny tcp from any to any 137 in via $pif
$cmd 00321 deny tcp from any to any 138 in via $pif
$cmd 00322 deny tcp from any to any 139 in via $pif
$cmd 00323 deny tcp from any to any 81 in via $pif

# Deny any late arriving packets
$cmd 00330 deny all from any to any frag in via $pif

# Deny ACK packets that did not match the dynamic rule table
$cmd 00332 deny tcp from any to any established in via $pif

# Allow traffic in from ISP's DHCP server. This rule must contain
# the IP address of your ISP.s DHCP server as it.s the only
# authorized source to send this packet type.
# Only necessary for cable or DSL configurations.
# This rule is not needed for .user ppp. type connection to
# the public Internet. This is the same IP address you captured
# and used in the outbound section.
#$cmd 00360 allow udp from any to x.x.x.x 67 in via $pif keep-state
```

```
# Allow in standard www function because I have apache server
$cmd 00400 allow tcp from any to me 80 in via $pif setup limit src-addr 2

# Allow in secure FTP, Telnet, and SCP from public Internet
$cmd 00410 allow tcp from any to me 22 in via $pif setup limit src-addr 2

# Allow in non-secure Telnet session from public Internet
# labeled non-secure because ID & PW are passed over public
# Internet as clear text.
# Delete this sample group if you do not have telnet server enabled.
$cmd 00420 allow tcp from any to me 23 in via $pif setup limit src-addr 2

# Reject & Log all incoming connections from the outside
$cmd 00499 deny log all from any to any in via $pif

# Everything else is denied by default
# deny and log all packets that fell through to see what they are
$cmd 00999 deny log all from any to any
################ End of IPFW rules file ################################
```

30.6.5.7. Ein Beispiel für zustandshafte NAT-Regeln

Es müssen einige zusätzliche Konfigurationseinstellungen vorgenommen werden, um die die NAT-Funktion von IPFW zu nutzen. Die Kernelquellen müssen mit der Option `IPDIVERT` (im IPFIREWALL-Abschnitt der Kernelkonfigurationsdatei) neu gebaut werden, um den benötigten angepassten Kernel zu erzeugen.

Zusätzlich werden folgende Optionen in der `/etc/rc.conf` benötigt:

```
natd_enable="YES"                   # Enable NATD function
natd_interface="rl0"                # interface name of public Internet NIC
natd_flags="-dynamic -m"            # -m = preserve port numbers if possible
```

Zustandshafte Regeln bei aktiviertem `divert natd` (*Network Address Translation*) verkomplizieren die Formulierung des Regelwerkes beträchtlich. Damit Ihre Firewall funktioniert, kommt es insbesondere auf die Position der Ausdrücke `check-state` sowie `divert natd` an. Sie können nicht länger einen einfachen, kaskadierenden Ablauf verwenden (also einen Regelsatz, bei dem einfach auf eine Regel nach der anderen geprüft wird. Vielmehr wird der neue Aktionstyp `skipto` benötigt. Dies erfordert, dass jede Regel über eine eindeutige Nummer verfügt, um so eindeutige Sprungziele zu erhalten.

Im Folgenden wird anhand eines umkommentierten Beispiels der Paketfluss durch das Regelwerk verdeutlicht.

Die Verarbeitung beginnt mit der ersten Regel (also am Anfang der Regeldatei. Sie setzt sich Regel für Regel weiter fort, bis das Ende der Datei erreicht ist oder eine Regel für das Paket einen Treffer erzielt und das Paket so die Firewall verlassen kann. Achten Sie besonders auf die Position der Regeln mit den Nummern `100, 101, 450, 500` sowie `510`. Diese Regeln steuern die Adressumsetzung ausgehender und eingehender Pakete, so dass deren entsprechende Einträge in der Zustandstabelle immer die private LAN-Adressen abbilden. Zusätzlich werden in allen Regeln die Richtung des Pakets (eingehend oder ausgehend) so die vom Paket zu verwendende Netzwerkschnittstelle definiert. Ausgehende Anfragen, die eine Sitzung starten, rufen immer `skipto rule 500`, damit NAT verwendet werden kann.

Nehmen wir nun an, dass ein Nutzer einen Webbrowser verwendet, um eine Internetseite aufzurufen. Derartige Anfragen werden in der Regel über Port 80 geleitet. Die zugehörigen Pakete werden durch die Firewall verarbeitet. Regel 100 trifft nicht zu, denn das Paket geht nach außen, nicht nach innen. Regel 101 trifft ebenfalls nicht zu, denn es handelt sich um das erste Paket. Folglich wird die Sitzung erst initiiert und kann somit noch nicht in der Zustandstabelle enthalten sein kann. Die erste Regel, die zutrifft, ist Regel 125. Das Paket will das lokale Netzwerk über die Schnittstelle zum öffentlichen Internet (das heißt nach außen) verlassen, es hat aber noch die Quelladresse des privaten lokalen Netzwerks. Da Regel 125 zutrifft, werden zwei Aktionen ausgeführt: Die Option `keep-state` bewirkt, dass das Paket in der internen Tabelle für zustandshafte, dynamische Regeln registriert wird. Danach wird der Aktionsteil der Regel ausgeführt. Dieser ist Bestandteil der Informationen, die in die in der Tabelle für dynamische Regeln aufgenommen wird und lautet `skipto rule 500`. Die Regel 500 führt NATs auf die IP-Adresse des Paketes durch. Danach verlässt das Paket das LAN nach außen in Richtung des öffentlichen Internets. Dieser

letzte Teil ist für funktionierendes NAT von entscheidender Bedeutung. Nachdem dieses Paket am Bestimmungsort angekommen ist, wird dort eine Antwort generiert und zurückgeschickt. Dieses Paket wird auf die gleiche Art und Weise durch das gegebene Regelwerk verarbeitet. Dieses Mal trifft Regel 100 auf das Paket zu, damit wird die Bestimmungsadresse auf die zugehörige (lokale) LAN-Adresse (rück-)abgebildet. Danach wird es von der `check-state`-Regel verarbeitet, die Zustandstabelle erkennt, dass eine zugehörige aktive Sitzung vorliegt und das Paket wird freigegeben und in das LAN geleitet. Es wird innerhalb des LANs von dem PC, der die zugehörige Sitzung hält, empfangen, der ein neues Paket absendet und ein weiteres Datensegment vom entfernten Server anfordert. Dieses Mal wird bei der Prüfung der `check-state` -Regel ein nach außen gehender zugehöriger Eintrag in der Zustandstabelle gefunden und die entsprechende Aktion (also `skipto 500`) wird ausgeführt. Das Paket springt zu Regel 500 und wird durch diese Regel für das öffentliche Internet freigegeben.

Innerhalb des durch die Firewall geschützten Netzwerks werden alle eingehenden Pakete, die zu einer existierenden Sitzung gehören, durch die Regel `check-state` sowie entsprechend platzierte `divert natd` -Regeln verarbeitet. Die notwendige Arbeit beschränkt sich darauf, alle „schlechten“ Pakete zu blockieren und nur authorisierten Diensten zugehörige Pakete durchzulassen. In Umkehrung des bisherigen Beispiels nehmen wir nun, dass auf dem Rechner, auf dem die Firewall läuft, auch ein Apache Webserver läuft, auf den von außen, also aus dem öffentlichen Internet, zugegriffen werden kann. Das erste von außen eintreffende Paket (das auch eine neue Sitzung startet) erfüllt Regel 100. Die Zieladresse des Paketes wird daher auf die LAN-Adresse des Firewallrechners abgebildet. Das Paket wird dann weiter auf alle in der Firewall definierten Regeln geprüft und trifft schließlich auf Regel 425. Durch diese Regel werden zwei Aktionen ausgelösst: Erstens wird aus dem Paket eine dynamische Regel generiert und in die Zustandstabelle geschrieben. Zusätzlich wird jedoch die Anzahl neuer Sitzungsanfragen (von der gleichen Quell-IP-Adresse) auf 2 begrenzt, um so DoS-Angriffe auf Dienste, die auf diesem Port laufen, zu verhindern. Die Aktion dieser Regel ist `allow`, daher wird das Paket freigegeben und in das LAN weitergeleitet. Das als Antwort generierte Paket wird durch die `check-state` -Regel als zu einer Sitzung gehörend erkannt. Damit wird es der Regel 500 zugeführt, NAT wird durchgeführt und über die Schnittstelle zum öffentlichen Internet nach außen geroutet.

Beispiel 1 für einen Regelsatz:

```
#!/bin/sh
cmd="ipfw -q add"
skip="skipto 500"
pif=rl0
ks="keep-state"
good_tcpo="22,25,37,43,53,80,443,110,119"

ipfw -q -f flush

$cmd 002 allow all from any to any via xl0  # exclude LAN traffic
$cmd 003 allow all from any to any via lo0  # exclude loopback traffic

$cmd 100 divert natd ip from any to any in via $pif
$cmd 101 check-state

# Authorized outbound packets
$cmd 120 $skip udp from any to xx.168.240.2 53 out via $pif $ks
$cmd 121 $skip udp from any to xx.168.240.5 53 out via $pif $ks
$cmd 125 $skip tcp from any to any $good_tcpo out via $pif setup $ks
$cmd 130 $skip icmp from any to any out via $pif $ks
$cmd 135 $skip udp from any to any 123 out via $pif $ks


# Deny all inbound traffic from non-routable reserved address spaces
$cmd 300 deny all from 192.168.0.0/16  to any in via $pif  #RFC 1918 private IP
$cmd 301 deny all from 172.16.0.0/12   to any in via $pif  #RFC 1918 private IP
$cmd 302 deny all from 10.0.0.0/8      to any in via $pif  #RFC 1918 private IP
$cmd 303 deny all from 127.0.0.0/8     to any in via $pif  #loopback
$cmd 304 deny all from 0.0.0.0/8       to any in via $pif  #loopback
$cmd 305 deny all from 169.254.0.0/16  to any in via $pif  #DHCP auto-config
$cmd 306 deny all from 192.0.2.0/24    to any in via $pif  #reserved for docs
$cmd 307 deny all from 204.152.64.0/23 to any in via $pif  #Sun cluster
$cmd 308 deny all from 224.0.0.0/3     to any in via $pif  #Class D & E multicast
```

```
# Authorized inbound packets
$cmd 400 allow udp from xx.70.207.54 to any 68 in $ks
$cmd 420 allow tcp from any to me 80 in via $pif setup limit src-addr 1


$cmd 450 deny log ip from any to any

# This is skipto location for outbound stateful rules
$cmd 500 divert natd ip from any to any out via $pif
$cmd 510 allow ip from any to any

######################## end of rules  #################
```

Das folgende Beispiel ist praktisch identisch mit dem ersten Regelsatz. Allerdings wurden die Regel umfassend kommentiert und umgeschrieben, damit sie für weniger erfahrene Benutzer leichter verständlich werden.

Beispiel 2 für einen Regelsatz:

```
#!/bin/sh
################ Start of IPFW rules file ###############################
# Flush out the list before we begin.
ipfw -q -f flush

# Set rules command prefix
cmd="ipfw -q add"
skip="skipto 800"
pif="rl0"     # public interface name of NIC
              # facing the public Internet

########################################################################
# No restrictions on Inside LAN Interface for private network
# Change xl0 to your LAN NIC interface name
########################################################################
$cmd 005 allow all from any to any via xl0

########################################################################
# No restrictions on Loopback Interface
########################################################################
$cmd 010 allow all from any to any via lo0

########################################################################
# check if packet is inbound and nat address if it is
########################################################################
$cmd 014 divert natd ip from any to any in via $pif

########################################################################
# Allow the packet through if it has previous been added to the
# the "dynamic" rules table by a allow keep-state statement.
########################################################################
$cmd 015 check-state

########################################################################
# Interface facing Public Internet (Outbound Section)
# Check session start requests originating from behind the
# firewall on the private network or from this gateway server
# destined for the public Internet.
########################################################################

# Allow out access to my ISP's Domain name server.
# x.x.x.x must be the IP address of your ISP's DNS
# Dup these lines if your ISP has more than one DNS server
# Get the IP addresses from /etc/resolv.conf file
$cmd 020 $skip tcp from any to x.x.x.x 53 out via $pif setup keep-state


# Allow out access to my ISP's DHCP server for cable/DSL configurations.
```

```
$cmd 030 $skip udp from any to x.x.x.x 67 out via $pif keep-state

# Allow out non-secure standard www function
$cmd 040 $skip tcp from any to any 80 out via $pif setup keep-state

# Allow out secure www function https over TLS SSL
$cmd 050 $skip tcp from any to any 443 out via $pif setup keep-state

# Allow out send & get email function
$cmd 060 $skip tcp from any to any 25 out via $pif setup keep-state
$cmd 061 $skip tcp from any to any 110 out via $pif setup keep-state

# Allow out FreeBSD (make install & CVSUP) functions
# Basically give user root "GOD" privileges.
$cmd 070 $skip tcp from me to any out via $pif setup keep-state uid root

# Allow out ping
$cmd 080 $skip icmp from any to any out via $pif keep-state

# Allow out Time
$cmd 090 $skip tcp from any to any 37 out via $pif setup keep-state

# Allow out nntp news (i.e. news groups)
$cmd 100 $skip tcp from any to any 119 out via $pif setup keep-state

# Allow out secure FTP, Telnet, and SCP
# This function is using SSH (secure shell)
$cmd 110 $skip tcp from any to any 22 out via $pif setup keep-state

# Allow out whois
$cmd 120 $skip tcp from any to any 43 out via $pif setup keep-state

# Allow ntp time server
$cmd 130 $skip udp from any to any 123 out via $pif keep-state

################################################################
# Interface facing Public Internet (Inbound Section)
# Check packets originating from the public Internet
# destined for this gateway server or the private network.
################################################################

# Deny all inbound traffic from non-routable reserved address spaces
$cmd 300 deny all from 192.168.0.0/16  to any in via $pif  #RFC 1918 private IP
$cmd 301 deny all from 172.16.0.0/12   to any in via $pif  #RFC 1918 private IP
$cmd 302 deny all from 10.0.0.0/8      to any in via $pif  #RFC 1918 private IP
$cmd 303 deny all from 127.0.0.0/8     to any in via $pif  #loopback
$cmd 304 deny all from 0.0.0.0/8       to any in via $pif  #loopback
$cmd 305 deny all from 169.254.0.0/16  to any in via $pif  #DHCP auto-config
$cmd 306 deny all from 192.0.2.0/24    to any in via $pif  #reserved for docs
$cmd 307 deny all from 204.152.64.0/23 to any in via $pif  #Sun cluster
$cmd 308 deny all from 224.0.0.0/3     to any in via $pif  #Class D & E multicast

# Deny ident
$cmd 315 deny tcp from any to any 113 in via $pif

# Deny all Netbios service. 137=name, 138=datagram, 139=session
# Netbios is MS/Windows sharing services.
# Block MS/Windows hosts2 name server requests 81
$cmd 320 deny tcp from any to any 137 in via $pif
$cmd 321 deny tcp from any to any 138 in via $pif
$cmd 322 deny tcp from any to any 139 in via $pif
$cmd 323 deny tcp from any to any 81  in via $pif

# Deny any late arriving packets
$cmd 330 deny all from any to any frag in via $pif
```

```
# Deny ACK packets that did not match the dynamic rule table
$cmd 332 deny tcp from any to any established in via $pif

# Allow traffic in from ISP's DHCP server. This rule must contain
# the IP address of your ISP's DHCP server as it's the only
# authorized source to send this packet type.
# Only necessary for cable or DSL configurations.
# This rule is not needed for 'user ppp' type connection to
# the public Internet. This is the same IP address you captured
# and used in the outbound section.
$cmd 360 allow udp from x.x.x.x to any 68 in via $pif keep-state

# Allow in standard www function because I have Apache server
$cmd 370 allow tcp from any to me 80 in via $pif setup limit src-addr 2

# Allow in secure FTP, Telnet, and SCP from public Internet
$cmd 380 allow tcp from any to me 22 in via $pif setup limit src-addr 2

# Allow in non-secure Telnet session from public Internet
# labeled non-secure because ID & PW are passed over public
# Internet as clear text.
# Delete this sample group if you do not have telnet server enabled.
$cmd 390 allow tcp from any to me 23 in via $pif setup limit src-addr 2

# Reject & Log all unauthorized incoming connections from the public Internet
$cmd 400 deny log all from any to any in via $pif

# Reject & Log all unauthorized out going connections to the public Internet
$cmd 450 deny log all from any to any out via $pif

# This is skipto location for outbound stateful rules
$cmd 800 divert natd ip from any to any out via $pif
$cmd 801 allow ip from any to any

# Everything else is denied by default
# deny and log all packets that fell through to see what they are
$cmd 999 deny log all from any to any
################ End of IPFW rules file ###############################
```

Kapitel 31. Weiterführende Netzwerkthemen

Übersetzt von Johann Kois.

31.1. Übersicht

Dieses Kapitel beschreibt verschiedene weiterführende Netzwerkthemen.

Nachdem Sie dieses Kapitel gelesen haben, werden Sie

- Die Grundlagen von Gateways und Routen kennen.
- Wissen, wie man USB Tethering einrichtet.
- Bluetooth®- sowie drahtlose, der Norm IEEE® 802.11 entsprechende, Geräte mit FreeBSD verwenden können.
- Eine Bridge unter FreeBSD einrichten können.
- Wissen, wie man mithilfe von PXE über ein Netzwerk von einem NFS Root-Dateisystem bootet.
- IPv6 auf einem FreeBSD-Rechner einrichten können.
- Das Common Address Redundancy Protocol (CARP) unter FreeBSD einsetzen können.

Bevor Sie dieses Kapitel lesen, sollten Sie

- Die Grundlagen der `/etc/rc` -Skripte verstanden haben.
- Mit der grundlegenden Netzwerkterminologie vertraut sein.
- Einen neuen FreeBSD-Kernel konfigurieren und installieren können (Kapitel 9, *Konfiguration des FreeBSD-Kernels*).
- Wissen, wie man zusätzliche Software von Drittherstellern installiert (Kapitel 5, *Installieren von Anwendungen: Pakete und Ports*).

31.2. Gateways und Routen

Beigetragen von Coranth Gryphon.

Der Mechanismus mit dem ein Rechner einen Rechner über ein Netzwerk finden kann, wird als *Routing* bezeichnet. Eine „Route“ besteht aus einem definierten Adresspaar: Einem „Ziel“ und einem „Gateway“. Die Route zeigt an, dass Pakete über das *Gateway* zum *Ziel* gelangen können. Es gibt drei Arten von Zielen: Einzelne Rechner (Hosts), Subnetze und das „Standard“ziel. Die „Standardroute“ wird verwendet, wenn keine andere Route zutrifft. Außerdem gibt es drei Arten von Gateways: Einzelne Rechner (Hosts), Schnittstellen (Interfaces, auch als „Links“ bezeichnet), sowie Ethernet Hardware-Adressen (MAC). Bekannte Adressen werden in einer Routingtabelle gespeichert.

Dieser Abschnitt bietet einen Überblick über die Grundlagen des Routings. Er demonstriert, wie ein FreeBSD-System als Router konfiguriert werden kann und bietet einige Tipps zur Fehlerbehebung.

31.2.1. Grundlagen des Routings

netstat(1) zeigt die Routingtabellen eines FreeBSD-Systems an:

```
% netstat -r
```

```
Routing tables

Internet:
Destination      Gateway            Flags     Refs     Use     Netif Expire
default          outside-gw         UGS         37     418       em0
localhost        localhost          UH           0     181       lo0
test0            0:e0:b5:36:cf:4f   UHLW         5   63288       re0     77
10.20.30.255     link#1             UHLW         1    2421
example.com      link#1             UC           0       0
host1            0:e0:a8:37:8:1e    UHLW         3    4601       lo0
host2            0:e0:a8:37:8:1e    UHLW         0       5       lo0 =>
host2.example.com link#1            UC           0       0
224              link#1             UC           0       0
```

Die Einträge in diesem Beispiel sind wie folgt:

default
: Die erste Route in der Ausgabe gibt die Standardroute (`default`) an. Wenn sich der lokale Rechner mit einem entfernten Rechner verbinden will, wird die Routingtabelle überprüft, um festzustellen, ob bereits ein bekannter Pfad vorhanden ist. Wird für den entfernten Rechner ein Eintrag in der Routingtabelle gefunden, so prüft das System ob es sich über die angegebene Schnittstelle verbinden kann.

  Wenn das Zielsystem mit keinem Eintrag übereinstimmt, oder wenn alle bekannten Routen fehlschlagen, verwendet das System die Standardroute. Für die Rechner im lokalen Netzwerk ist das Feld `Gateway` auf das System gesetzt, welches direkt mit dem Internet verbunden ist. `UG` in der Spalte `Flags` zeigt an, dass das Gateway einsatzbereit ist.

  Die Standardroute für einen Rechner, der selbst als Gateway zur Außenwelt fungiert, ist der Gateway-Rechner des Internetanbieters (ISP).

localhost
: Die zweite Route zeigt die `localhost` Route. Die festgelegte Schnittstelle in der `Netif`-Spalte für `localhost` ist `lo0`, das auch als loopback-Gerät bekannt ist. Das bedeutet, dass der gesamte Datenverkehr für dieses Ziel intern bleibt, anstatt ihn über ein Netzwerk zu versenden.

MAC-Adresse
: Bei den mit `0:e0:` beginnenden Adressen handelt es sich um MAC-Adressen. FreeBSD identifiziert Rechner im lokalen Netz, im Beispiel `test0`, automatisch und fügt eine direkte Route über die Ethernet-Schnittstelle `re0` zu diesem Rechner hinzu. Außerdem existiert in der Spalte `Expire` ein Timeout, der verwendet wird, wenn dieser Rechner in einem definierten Zeitraum nicht reagiert. Wenn dies passiert, wird die Route zu diesem Rechner automatisch gelöscht. Rechner im lokalen Netz werden über das Routing Information Protocol (RIP) identifiziert, welches den kürzesten Weg zu den jeweiligen Rechnern berechnet.

Subnetz
: FreeBSD wird automatisch Subnetzrouten für das lokale Subnetz hinzufügen. In diesem Beispiel ist `10.20.30.255` die Broadcast-Adresse für das Subnetz `10.20.30`, und `example.com` ist der zu diesem Subnetz gehörige Domainname. Das Ziel `link#1` bezieht sich auf die erste Ethernet-Karte im Rechner.

  Routen für Rechner im lokalen Netz und lokale Subnetze werden automatisch durch den routed(8) Daemon konfiguriert. Ist dieser nicht gestartet, existieren nur statische Routen, die vom Administrator definiert werden.

Host
: Die Zeile `host1` bezieht sich auf den Rechner, der durch seine Ethernetadresse bekannt ist. Da es sich um den sendenden Rechner handelt, verwendet FreeBSD automatisch das Loopback-Gerät (`lo0`), anstatt den Datenverkehr über die Ethernet-Schnittstelle zu senden.

  Die zwei `host2` Zeilen repräsentieren Aliase, die mit ifconfig(8) erstellt wurden. Das Symbol `=>` nach der `lo0`-Schnittstelle sagt aus, dass zusätzlich zur Loopback-Adresse auch ein Alias eingestellt ist. Solche Routen sind

nur auf Rechnern vorhanden, die den Alias bereitstellen. Alle anderen Rechner im lokalen Netz haben für solche Routen nur eine `link#1` Zeile.

224
Die letzte Zeile (Zielsubnetz `224`) behandelt Multicasting.

Schließlich gibt es für Routen noch verschiedene Attribute, die sich in der Spalte `Flags` befinden. Tabelle 31.1, „Allgemeine Attribute in Routingtabellen“ fasst einige dieser Flags und deren Bedeutung zusammen:

Tabelle 31.1. Allgemeine Attribute in Routingtabellen

Attribut	Bedeutung
U	Die Route ist aktiv (up).
H	Das Ziel der Route ist ein einzelner Rechner (Host).
G	Alle Daten, die an dieses Ziel gesendet werden, werden von dem Gateway an ihr jeweiliges Ziel weitergeleitet.
S	Diese Route wurde statisch konfiguriert.
C	Erzeugt eine neue Route, basierend auf der Route für den Rechner, mit dem wir uns verbinden. Diese Routenart wird normalerweise für lokale Netzwerke verwendet.
W	Eine Route, die automatisch konfiguriert wurde. Sie basiert auf einer lokalen Netzwerkroute (Clone).
L	Die Route beinhaltet einen Verweis auf eine Ethernetkarte (Link).

In FreeBSD kann die Standardroute durch die Angabe der IP-Adresse des Standard-Gateways in `/etc/rc.conf` definiert werden:

```
defaultrouter="10.20.30.1"
```

Die Standardroute kann mit `route` auch manuell gesetzt werden:

```
# route add default 10.20.30.1
```

Beachten Sie, dass manuell hinzugefügte Routen bei einem Neustart des Systems verloren gehen. Weitere Informationen zum Bearbeiten von Netzwerk-Routingtabellen finden Sie in route(8).

31.2.2. Statische Routen einrichten

Beigetragen von Al Hoang.

Ein FreeBSD-System kann als Standard-Gateway bzw. Router für ein Netzwerk konfiguriert werden, wenn es sich um einen Dual-Homed-Host handelt. Ein Dual-Homed-Host ist ein Rechner, der sich in mindestens zwei verschiedenen Netzwerken befindet. Typischerweise ist jedes Netzwerk über eine separate Netzwerkschnittstelle verbunden. Mit IP Aliasing können mehrere Adressen, die jeweils zu einem andren Subnetz gehören, an eine physikalische Schnittstelle gebunden werden.

Damit Pakete zwischen den Schnittstellen weitergeleitet werden können, muss das FreeBSD-System als Router konfiguriert werden. Internetstandards und gute Ingenieurspraxis sorgten dafür, dass diese Funktion in FreeBSD in der Voreinstellung deaktiviert ist. Sie kann jedoch aktiviert werden, indem folgende Zeile in `/etc/rc.conf` hinzugefügt wird:

```
gateway_enable="YES"          # Auf YES setzen, wenn der Rechner als Gateway arbeiten ↺
soll
```

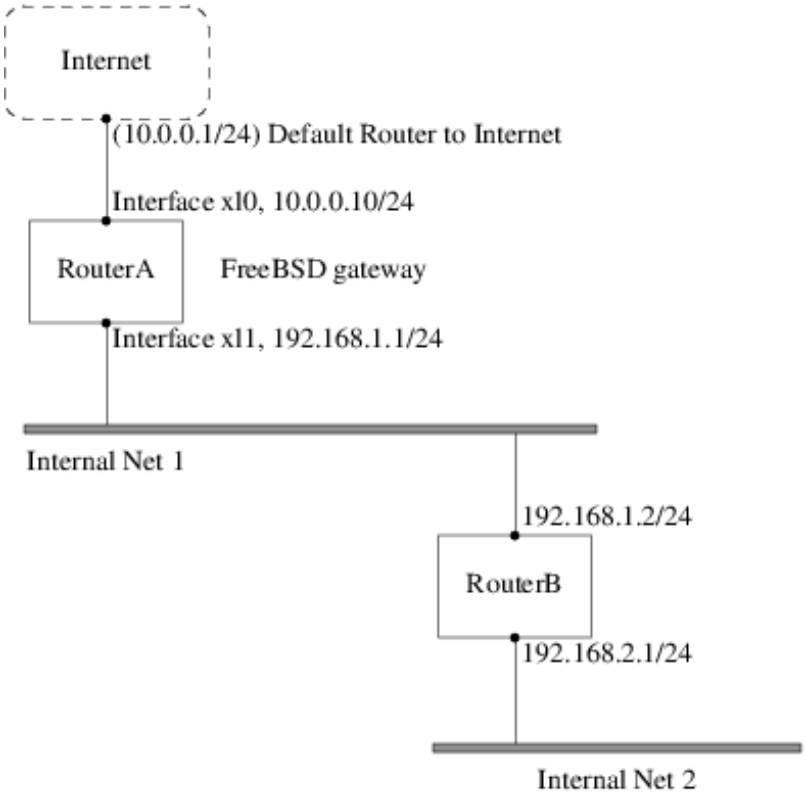
Um das Routing zu aktivieren, setzen Sie die sysctl(8)-Variable `net.inet.ip.forwarding` auf `1`. Um das Routing zu stoppen, muss die Variable wieder auf `0` gesetzt werden.

Die Routingtabelle eines Routers benötigt zusätzliche Routen, damit er weiß, wie er andere Netzwerke erreichen kann. Die Routen können entweder manuell als statische Routen hinzugefügt werden, oder aber der Router lernt automatisch die Routen anhand des Routing-Protokolls. Statische Routen eignen sich für kleine Netzwerke und dieser Abschnitt beschreibt, wie Sie eine statische Route für ein kleines Netzwerk hinzufügen.

Anmerkung

In großen Netzwerken sind statische Routen schlecht skalierbar. FreeBSD beinhaltet den BSD-Routing-Daemon routed(8), der die Protokolle RIP (Version 1 und Version 2) sowie IRDP unterstützt. Die Routing-Protokolle BGP und OSPF können über den Port oder das Paket net/zebra installiert werden.

Nehmen wir an, dass wir über folgendes Netzwerk verfügen:



`RouterA`, ein FreeBSD-Rechner, dient als Router für den Zugriff auf das Internet. Die Standardroute ist auf `10.0.0.1` gesetzt, damit ein Zugriff auf das Internet möglich wird. `RouterB` ist bereits konfiguriert, da er `192.168.1.1` als Standard-Gateway benutzt.

Bevor die statischen Routen hinzugefügt werden, sieht die Routingtabelle auf `RouterA` in etwa so aus:

```
% netstat -nr
Routing tables

Internet:
Destination        Gateway            Flags    Refs      Use  Netif  Expire
default            10.0.0.1           UGS         0    49378    xl0
127.0.0.1          127.0.0.1          UH          0        6    lo0
10.0.0/24          link#1             UC          0        0    xl0
192.168.1/24       link#2             UC          0        0    xl1
```

Mit dieser Routingtabelle hat `RouterA` keine Route zum Netzwerk `192.168.2.0/24` . Der folgende Befehl wird das interne Netz 2 in die Routingtabelle von `RouterA` aufnehmen und dabei `192.168.1.2` als nächsten Zwischenschritt (*Hop*) verwenden:

```
# route add -net 192.168.2.0/24 192.168.1.2
```

Ab sofort kann `RouterA` alle Rechner des Netzwerks `192.168.2.0/24` erreichen. Allerdings gehen die Routing-Informationen verloren, wenn das FreeBSD-System neu gestartet wird. Um statische Routen dauerhaft einzurichten, müssen diese in `/etc/rc.conf` eingetragen werden:

```
# Add Internal Net 2 as a persistent static route
static_routes="internalnet2"
route_internalnet2="-net 192.168.2.0/24 192.168.1.2"
```

Die Variable `static_routes` enthält eine Reihe von Strings, die durch Leerzeichen getrennt sind. Jeder String bezieht sich auf den Namen einer Route. Die Variable `route_internalnet2` enthält die statische Route.

Wird mit der Variablen `static_routes` mehr als eine Variable angegeben, so werden auch mehrere Routen angelegt. Im folgenden Beispiel werden statische Routen zu den Netzwerken `192.168.0.0/24` und `192.168.1.0/24` angelegt.

```
static_routes="net1 net2"
route_net1="-net 192.168.0.0/24 192.168.0.1"
route_net2="-net 192.168.1.0/24 192.168.1.1"
```

31.2.3. Problembehandlung

Wenn ein Adressraum einem Netzwerk zugeordnet wird, konfiguriert der Dienstanbieter seine Routing-Tabellen, so dass der gesamte Verkehr für das Netzwerk über die Verbindung zu der Seite gesendet wird. Aber woher wissen externe Webseiten, dass sie die Daten an das Netzwerk des ISP senden sollen?

Es gibt ein System, das alle zugewiesenen Adressräume verwaltet und die Verbindung zum Internet-Backbone definiert. Der „Backbone“ ist das Netz aus Hauptverbindungen, die den Internetverkehr in der ganzen Welt transportieren und verteilen. Jeder Backbone-Rechner verfügt über eine Kopie von Master-Tabellen, die den Verkehr für ein bestimmtes Netzwerk hierarchisch vom Backbone über eine Kette von Dienstanbietern bis hin zu einem bestimmten Netzwerk leiten.

Es ist die Aufgabe des Dienstanbieters, den Backbone-Seiten mitzuteilen, dass sie mit einer Seite verbunden wurden. Dieser Vorgang wird als *Bekanntmachung von Routen* (*routing propagation*) bezeichnet.

Manchmal kommt es zu Problemen bei der Bekanntmachung von Routen, und einige Seiten sind nicht in der Lage, sich zu verbinden. Der vielleicht nützlichste Befehl, um festzustellen wo das Routing nicht funktioniert, ist `traceroute`. Das Programm ist nützlich, falls `ping` fehlschlägt.

Rufen Sie `traceroute` mit dem Namen des entfernten Rechners auf, mit dem eine Verbindung aufgebaut werden soll. Die Ausgabe zeigt die Gateway-Rechner entlang des Verbindungspfades an. Schließlich wird der Zielrechner erreicht oder es kommt zu einem Verbindungsabbruch. Weitere Informationen finden Sie in traceroute(8).

31.2.4. Multicast-Routing

FreeBSD unterstützt sowohl Multicast-Anwendungen als auch Multicast-Routing. Multicast-Anwendungen benötigen keine spezielle Konfiguration, um auf FreeBSD lauffähig zu sein. Damit Multicast-Routing unterstützt wird, muss die folgende Option in der Kernelkonfiguration aktiviert werden:

```
options MROUTING
```

Der Multicast-Routing-Daemon mrouted kann als Port oder Paket net/mroute installiert werden. Dieser Daemon implementiert das DVMRP Multicast-Routing-Protokoll. Um die Tunnel und DVMRP einzurichten, muss `/usr/local/etc/mrouted.conf` bearbeitet werden. Bei der Installation von mrouted wird auch map-mbone und mrinfo sowie die zugehörigen Manualpages installiert, in denen Sie auch Konfigurationsbeispiele finden können.

Anmerkung

DVMRP wurde in vielen Multicast-Installationen weitgehend durch das PIM-Protokoll ersetzt. Weitere Informationen finden Sie in pim(4).

31.3. Drahtlose Netzwerke

Loader, Marc Fonvieille und Murray Stokely.

31.3.1. Grundlagen

Die meisten drahtlosen Netzwerke basieren auf dem Standard IEEE® 802.11. Ein einfaches drahtloses Netzwerk besteht aus Stationen, die im 2,4 GHz- oder im 5 GHz-Band miteinander kommunizieren. Es ist aber auch möglich, dass regional andere Frequenzen, beispielsweise im 2,3 GHz- oder 4,9 GHz-Band, verwendet werden.

802.11-Netzwerke können auf zwei verschiedene Arten aufgebaut sein: Im *Infrastruktur-Modus* agiert eine Station als Master, mit dem sich alle anderen Stationen verbinden. Die Summe aller Stationen wird als Basic Service Set (BSS), die Master-Station hingegen als Access Point (AP) bezeichnet. In einem BSS läuft jedwede Kommunikation über den Access Point. Die zweite Form drahtloser Netzwerke sind die sogenannten *Ad-hoc-Netzwerke* (auch als IBSS bezeichnet), in denen es keinen Access Point gibt und in denen die Stationen direkt miteinander kommunizieren.

Die ersten 802.11-Netzwerke arbeiteten im 2,4 GHz-Band und nutzten dazu Protokolle der IEEE®-Standards 802.11 sowie 802.11b. Diese Standards legen unter anderem Betriebsfrequenzen sowie Merkmale des MAC-Layers (wie Frames und Transmissionsraten) fest. Später kam der Standard 802.11a hinzu, der im 5 GHz-Band, im Gegensatz zu den ersten beiden Standards aber mit unterschiedlichen Signalmechanismen und höheren Transmissionsraten arbeitet. Der neueste Standard 802.11g implementiert die Signal- und Transmissionsmechanismen von 802.11a im 2,4 GHz-Band, ist dabei aber abwärtskompatibel zu 802.11b-Netzwerken.

Unabhängig von den zugrundeliegenden Transportmechanismen verfügen 802.11-Netzwerke über diverse Sicherheitsmechanismen. Der ursprüngliche 802.11-Standard definierte lediglich ein einfaches Sicherheitsprotokoll namens WEP. Dieses Protokoll verwendet einen fixen, gemeinsam verwendeten Schlüssel sowie die RC4-Kryptografie-Chiffre, um Daten verschlüsselt über das drahtlose Netzwerk zu senden. Alle Stationen des Netzwerks müssen sich auf den gleichen fixen Schlüssel einigen, um miteinander kommunizieren zu können. Dieses Schema ist sehr leicht zu knacken und wird deshalb heute kaum mehr eingesetzt. Aktuelle Sicherheitsmechanismen bauen auf dem Standard IEEE® 802.11i auf, der neue kryptographische Schlüssel (Chiffren), ein neues Protokoll für die Anmeldung von Stationen an einem Access Point, sowie Mechanismen zum Austausch von Schlüsseln als Vorbereitung der Kommunikation zwischen verschiedenen Geräten festlegt. Kryptografische Schlüssel werden in regelmäßigen Abständen aktualisiert. Außerdem gibt es Mechanismen zur Feststellung und Prävention von Einbruchsversuchen. Ein weiteres häufig verwendetes Sicherheitsprotokoll ist WPA. Dabei handelt es sich um einen Vorläufer von 802.11i, der von einem Industriekonsortium als Zwischenlösung bis zur endgültigen Verabschiedung von 802.11i entwickelt wurde. WPA definiert eine Untergruppe der Anforderungen des 802.11i-Standards und ist für den Einsatz in älterer Hardware vorgesehen. WPA benötigt nur den TKIP-Chiffre, welcher auf dem ursprünglichen WEP-Code basiert. 802.11i erlaubt zwar auch die Verwendung von TKIP, benötigt aber zusätzlich eine stärkere Chiffre (AES-CCM) für die Datenverschlüsselung. AES war für WPA nicht vorgesehen, weil man es als zu rechenintensiv für den Einsatz in älteren Geräten ansah.

Ein weiterer zu beachtender Standard ist 802.11e. Dieser definiert Protokolle zur Übertragung von Multimedia-Anwendungen, wie das Streaming von Videodateien oder Voice-over-IP (VoIP) in einem 802.11-Netzwerk. Analog zu 802.11i verfügt auch 802.11e über eine vorläufige Spezifikation namens WMM (ursprünglich WME), die von einem Industriekonsortium als Untergruppe von 802.11e spezifiziert wurde, um Multimedia-Anwendungen bereits vor der endgültigen Verabschiedung des 802.11e-Standards implementieren zu können. 802.11e sowie WME/WMM erlauben eine Prioritätenvergabe beim Datentransfer im einem drahtlosen Netzwerk. Möglich wird dies durch den

Einsatz von Quality of Service-Protokollen (QoS) und erweiterten Medienzugriffsprotokollen. Werden diese Protokolle korrekt implementiert, erlauben sie hohe Datenübertragungsraten und einen priorisierten Datenfluss.

FreeBSD unterstützt die Standards 802.11a, 802.11b und 802.11g. Ebenfalls unterstützt werden WPA sowie die Sicherheitsprotokolle gemäß 802.11i (sowohl für 11a, 11b als auch 11g). QoS und Verkehrspriorisierung, die von den WME/WMM-Protokollen benötigt werden, werden für einen begrenzten Satz von drahtlosen Geräten unterstützt.

31.3.2. Basiskonfiguration

31.3.2.1. Kernelkonfiguration

Um ein drahtloses Netzwerk zu nutzen, wird eine drahtlose Netzwerkkarte benötigt und ein Kernel, der drahtlose Netzwerke unterstützt. Der Kernel unterstützt den Einsatz von Kernelmodulen. Daher muss nur die Unterstützung für die verwendeten Geräte aktiviert werden.

Die meisten drahtlosen Geräte verwenden Bauteile von Atheros und werden deshalb vom ath(4)-Treiber unterstützt. Um diesen Treiber zu verwenden, muss die folgende Zeile in `/boot/loader.conf` hinzugefügt werden:

```
if_ath_load="YES"
```

Der Atheros-Treiber besteht aus drei Teilen: dem Treiber selbst (ath(4)), dem Hardware-Support-Layer für die chipspezifischen Funktionen (ath_hal(4)) sowie einem Algorithmus zur Auswahl der Frame-Übertragungsrate (ath_rate_sample). Wenn diese Unterstützung als Kernelmodul geladen wird, kümmert sich das Modul automatisch um Abhängigkeiten. Um die Unterstützung für ein anderes drahtloses Gerät zu laden, geben Sie das entsprechende Modul für dieses Gerät an. Dieses Beispiel zeigt die Verwendung von Geräten, die auf Bauteilen von Intersil Prism basieren und den Treiber wi(4) benötigen:

```
if_wi_load="YES"
```

Anmerkung

Die Beispiele in diesem Abschnitt verwenden den ath(4)-Treiber. Verwenden Sie ein anderes Gerät, muss der Gerätename an die Konfiguration angepasst werden. Eine Liste aller verfügbaren Treiber und unterstützten drahtlosen Geräte finden sich in den FreeBSD Hardware Notes unter Release Information der FreeBSD Homepage. Gibt es keinen nativen FreeBSD-Treiber für das drahtlose Gerät, kann möglicherweise mit NDIS ein Windows®-Treiber verwendet werden.

Zusätzlich müssen die Module zur Verschlüsselung des drahtlosen Netzwerks geladen werden. Diese werden normalerweise dynamisch vom wlan(4)-Modul geladen. Im folgenden Beispiel erfolgt allerdings eine manuelle Konfiguration. Folgende Module sind verfügbar: wlan_wep(4), wlan_ccmp(4) und wlan_tkip(4). Sowohl wlan_ccmp(4) als auch wlan_tkip(4) werden nur benötigt, wenn WPA und/oder die Sicherheitsprotokolle von 802.11i verwendet werden. Wenn das Netzwerk keine Verschlüsselung verwendet, wird die wlan_wep(4)-Unterstützung nicht benötigt. Um diese Module beim Systemstart zu laden, fügen Sie folgende Zeilen in `/boot/loader.conf` ein:

```
wlan_wep_load="YES"
wlan_ccmp_load="YES"
wlan_tkip_load="YES"
```

Sobald diese Einträge in `/boot/loader.conf` vorhanden sind, muss das FreeBSD-System neu gestartet werden. Alternativ können die Kernelmodule auch manuell mit kldload(8) geladen werden.

Anmerkung

Benutzer, die keine Kernelmodule verwenden wollen, können die benötigten Treiber auch in den Kernel kompilieren. Dazu müssen die folgenden Zeilen in die Kernelkonfigurationsdatei aufgenommen werden:

```
device wlan              # 802.11 support
device wlan_wep          # 802.11 WEP support
device wlan_ccmp         # 802.11 CCMP support
device wlan_tkip         # 802.11 TKIP support
device wlan_amrr         # AMRR transmit rate control algorithm
device ath               # Atheros pci/cardbus NIC's
device ath_hal           # pci/cardbus chip support
options AH_SUPPORT_AR5416 # enable AR5416 tx/rx descriptors
device ath_rate_sample   # SampleRate tx rate control for ath
```

Mit diesen Informationen in der Kernelkonfigurationsdatei kann der Kernel neu gebaut, und das FreeBSD-System anschließend neu gestartet werden.

Informationen über das drahtlose Gerät sollten in den Boot-Meldungen folgendermaßen angezeigt werden:

```
ath0: <Atheros 5212> mem 0x88000000-0x8800ffff irq 11 at device 0.0 on cardbus1
ath0: [ITHREAD]
ath0: AR2413 mac 7.9 RF2413 phy 4.5
```

31.3.3. Infrastruktur-Modus

Drahtlose Netzwerke werden in der Regel im Infrastruktur-Modus (BSS) betrieben. Dazu werden mehrere drahtlose Access Points zu einem gemeinsamen drahtlosen Netzwerk verbunden. Jedes dieser drahtlosen Netzwerke hat einen eigenen Namen, der als >SSID> bezeichnet wird. Alle Clients eines drahtlosen Netzwerks verbinden sich in diesem Modus mit einem Access Point.

31.3.3.1. FreeBSD-Clients

31.3.3.1.1. Einen Access Point finden

Um nach verfügbaren drahtlosen Netzwerken zu suchen verwenden Sie ifconfig(8). Dieser Scanvorgang kann einen Moment dauern, da jede verfügbare Frequenz auf verfügbare Access Points hin überprüft werden muss. Nur der Super-User kann einen Scanvorgang starten:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 up scan
SSID/MESH ID    BSSID              CHAN RATE   S:N     INT CAPS
dlinkap         00:13:46:49:41:76   11   54M -90:96   100 EPS  WPA WME
freebsdap       00:11:95:c3:0d:ac    1   54M -83:96   100 EPS  WPA
```

Anmerkung

Die Netzwerkkarte muss in den Status `up` versetzt werden, bevor der erste Scanvorgang gestartet werden kann. Für spätere Scans ist dies aber nicht mehr erforderlich.

Als Ergebnis erhalten Sie eine Liste mit allen gefundenen BSS/IBSS-Netzwerken. Zusätzlich zum Namen des Netzwerks, der `SSID`, wird auch die `BSSID` ausgegeben. Dabei handelt es sich um die MAC-Adresse des Access Points. Das Feld `CAPS` gibt den Typ des Netzwerks sowie die Fähigkeiten der Stationen innerhalb des Netzwerks an:

Tabelle 31.2. Station Capability Codes

Capability Code	Bedeutung
E	Extended Service Set (ESS). Zeigt an, dass die Station Teil eines Infrastruktur-Netzwerks ist, und nicht eines IBSS/Ad-hoc-Netzwerks.
I	IBSS/Ad-hoc-Netzwerk. Die Station ist Teil eines Ad-hoc-Netzwerks und nicht eines ESS-Netzwerks.
P	Privacy. Alle Datenframes, die innerhalb des BSS ausgetauscht werden, sind verschlüsselt. Dieses BSS verwendet dazu kryptographische Verfahren wie WEP, TKIP oder AES-CCMP.
S	Short Preamble. Das Netzwerk verwendet eine kurze Präambel (definiert in 802.11b High Rate/DSSS PHY). Eine kurze Präambel verwendet ein 56 Bit langes Sync-Feld, im Gegensatz zu einer langen Präambel, die ein 128 Bit langes Sync-Feld verwendet.
s	Short slot time. Das 802.11g-Netzwerk verwendet eine kurze Slotzeit, da es in diesem Netzwerk keine veralteten (802.11b) Geräte gibt.

Um eine Liste der bekannten Netzwerke auszugeben, verwenden Sie den folgenden Befehl:

```
# ifconfig wlan0 list scan
```

Diese Liste kann entweder automatisch durch das drahtlose Gerät oder manuell durch eine `scan`-Aufforderung aktualisiert werden. Veraltete Informationen werden dabei automatisch entfernt.

31.3.3.1.2. Basiseinstellungen

Dieser Abschnitt beschreibt, wie Sie eine drahtlose Netzwerkkarte ohne Verschlüsselung unter FreeBSD einrichten. Nachdem Sie sich mit den Informationen dieses Abschnitts vertraut gemacht haben, sollten Sie das drahtlose Netzwerk mit WPA verschlüsseln.

Das Einrichten eines drahtlosen Netzwerks erfolgt in drei Schritten: Der Auswahl eines Access Points, die Anmeldung der Station sowie der Konfiguration der IP-Adresse.

31.3.3.1.2.1. Einen Access Point auswählen

Im Normalfall wird sich die Station automatisch mit einem der zur Verfügung stehenden Access Points verbinden. Dazu muss lediglich das drahtlose Gerät aktiviert, oder in `/etc/rc.conf` eingetragen sein:

```
wlans_ath0="wlan0"
ifconfig_wlan0="DHCP"
```

Stehen mehrere Access Points zur Verfügung, kann ein spezifischer durch Angabe der SSID gewählt werden:

```
wlans_ath0="wlan0"
ifconfig_wlan0="ssid Ihre_SSID DHCP"
```

Gibt es in einem Netzwerk mehrere Access Points mit der gleichen SSID, was das Routing vereinfacht, kann es notwending sein, dass ein bestimmtes Gerät verbunden werden muss. Dazu muss lediglich die BSSID des Access Points angeben werden. Die Angabe der SSID ist hierbei nicht zwingend notwendig:

```
wlans_ath0="wlan0"
ifconfig_wlan0="ssid Ihre_SSID bssid xx:xx:xx:xx:xx:xx DHCP"
```

Es gibt noch weitere Möglichkeiten, den Zugriff auf bestimmte Access Point zu beschränken, beispielsweise durch die Begrenzung der Frequenzen, auf denen eine Station nach einem Access Point sucht. Sinnvoll ist ein solches Vorgehen beispielsweise, wenn das drahtlose Gerät in verschiedenen Frequenzbereichen arbeiten kann, da in die-

sem Fall das Prüfen aller Frequenzen sehr zeitintensiv sein kann. Um nur innerhalb eines bestimmten Frequenzbereichs nach einem Access Point zu suchen, verwenden Sie die Option `mode`:

```
wlans_ath0="wlan0"
ifconfig_wlan0="mode 11g ssid Ihre_SSID DHCP"
```

In diesem Beispiel sucht das drahtlose Gerät nur im 2,4 GHz-Band (802.11g), aber nicht innerhalb des 5 GHz-Bandes nach einem Access Point. Mit der Option `channel` kann eine bestimmte Frequenz vorgegeben werden, auf der gesucht werden soll. Die Option `chanlist` erlaubt die Angabe mehrerer erlaubter Frequenzen. Eine umfassende Beschreibung dieser Optionen finden Sie in ifconfig(8).

31.3.3.1.2.2. Authentifizierung

Sobald ein Access Point gefunden wurde, muss sich die Station am Access Point authentifizieren, bevor Daten übertragen werden können. Dazu gibt es verschiedene Möglichkeiten. Am häufigsten wird die sogenannte *offene Authentifizierung* verwendet. Dabei wird es jeder Station erlaubt, sich mit einem Netzwerk zu verbinden und Daten zu übertragen. Aus Sicherheitsgründen sollte diese Methode allerdings nur zu Testzwecken bei der erstmaligen Einrichtung eines drahtlosen Netzwerks verwendet werden. Andere Authentifizierungsmechanismen erfordern den Austausch kryptographischer Informationen, bevor sie die Übertragung von Daten erlauben. Dazu gehören der Austausch fixer (vorher vereinbarter) Schlüssel oder Kennwörter, sowie der Einsatz komplexerer Verfahren mit Backend-Diensten wie RADIUS. Die offene Authentifizierung ist die Voreinstellung. Am zweithäufigsten kommt das im Abschnitt 31.3.3.1.3.1, „WPA-PSK“ beschriebene WPA-PSK zum Einsatz, welches auch als WPA Personal bezeichnet wird.

Anmerkung

Kommt eine Apple® AirPort® Extreme-Basisstation als Access Point zum Einsatz, muss sowohl die Shared-Key-Authentifizierung als auch ein WEP-Schlüssel konfiguriert werden. Die entsprechende Konfiguration erfolgt entweder in `/etc/rc.conf` oder über das Programm wpa_supplicant(8). Für eine einzelne AirPort®-Basisstation kann der Zugriff wie folgt konfiguriert werden:

```
wlans_ath0="wlan0"
ifconfig_wlan0="authmode shared wepmode on weptxkey 1 wepkey 01234567 ↺
DHCP"
```

Normalerweise sollte Shared-Key-Authentifizierung nicht verwendet werden, da diese die Sicherheit des WEP-Schlüssel noch weiter verringert. Wenn WEP für Kompatibilität mit älteren Geräten verwendet werden muss, ist es besser, WEP mit offener Authentifizierung zu verwenden. Weitere Informationen zu WEP finden Sie im Abschnitt 31.3.3.1.4, „WEP“.

31.3.3.1.2.3. Eine IP-Adresse über DHCP beziehen

Sobald ein Access Point ausgewählt ist und die Authentifizierungsparameter festgelegt sind, wird eine IP-Adresse benötigt. In der Regel wird die IP-Adresse über DHCP bezogen. Um dies zu erreichen, bearbeiten Sie `/etc/rc.conf` und fügen Sie `DHCP` für das drahtlose Gerät in die Konfiguration hinzu:

```
wlans_ath0="wlan0"
ifconfig_wlan0="DHCP"
```

Das drahtlose Gerät kann nun gestartet werden:

```
# service netif start
```

Nachdem das Gerät aktiviert wurde, kann mit ifconfig(8) der Status des Geräts `ath0` abgefragt werden:

```
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        ether 00:11:95:d5:43:62
```

```
        inet 192.168.1.100 netmask 0xffffff00 broadcast 192.168.1.255
        media: IEEE 802.11 Wireless Ethernet OFDM/54Mbps mode 11g
        status: associated
        ssid dlinkap channel 11 (2462 Mhz 11g) bssid 00:13:46:49:41:76
        country US ecm authmode OPEN privacy OFF txpower 21.5 bmiss 7
        scanvalid 60 bgscan bgscanintvl 300 bgscanidle 250 roam:rssi 7
        roam:rate 5 protmode CTS wme burst
```

`status: associated` besagt, dass sich das Gerät mit dem drahtlosen Netzwerk verbunden hat. `bssid 00:13:46:49:41:76` ist die MAC-Adresse des Access Points und `authmode OPEN` zeigt an, dass die Kommunikation nicht verschlüsselt wird.

31.3.3.1.2.4. Statische IP-Adressen

Wenn eine IP-Adresse nicht von einem DHCP-Server bezogen werden kann, vergeben Sie eine statische IP-Adresse. Ersetzten Sie dazu das oben gezeigte Schlüsselwort `DHCP` durch die entsprechende IP-Adresse. Beachten Sie dabei, dass Sie die anderen Konfigurationsparameter nicht versehentlich verändern:

```
wlans_ath0="wlan0"
ifconfig_wlan0="inet 192.168.1.100  netmask 255.255.255.0  ssid your_ssid_here "
```

31.3.3.1.3. WPA

Wi-Fi Protected Access (WPA) ist ein Sicherheitsprotokoll, das in 802.11-Netzwerken verwendet wird, um die fehlende Authentifizierung und Schwächen von WEP zu vermeiden. WPA stellt das aktuelle 802.1X-Authentifizierungsprotokoll dar und verwendet eine von mehreren Chiffren, um die Datensicherheit zu gewährleisten. Die einzige Chiffre, die von WPA verlangt wird, ist *Temporary Key Integrity Protocol* (TKIP). TKIP ist eine Chiffre, die die von WEP verwendete RC4-Chiffre um Funktionen zur Prüfung der Datenintegrität und zur Erkennung und Bekämpfung von Einbruchsversuchen erweitert. TKIP ist durch Softwaremodifikationen auch unter veralteter Hardware lauffähig. Im Vergleich zu WEP ist WPA zwar sehr viel sicherer, es ist aber dennoch nicht völlig immun gegen Angriffe. WPA definiert mit AES-CCMP noch eine weitere Chiffre als Alternative zu TKIP. AES-CCMP, welches häufig als WPA2 oder RSN bezeichnet wird, sollte bevorzugt eingesetzt werden.

WPA definiert Authentifizierungs- und Verschlüsselungsprotokolle. Die Authentifizierung erfolgt in der Regel über eine der folgenden Techniken: 802.1X gemeinsam mit einem Backend-Authentifizierungsdienst wie RADIUS, oder durch einen Minimal-Handshake zwischen der Station und dem Access Point mit einem vorher vereinbarten gemeinsamen Schlüssel. Die erste Technik wird als WPA Enterprise, die zweite hingegen als WPA Personal bezeichnet. Da sich der Aufwand für das Aufsetzen eines RADIUS-Backend-Servers für die meisten drahtlosen Netzwerke nicht lohnt, wird WPA in der Regel als WPA-PSK konfiguriert.

Die Kontrolle der drahtlosen Verbindung sowie das Aushandeln des Schlüssel, oder die Authentifizierung mit einem Server, erfolgt über wpa_supplicant(8). Dieses Programm benötigt eine Konfigurationsdatei, `/etc/wpa_supplicant.conf`. Weitere Informationen finden Sie in wpa_supplicant.conf(5).

31.3.3.1.3.1. WPA-PSK

WPA-PSK, das auch als WPA-Personal bekannt ist, basiert auf einem gemeinsamen, vorher vereinbarten Schlüssel (PSK), der aus einem Passwort generiert und danach als Master-Key des drahtlosen Netzwerks verwendet wird. Jeder Benutzer des drahtlosen Netzwerks verwendet daher *den gleichen* Schlüssel. WPA-PSK sollte nur in kleinen Netzwerken eingesetzt werden, in denen die Konfiguration eines Authentifizierungsservers nicht möglich oder erwünscht ist.

Warnung

Achten Sie darauf, immer starke Passwörter zu verwenden, die ausreichend lang sind und auch Sonderzeichen enthalten, damit diese nicht leicht erraten oder umgangen werden können.

Der erste Schritt zum Einsatz von WPA-PSK ist die Konfiguration der SSID und des gemeinsamen Schlüssels des Netzwerks in `/etc/wpa_supplicant.conf`:

```
network={
  ssid="freebsdap"
  psk="freebsdmall"
}
```

Danach wird in `/etc/rc.conf` definiert, dass WPA zur Verschlüsselung eingesetzt werden soll und dass die IP-Adresse über DHCP bezogen wird:

```
wlans_ath0="wlan0"
ifconfig_wlan0="WPA DHCP"
```

Nun kann das drahtlose Gerät aktiviert werden:

```
# service netif start
Starting wpa_supplicant.
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 5
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 6
DHCPOFFER from 192.168.0.1
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.1
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet OFDM/36Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 MHz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

Alternativ kann das drahtlose Gerät manuell, mit Hilfe der Informationen aus `/etc/wpa_supplicant.conf` konfiguriert werden:

```
# wpa_supplicant -i wlan0 -c /etc/wpa_supplicant.conf
Trying to associate with 00:11:95:c3:0d:ac (SSID='freebsdap' freq=2412 MHz)
Associated with 00:11:95:c3:0d:ac
WPA: Key negotiation completed with 00:11:95:c3:0d:ac [PTK=CCMP GTK=CCMP]
CTRL-EVENT-CONNECTED - Connection to 00:11:95:c3:0d:ac completed (auth) [id=0 id_str=]
```

Im zweiten Schritt starten Sie nun dhclient(8), um eine IP-Adresse vom DHCP-Server zu beziehen:

```
# dhclient wlan0
DHCPREQUEST on wlan0 to 255.255.255.255 port 67
DHCPACK from 192.168.0.1
bound to 192.168.0.254 -- renewal in 300 seconds.
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet OFDM/36Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 MHz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

Anmerkung

Enthält `/etc/rc.conf` bereits die Zeile `ifconfig_wlan0="DHCP"`, wird dhclient(8) automatisch gestartet, nachdem wpa_supplicant(8) sich mit dem Access Point verbunden hat.

Sollte der Einsatz von DHCP nicht möglich oder nicht gewünscht sein, konfigurieren Sie eine statische IP-Adresse, nachdem wpa_supplicant(8) die Station authentifiziert hat:

```
# ifconfig wlan0 inet 192.168.0.100  netmask 255.255.255.0
# ifconfig wlan0
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.100 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet OFDM/36Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 MHz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

Falls DHCP nicht verwendet wird, müssen zusätzlich noch das Standard-Gateway sowie der Nameserver manuell festgelegt werden:

```
# route add default  your_default_router
# echo "nameserver  your_DNS_server " >> /etc/resolv.conf
```

31.3.3.1.3.2. WPA und EAP-TLS

Die zweite Möglichkeit, WPA einzusetzen, ist die Verwendung eines 802.1X-Backend-Authentifizierungsservers. Diese Variante wird als WPA-Enterprise bezeichnet, um sie vom weniger sicheren WPA-Personal abzugrenzen. Die bei WPA-Enterprise verwendete Authentifizierung basiert auf dem *Extensible Authentication Protocol* (EAP).

EAP selbst bietet keine Verschlüsselung, sondern operiert in einem verschlüsselten Tunnel. Es gibt verschiedene auf EAP basierende Authentifizierungsmethoden, darunter EAP-TLS, EAP-TTLS und EAP-PEAP.

EAP mit *Transport Layers Security* (EAP-TLS) ist ein sehr gut unterstütztes Authentifizierungsprotokoll, da es sich dabei um die erste EAP-Methode handelt, die von der Wi-Fi Alliance zertifiziert wurde. EAP-TLS erfordert drei Zertifikate: Das auf allen Rechnern installierte CA-Zertifikat, das Server-Zertifikat des Authentifizierungsservers, sowie ein Client-Zertifikat für jeden drahtlosen Client. Sowohl der Authentifizierungsservers als auch die drahtlosen Clients authentifizieren sich gegenseitig durch ihre Zertifikate, wobei sie überprüfen, ob diese Zertifikate auch von der Zertifizierungs-Authorität (CA) des jeweiligen Unternehmens signiert wurden.

Die Konfiguration erfolgt (analog zu WPA-PSK) über `/etc/wpa_supplicant.conf`:

```
network={
  ssid="freebsdap" ❶
  proto=RSN  ❷
  key_mgmt=WPA-EAP ❸
  eap=TLS ❹
  identity="loader" ❺
  ca_cert="/etc/certs/cacert.pem" ❻
  client_cert="/etc/certs/clientcert.pem" ❼
  private_key="/etc/certs/clientkey.pem" ❽
  private_key_passwd="freebsdmallclient" ❾
}
```

❶ Der Name des Netzwerks (SSID).

❷ Das als WPA2 bekannte RSN IEEE® 802.11i Protokoll wird verwendet.
❸ Die `key_mgmt` -Zeile bezieht sich auf das verwendete Key-Management-Protokoll. In diesem Beispiel wird WPA gemeinsam mit der EAP-Authentifizierung verwendet.
❹ Die für die Verbindung verwendete EAP-Methode.
❺ Das `identity`-Feld enthält den von EAP verwendeten Identifizierungsstring.
❻ Das Feld `ca_cert` gibt den Pfad zum CA-Zertifikat an. Diese Datei wird zur Verifizierung des Server-Zertifikats benötigt.
❼ Die `client_cert`-Zeile gibt den Pfad zum Client-Zertifikat an. Jeder Client hat ein eigenes, innerhalb des Netzwerks eindeutiges, Zertifikat.
❽ Das Feld `private_key` gibt den Pfad zum privaten Schlüssel des Client-Zertifikat an.
❾ Das Feld `private_key_passwd` enthält die Passphrase für den privaten Schlüssel.

Danach fügen Sie die folgende Zeile in `/etc/rc.conf` ein:

```
wlans_ath0="wlan0"
ifconfig_wlan0="WPA DHCP"
```

Nun können Sie das drahtlose Gerät aktivieren:

```
# service netif start
Starting wpa_supplicant.
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 7
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 15
DHCPACK from 192.168.0.20
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet DS/11Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 MHz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

Alternativ kann das drahtlose Gerät manuell mit wpa_supplicant(8) und ifconfig(8) aktiviert werden.

31.3.3.1.3.3. WPA mit EAP-TTLS

Bei EAP-TLS müssen sowohl der Authentifizierungsserver als auch die Clients jeweils ein eigenes Zertifikat aufweisen. Bei EAP-TTLS ist das Client-Zertifikat optional. EAP-TTLS geht dabei vor wie ein Webserver, der einen sicheren SSL-Tunnel erzeugen kann, ohne dass der Besucher dabei über ein clientseitiges Zertifikat verfügen muss. EAP-TTLS verwendet einen verschlüsselten TLS-Tunnel zum sicheren Transport der Authentifizierungsdaten.

Die erforderliche Konfiguration erfolgt in `/etc/wpa_supplicant.conf`:

```
network={
  ssid="freebsdap"
  proto=RSN
  key_mgmt=WPA-EAP
  eap=TTLS ❶
  identity="test" ❷
  password="test" ❸
  ca_cert="/etc/certs/cacert.pem" ❹
  phase2="auth=MD5" ❺
}
```

❶ Die für die Verbindung verwendete EAP-Methode.
❷ Das `identity`-Feld enthält den Identifizierungsstring für die EAP-Authentifizierung innerhalb des verschlüsselten TLS-Tunnels.

❸ Das `password`-Feld enthält die Passphrase für die EAP-Authentifizierung.
❹ Das Feld `ca_cert` gibt den Pfad zum CA-Zertifikat an. Diese Datei wird zur Verifizierung des Server-Zertifikats benötigt.
❻ Die innerhalb des verschlüsselten TLS-Tunnels verwendete Authentifizierungsmethode. In Fall von PEAP ist dies `auth=MSCHAPV2`.

Folgende Zeilen müssen in `/etc/rc.conf` aufgenommen werden:

```
wlans_ath0="wlan0"
ifconfig_wlan0="WPA DHCP"
```

Nun kann das drahtlose Gerät aktiviert werden:

```
# service netif start
Starting wpa_supplicant.
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 7
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 15
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 21
DHCPACK from 192.168.0.20
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet DS/11Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 MHz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

31.3.3.1.3.4. WPA mit EAP-PEAP

Anmerkung

PEAPv0/EAP-MSCHAPv2 ist die gängigste PEAP-Methode. In diesem Kapitel wird der Begriff PEAP stellvertretend für diese Methode verwendet.

Protected EAP (PEAP) wurde als Alternative zu EAP-TTLS entwickelt und ist nach EAP-TLS der meist genutzte EAP-Standard. In einem Netzwerk mit verschiedenen Betriebssystemen sollte PEAP das am besten unterstützte Standard nach EAP-TLS sein.

PEAP arbeitet ähnlich wie EAP-TTLS. Es verwendet ein serverseitiges Zertifikat, um einen verschlüsselten TLS-Tunnel, über den die sichere Authentifizierung zwischen den Clients und dem Authentifizierungsserver erfolgt. In Sachen Sicherheit unterscheiden sich EAP-TTLS und PEAP allerdings: PEAP überträgt den Benutzernamen im Klartext und verschlüsselt nur das Passwort, während EAP-TTLS sowohl den Benutzernamen, als auch das Passwort über den TLS-Tunnel überträgt.

Um EAP-PEAP zu konfigurieren, fügen Sie die folgenden Zeilen in `/etc/wpa_supplicant.conf` ein:

```
network={
  ssid="freebsdap"
  proto=RSN
  key_mgmt=WPA-EAP
  eap=PEAP ❶
  identity="test" ❷
  password="test" ❸
  ca_cert="/etc/certs/cacert.pem" ❹
```

```
  phase1="peaplabel=0" ❺
  phase2="auth=MSCHAPV2" ❻
}
```

❶ Die für die Verbindung verwendete EAP-Methode.
❷ Das `identity`-Feld enthält den Identifizierungsstring für die innerhalb des verschlüsselten TLS-Tunnels erfolgende EAP-Authentifizierung.
❸ Das Feld `password` enthält die Passphrase für die EAP-Authentifizierung.
❹ Das Feld `ca_cert` gibt den Pfad zum CA-Zertifikat an. Diese Datei wird zur Verifizierung des Server-Zertifikats benötigt.
❺ Dieses Feld enthält die Parameter für die erste Phase der Authentifizierung, den TLS-Tunnel. Je nachdem, welcher Authentifizierungsserver benutzt wird, kann ein spezifisches Label für die Authentifizierung verwendet werden. Meistens lautet das Label „client EAP encryption“, dass durch `peaplabel=0` gesetzt wird. Weitere Informationen finden Sie in wpa_supplicant.conf(5).
❻ Das innerhalb des verschlüsselten TLS-Tunnels verwendete Authentifizierungsprotokoll. In unserem Beispiel handelt es sich dabei um `auth=MSCHAPV2`.

Danach fügen Sie die folgende Zeile in `/etc/rc.conf` ein:

```
ifconfig_ath0="WPA DHCP"
```

Nun kann das drahtlose Gerät aktiviert werden.

```
# service netif start
Starting wpa_supplicant.
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 7
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 15
DHCPREQUEST on wlan0 to 255.255.255.255 port 67 interval 21
DHCPACK from 192.168.0.20
bound to 192.168.0.254 -- renewal in 300 seconds.
wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.254 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet DS/11Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 MHz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode WPA2/802.11i privacy ON deftxkey UNDEF
      AES-CCM 3:128-bit txpower 21.5 bmiss 7 scanvalid 450 bgscan
      bgscanintvl 300 bgscanidle 250 roam:rssi 7 roam:rate 5 protmode CTS
      wme burst roaming MANUAL
```

31.3.3.1.4. WEP

Wired Equivalent Privacy (WEP) ist Teil des ursprünglichen 802.11-Standards. Es enthält keinen Authentifzierungsmechanismus und verfügt lediglich über eine schwache Zugriffskontrolle, die sehr leicht umgangen werden kann.

WEP kann über ifconfig(8) aktiviert werden:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 inet 192.168.1.100 netmask 255.255.255.0 \
           ssid my_net wepmode on weptxkey 3 wepkey 3:0x3456789012
```

- `weptxkey` definiert den WEP-Schlüssel, der für die Datenübertragung verwendet wird. Dieses Beispiel verwendet den dritten Schlüssel. Der gleiche Schlüssel muss auch am Access Point eingestellt sein. Kennen Sie den vom Access Point verwendeten Schlüssel nicht, sollten Sie zuerst den Wert 1 (den ersten Schlüssel) für diese Variable verwenden.

- `wepkey` legt den zu verwendenden WEP-Schlüssel in der Form *`Nummer:Schlüssel`* fest. Schlüssel 1 wird standardmäßig verwendet. Die "Nummer" muss nur angegeben werden, wenn ein anderer als der erste Schlüssel verwendet werden soll.

Anmerkung

Ersetzen Sie `0x3456789012` durch den am Access Point konfigurierten Schlüssel.

Weitere Informationen finden Sie in ifconfig(8).

Das Programm wpa_supplicant(8) eignet sich ebenfalls dazu, WEP für drahtlose Geräte zu aktivieren. Obige Konfiguration lässt sich dabei durch die Aufnahme der folgenden Zeilen in `/etc/wpa_supplicant.conf` realisieren:

```
network={
  ssid="my_net"
  key_mgmt=NONE
  wep_key3=3456789012
  wep_tx_keyidx=3
}
```

Danach müssen Sie das Programm noch aufrufen:

```
# wpa_supplicant -i  wlan0 -c /etc/wpa_supplicant.conf
Trying to associate with 00:13:46:49:41:76 (SSID='dlinkap' freq=2437 MHz)
Associated with 00:13:46:49:41:76
```

31.3.4. Ad-hoc-Modus

Der IBSS-Modus, der auch als Ad-hoc-Modus bezeichnet wird, ist für Punkt-zu-Punkt-Verbindungen vorgesehen. Um beispielsweise eine Ad-hoc-Verbindung zwischen den Rechnern `A` und `B` aufzubauen, werden lediglich zwei IP-Adressen und eine SSID benötigt.

Auf Rechner `A`:

```
# ifconfig wlan0 create wlandev ath0 wlanmode adhoc
# ifconfig wlan0 inet 192.168.0.1 netmask 255.255.255.0 ssid freebsdap
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:c3:0d:ac
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <adhoc>
      status: running
      ssid freebsdap channel 2 (2417 Mhz 11g) bssid 02:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy OFF txpower 21.5 scanvalid 60
      protmode CTS wme burst
```

Der `adhoc` -Parameter zeigt an, dass die Schnittstelle im IBSS-Modus läuft.

Rechner `B` sollte nun in der Lage sein, Rechner `A` zu finden:

```
# ifconfig wlan0 create wlandev ath0 wlanmode adhoc
# ifconfig wlan0 up scan
  SSID/MESH ID    BSSID              CHAN RATE   S:N     INT CAPS
  freebsdap       02:11:95:c3:0d:ac    2   54M -64:-96  100 IS   WME
```

Der Wert `I` (Spalte CAPS) in dieser Ausgabe bestätigt, dass sich Rechner `A` im Ad-hoc-Modus befindet. Nun müssen Sie noch Rechner `B` eine andere IP-Adresse zuweisen:

```
# ifconfig wlan0 inet 192.168.0.2 netmask 255.255.255.0 ssid freebsdap
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.2 netmask 0xffffff00 broadcast 192.168.0.255
```

```
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <adhoc>
      status: running
      ssid freebsdap channel 2 (2417 Mhz 11g) bssid 02:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy OFF txpower 21.5 scanvalid 60
      protmode CTS wme burst
```

Damit sind die Rechner `A` und `B` bereit und können untereinander Daten austauschen.

31.3.5. FreeBSD Host Access Points

FreeBSD kann als Access Point (AP) agieren. Dies verhindert, dass man sich einen Hardware AP kaufen oder ein Ad-hoc Netzwerk laufen lassen muss. Dies kann sinnvoll sein, falls der FreeBSD-Computer als Gateway zu einem anderen Netzwerk, wie dem Internet, fungiert.

31.3.5.1. Grundeinstellungen

Bevor Sie einen FreeBSD-Computer als AP konfigurieren, muss der Kernel mit der entsprechenden Netzwerkunterstützung für die drahtlose Karte, sowie die Sicherheitsprotokolle konfiguriert werden. Weitere Informationen finden Sie im Abschnitt 31.3.2, „Basiskonfiguration“.

Anmerkung

Die Verwendung der NDIS Treiber für Windows® erlauben zur Zeit keinen AP-Modus. Nur die nativen FreeBSD-Wireless-Treiber unterstützen den AP-Modus.

Nachdem die Netzwerkunterstützung geladen ist, überprüfen Sie, ob das Wireless-Gerät den hostbasierenden Access-Point Modus, der auch als hostap-Modus bekannt ist, unterstützt:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 list caps
drivercaps=6f85edc1<STA,FF,TURBOP,IBSS,HOSTAP,AHDEMO,TXPMGT,SHSLOT,SHPREAMBLE,MONITOR,MBSS,WPA1,WPA2,BURST,WM
cryptocaps=1f<WEP,TKIP,AES,AES_CCM,TKIPMIC>
```

Diese Ausgabe zeigt die Eigenschaften der Karte. Das Wort `HOSTAP` bestätigt, dass diese Wireless-Karte als AP agieren kann. Die verschiedenen unterstützten Algorithmen werden ebenfalls angezeigt: WEP, TKIP und AES. Diese Informationen zeigen an, welche Sicherheitsprotokolle auf dem AP nutzbar sind.

Das Wireless-Gerät kann nur während der Erzeugung des Pseudo-Geräts in den hostap-Modus gesetzt werden. Zuvor erstellte Pseudo-Geräte müssen also vorher zerstört werden:

```
# ifconfig wlan0 destroy
```

Danach muss das Gerät erneut erstellt werden, bevor die restlichen Netzwerkparameter konfiguriert werden können:

```
# ifconfig wlan0 create wlandev ath0 wlanmode hostap
# ifconfig wlan0 inet 192.168.0.1 netmask 255.255.255.0 ssid freebsdap mode 11g channel 1
```

Benutzen Sie danach erneut ifconfig(8), um den Status der `wlan0`-Schnittstelle abzufragen:

```
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:c3:0d:ac
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <hostap>
      status: running
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy OFF txpower 21.5 scanvalid 60
      protmode CTS wme burst dtimperiod 1 -dfs
```

Die `hostap`-Parameter geben die Schnittstelle an, die im hostbasierenden Access Point Modus läuft.

Die Konfiguration der Schnittstelle kann durch Hinzufügen der folgenden Zeilen in die Datei `/etc/rc.conf` automatisch während des Bootvorganges erfolgen:

```
wlans_ath0="wlan0"
create_args_wlan0="wlanmode hostap"
ifconfig_wlan0="inet 192.168.0.1 netmask 255.255.255.0 ssid freebsdap mode 11g channel 1"
```

31.3.5.2. Hostbasierender Access Point ohne Authentifizierung oder Verschlüsselung

Obwohl es nicht empfohlen wird, einen AP ohne jegliche Authentifizierung oder Verschlüsselung laufen zu lassen, ist es eine einfache Art zu testen, ob der AP funktioniert. Diese Konfiguration ist auch wichtig für die Fehlersuche bei Client-Problemen.

Nachdem der AP konfiguriert wurde, ist es möglich von einem anderen drahtlosen Computer eine Suche nach dem AP zu starten:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 up scan
SSID/MESH ID    BSSID              CHAN RATE   S:N     INT CAPS
freebsdap       00:11:95:c3:0d:ac    1   54M -66:-96  100 ES   WME
```

Der Client-Rechner hat den AP gefunden und kann nun eine Verbindung aufbauen:

```
# ifconfig wlan0 inet 192.168.0.2 netmask 255.255.255.0 ssid freebsdap
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:d5:43:62
      inet 192.168.0.2 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet OFDM/54Mbps mode 11g
      status: associated
      ssid freebsdap channel 1 (2412 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy OFF txpower 21.5 bmiss 7
      scanvalid 60 bgscan bgscanintvl 300 bgscanidle 250 roam:rssi 7
      roam:rate 5 protmode CTS wme burst
```

31.3.5.3. WPA-hostbasierter Access Point

Dieser Abschnitt beschäftigt sich mit der Konfiguration eines FreeBSD-AP mit dem WPA-Sicherheitsprotokoll. Weitere Einzelheiten zu WPA und der Konfiguration von Clients mit WPA finden Sie im Abschnitt 31.3.3.1.3, „WPA“.

Der hostapd-Dienst wird genutzt, um die Client-Authentifizierung und das Schlüsselmanagement auf dem AP mit aktiviertem WPA zu nutzen.

Die folgende Konfiguration wird auf dem FreeBSD-Computer ausgeführt, der als AP agiert. Nachdem der AP korrekt arbeitet, sollte hostapd(8) automatisch beim Booten durch folgende Zeile in `/etc/rc.conf` aktiviert werden:

```
hostapd_enable="YES"
```

Bevor Sie versuchen hostapd(8) zu konfigurieren, konfigurieren Sie zunächst die Grundeinstellungen, wie im Abschnitt 31.3.5.1, „Grundeinstellungen“ beschrieben.

31.3.5.3.1. WPA-PSK

WPA-PSK ist für kleine Netzwerke gedacht, in denen die Verwendung eines Authentifizierungs-Backend-Server nicht möglich oder nicht erwünscht ist.

Die Konfiguration wird in `/etc/hostapd.conf` durchgeführt:

```
interface=wlan0 ❶
debug=1 ❷
ctrl_interface=/var/run/hostapd ❸
ctrl_interface_group=wheel ❹
ssid=freebsdap ❺
wpa=1 ❻
wpa_passphrase=freebsdmall ❼
```

```
wpa_key_mgmt=WPA-PSK ❽
wpa_pairwise=CCMP TKIP ❾
```

❶ Dieses Feld zeigt die Wireless-Schnittstelle an, die für den Access Point verwendet wird an.
❷ Dieses Feld legt den debuglevel von hostapd(8) während der Ausführung fest. Ein Wert von 1 ist der kleinste zulässige Wert.
❸ Das `ctrl_interface`-Feld gibt den Pfadnamen des Verzeichnisses an, der von hostapd(8) genutzt wird, um die Domain-Socket-Dateien zu speichern, die für die Kommunikation mit externen Programmen, wie z.B. hostapd_cli(8), benutzt werden. In diesem Beispiel wird der Standardwert verwendet.
❹ Die Zeile `ctrl_interface_group` legt fest, welche Gruppe die Erlaubnis hat, die Schnittstellendateien zu kontrollieren.
❺ Dieses Feld setzt den Netzwerknamen.
❻ Das `wpa`-Feld aktiviert WPA und gibt an welches WPA-Authentifizierungprotokoll benötigt wird. Ein Wert von 1 konfiguriert den AP mit WPA-PSK.
❼ Das `wpa_passphrase`-Feld beinhaltet das ASCII-Passwort für die WPA-Authentifizierung.

> **Warnung**
>
> Achten Sie darauf, immer starke Passwörter zu verwenden, die ausreichend lang sind und auch Sonderzeichen enthalten, damit diese nicht leicht erraten oder umgangen werden können.

❽ Die `wpa_key_mgmt` Zeile zeigt das verwendete Schlüsselmanagement-Protokoll. Dieses Beispiel nutzt WPA-PSK.
❾ Das `wpa_pairwise` Feld zeigt die zulässigen Verschlüsselungsverfahren des AP. Hier werden beide, TKIP (WPA) und CCMP (WPA2) akzeptiert. CCMP-Verschlüsselung ist eine Alternative zu TKIP und sollte wenn möglich eingesetzt werden. TKIP sollte nur da eingesetzt werden, wo kein CCMP möglich ist.

Als nächstes wird hostapd gestartet:

```
# service hostapd forcestart
```

```
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 2290
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      inet6 fe80::211:95ff:fec3:dac%ath0 prefixlen 64 scopeid 0x4
      ether 00:11:95:c3:0d:ac
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <hostap>
      status: associated
      ssid freebsdap channel 1 bssid 00:11:95:c3:0d:ac
      authmode WPA2/802.11i privacy MIXED deftxkey 2 TKIP 2:128-bit txpowmax 36 ↺
protmode CTS dtimperiod 1 bintval 100
```

Sobald der AP läuft, können sich die Clients mit ihm verbinden. Weitere Informationen finden Sie im Abschnitt 31.3.3.1.3, „WPA“. Es ist möglich zu sehen, welche Stationen mit dem AP verbunden sind. Geben Sie dazu `ifconfig wlan0 list sta` ein.

31.3.5.4. WEP-hostbasierter Access Point

Es ist nicht empfehlenswert, einen AP mit WEP zu konfigurieren, da es keine Authentifikationsmechanismen gibt und WEP leicht zu knacken ist. Einige ältere drahtlose Karten unterstützen nur WEP als Sicherheitsprotokoll. Diese Karten können nur mit einem AP ohne Authentifikation oder Verschlüsselung genutzt werden.

Das Wireless-Gerät kann nun in den hostap-Modus versetzt werden und mit der korrekten SSID und IP-Adresse konfiguriert werden:

```
# ifconfig wlan0 create wlandev ath0 wlanmode hostap
# ifconfig wlan0 inet 192.168.0.1 netmask 255.255.255.0 \
```

```
  ssid freebsdap  wepmode on weptxkey  3 wepkey 3:0x3456789012  mode 11g
```

- Der `weptxkey` zeigt an, welcher WEP-Schlüssel bei der Übertragung benutzt wird. In diesem Beispiel wird der dritte Schlüssel benutzt, da die Nummerierung bei 1 beginnt. Dieser Parameter muss angegeben werden, damit die Daten verschlüsselt werden.

- Der `wepkey` gibt den gewählten WEP-Schlüssel an. Er sollte im folgenden Format *`index:key`* vorliegen. Wenn kein Index vorhanden ist, wird der Schlüssel `1` benutzt. Ansonsten muss der Index manuell festgelegt werden.

Benutzen Sie ifconfig(8) um den Status der `wlan0`-Schnittstelle erneut anzuzeigen:

```
# ifconfig wlan0
  wlan0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
      ether 00:11:95:c3:0d:ac
      inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
      media: IEEE 802.11 Wireless Ethernet autoselect mode 11g <hostap>
      status: running
      ssid freebsdap channel 4 (2427 Mhz 11g) bssid 00:11:95:c3:0d:ac
      country US ecm authmode OPEN privacy ON deftxkey 3 wepkey 3:40-bit
      txpower 21.5 scanvalid 60 protmode CTS wme burst dtimperiod 1 -dfs
```

Es ist möglich, von einem anderen drahtlosen Computer eine Suche nach dem AP zu starten:

```
# ifconfig wlan0 create wlandev ath0
# ifconfig wlan0 up scan
SSID            BSSID              CHAN RATE  S:N   INT CAPS
freebsdap       00:11:95:c3:0d:ac    1   54M 22:1   100 EPS
```

Der Client-Rechner hat den AP gefunden und kann nun eine Verbindung aufbauen. Weitere Informationen finden Sie im Abschnitt 31.3.3.1.4, „WEP“.

31.3.6. Benutzung von drahtgebundenen und drahtlosen Verbindungen

Eine Verbindung per Kabel bietet eine bessere Leistung und eine höhere Zuverlässigkeit, während die Wireless-Verbindung eine höhere Flexibilität und Mobilität bietet. Benutzer von Laptops wollen normalerweise beides nutzen und zwischen beiden Verbindungen hin und her schalten.

Unter FreeBSD ist es möglich zwei oder mehr Netzwerkschnittstellen in einem „failover“-Mode zu kombinieren. Diese Konfiguration nutzt die beste verfügbare Verbindung aus einer Gruppe von Netzwerkverbindungen. Sobald sich der Linkstatus ändert, wechselt das Betriebssystem automatisch auf eine andere Verbindung.

Link-Aggregation und Failover werden im Abschnitt 31.7, „Link-Aggregation und Failover“ behandelt. Ein Beispiel für die Verwendung von kabelgebundenen und drahtlosen Verbindungen gibt es im Beispiel 31.3, „Failover Modus zwischen Ethernet- und Wireless-Schnittstellen“.

31.3.7. Problembehandlung

Dieser Abschnitt beschreibt eine Reihe von Maßnahmen zur Behebung von alltäglichen Problemen mit Drahtlosnetzwerken.

- Wird der Access Point bei der Suche nicht gefunden, überprüfen Sie, dass die Konfiguration des drahtlosen Geräts nicht die Anzahl der Kanäle beschränkt.

- Wenn sich das Gerät nicht mit dem Access Point verbinden kann, überprüfen Sie, ob die Konfiguration der Station auch der des Access Points entspricht. Dazu gehören auch die Authentifzierungsmethode und die Sicherheitsprotokolle. Halten Sie die Konfiguration so einfach wie möglich. Verwenden Sie ein Sicherheitsprotokoll wie WPA oder WEP, können Sie testweise den Access Point auf *offene Authentifizierung* und *keine Sicherheit* einstellen.

- Sobald sich das Gerät mit dem Access Point verbinden kann, prüfen Sie die Sicherheitseinstellungen mit einfachen Werkzeugen wie ping(8).

  Für die Fehlersuche steht wpa_supplicant(8) zur Verfügung. Starten Sie das Programm manuell mit der Option `-dd` und durchsuchen Sie anschließend die Systemprotokolle nach eventuellen Fehlermeldungen.

- Zusätzlich gibt es auch zahlreiche Low-Level-Debugging-Werkzeuge. Die Ausgabe von Debugging-Informationen des 802.11 Protocol Support Layers lassen sich mit dem Programm wlandebug(8) aktivieren. Unter FreeBSD 9.1 finden Sie dieses Programm unter `/usr/src/tools/tools/net80211` . Um beispielsweise während der Suche nach Access Points und des Aufbaus von 802.11-Verbindungen (*Handshake*) auftretende Systemmeldungen auf die Konsole auszugeben, verwenden Sie den folgenden Befehl:

  ```
  # wlandebug -i ath0 +scan+auth+debug+assoc
    net.wlan.0.debug: 0 => 0xc80000<assoc,auth,scan>
  ```

  Der 802.11-Layer liefert umfangreiche Statistiken, die mit dem Werkzeug `wlanstats`, das sich in `/usr/src/tools/tools/net80211` befindet, abgerufen werden können. Diese Statistiken sollten alle Fehler identifizieren, die im 802.11-Layer auftreten. Beachten Sie aber, dass einige Fehler bereits im darunterliegenden Gerätetreiber auftreten und daher in diesen Statistiken nicht enthalten sind. Wie Sie Probleme des Gerätetreibers identifizieren, entnehmen Sie bitte der Dokumentation des Gerätetreibers.

Wenn die oben genannten Informationen nicht helfen das Problem zu klären, erstellen Sie einen Problembericht, der die Ausgabe der weiter oben genannten Werkzeuge beinhaltet.

31.4. USB Tethering

Viele Mobiltelefone bieten die Möglichkeit, ihre Datenverbindung über USB (oft "Tethering" genannt) zu teilen. Diese Funktion verwendet das RNDIS oder CDC-Protokoll.

Bevor Sie ein Gerät anschließen, laden Sie den entsprechenden Treiber in den Kernel:

```
# kldload if_urndis
# kldload cdce
```

Sobald das Gerät angeschlossen ist, steht es Ihnen unter `ue0` wie ein normales Netzwerkgerät zur Verfügung. Stellen Sie sicher, dass die Option „USB Tethering“ auf dem Gerät aktiviert ist.

31.5. Bluetooth

Beigetragen von Pav Lucistnik.

Bluetooth ermöglicht die Bildung von persönlichen Netzwerken über drahtlose Verbindungen bei einer maximalen Reichweite von 10 Metern und operiert im unlizensierten 2,4-GHz-Band. Solche Netzwerke werden normalerweise spontan gebildet, wenn sich mobile Geräte, wie Mobiltelefone, Handhelds oder Notebooks miteinander verbinden. Im Gegensatz zu Wireless LAN ermöglicht Bluetooth auch höherwertige Dienste, wie FTP-ähnliche Dateiserver, Filepushing, Sprachübertragung, Emulation von seriellen Verbindungen und andere mehr.

Dieses Kapitel beschreibt die Verwendung von USB-Bluetooth-Adaptern in FreeBSD. Weiterhin werden verschiedene Bluetooth-Protokolle und Programme vorgestellt.

31.5.1. Die Bluetooth-Unterstützung aktivieren

Der Bluetooth-Stack von FreeBSD verwendet das netgraph(4)-Framework. Viele Bluetooth-USB-Adapter werden durch den ng_ubt(4)-Treiber unterstützt. Auf dem Chip BCM2033 von Broadcom basierende Bluetooth-Geräte werden von den Treibern ubtbcmfw(4) sowie ng_ubt(4) unterstützt. Die Bluetooth-PC-Card 3CRWB60-A von 3Com verwendet den ng_bt3c(4)-Treiber. Serielle sowie auf UART basierende Bluetooth-Geräte werden von sio(4), ng_h4(4) sowie hcseriald(8) unterstützt.

Bevor ein Gerät angeschlossen wird, muss der entsprechende Treiber in den Kernel geladen werden. Hier verwendet das Gerät den ng_ubt(4)-Treiber:

```
# kldload ng_ubt
```

Ist das Bluetooth-Gerät beim Systemstart angeschlossen, kann das entsprechende Modul bei Booten geladen werden, indem der entsprechende Treiber in `/boot/loader.conf` hinzugefügt wird:

```
ng_ubt_load="YES"
```

Sobald der Treiber geladen ist, schließen Sie den USB-Adapter an. Eine Meldung ähnlich der folgenden wird auf der Konsole und in `/var/log/messages` erscheinen:

```
ubt0: vendor 0x0a12 product 0x0001, rev 1.10/5.25, addr 2
ubt0: Interface 0 endpoints: interrupt=0x81, bulk-in=0x82, bulk-out=0x2
ubt0: Interface 1 (alt.config 5) endpoints: isoc-in=0x83, isoc-out=0x3,
      wMaxPacketSize=49, nframes=6, buffer size=294
```

Verwenden Sie das Startskript zum Starten und Beenden des Bluetooth-Stacks. Es ist empfehlenswert, den Bluetooth-Stack zu beenden, bevor Sie den Adapter entfernen. Wenn Sie den Bluetooth-Stack starten, erhalten Sie eine Meldung ähnlich der folgenden:

```
# service bluetooth start ubt0
BD_ADDR: 00:02:72:00:d4:1a
Features: 0xff 0xff 0xf 00 00 00 00 00
<3-Slot> <5-Slot> <Encryption> <Slot offset>
<Timing accuracy> <Switch> <Hold mode> <Sniff mode>
<Park mode> <RSSI> <Channel quality> <SCO link>
<HV2 packets> <HV3 packets> <u-law log> <A-law log> <CVSD>
<Paging scheme> <Power control> <Transparent SCO data>
Max. ACL packet size: 192 bytes
Number of ACL packets: 8
Max. SCO packet size: 64 bytes
Number of SCO packets: 8
```

31.5.2. Suche nach anderen Bluetooth-Geräten

Das *Host Controller Interface* (HCI) bietet eine einheitliche Methode für den Zugriff auf Bluetooth-Baissband-Funktionen. In FreeBSD wird ein netgraph HCI-Knoten für jedes Bluetooth-Gerät erstellt. Weitere Einzelheiten finden Sie in ng_hci(4).

Eine der wichtigsten Aufgaben ist das Auffinden von sich in Reichweite befindenden Bluetooth-Geräten. Diese Funktion wird als *inquiry* bezeichnet. Inquiry sowie andere mit HCI in Verbindung stehende Funktionen werden von hccontrol(8) zur Verfügung gestellt. Das folgende Beispiel zeigt, wie man herausfindet, welche Bluetooth-Geräte sich in Reichweite befinden. Eine solche Abfrage dauert nur wenige Sekunden. Beachten Sie, dass ein Gerät nur dann antwortet, wenn es sich im Modus *discoverable* befindet.

```
% hccontrol -n ubt0hci inquiry
Inquiry result, num_responses=1
Inquiry result #0
       BD_ADDR: 00:80:37:29:19:a4
       Page Scan Rep. Mode: 0x1
       Page Scan Period Mode: 00
       Page Scan Mode: 00
       Class: 52:02:04
       Clock offset: 0x78ef
Inquiry complete. Status: No error [00]
```

`BD_ADDR` stellt, ähnlich der MAC-Adresse einer Netzwerkkarte, die eindeutige Adresse eines Bluetooth-Gerätes dar. Diese Adresse ist für die Kommunikation mit dem Gerät nötig. Es ist aber auch möglich, BD_ADDR einen Klartextnamen zuzuweisen. `/etc/bluetooth/hosts` enthält Informationen über die bekannten Bluetooth-Rechner. Das folgende Beispiel zeigt, wie man den Klartextnamen eines entfernten Geräts in Erfahrung bringen kann:

```
% hccontrol -n ubt0hci remote_name_request 00:80:37:29:19:a4
BD_ADDR: 00:80:37:29:19:a4
Name: Pav's T39
```

Wenn Sie ein entferntes Bluetooth-Gerät abfragen, wird dieses den Rechner unter dem Namen „your.host.name (ubt0)“ finden. Dieser Name kann aber jederzeit geändert werden.

Bluetooth ermöglicht Punkt-zu-Punkt-Verbindungen an denen nur zwei Bluetooth-Geräte beteiligt sind, aber auch Punkt-zu-Multipunkt-Verbindungen, bei denen eine Verbindung von mehreren Bluetooth-Geräten gemeinsam genutzt wird. Das folgende Beispiel zeigt, wie man die aktiven Basisbandverbindungen des lokalen Gerätes anzeigen kann:

```
% hccontrol -n ubt0hci read_connection_list
Remote BD_ADDR    Handle Type Mode Role Encrypt Pending Queue State
00:80:37:29:19:a4     41  ACL    0 MAST    NONE       0     0 OPEN
```

Ein *connection handle* ist für die Beendigung einer Basisbandverbindung nützlich. Im Normalfall werden inaktive Verbindungen aber automatisch vom Bluetooth-Stack getrennt.

```
# hccontrol -n ubt0hci disconnect 41
Connection handle: 41
Reason: Connection terminated by local host [0x16]
```

Rufen Sie `hccontrol help` auf, wenn Sie eine komplette Liste aller verfügbaren HCI-Befehle benötigen. Die meisten dieser Befehle müssen nicht als `root` ausgeführt werden.

31.5.3. Erstmaliger Verbindungsaufbau zwischen zwei Bluetooth-Geräten (Pairing)

In der Voreinstellung nutzt Bluetooth keine Authentifizierung, daher kann sich jedes Bluetoothgerät mit jedem anderen Gerät verbinden. Ein Bluetoothgerät, wie beispielsweise ein Mobiltelefon, kann jedoch für einen bestimmten Dienst, etwa eine Einwählverbindung, eine Authentifizierung anfordern. Bluetooth verwendet zu diesem Zweck *PIN-Codes*. Ein PIN-Code ist ein maximal 16 Zeichen langer ASCII-String. Damit eine Verbindung zustande kommt, muss auf beiden Geräten der gleiche PIN-Code verwendet werden. Nachdem der Code eingegeben wurde, erzeugen beide Geräte einen *link key*, der auf den Geräten gespeichert wird. Beim nächsten Verbindungsaufbau wird der zuvor erzeugte Link Key verwendet. Diesen Vorgang bezeichnet man als *Pairing*. Geht der Link Key auf einem Gerät verloren, muss das Pairing wiederholt werden.

Der hcsecd(8)-Daemon verarbeitet Bluetooth-Authentifzierungsanforderungen und wird über die Datei `/etc/bluetooth/hcsecd.conf` konfiguriert. Der folgende Ausschnitt dieser Datei zeigt die Konfiguration für ein Mobiltelefon, das den PIN-Code „1234“ verwendet:

```
device {
        bdaddr  00:80:37:29:19:a4;
        name    "Pav's T39";
        key     nokey;
        pin     "1234";
      }
```

Von der Länge abgesehen, unterliegen PIN-Codes keinen Einschränkungen. Einige Geräte, beispielsweise Bluetooth-Headsets, haben einen festen PIN-Code eingebaut. Die Option `-d` sorgt dafür, dass der hcsecd(8)-Daemon im Vordergrund läuft. Dadurch kann der Ablauf einfach verfolgt werden. Stellen Sie das entfernte Gerät auf *receive pairing* und initiieren Sie die Bluetoothverbindung auf dem entfernten Gerät. Sie erhalten die Meldung, dass Pairing akzeptiert wurde und der PIN-Code benötigt wird. Geben Sie den gleichen PIN-Code ein, den Sie in `hcsecd.conf` festgelegt haben. Der Computer und das entfernte Gerät sind nun miteinander verbunden. Alternativ können Sie das Pairing auch auf dem entfernten Gerät initiieren.

hcsecd(8) kann durch das Einfügen der folgenden Zeile in `/etc/rc.conf` beim Systemstart automatisch aktiviert werden:

```
hcsecd_enable="YES"
```

Es folgt nun eine beispielhafte Ausgabe des hcsecd(8)-Daemons:

```
hcsecd[16484]: Got Link_Key_Request event from 'ubt0hci', remote bdaddr 0:80:37:29:19:a4
```

```
hcsecd[16484]: Found matching entry, remote bdaddr 0:80:37:29:19:a4, name 'Pav's T39', ↺
link key doesn't exist
hcsecd[16484]: Sending Link_Key_Negative_Reply to 'ubt0hci' for remote bdaddr ↺
0:80:37:29:19:a4
hcsecd[16484]: Got PIN_Code_Request event from 'ubt0hci', remote bdaddr 0:80:37:29:19:a4
hcsecd[16484]: Found matching entry, remote bdaddr 0:80:37:29:19:a4, name 'Pav's T39', ↺
PIN code exists
hcsecd[16484]: Sending PIN_Code_Reply to 'ubt0hci' for remote bdaddr 0:80:37:29:19:a4
```

31.5.4. Einwahlverbindungen und Netzwerkverbindungen mit PPP-Profilen einrichten

Ein *Dial-Up Networking*-Profil (DUN) kann dazu benutzt werden, ein Mobiltelefon als drahtloses Modem zu nutzen, um sich über einen Einwahlprovider mit dem Internet zu verbinden. Es kann auch dazu genutzt werden, einen Computer so zu konfigurieren, dass dieser Datenabfragen empfängt.

Der Zugriff auf ein Netzwerk über ein PPP-Profil kann einen Zugriff auf das LAN für ein oder mehrere Bluetooth-Geräte bieten. Eine PC-zu-PC-Verbindung unter Verwendung einer PPP-Verbindung über eine serielle Verbindung ist ebenfalls möglich.

Diese Profile werden unter FreeBSD durch ppp(8) sowie rfcomm_pppd(8) implementiert - einem Wrapper, der Bluetooth-Verbindungen unter PPP nutzbar macht. Bevor ein Profil verwendet werden kann, muss ein neuer PPP-Abschnitt in `/etc/ppp/ppp.conf` erzeugt werden. Beispielkonfigurationen zu diesem Thema finden Sie in rfcomm_pppd(8).

Dieses Beispiel verwendet rfcomm_pppd(8), um eine Verbindung zu einem entfernten Gerät mit der `BD_ADDR` `00:80:37:29:19:a4` auf dem RFCOMM-Kanal `DUN` aufzubauen:

```
# rfcomm_pppd -a 00:80:37:29:19:a4 -c -C dun -l rfcomm-dialup
```

Die aktuelle Kanalnummer des entfernten Geräts erhalten Sie über das SDP-Protokoll. Es ist auch möglich, manuell einen RFCOMM-Kanal festzulegen. In diesem Fall führt rfcomm_pppd(8) keine SDP-Abfrage durch. Verwenden Sie sdpcontrol(8), um die RFCOMM-Kanäle des entfernten Geräts herauszufinden.

Der sdpd(8)-Server muss laufen, damit ein Netzzugriff mit dem PPP LAN-Profil möglich ist. Außerdem muss für den LAN-Client ein neuer Eintrag in `/etc/ppp/ppp.conf` erzeugt werden. Beispielkonfigurationen zu diesem Thema finden Sie in rfcomm_pppd(8). Danach starten Sie den RFCOMM PPP-Server über eine gültige RFCOMM-Kanalnummer. Der RFCOMM PPP-Server bindet dadurch den Bluetooth-LAN-Dienst an den lokalen SDP-Daemon. Das folgende Beispiel zeigt, wie man den RFCOMM PPP-Server startet.

```
# rfcomm_pppd -s -C 7 -l rfcomm-server
```

31.5.5. Bluetooth-Protokolle

Dieser Abschnitt gibt einen Überblick über die verschiedenen Bluetooth-Protokolle, ihre Funktionen sowie weitere Programme.

31.5.5.1. Das Logical Link Control and Adaptation Protocol (L2CAP)

Das *Logical Link Control and Adaptation Protocol* (L2CAP) bietet höherwertigen Protokollen verbindungsorientierte und verbindungslose Datendienste an. L2CAP erlaubt höherwertigen Protokollen und Programmen den Versand und Empfang von L2CAP-Datenpaketen mit einer Länge von bis zu 64 Kilobytes.

L2CAP arbeitet *kanal*basiert. Ein Kanal ist eine logische Verbindung innerhalb einer Basisbandverbindung. Jeder Kanal ist dabei an ein einziges Protokoll gebunden. Mehrere Geräte können an das gleiche Protokoll gebunden sein, es ist aber nicht möglich, einen Kanal an mehrere Protokolle zu binden. Jedes über einen Kanal ankommende L2CAP-Paket wird an das entsprechende höherwertige Protokoll weitergeleitet. Mehrere Kanäle können sich die gleiche Basisbandverbindung teilen.

Unter FreeBSD wird eine netgraph-Gerätedatei vom Typ *l2cap* für jedes einzelne Bluetooth-Gerät erzeugt. Diese Gerätedatei ist normalerweise mit der Bluetooth-HCI-Gerätedatei (downstream) sowie der Bluetooth-Socket-Gerä-

tedatei (upstream) verbunden. Der Standardname für die L2CAP-Gerätedatei lautet „devicel2cap“. Weitere Details finden Sie in ng_l2cap(4).

Ein nützlicher Befehl zum Anpingen von anderen Geräten ist l2ping(8). Einige Bluetooth-Geräte senden allerdings nicht alle erhaltenen Daten zurück. Die Ausgabe `0 bytes` im folgenden Beispiel ist also kein Fehler:

```
# l2ping -a 00:80:37:29:19:a4
0 bytes from 0:80:37:29:19:a4 seq_no=0 time=48.633 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=1 time=37.551 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=2 time=28.324 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=3 time=46.150 ms result=0
```

Das Programm l2control(8) liefert Informationen über L2CAP-Dateien. Das folgende Beispiel zeigt, wie man die Liste der logischen Verbindungen (Kanäle) sowie die Liste der Basisbandverbindungen abfragen kann:

```
% l2control -a 00:02:72:00:d4:1a read_channel_list
L2CAP channels:
Remote BD_ADDR     SCID/ DCID   PSM  IMTU/ OMTU State
00:07:e0:00:0b:ca    66/   64     3   132/  672 OPEN
% l2control -a 00:02:72:00:d4:1a read_connection_list
L2CAP connections:
Remote BD_ADDR    Handle Flags Pending State
00:07:e0:00:0b:ca     41 O           0 OPEN
```

btsockstat(1) ist ein weiteres Diagnoseprogramm. Es funktioniert ähnlich wie netstat(1), arbeitet aber mit Bluetooth-Datenstrukturen. Das folgende Beispiel zeigt die gleiche Liste der logischen Verbindungen wie l2control(8) im vorherigen Beispiel.

```
% btsockstat
Active L2CAP sockets
PCB      Recv-Q Send-Q Local address/PSM       Foreign address   CID   State
c2afe900      0      0 00:02:72:00:d4:1a/3     00:07:e0:00:0b:ca 66    OPEN
Active RFCOMM sessions
L2PCB    PCB      Flag MTU   Out-Q DLCs State
c2afe900 c2b53380 1    127   0     Yes  OPEN
Active RFCOMM sockets
PCB      Recv-Q Send-Q Local address     Foreign address   Chan DLCI State
c2e8bc80      0    250 00:02:72:00:d4:1a 00:07:e0:00:0b:ca 3    6    OPEN
```

31.5.5.2. Radio Frequency Communication (RFCOMM)

Das RFCOMM-Protokoll emuliert serielle Verbindungen über das L2CAP-Protokoll. Bei RFCOMM handelt es sich um ein einfaches Transportprotokoll, das um Funktionen zur Emulation der 9poligen Schaltkreise von mit RS-232 (EIATIA-232-E) kompatiblen seriellen Ports ergänzt wurde. Es erlaubt bis zu 60 simultane Verbindungen (RFCOMM-Kanäle) zwischen zwei Bluetooth-Geräten.

Eine RFCOMM-Kommunikation besteht aus zwei Anwendungen (den Kommunikationsendpunkten), die über das Kommunikationssegment miteinander verbunden sind. RFCOMM unterstützt Anwendungen, die auf serielle Ports angewiesen sind. Das Kommunikationssegment entspricht der direkten Bluetooth-Verbindung zwischen den beiden Geräten.

RFCOMM kümmert sich um die direkte Verbindung von zwei Geräten, oder um die Verbindung zwischen einem Gerät und einem Modem über eine Netzwerkverbindung. RFCOMM unterstützt auch andere Konfigurationen. Ein Beispiel dafür sind Module, die drahtlose Bluetooth-Geräte mit einer verkabelten Schnittstelle verbinden können.

Unter FreeBSD ist das RFCOMM-Protokoll im Bluetooth Socket-Layer implementiert.

31.5.5.3. Das Service Discovery Protocol (SDP)

Das *Service Discovery Protocol* (SDP) erlaubt es Clientanwendungen, von Serveranwendungen angebotene Dienste sowie deren Eigenschaften abzufragen. Zu diesen Eigenschaften gehören die Art oder die Klasse der angebotenen Dienste sowie der Mechanismus oder das Protokoll, die zur Nutzung des Dienstes notwendig sind.

SDP ermöglicht Verbindungen zwischen einem SDP-Server und einem SDP-Client. Der Server enthält eine Liste mit den Eigenschaften der vom Server angebotenen Dienste. Jeder Eintrag beschreibt jeweils einen einzigen Serverdienst. Ein Client kann diese Informationen durch eine SDP-Anforderung vom SDP-Server beziehen. Wenn der Client oder eine Anwendung des Clients einen Dienst nutzen will, muss eine separate Verbindung mit dem Dienstanbieter aufgebaut werden. SDP bietet einen Mechanismus zum Auffinden von Diensten und deren Eigenschaften an, es bietet aber keine Mechanismen zur Verwendung dieser Dienste.

Normalerweise sucht ein SDP-Client nur nach Diensten, die bestimmte geforderte Eigenschaften erfüllen. Es ist aber auch möglich, anhand der Dienstbeschreibungen eine allgemeine Suche nach den von einem SDP-Server angebotenen Diensten durchzuführen. Diesen Vorgang bezeichnet man als *Browsing*.

Der Bluetooth-SDP-Server sdpd(8) und der Kommandozeilenclient sdpcontrol(8) sind bereits in der Standardinstallation von FreeBSD enthalten. Das folgende Beispiel zeigt, wie eine SDP-Abfrage durchgeführt wird:

```
% sdpcontrol -a 00:01:03:fc:6e:ec browse
Record Handle: 00000000
Service Class ID List:
        Service Discovery Server (0x1000)
Protocol Descriptor List:
        L2CAP (0x0100)
                Protocol specific parameter #1: u/int/uuid16 1
                Protocol specific parameter #2: u/int/uuid16 1

Record Handle: 0x00000001
Service Class ID List:
        Browse Group Descriptor (0x1001)

Record Handle: 0x00000002
Service Class ID List:
        LAN Access Using PPP (0x1102)
Protocol Descriptor List:
        L2CAP (0x0100)
        RFCOMM (0x0003)
                Protocol specific parameter #1: u/int8/bool 1
Bluetooth Profile Descriptor List:
        LAN Access Using PPP (0x1102) ver. 1.0
```

Beachten Sie, dass jeder Dienst eine Liste seiner Eigenschaften, wie etwa den RFCOMM-Kanal, zurückgibt. Je nach dem, welche Dienste der Benutzer benötigt, sollten einige dieser Eigenschaften notiert werden. Einige Bluetooth-Implementationen unterstützen kein *Service Browsing* und geben daher eine leere Liste zurück. Ist dies der Fall, ist es dennoch möglich, nach einem bestimmten Dienst zu suchen. Das folgende Beispiel demonstriert die Suche nach dem OBEX Object Push (OPUSH) Dienst:

```
% sdpcontrol -a 00:01:03:fc:6e:ec search OPUSH
```

Unter FreeBSD ist es die Aufgabe des sdpd(8)-Servers, Bluetooth-Clients verschiedene Dienste anzubieten. Sie können diesen Server durch das Einfügen der folgenden Zeile in `/etc/rc.conf` aktivieren:

```
sdpd_enable="YES"
```

Nun kann der sdpd(8)-Daemon durch folgene Eingabe gestartet werden:

```
# service sdpd start
```

Der lokale Server, der den entfernten Clients Bluetooth-Dienste anbieten soll, bindet diese Dienste an den lokalen SDP-Daemon. Ein Beispiel für eine solche Anwendung ist rfcomm_pppd(8). Einmal gestartet, wird der Bluetooth-LAN-Dienst an den lokalen SDP-Daemon gebunden.

Die Liste der vorhandenen Dienste, die am lokalen SDP-Server registriert sind, lässt sich durch eine SDP-Abfrage über einen lokalen Kontrollkanal abfragen:

```
# sdpcontrol -l browse
```

31.5.5.4. OBEX Object-Push (OPUSH)

OBEX ist ein häufig verwendetes Protokoll für den Dateitransfer zwischen Mobilgeräten. Sein Hauptzweck ist die Kommunikation über die Infrarotschnittstelle. Es dient daher zum Datentransfer zwischen Notebooks oder PDAs sowie zum Austausch von Visitenkarten oder Kalendereinträgen zwischen Mobiltelefonen und anderen Geräten mit PIM-Funktionen.

Server und Client von OBEX werden durch obexapp bereitgestellt, das als Paket oder Port comms/obexapp installiert werden kann.

Mit dem OBEX-Client werden Objekte zum OBEX-Server geschickt oder angefordert. Ein Objekt kann etwa eine Visitenkarte oder ein Termin sein. Der OBEX-Client fordert über SDP die Nummer des RFCOMM-Kanals vom entfernten Gerät an. Dies kann auch durch die Verwendung des Servicenamens anstelle der RFCOMM-Kanalnummer erfolgen. Folgende Dienste werden unterstützt: `>IrMC`, `FTRN` und `OPUSH`. Es ist möglich, den RFCOMM-Kanal als Nummer anzugeben. Es folgt ein Beispiel für eine OBEX-Sitzung, bei der ein Informationsobjekt vom Mobiltelefon angefordert und ein neues Objekt (hier eine Visitenkarte) an das Telefonbuch des Mobiltelefons geschickt wird:

```
% obexapp -a 00:80:37:29:19:a4 -C IrMC
obex> get telecom/devinfo.txt
Success, response: OK, Success (0x20)
obex> put new.vcf
Success, response: OK, Success (0x20)
obex> di
Success, response: OK, Success (0x20)
```

Um OBEX-Push-Dienste anbieten zu können, muss der sdpd-Server gestartet sein. Ein Wurzelverzeichnis, in dem alle ankommenden Objekte gespeichert werden, muss zusätzlich angelegt werden. In der Voreinstellung ist dies `/var/spool/obex` . Starten Sie den OBEX-Server mit einer gültigen Kanalnummer. Der OBEX-Server registriert nun den OBEX-Push-Dienst mit dem lokalen SDP-Daemon. Das folgende Beispiel zeigt, wie der OBEX-Server gestartet wird:

```
# obexapp -s -C 10
```

31.5.5.5. Das Serial-Port Profil (SPP)

Das *Serial Port Profile* (SSP) ermöglicht es Bluetooth-Geräten eine serielle Kabelverbindung zu emulieren. Anwendungen sind dadurch in der Lage, über eine virtuelle serielle Verbindung Bluetooth als Ersatz für eine Kabelverbindung zu nutzen.

rfcomm_sppd(1) implementiert unter FreeBSD SSP und ein Pseudo-tty, das als virtuelle serielle Verbindung verwendet wird. Das folgende Beispiel zeigt, wie man eine Verbindung mit einem entfernten Serial-Port-Dienst herstellt. Ein RFCOMM-Kanal muss dabei nicht angegeben werden, da rfcomm_sppd(1) den Kanal über SDP abfragen kann. Um dies zu umgehen, geben Sie einen RFCOMM-Kanal auf der Kommandozeile an.

```
# rfcomm_sppd -a 00:07:E0:00:0B:CA -t /dev/ttyp6
rfcomm_sppd[94692]: Starting on /dev/ttyp6...
```

Sobald die Verbindung hergestellt ist, kann pseudo-tty als serieller Port verwenden werden.

```
# cu -l ttyp6
```

31.5.6. Problembehandlung

Wenn FreeBSD eine neue Verbindung akzeptiert, versucht es, die Rolle zu tauschen, um zum Master zu werden. Einige ältere Geräte, die dies nicht unterstützen, können keine Verbindung aufbauen. Da der Rollentausch ausgeführt wird sobald eine neue Verbindung aufgebaut wird, ist es nicht möglich, das entfernte Gerät zu fragen ob es den Rollentausch unterstützt. Es gibt jedoch eine HCI-Option, die dieses Verhalten deaktiviert:

```
# hccontrol -n ubt0hci write_node_role_switch 0
```

Verwenden Sie hcidump, das als Paket Port comms/hcidump installiert werden kann, um Bluetooth-Pakete anzuzeigen. Dieses Programm hat Ähnlichkeiten mit tcpdump(1) und kann zur Anzeige der Bluetooth-Pakete in einem Terminal, oder zur Speicherung von Paketen in einer Datei (Dump) verwendet werden.

31.6. LAN-Kopplung mit einer Bridge

Geschrieben von Andrew Thompson.

Manchmal ist es nützlich, ein Netzwerk, wie ein Ethernetsegment, in separate Netzwerke aufzuteilen, ohne gleich IP-Subnetze zu erzeugen, die über einen Router miteinander verbunden sind. Ein Gerät, das zwei Netze auf diese Weise verbindet, wird als „Bridge“ bezeichnet.

Eine Bridge arbeitet, indem sie die MAC-Adressen der Geräte in ihren Netzwerksegmenten lernt. Der Verkehr wird nur dann zwischen zwei Segmenten weitergeleitet, wenn sich Sender und Empfänger in verschiedenen Netzwerkssegmenten befinden. Jedes FreeBSD-System mit zwei Netzwerkkarten kann als Bridge fungieren.

Bridging kann in den folgenden Situationen sinnvoll sein:

Verbinden von Netzwerken
: Die Hauptaufgabe einer Bridge ist die Verbindung von zwei oder mehreren Netzwerksegmenten. Es gibt viele Gründe, eine hostbasierte Bridge einzusetzen, anstelle von Netzwerkkomponenten, wie beispielsweise Kabelverbindungen oder Firewalls. Eine Bridge kann außerdem ein drahtloses Gerät mit einem Kabelnetzwerk verbinden. Diese Fähigkeit der Bridge wird als *HostAP-Modus* bezeichnet. Die Bridge agiert in diesem Fall als Access Point für das drahtlose Gerät.

Filtering / Traffic Shaping Firewall
: Eine Bridge kann eingesetzt werden, wenn Firewallfunktionen benötigt werden, ohne dabei Routing oder *Network Adress Translation* (NAT) zu verwenden.

  Ein Beispiel dafür wäre ein kleines Unternehmen, das über DSL oder ISDN an einen ISP angebunden ist. Es verfügt über 13 erreichbare IP-Adressen und das Netzwerk besteht aus 10 Rechnern. In dieser Situation ist der Einsatz von Subnetzen sowie einer routerbasierten Firewall aufgrund der IP-Adressierung schwierig. Eine Bridge-basierte Firewall kann hingegen ohne Probleme konfiguriert werden.

Netzwerküberwachung
: Eine Bridge kann zwei Netzwerksegmente miteinander verbinden und danach alle Ethernet-Rahmen überprüfen, die zwischen den beiden Netzwerksegmenten ausgetauscht werden. Dazu verwendet man entweder bpf(4) und tcpdump(1) auf dem Netzgerät der Bridge oder schickt Kopien aller Rahmen an ein zusätzliches Netzgerät, das als *Span Port* bekannt ist.

Layer 2 VPN
: Zwei Ethernetnetzwerke können über einen IP-Link miteinander verbunden werden, indem die beiden Netzwerke über einen EtherIP-Tunnel gekoppelt werden, oder eine tap(4)-basierte Lösung wie OpenVPN eingesetzt wird.

Layer 2 Redundanz
: Die Systeme eines Netzwerks können über das *Spanning Tree Protocol* (STP) redundant miteinander verbunden sein, um redundante Pfade zu blockieren.

Dieser Abschnitt beschreibt, wie ein FreeBSD-System mit Hilfe von if_bridge(4) als Bridge konfiguriert wird. Ein netgraph-Bridge-Treiber ist ebenfalls verfügbar und wird in ng_bridge(4) beschrieben.

Anmerkung

Paketfilter können mit allen Firewallpaketen verwendet werden, die das pfil(9)-Framework benutzen. Eine Bridge kann auch als *Traffic Shaper* verwendet werden, wenn Sie altq(4) oder dummynet(4) einsetzen.

31.6.1. Die Bridge aktivieren

In FreeBSD handelt es sich bei if_bridge(4) um ein Kernelmodul, das von ifconfig(8) automatisch geladen wird, wenn eine Bridge-Schnittstelle erzeugt wird. Es ist auch möglich, die Unterstützung für den Treiber in den Kernel zu kompilieren, indem die Zeile `device if_bridge` in die Kernelkonfigurationsdatei hinzugefügt wird.

Eine Bridge wird durch das Klonen von Schnittstellen erzeugt. Um eine Bridge zu erzeugen, verwenden Sie:

```
# ifconfig bridge create
bridge0
# ifconfig bridge0
bridge0: flags=8802<BROADCAST,SIMPLEX,MULTICAST> metric 0 mtu 1500
        ether 96:3d:4b:f1:79:7a
        id 00:00:00:00:00:00 priority 32768 hellotime 2 fwddelay 15
        maxage 20 holdcnt 6 proto rstp maxaddr 100 timeout 1200
        root id 00:00:00:00:00:00 priority 0 ifcost 0 port 0
```

Wenn eine Bridge erzeugt wird, erhält sie automatisch eine zufällig generierte Ethernet-Adresse. Die Parameter `maxaddr` sowie `timeout` legen fest, wie viele MAC-Adressen die Bridge in ihrer Forward-Tabelle halten kann und wie viele Sekunden jeder Eintrag erhalten bleiben soll, nachdem er zuletzt verwendet wurde. Die restlichen Parameter sind für die Konfiguration von STP notwendig.

Im nächsten Schritt werden die Schnittstellen, die die Bridge verbinden soll, zugewiesen. Damit die Bridge Datenpakete weiterleiten kann, müssen sowohl die Bridge als auch die Schnittstellen der zu verbindenden Netzwerksegmente aktiviert sein:

```
# ifconfig bridge0 addm fxp0 addm fxp1 up
# ifconfig fxp0 up
# ifconfig fxp1 up
```

Jetzt ist die Bridge in der Lage, Ethernet-Rahmen zwischen den Schnittstellen `fxp0` und `fxp1` weiterzuleiten. Um diese Konfiguration beim Systemstart automatisch zu aktivieren, müssen die folgenden Zeilen in `/etc/rc.conf` hinzugefügt werden:

```
cloned_interfaces="bridge0"
ifconfig_bridge0="addm fxp0 addm fxp1 up"
ifconfig_fxp0="up"
ifconfig_fxp1="up"
```

Wenn die Bridge eine IP-Adresse benötigt, muss diese der Schnittstelle der Bridge zugewiesen werden und nicht der Schnittstelle der gekoppelten Netzwerksegmente. Die IP-Adresse kann manuell gesetzt, oder über DHCP bezogen werden. Dieses Beispiel verwendet eine statische IP-Adresse:

```
# ifconfig bridge0 inet 192.168.0.1/24
```

Es ist auch möglich der Bridge-Schnittstelle eine IPv6-Adresse zuzuweisen. Um die Änderungen dauerhaft zu speichern, fügen Sie die Adressinformationen in `/etc/rc.conf` ein.

Anmerkung

Nachdem ein Paketfilter aktiviert wurde, können Datenpakete, die von den Schnittstellen der gekoppelten Netzwerksegmente gesendet und empfangen werden, über die Bridge weitergeleitet oder nach bestimmten Regeln gefiltert oder auch komplett geblockt werden. Ist die Richtung des Paketflusses wichtig, ist es am besten, eine Firewall auf den Schnittstellen der einzelnen Netzwerksegmente einzurichten und nicht auf der Bridge selbst.

Eine Bridge verfügt über verschiedene Optionen zur Weiterleitung von Nicht-IP- und IP-Paketen, sowie Paketfilterung auf Layer 2 mittels ipfw(8). Weitere Informationen finden Sie in if_bridge(4).

31.6.2. Spanning Tree aktivieren

Damit ein Ethernet-Netzwerk richtig funktioniert, kann nur ein aktiver Pfad zwischen zwei Geräten existieren. Das STP-Protokoll erkennt Schleifen in einer Netzwerktopologie und setzt redundante Pfade in einen blockierten Zustand. Sollte eine der aktiven Verbindungen ausfallen, berechnet STP einen anderen Baum und ermöglicht es dann einem blockierten Pfad, alle Netzwerkverbindungen wiederherzustellen.

Das *Rapid Spanning Tree Protocol* (RSTP oder 802.1w), ist abwärtskompatibel zum veralteten STP. RSTP arbeitet schneller und tauscht Informationen mit benachbarten Switchen aus, um Pakete korrekt weiterzuleiten und eine Schleifenbildung zu verhindern. FreeBSD unterstützt die Betriebsmodi RSTP und STP, wobei RSTP als Standardmodus voreingestellt ist.

STP kann auf den Schnittstellen der durch die Bridge verbundenen Netzwerksegmente mittels ifconfig(8) aktiviert werden. Für eine Bridge, die die Schnittstellen `fxp0` und `fxp1` verbindet, aktivieren Sie STP wie folgt:

```
# ifconfig bridge0 stp fxp0 stp fxp1
bridge0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        ether d6:cf:d5:a0:94:6d
        id 00:01:02:4b:d4:50 priority 32768 hellotime 2 fwddelay 15
        maxage 20 holdcnt 6 proto rstp maxaddr 100 timeout 1200
        root id 00:01:02:4b:d4:50 priority 32768 ifcost 0 port 0
        member: fxp0 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 3 priority 128 path cost 200000 proto rstp
                role designated state forwarding
        member: fxp1 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 4 priority 128 path cost 200000 proto rstp
                role designated state forwarding
```

Diese Bridge hat die Spanning-Tree-ID `00:01:02:4b:d4:50` und die Priorität `32768`. Da diese ID mit der `Root-ID` identisch ist, handelt es sich um die Root-Bridge dieses Netzwerks.

Auf einer anderen Bridge des Netzwerks ist STP ebenfalls aktiviert:

```
bridge0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        ether 96:3d:4b:f1:79:7a
        id 00:13:d4:9a:06:7a priority 32768 hellotime 2 fwddelay 15
        maxage 20 holdcnt 6 proto rstp maxaddr 100 timeout 1200
        root id 00:01:02:4b:d4:50 priority 32768 ifcost 400000 port 4
        member: fxp0 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 4 priority 128 path cost 200000 proto rstp
                role root state forwarding
        member: fxp1 flags=1c7<LEARNING,DISCOVER,STP,AUTOEDGE,PTP,AUTOPTP>
                port 5 priority 128 path cost 200000 proto rstp
                role designated state forwarding
```

Die Zeile `root id 00:01:02:4b:d4:50 priority 32768 ifcost 400000 port 4` zeigt an, dass die Root-Bridge die ID `00:01:02:4b:d4:50` hat. Die Pfadkosten hin zur Root-Bridge betragen `400000`, wobei der Pfad zur Root-Bridge über `port 4` geht, der wiederum der Schnittstelle `fxp0` entspricht.

31.6.3. Parameter der Bridge-Schnittstelle

Einige Paramenter von `ifconfig` dienen ausschließlich der Konfiguration von Bridge-Schnittstellen. Dieser Abschnitt fasst die Verwendung dieser Parameter zusammen. Die vollständige Liste der verfügbaren Parameter wird in ifconfig(8) beschrieben.

private
: Eine private Schnittstelle leitet keine Daten an einen Port weiter, bei dem es sich ebenfalls um eine private Schnittstelle handelt. Der Datenverkehr wird dabei komplett blockiert, auch Ethernet-Rahmen und ARP-Pakete werden nicht weitergeleitet. Wollen Sie hingegen nur spezifische Datenpakete blockieren, sollten Sie eine Firewall einsetzen.

span
: Ein *Span Port* übertragt eine Kopie jedes Ethernet-Rahmens, der an der Bridge ankommt. Auf einer Bridge können beliebig viele Span Ports festgelegt werden. Wird eine Schnittstelle als Span Port konfiguriert, kann sie nicht mehr als normaler Bridge-Port verwendet werden. Eine derartige Konfiguration ist beispielsweise sinnvoll, um den Datenverkehr, der in einem Netzwerk über die Bridge läuft, auf einen Rechner zu übertragen, der mit einem Span Port der Bridge verbunden ist. Um beispielsweise eine Kopie aller Ethernet-Rahmen über die Schnittstelle `fxp0` zu übertragen:

```
# ifconfig bridge0 span fxp4
```

sticky
: Wenn die Schnittstelle eines über eine Bridge verbundenen Netzwerksegments als *sticky* gekennzeichnet wird, werden alle dynamisch gelernten Adressen als statische Adressen behandelt, sobald sie in den Forward-Cache der Bridge aufgenommen wurden. Sticky-Einträge werden niemals aus dem Cache entfernt oder ersetzt. Selbst dann nicht, wenn die Adresse von einer anderen Schnittstelle verwendet wird. Sie können dadurch die Vorteile statischer Adresseinträge nutzen, ohne die Forward-Tabelle vor dem Einsatz der Bridge mit statischen Einträgen füllen zu müssen. Clients, die sich in einem bestimmten von der Bridge verwalteten Segmente befinden, können dabei nicht in ein anderes Segment wechseln.

  Ein Beispiel für den Einsatz von Sticky-Adressen ist die Kombination einer Bridge mit mehreren VLANs, um einen Router zu konfigurieren, der einzelne Kundennetzwerke voneinander trennt, ohne dabei IP-Adressbereiche zu verschwenden. Für das folgende Beispiel nehmen wir an, dass sich der Client `CustomerA` im VLAN `vlan100` und der Client `CustomerB` im VLAN `vlan101` befinden. Die Bridge hat die IP-Adresse `192.168.0.1` :

```
# ifconfig bridge0 addm vlan100 sticky vlan100 addm vlan101 sticky vlan101
# ifconfig bridge0 inet 192.168.0.1/24
```

  In diesem Beispiel sehen beide Clients `192.168.0.1` als das Default-Gateway. Da der Brücken-Cache *sticky* ist, sind Sie nicht dazu in der Lage, die MAC-Adresse des anderen Kunden zu spoofen und dessen Datenverkehr abzufangen.

  Sie können die Kommunikation zwischen den VLANs vollständig unterbinden, wenn Sie private Schnittstellen oder eine Firewall einsetzen:

```
# ifconfig bridge0 private vlan100 private vlan101
```

  Die Kunden sind nun komplett voneinander isoliert und der komplette `/24`-Adressbereich kann zugewiesen werden, ohne dass Subnetze eingesetzt werden.

  Die maximale mögliche Anzahl an eindeutigen MAC-Adressen hinter einer Schnittstelle kann festgelegt werden. Sobald das Limit erreicht ist, werden Pakete mit einer unbekannten Quell-Adresse solange verworfen, bis ein existierender Eintrag gelöscht wird oder abläuft.

  Das folgende Beispiel setzt die maximale Anzahl von Netzgeräten für `CustomerA` für das VLAN `vlan100` auf 10.

```
# ifconfig bridge0 ifmaxaddr vlan100 10
```

Die Bridge unterstützt auch den Monitormodus. Dabei werden alle Pakete verworfen, nachdem sie von bpf(4) verarbeitet wurden. In diesem Modus erfolgt keine weitere Bearbeitung und auch keine Weiterleitung von Datenpaketen. Es ist daher möglich, die Eingabe von zwei oder mehr Netzwerkschnittstellen in einen einzigen gemeinsamen bpf(4)-Stream zu vereinen. Ein solcher Datenstrom ist beispielsweise nützlich, um den Datenverkehr für „network taps“ zu rekonstruieren, die ihre RX/TX-Signale über verschiedene Schnittstellen senden. Um beispielsweise die Eingabe von vier Netzwerkschnittstellen in einzigen gemeinsamen Datenstrom zu vereinen:

```
# ifconfig bridge0 addm fxp0 addm fxp1 addm fxp2 addm fxp3 monitor up
# tcpdump -i bridge0
```

31.6.4. SNMP-Monitoring

Die Schnittstelle der Bridge sowie die STP-Parameter können durch den im Basissystem enthaltenen bsnmpd(1) überwacht werden. Die exportierten Bridge-MIBs entsprechen den IETF-Standards, daher kann ein beliebiger SNMP-Client oder ein beliebiges Monitoring-Werkzeug eingesetzt werden, um die benötigten Daten zu erhalten.

Um das Monitoring auf der Bridge zu aktivieren, kommentieren Sie diese Zeile in `/etc/snmp.config` aus, indem Sie das Zeichen # entfernen:

```
begemotSnmpdModulePath."bridge" = "/usr/lib/snmp_bridge.so"
```

Weitere Konfigurationsparameter wie Community-Namen und Zugriffslisten müssen ebenfalls in dieser Datei angepasst werden. Weitere Informationen finden Sie in bsnmpd(1) und snmp_bridge(3). Nachdem die Änderungen gespeichert wurden, fügen Sie folgende Zeile in `/etc/rc.conf` hinzu:

```
bsnmpd_enable="YES"
```

Danach starten Sie bsnmpd(1):

```
# service bsnmpd start
```

Die folgenden Beispiele verwenden das Softwarepaket Net-SNMP (net-mgmt/net-snmp), um die Bridge vom Client aus abzufragen. Alternativ kann auch der Port net-mgmt/bsnmptools benutzt werden. Auf dem SNMP-Client müssen danach die folgenden Zeilen in `$HOME/.snmp/snmp.conf` hinzugefügt werden, um die MIB-Definitionen der Bridge in Net-SNMP zu importieren:

```
mibdirs +/usr/share/snmp/mibs
mibs +BRIDGE-MIB:RSTP-MIB:BEGEMOT-MIB:BEGEMOT-BRIDGE-MIB
```

Um eine einzelne Bridge über den IETF BRIDGE-MIB (RFC4188) zu überwachen:

```
% snmpwalk -v 2c -c public bridge1.example.com mib-2.dot1dBridge
BRIDGE-MIB::dot1dBaseBridgeAddress.0 = STRING: 66:fb:9b:6e:5c:44
BRIDGE-MIB::dot1dBaseNumPorts.0 = INTEGER: 1 ports
BRIDGE-MIB::dot1dStpTimeSinceTopologyChange.0 = Timeticks: (189959) 0:31:39.59 centi-
seconds
BRIDGE-MIB::dot1dStpTopChanges.0 = Counter32: 2
BRIDGE-MIB::dot1dStpDesignatedRoot.0 = Hex-STRING: 80 00 00 01 02 4B D4 50
...
BRIDGE-MIB::dot1dStpPortState.3 = INTEGER: forwarding(5)
BRIDGE-MIB::dot1dStpPortEnable.3 = INTEGER: enabled(1)
BRIDGE-MIB::dot1dStpPortPathCost.3 = INTEGER: 200000
BRIDGE-MIB::dot1dStpPortDesignatedRoot.3 = Hex-STRING: 80 00 00 01 02 4B D4 50
BRIDGE-MIB::dot1dStpPortDesignatedCost.3 = INTEGER: 0
BRIDGE-MIB::dot1dStpPortDesignatedBridge.3 = Hex-STRING: 80 00 00 01 02 4B D4 50
BRIDGE-MIB::dot1dStpPortDesignatedPort.3 = Hex-STRING: 03 80
BRIDGE-MIB::dot1dStpPortForwardTransitions.3 = Counter32: 1
RSTP-MIB::dot1dStpVersion.0 = INTEGER: rstp(2)
```

Der Wert der Variable `dot1dStpTopChanges.0` ist hier 2, die STP-Topologie der Bridge wurde also bereits zweimal geändert. Unter einer Änderung versteht man die Anpassung eines oder mehrerer Links und die Kalkulation eines neuen Baums. Der Wert der Variable `dot1dStpTimeSinceTopologyChange.0` gibt an, wann dies zuletzt geschah.

Um mehrere Bridge-Schnittstellen zu überwachen, kann der private BEGEMOT-BRIDGE-MIB eingesetzt werden:

```
% snmpwalk -v 2c -c public bridge1.example.com
enterprises.fokus.begemot.begemotBridge
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseName."bridge0" = STRING: bridge0
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseName."bridge2" = STRING: bridge2
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseAddress."bridge0" = STRING: e:ce:3b:5a:9e:13
```

```
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseAddress."bridge2" = STRING: 12:5e:4d:74:d:fc
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseNumPorts."bridge0" = INTEGER: 1
BEGEMOT-BRIDGE-MIB::begemotBridgeBaseNumPorts."bridge2" = INTEGER: 1
...
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTimeSinceTopologyChange."bridge0" = Timeticks: ↺
(116927) 0:19:29.27 centi-seconds
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTimeSinceTopologyChange."bridge2" = Timeticks: ↺
(82773) 0:13:47.73 centi-seconds
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTopChanges."bridge0" = Counter32: 1
BEGEMOT-BRIDGE-MIB::begemotBridgeStpTopChanges."bridge2" = Counter32: 1
BEGEMOT-BRIDGE-MIB::begemotBridgeStpDesignatedRoot."bridge0" = Hex-STRING: 80 00 00 40 ↺
95 30 5E 31
BEGEMOT-BRIDGE-MIB::begemotBridgeStpDesignatedRoot."bridge2" = Hex-STRING: 80 00 00 50 ↺
8B B8 C6 A9
```

Um die über den `mib-2.dot1dBridge`-Subtree überwachte Bridge-Schnittstelle zu ändern:

```
% snmpset -v 2c -c private bridge1.example.com
BEGEMOT-BRIDGE-MIB::begemotBridgeDefaultBridgeIf.0 s bridge2
```

31.7. Link-Aggregation und Failover

Geschrieben von Andrew Thompson.
Übersetzt von Benedict Reuschling und Sharon Bahagi.

Die von FreeBSD unterstützte lagg(4)-Schnittstelle erlaubt die Gruppierung von mehreren Netzwerkadaptern als eine virtuelle Schnittstelle, mit dem Ziel, Ausfallsicherheit (Failover) und Link Aggregation bereitzustellen. Bei Failover kann der Verkehr auch dann weiter fließen, wenn nur eine Schnittstelle verfügbar ist. Link Aggregation funktioniert am besten mit Switches, die LCAP unterstützen, da dieses Protokoll den Datenverkehr bidirektional verteilt, während es auch auf den Ausfall einzelner Verbindungen reagiert.

Die von der lagg-Schnittstelle unterstützten Protokolle bestimmen, welche Ports für den ausgehenden Datenverkehr benutzt werden, und ob ein bestimmter Port eingehenden Datenverkehr akzeptiert. Die folgenden Protokolle werden von lagg(4) unterstützt:

Failover (Ausfallsicherheit)
: Dieser Modus sendet und empfängt Datenverkehr nur auf dem Masterport. Sollte der Masterport nicht zur Verfügung stehen, wird der nächste aktive Port verwendet. Der zuerst hinzugefügte Adapter der virtuellen Schnittstelle wird zum Masterport, jeder weitere Adapter dient als Gerät zur Ausfallsicherheit. Wenn ein Failover auf einem Nicht-Master Port stattfindet, wird der ursprüngliche Port wieder zum Master-Port, sobald er wieder verfügbar ist.

fec / loadbalance (Lastverteilung)
: Cisco® Fast EtherChannel® (FEC) findet sich auf älteren Cisco® Switches. Es bietet eine statische Konfiguration und handelt weder Aggregation mit der Gegenstelle aus, noch werden Frames zur Überwachung der Verbindung ausgetauscht. Wenn der Switch LACP unterstützt, sollte diese Option auch verwendet werden.

lacp
: Das IEEE® 802.3ad Link-Aggregation Control Protokoll (LACP). Mit LACP wird eine Menge von aggregierbaren Verbindungen mit der Gegenstelle in einer oder mehreren Link Aggregated Groups (LAG) ausgehandelt. Jede LAG besteht aus Ports der gleichen Geschwindigkeit, eingestellt auf Voll-Duplex-Betrieb. Der Verkehr wird über die Ports in der LAG mit der größten Gesamtgeschwindigkeit balanciert. Typischerweise gibt es nur eine LAG, die alle Ports enthält. Im Falle von Änderungen in der physischen Anbindung wird LACP schnell zu einer neuen Konfiguration konvergieren.

  LACP balanciert ausgehenden Verkehr über die aktiven Ports basierend auf der gehashten Protokollheaderinformation und akzeptiert eingehenden Verkehr auf jedem aktiven Port. Der Hash beinhaltet die Ethernet-Quell- und Zieladresse, und, soweit verfügbar, den VLAN-Tag, sowie die IPv4 oder IPv6 Quell- und Zieladresse.

roundrobin
: Dieser Modus verteilt ausgehenden Verkehr mittels einer Round-Robin-Zuteilung über alle aktiven Ports und akzeptiert eingehenden Verkehr auf jedem aktiven Port. Da dieser Modus die Reihenfolge von Ethernet-Rahmen verletzt, sollte er mit Vorsicht eingesetzt werden.

31.7.1. Beispiele

Dieser Abschnitt zeigt, wie man einen Cisco® Switch und ein FreeBSD-System für LACP Load Balancing konfiguriert. Weiterhin wird gezeigt, wie man zwei Ethernet-Schnittstellen, sowie eine Ethernet- und eine Drahtlos-Schnittstelle für den Failover-Modus konfigurieren kann.

Beispiel 31.1. LACP Aggregation mit einem Cisco® Switch

Dieses Beispiel verbindet zwei fxp(4) Ethernet-Schnittstellen einer FreeBSD-Maschine zu den ersten zwei Ethernet-Ports auf einem Cisco® Switch als eine einzelne, lastverteilte und ausfallsichere Verbindung. Weitere Adapter können hinzugefügt werden, um den Durchsatz zu erhöhen und die Ausfallsicherheit zu steigern. Ersetzen Sie die Namen der Cisco®-Ports, Ethernet-Geräte, channel-group Nummern und IP-Adressen im Beispiel durch Namen, die mit Ihrer lokalen Konfiguration übereinstimmen.

Da die Reihenfolge der Frames bei Ethernet zwingend eingehalten werden muss, fließt auch jeglicher Verkehr zwischen zwei Stationen über den gleichen physischen Kanal, was die maximale Geschwindigkeit der Verbindung auf die eines einzelnen Adapters beschränkt. Der Übertragungsalgorithmus versucht, so viele Informationen wie möglich zu verwenden, um die verschiedenen Verkehrsflüsse zu unterscheiden und balanciert diese über die verfügbaren Adapter.

Fügen Sie auf dem Cisco®-Switch die Adapter *`FastEthernet0/1`* und *`FastEthernet0/2`* zu der channel-group *`1`* hinzu:

```
interface
FastEthernet0/1
 channel-group  1 mode active
 channel-protocol lacp
!
interface  FastEthernet0/2
 channel-group  1 mode active
 channel-protocol lacp
```

Erstellen Sie auf der FreeBSD Maschine die lagg(4)-Schnittstelle unter Verwendung von *`fxp0`* und *`fxp1`* und starten Sie die Schnittstelle mit der IP-Adresse *`10.0.0.3/24`* :

```
# ifconfig  fxp0  up
# ifconfig  fxp1  up
# ifconfig  lagg0 create
# ifconfig  lagg0 up laggproto lacp laggport   fxp0 laggport  fxp1 10.0.0.3/24
```

Überprüfen Sie den Status der virtuellen Schnittstelle:

```
# ifconfig  lagg0
lagg0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=8<VLAN_MTU>
        ether 00:05:5d:71:8d:b8
        media: Ethernet autoselect
        status: active
        laggproto lacp
        laggport: fxp1 flags=1c<ACTIVE,COLLECTING,DISTRIBUTING>
        laggport: fxp0 flags=1c<ACTIVE,COLLECTING,DISTRIBUTING>
```

Ports, die als *ACTIVE* markiert sind, sind Teil der aktiven Aggregations-Gruppe, die mit dem Switch ausgehandelt wurde. Der Verkehr wird über diese Gruppe übertragen und empfangen. Benutzen Sie ifconfig(8) mit `-v`, um sich die LAG-Bezeichner anzeigen zu lassen.

Um den Status der Ports auf dem Switch anzuzeigen, benutzen Sie **`show lacp neighbor`**:

```
switch# show lacp neighbor
Flags:  S - Device is requesting Slow LACPDUs
        F - Device is requesting Fast LACPDUs
        A - Device is in Active mode       P - Device is in Passive mode

Channel group 1 neighbors

Partner's information:

                  LACP port                        Oper    Port     Port
Port      Flags   Priority  Dev ID         Age     Key     Number   State
Fa0/1     SA      32768     0005.5d71.8db8  29s    0x146   0x3      0x3D
Fa0/2     SA      32768     0005.5d71.8db8  29s    0x146   0x4      0x3D
```

Benutzen Sie **`show lacp neighbor detail`**, um weitere Informationen zu erhalten.

Damit diese Konfiguration auch nach einem Neustart erhalten bleibt, fügen Sie auf dem FreeBSD-System folgende Einträge in `/etc/rc.conf` hinzu:

```
ifconfig_fxp0="up"
ifconfig_fxp1="up"
cloned_interfaces="lagg0
ifconfig_lagg0="laggproto lacp laggport fxp0 laggport fxp1 10.0.0.3/24 "
```

Beispiel 31.2. Ausfallsicherer Modus

Der ausfallsichere Modus kann verwendet werden, um zu einer zweiten Schnittstelle zu wechseln, sollte die Verbindung mit der Master-Schnittstelle ausfallen. Um den ausfallsicheren Modus zu konfigurieren, aktivieren Sie zunächst die zugrunde liegenden physikalischen Schnittstellen. Erstellen Sie dann die lagg(4)-Schnittstelle mit *`fxp0`* als Master-Schnittstelle und *`fxp1`* als sekundäre Schnittstelle. Der virtuellen Schnittstelle wird die IP-Adresse *`10.0.0.15/24`* zugewiesen:

```
# ifconfig fxp0 up
# ifconfig fxp1 up
# ifconfig lagg0 create
# ifconfig lagg0 up laggproto failover laggport  fxp0 laggport fxp1 10.0.0.15/24
```

Die virtuelle Schnittstelle sollte in etwa so aussehen:

```
# ifconfig lagg0
lagg0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=8<VLAN_MTU>
        ether 00:05:5d:71:8d:b8
        inet 10.0.0.15 netmask 0xffffff00 broadcast 10.0.0.255
        media: Ethernet autoselect
        status: active
        laggproto failover
        laggport: fxp1 flags=0<>
        laggport: fxp0 flags=5<MASTER,ACTIVE>
```

Der Verkehr wird auf *`fxp0`* übertragen und empfangen. Wenn die Verbindung auf *`fxp0`* abbricht, wird *`fxp1`* die Verbindung übernehmen. Sobald die Verbindung auf der Master-Schnittstelle wiederhergestellt ist, wird diese wieder als aktive Schnittstelle genutzt.

Damit diese Konfiguration auch nach einem Neustart erhalten bleibt, fügen Sie folgende Einträge in `/etc/rc.conf` hinzu:

```
ifconfig_fxp0="up"
```

```
ifconfig_fxp1="up"
cloned_interfaces="lagg0
ifconfig_lagg0="laggproto failover laggport fxp0 laggport fxp1 10.0.0.15/24 "
```

Beispiel 31.3. Failover Modus zwischen Ethernet- und Wireless-Schnittstellen

Für Laptop-Benutzer ist es normalerweise wünschenswert, „wireless“ als sekundäre Schnittstelle einzurichten, die verwendet wird, wenn die Ethernet-Verbindung nicht verfügbar ist. Mit lagg(4) ist es möglich, ein Failover mit einer IP-Adresse zu konfigurieren, welches die Ethernet-Verbindung aus Performance- und Sicherheitsgründen bevorzugt, während es gleichzeitig möglich bleibt, Daten über die drahtlose Verbindung zu übertragen.

Dies wird durch das Überschreiben der physikalischen MAC-Adresse der drahtlosen Schnittstelle, durch die der Ethernet-Schnittstelle erreicht.

In diesem Beispiel ist die Ethernet-Schnittstelle *bge0* die Master-Schnittstelle und die drahtlose Schnittstelle *wlan0* die Failover-Schnittstelle. *wlan0* wurde von der *iwn0* mit der MAC-Adresse der Ethernet-Schnittstelle eingerichtet. Im ersten Schritt wird die MAC-Adresse der Ethernet-Schnittstelle ermittelt:

```
# ifconfig bge0
bge0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
 options=19b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM,TSO4>
 ether 00:21:70:da:ae:37
 inet6 fe80::221:70ff:feda:ae37%bge0 prefixlen 64 scopeid 0x2
 nd6 options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL>
 media: Ethernet autoselect (1000baseT <full-duplex>)
 status: active
```

Ersetzen Sie *bge0* durch den Namen der Ethernet-Schnittstelle des Systems. Die `ether`-Zeile wird die MAC-Adresse der angegebenen Schnittstelle enthalten. Ändern Sie nun die MAC-Adresse der zugrunde liegenden drahtlosen Schnittstelle:

```
# ifconfig iwn0 ether 00:21:70:da:ae:37
```

Starten Sie die drahtlose Schnittstelle, aber ohne eine IP-Adresse zu setzen:

```
# ifconfig wlan0 create wlandev iwn0 ssid my_router up
```

Stellen Sie sicher, dass die *bge0*-Schnittstelle aktiv ist. Erstellen Sie die lagg(4)-Schnittstelle mit *bge0* als Master und *wlan0* als Failover:

```
# ifconfig bge0 up
# ifconfig lagg0 create
# ifconfig lagg0 up laggproto failover laggport  bge0 laggport wlan0
```

Die virtuelle Schnittstelle sollte in etwa so aussehen:

```
# ifconfig lagg0
lagg0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=8<VLAN_MTU>
        ether 00:21:70:da:ae:37
        media: Ethernet autoselect
        status: active
        laggproto failover
        laggport: wlan0 flags=0<>
        laggport: bge0 flags=5<MASTER,ACTIVE>
```

Starten Sie dann den DHCP-Client, um eine IP-Adresse zu erhalten:

```
# dhclient lagg0
```

Damit diese Konfiguration auch nach einem Neustart erhalten bleibt, fügen Sie folgende Einträge in `/etc/rc.conf` hinzu:

```
ifconfig_bge0="up"
ifconfig_iwn0="ether 00:21:70:da:ae:37 "
wlans_iwn0="wlan0"
ifconfig_wlan0="WPA"
cloned_interfaces="lagg0"
ifconfig_lagg0="laggproto failover laggport
bge0 laggport wlan0
DHCP"
```

31.8. Plattenloser Betrieb mit PXE

Aktualisiert von Jean-François Dockès.
Reorganisiert und erweitert von Alex Dupre.

Das Intel® Preboot eXecution Environment (PXE) erlaubt es dem Betriebssystem über das Netzwerk zu starten. Zum Beispiel kann ein FreeBSD-System, ohne lokale Festplatte, über das Netzwerk gestartet und betrieben werden. Die Dateisysteme werden dabei über einen NFS-Server eingehangen. PXE-Unterstützung steht in der Regel im BIOS zur Verfügung. Um PXE beim Systemstart zu verwenden, müssen Sie im BIOS des Rechners die Option `Über das Netzwerk starten` aktivieren. Alternativ können Sie während der PC-Initialisierung auch eine Funktionstaste drücken.

Um die notwendigen Dateien für ein Betriebssystem für den Start über das Netzwerk bereitzustellen, benötigt ein PXE-Setup einen richtig konfigurierten DHCP-, TFTP- und NFS-Server, wobei:

- Die initialen Parameter, wie IP-Adresse, Dateiname und Speicherort der ausführbaren Bootdateien, Servername sowie Root-Pfad vom DHCP-Server bezogen werden.
- Der Loader für das Betriebssystem über TFTP gestartet wird.
- Die Dateisysteme über NFS geladen werden.

Sobald das Gastsystem über PXE startet, erhält es vom DHCP-Server Informationen, wo der initiale Bootloader per TFTP zu bekommen ist. Nachdem das Gastsystem diese Informationen erhalten hat, lädt es den Bootloader über TFTP herunter und führt diesen anschließend aus. In FreeBSD ist `/boot/pxeboot` der Bootloader. Nachdem `/boot/pxeboot` ausgeführt und der FreeBSD-Kernel geladen wurde, wird mit dem Rest der FreeBSD-Bootsequenz, wie in Kapitel 13, *FreeBSDs Bootvorgang* beschrieben, fortgefahren.

Dieser Abschnitt beschreibt, wie Sie diese Dienste auf einem FreeBSD-System so konfigurieren, sodass andere Systeme FreeBSD über PXE starten können. Weitere Informationen finden Sie in diskless(8).

Achtung

Wie beschrieben, ist das System, welches diese Dienste bereitstellt, unsicher. Daher sollte es in einem geschützten Bereich des Netzwerks aufgestellt und von anderen Hosts als nicht vertrauenswürdig eingestuft werden.

31.8.1. Konfiguration der PXE-Umgebung

Beigetragen von Craig Rodrigues.

Die in diesem Abschnitt dargestellten Schritte konfigurieren die in FreeBSD enthaltenen NFS- und TFTP-Server. Der folgende Abschnitt beschreibt die Installation und Konfiguration des DHCP-Servers. In diesem Beispiel verwenden wir `/b/tftpboot/FreeBSD/install` , welches die Dateien für PXE-Benutzer enthält. Es ist wichtig, dass dieses Verzeichnis existiert und das der gleiche Verzeichnisname ebenfalls in `/etc/inetd.conf` und `/usr/local/etc/dhcpd.conf` gesetzt wird.

1. Erstellen Sie das Root-Verzeichnis, welches eine FreeBSD-Installation enthält und über NFS eingehangen werden kann:

```
# export NFSROOTDIR=/b/tftpboot/FreeBSD/install
# mkdir -p ${NFSROOTDIR}
```

2. Aktivieren Sie den NFS-Server, indem Sie folgende Zeile in `/etc/rc.conf` hinzufügen:

```
nfs_server_enable="YES"
```

   Exportieren Sie das Root-Verzeichnis über NFS, indem Sie folgende Zeile in `/etc/exports` hinzufügen:

```
/b -ro -alldirs
```

3. Starten Sie den NFS-Server:

```
# service nfsd start
```

4. Aktivieren Sie inetd(8), indem Sie folgende Zeile in `/etc/rc.conf` hinzufügen:

```
inetd_enable="YES"
```

5. Kommentieren Sie die folgende Zeile in `/etc/inetd.conf` aus, indem Sie sicherstellen, dass die Zeile nicht mit einem #-Zeichen beginnt:

```
tftp dgram udp wait root /usr/libexec/tftp tftp -l -s /b/tftpboot
```

Anmerkung

Einige PXE-Versionen benötigen die TCP-Version von TFTP. In diesem Fall können Sie die zweite `tftp`-Zeile, welche `stream tcp` enthält, auskommentieren.

6. Starten Sie inetd(8):

```
# service inetd start
```

7. Erstellen Sie einen neues Basissystem und einen FreeBSD-Kernel. Detaillierte Anweisungen hierzu finden Sie im Abschnitt 24.7, „Das komplette Basissystem neu bauen“:

```
# cd /usr/src
# make buildworld
# make buildkernel
```

8. Installieren Sie FreeBSD in das Verzeichnis, welches über NFS eingehangen ist:

```
# make installworld DESTDIR=${NFSROOTDIR}
# make installkernel DESTDIR=${NFSROOTDIR}
# make distribution DESTDIR=${NFSROOTDIR}
```

9. Testen Sie den TFTP-Server und vergewissern Sie sich, dass Sie den Bootloader herunterladen können, der über PXE bereitgestellt wird:

```
# tftp localhost
tftp> get FreeBSD/install/boot/pxeboot
Received 264951 bytes in 0.1 seconds
```

10. Bearbeiten Sie `${NFSROOTDIR}/etc/fstab` und erstellen Sie einen Eintrag, um das Root-Dateisystem über NFS einzuhängen:

```
# Device                                        Mountpoint    FSType    Options    Dump
  Pass
myhost.example.com :/b/tftpboot/FreeBSD/install    /         nfs       ro         0
  0
```

    Ersetzen Sie *myhost.example.com* durch den Hostnamen oder die IP-Adresse des NFS-Servers. In diesem Beispiel wird das Root-Dateisystem schreibgeschützt eingehangen, um ein potenzielles Löschen des Inhalts durch die NFS-Clients zu verhindern.

11. Setzen Sie das root-Passwort in der PXE-Umgebung für Client-Maschinen, die über PXE starten:

```
# chroot ${NFSROOTDIR}
# passwd
```

12. Falls erforderlich, aktivieren Sie ssh(1) root-Logins für Client-Maschinen, die über PXE starten, indem Sie die Option `PermitRootLogin` in `${NFSROOTDIR}/etc/ssh/sshd_config` aktivieren. Dies ist in sshd_config(5) dokumentiert.

13. Führen Sie alle weiteren Anpassungen der PXE-Umgebung in ${NFSROOTDIR} durch, wie zum Beispiel die Installation weiterer Pakete, oder dass Bearbeiten der Passwortdatei mit vipw(8).

Booten Sie von einem NFS-Root-Volume, so erkennt `/etc/rc` dies und startet daraufhin das `/etc/rc.initdiskless` Skript. Lesen Sie die Kommentare in diesem Skript um zu verstehen, was dort vor sich geht. Weil das NFS-Root-Verzeichnis schreibgeschützt ist, wir aber Schreibzugriff für `/etc` und `/var` benötigen, müssen wir diese Verzeichnisse über Speicher-Dateisysteme (memory backed file system) einbinden.

```
# chroot ${NFSROOTDIR}
# mkdir -p conf/base
# tar -c -v -f conf/base/etc.cpio.gz --format cpio --gzip etc
# tar -c -v -f conf/base/var.cpio.gz --format cpio --gzip var
```

31.8.2. Konfiguration des DHCP-Servers

Der DHCP-Server muss nicht auf der selben Maschine laufen wie der TFTP- und NFS-Server, aber er muss über das Netzwerk erreichbar sein.

DHCP ist nicht Bestandteil des FreeBSD Basissystems, kann jedoch über den Port net/isc-dhcp42-server oder als Paket nachinstalliert werden.

Einmal installiert, bearbeiten Sie die Konfigurationsdatei `/usr/local/etc/dhcpd.conf` . Konfigurieren Sie die `next-server` , `filename` und `root-path` Einstellungen, wie in diesem Beispiel zu sehen ist:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
range 192.168.0.2 192.168.0.3;
option subnet-mask 255.255.255.0;
option routers 192.168.0.1;
option broadcast-address 192.168.0.255;
option domain-name-servers 192.168.35.35, 192.168.35.36;
option domain-name "example.com";

# IP address of TFTP server
next-server 192.168.0.1 ;
```

```
# path of boot loader obtained via tftp
filename "FreeBSD/install/boot/pxeboot  ";

# pxeboot boot loader will try to NFS mount this directory for root FS
option root-path "192.168.0.1:/b/tftpboot/FreeBSD/install/   ";
}
```

Die Anweisung `next-server` wird benutzt, um die IP-Adresse des TFTP-Servers anzugeben.

Die Anweisung `filename` definiert den Pfad zu `/boot/pxeboot`. Da hier der relative Dateiname verwendet wird, bedeutet das, dass `/b/tftpboot` nicht im Pfad enthalten ist.

Die Option `root-path` bestimmt den Pfad zum NFS root-Dateisystem.

Sobald die Änderungen gespeichert werden, aktivieren Sie NFS beim Systemstart, indem Sie die folgende Zeile in `/etc/rc.conf` hinzufügen:

```
dhcpd_enable="YES"
```

Starten Sie anschließend den DHCP-Dienst:

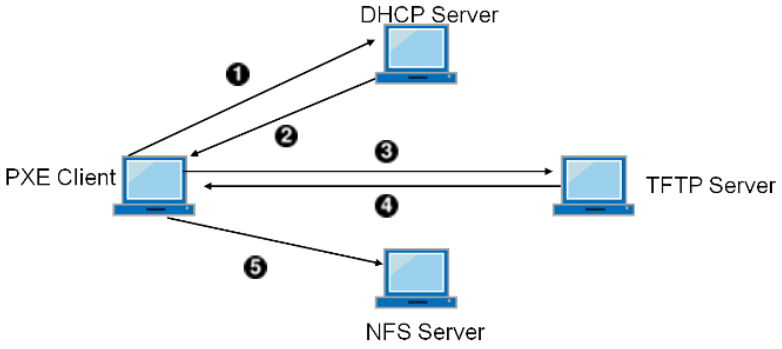
```
# service isc-dhcpd start
```

31.8.3. Fehlersuche bei PXE Problemen

Sobald alle Dienste konfiguriert und gestartet wurden, sollten PXE-Clients in der Lage sein, FreeBSD automatisch über das Netzwerk zu starten. Wenn ein bestimmter Client beim hochfahren keine Verbindung herstellen kann, sehen Sie im BIOS nach, ob die Option für den Start über das Netzwerk konfiguriert ist.

Dieser Abschnitt gibt einige Tipps zu Fehlerbehebung und zeigt, wie Sie Konfigurationsprobleme eingrezen können für den Fall, dass PXE-Clients nicht in der Lage sind über das Netzwerk zu starten.

1. Benutzen Sie den net/wireshark Port um Fehler im Netzwerkverkehr während des Bootvorgangs von PXE zu finden. Der Bootvorgang wird im folgenden Diagramm schematisch dargestellt.



   ❶ Client sendet eine `DHCPDISCOVER` Nachricht.
   ❷ Der DHCP-Server antwortet mit einer IP-Adresse, sowie den Werten für `next-server`, `filename` und `root-path`.
   ❸ Der Client sendet eine TFTP-Anfrage an `next-server`, mit der Bitte `filename` zu empfangen.
   ❹ Der TFTP-Server antwortet und sendet `filename` zum Client.
   ❺ Der Client führt `filename`, sprich pxeboot(8) aus, was wiederum den Kernel lädt. Wenn der Kernel ausgeführt wird, wird das Root-Dateisystem, welches in `root-path` spezifiziert ist, über NFS eingebunden.

   Abbildung 31.1. PXE-Bootvorgang mit NFS Root Mount

2. Schauen Sie in `/var/log/xferlog` auf Ihrem TFTP-Server und vergewissern Sie sich, dass die `pxeboot`-Datei von der richtigen Adresse heruntergeladen wurde. Um die obige Konfiguration von `/usr/local/etc/dhcpd.conf` zu testen, geben Sie folgendes ein:

```
# tftp 192.168.0.1
tftp> get FreeBSD/install/boot/pxeboot
Received 264951 bytes in 0.1 seconds
```

Weitere Informationen finden Sie in tftpd(8) und tftp(1). Die `BUGS`-Sektionen dieser Seiten dokumentieren einige Einschränkungen von TFTP.

3. Achten Sie darauf, dass Sie das Root-Dateisystem über NFS einhängen können. Auch hier können Sie Ihre Einstellungen aus `/usr/local/etc/dhcpd.conf` wie folgt testen:

```
# mount -t nfs 192.168.0.1:/b/tftpboot/FreeBSD/install /mnt
```

31.9. IPv6

Beigetragen von Aaron Kaplan.
Überarbeitet und erweitert von Tom Rhodes.
Erweitert von Brad Davis.

IPv6 ist die neueste Version des bekannten IP-Protokolls, das auch als IPv4 bezeichnet wird. IPv6 bietet gegenüber IPv4 mehrere Vorteile sowie viele neue Funktionen:

- IPv6 hat einen 128 Bit großen Adressraum, der 340.282.366.920.938.463.463.374.607.431.768.211.456 Adressen erlaubt. Dies behebt das Problem der immer knapper werdenden IPv4-Adressen und einer eventuellen Erschöpfung des IPv4-Adressraums.
- Router speichern nur noch Netzwerk-Aggregationsadressen in ihren Routingtabellen. Dadurch reduziert sich die durchschnittliche Größe einer Routingtabelle auf 8192 Einträge. Dies ist mit den Problemen bei der Skalierbarkeit von IPv4 verbunden, da jeder zugeordnete Block von IPv4-Adressen erfordert, dass Routing-Informationen zwischen vielen Routern im Internet ausgetauscht werden müssen. Die Routing-Tabellen wurden mit der Zeit so groß, dass ein effizientes Routing jetzt kaum noch möglich ist.
- Die automatische Konfiguration von Adressen, die im RFC2462 beschrieben wird.
- Verpflichtende Multicast-Adressen.
- Integriertes IPsec (IP-Security).
- Eine vereinfachte Headerstruktur.
- Unterstützung für mobile IP-Adressen.
- Die Umwandlung von IPv4- in IPv6-Adressen.

FreeBSD enthält die IPv6-Referenzimplementation von KAME und erfüllt damit bereits alle für die Nutzung von IPv6 nötigen Voraussetzungen. Dieser Abschnitt konzentriert sich auf die Konfiguration und den Betrieb von IPv6.

31.9.1. Hintergrundinformationen zu IPv6-Adressen

Es gibt verschiedene Arten von IPv6-Adressen:

Unicast
: Ein Paket, das an eine Unicast-Adresse gesendet wird, kommt nur an der Schnittstelle an, die dieser Adresse zugeordnet ist.

Anycast
: Anycast-Adressen unterscheiden sich in ihrer Syntax nicht von Unicast-Adressen, sie wählen allerdings aus mehreren Schnittstellen eine Schnittstelle aus. Ein für eine Anycast-Adresse bestimmtes Paket kommt an der

nächstgelegenen (entsprechend der Router-Metrik) Schnittstelle an. Anycast-Adressen werden nur von Routern verwendet.

Mulitcast
: Multicast-Adressen bestimmen Gruppen, denen mehrere Schnittstellen angehören. Ein Paket, das an eine Multicast-Adresse geschickt wird, kommt an allen Schnittstellen an, die zur Multicast-Gruppe gehören. Die von IPv4 bekannte Broadcast-Adresse (normalerweise `xxx.xxx.xxx.255`) wird bei IPv6 durch Multicast-Adressen verwirklicht.

Die kanonische Form einer IPv6-Adresse lautet `x:x:x:x:x:x:x:x` , wobei jedes „x“ für einen 16-Bit-Hexadezimalwert steht. Ein Beispiel für eine IPv6-Adresse wäre etwa `FEBC:A574:382B:23C1:AA49:4592:4EFE:9982`.

Eine IPv6-Adresse enthält oft Teilzeichenfolgen aus lauter Nullen. Eine solche Zeichenfolge kann zu „::“ verkürzt werden. Bis zu drei führende Nullen eines Hexquads können ebenfalls weggelassen werden. `fe80::1` entspricht also der Adresse `fe80:0000:0000:0000:0000:0000:0000:0001` .

Eine weitere Möglichkeit ist die Darstellung der letzten 32 Bit in der bekannten IPv4-Notation. `2002::10.0.0.1` ist also eine andere Schreibweise für die (hexadezimale) kanonische Form `2002:0000:0000:0000:0000:0000:0a00:0001` , die wiederum der Adresse `2002::a00:1` entspricht.

Benutzen Sie ifconfig(8), um die IPv6-Adresse eines FreeBSD-Systems anzuzeigen:

```
# ifconfig
rl0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
        inet 10.0.0.10 netmask 0xffffff00 broadcast 10.0.0.255
        inet6 fe80::200:21ff:fe03:8e1%rl0 prefixlen 64 scopeid 0x1
        ether 00:00:21:03:08:e1
        media: Ethernet autoselect (100baseTX )
        status: active
```

Bei `fe80::200:21ff:fe03:8e1%rl0` handelt es sich um eine automatisch konfigurierte *link-local*-Adresse. Sie wird im Rahmen der automatischen Konfiguration aus der MAC-Adresse erzeugt.

Einige IPv6-Adressen sind reserviert. Eine Zusammenfassung dieser Adressen finden Sie in Tabelle 31.3, „Reservierte IPv6-Adressen“:

Tabelle 31.3. Reservierte IPv6-Adressen

IPv6-Adresse	Präfixlänge	Beschreibung	Anmerkungen
`::`	128 Bit	nicht festgelegt	entspricht `0.0.0.0` bei IPv4.
`::1`	128 Bit	Loopback-Adresse	entspricht `127.0.0.1` bei IPv4.
`::00:xx:xx:xx:xx`	96 Bit	Eingebettete IPv4-Adresse	Die niedrigen 32 Bit sind die kompatiblen IPv4-Adressen.
`::ff:xx:xx:xx:xx`	96 Bit	Eine auf IPv6 abgebildete IPv4-Adresse.	Die niedrigen 32 Bit sind IPv4-Adressen für Hosts, die kein IPv6 unterstützen.
`fe80::/10`	10 Bit	link-local	Entspricht 196.254.0.0/16 bei IPv4.
`fc00::/7`	7 Bit	unique-local	Diese einzigartigen Adressen sind für die lokale Kommunikation bestimmt und werden nur innerhalb von abgegrenzten Standorten (*Sites*) weitergeleitet.

IPv6-Adresse	Präfixlänge	Beschreibung	Anmerkungen
`ff00::`	8 Bit	Multicast	
`2000::-3fff::`	3 Bit	Globaler Unicast	Alle globalen Unicast-Adressen stammen aus diesem Pool. Die ersten 3 Bit lauten `001`.

Weitere Informationen zum Aufbau von IPv6-Adressen finden Sie im RFC3513.

31.9.2. IPv6 konfigurieren

Um ein FreeBSD-System als IPv6-Client zu konfigurieren, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
ifconfig_em0_ipv6="inet6 accept_rtadv"
rtsold_enable="YES"
```

Die erste Zeile ermöglicht der angegebenen Schnittstelle, Router-Solicitation-Nachrichten zu empfangen. Die zweite Zeile aktiviert den Router-Solicitation-Daemon rtsold(8).

Für FreeBSD 8.*x* (und älter) fügen Sie stattdessen folgende Zeile hinzu:

```
ipv6_enable="YES"
```

Falls die Schnittstelle eine statisch zugewiesene IPv6-Adresse benötigt, fügen Sie einen Eintrag mit der statischen Adresse und dem zugehörigen Präfix für das Subnetz hinzu:

```
ifconfig_fxp0_ipv6="inet6 2001:db8:4672:6565:2026:5043:2d42:5344    prefixlen 64"
```

Für FreeBSD 8.*x* fügen Sie dagegen folgende Zeile hinzu:

```
ipv6_ifconfig_fxp0="2001:db8:4672:6565:2026:5043:2d42:5344   "
```

Um einen Standardrouter festzulegen, fügen Sie die Adresse hinzu:

```
ipv6_defaultrouter="2001:db8:4672:6565::1 "
```

31.9.3. Verbindung zu einem Provider aufbauen

Um sich mit anderen IPv6-Netzwerken zu verbinden, benötigen Sie einen Provider oder einen Tunnel, der IPv6 unterstützt:

- Fragen Sie einen Internetprovider, ob er IPv6 anbietet.
- SixXS bietet weltweit IPv6-Tunnelverbindungen an.
- Hurricane Electric bietet weltweit IPv6-Tunnelverbindungen an.

Anmerkung

Die Verwendung des Ports `/usr/ports/net/freenet6` für Einwahlverbindungen.

Dieser Abschnitt beschreibt, wie Sie die Anweisungen eines Tunnel-Providers dauerhaft in `/etc/rc.conf` einrichten.

Der erste Eintrag in `/etc/rc.conf` erzeugt die generische Tunnelschnittstelle `gif0`:

```
gif_interfaces="gif0"
```

Als nächstes konfigurieren Sie die IPv4-Adressen der lokalen und entfernten Endpunkte. Ersetzen Sie *MY_IPv4_ADDR* und *REMOTE_IPv4_ADDR* durch die tatsächlichen IPv4-Adressen:

```
gifconfig_gif0="MY_IPv4_ADDR REMOTE_IPv4_ADDR  "
```

Um die zugewiesene IPv6-Adresse als Endpunkt für den IPv6-Tunnel zu verwenden, fügen Sie folgende Zeile für FreeBSD 9.*x* (und neuer) ein:

```
ifconfig_gif0_ipv6="inet6 MY_ASSIGNED_IPv6_TUNNEL_ENDPOINT_ADDR  "
```

Für FreeBSD 8.*x* (und älter) fügen Sie folgende Zeile ein:

```
ipv6_ifconfig_gif0=" MY_ASSIGNED_IPv6_TUNNEL_ENDPOINT_ADDR  "
```

Legen Sie dann die Standardroute für das andere Ende des IPv6-Tunnels fest. Ersetzen Sie *MY_IPv6_REMOTE_TUNNEL_ENDPOINT_ADDR* mit der Adresse des Standard-Gateways des Providers:

```
ipv6_defaultrouter="MY_IPv6_REMOTE_TUNNEL_ENDPOINT_ADDR  "
```

Wenn das FreeBSD-System IPv6-Verkehr zwischen dem Netzwerk und der Außenwelt routen muss, aktivieren Sie das Gateway mit dieser Zeile:

```
ipv6_gateway_enable="YES"
```

31.9.4. Bekanntmachung von Routen und automatische Rechnerkonfiguration

Dieser Abschnitt beschreibt die Einrichtung von rtadvd(8), das Sie bei der Bekanntmachung der IPv6-Standardroute unterstützt.

Um rtadvd(8) zu aktivieren, fügen Sie folgende Zeile in `/etc/rc.conf` ein:

```
rtadvd_enable="YES"
```

Es ist wichtig, die Schnittstelle anzugeben, über die IPv6-Routen bekanntgemacht werden sollen. Soll rtadvd(8) `fxp0` verwenden, ist folgender Eintrag nötig:

```
rtadvd_interfaces="fxp0"
```

Danach erzeugen Sie die Konfigurationsdatei `/etc/rtadvd.conf` . Dazu ein Beispiel:

```
fxp0:\
    :addrs#1:addr="2001:471:1f11:246::":prefixlen#64:tc=ether:
```

Ersetzen Sie dabei `fxp0` durch die zu verwendende Schnittstelle, und `2001:471:1f11:246::` durch das entsprechend zugewiesene Präfix.

Bei einem `/64`-Subnetz müssen keine weiteren Anpassungen vorgenommen werden. Anderenfalls muss `prefixlen#` auf den korrekten Wert gesetzt werden.

31.9.5. IPv6 und Abbildung von IPv6-Adressen

Wenn IPv6 auf einem Server aktiviert ist, kann es für die Kommunikation erforderlich sein, IPv4-Adressen auf IPv6-Adressen abzubilden. Diese Kompatibilität erlaubt es, das IPv4-Adressen als IPv6-Adressen dargestellt werden. Die Kommunikation von IPv6-Anwendungen mit IPv4 und umgekehrt kann jedoch ein Sicherheitsrisiko darstellen.

Diese Option dient nur der Kompatibilität und wird in den meisten Fällen nicht erforderlich sein. Die Option ermöglicht es IPv6-Anwendungen zusammen mit IPv4 in einer Dual-Stack-Umgebung zu funktionieren. Dies ist besonders nützlich für Anwendungen von Drittanbietern, die evtl. keine IPv6-Umgebungen unterstützen. Um diese Funktion zu aktivieren, fügen Sie folgendes in `/etc/rc.conf` hinzu:

```
ipv6_ip4mapping="YES"
```

Für einige Administratoren können die Informationen im RFC 3493 (Sektion 3.6 und 3.7) und RFC 4038 (Sektion 4.2) hilreich sein.

31.10. Common Address Redundancy Protocol (CARP)

Beigetragen von Tom Rhodes.
Aktualisiert von Allan Jude.

Das *Common Address Redundancy Protocol* (CARP) erlaubt es, mehreren Rechnern die gleiche IP-Adresse zuzuweisen und *Hochverfügbarkeit* bereitzustellen. Das bedeutet, dass ein oder mehrere Rechner ausfallen können und die anderen Rechner transparent einspringen, ohne das der Benutzer etwas von einem Ausfall mitbekommt. Neben der gemeinsamen IP-Adresse, haben die jeweiligen Rechner auch eine eindeutige IP-Adresse zur Verwaltung und Konfiguration, wie in den folgenden Beispielen zu sehen ist.

31.10.1. Hochverfügbarkeit mit CARP

CARP wird häufig verwendet, um einen oder mehrere Dienste hochverfügbar zu machen. Dieses Beispiel konfiguriert eine Failover-Unterstützung mit drei Servern (mit jeweils eigener, eindeutiger IP-Adresse), die alle den gleichen Web-Inhalt anbieten. Die Lastverteilung dieser Maschinen wird dabei über *Round Robin* DNS konfiguriert. Mit Ausnahme des Hostnamens und der IP-Management-Adresse sind Master- und Backup-Maschinen identisch konfiguriert. Die Server müssen die gleiche Konfiguration und die gleichen Dienste aktiviert haben. Tritt ein Failover auf, können Anfragen an den Dienst mit der gemeinsam genutzten IP-Adresse nur dann richtig beantwortet werden, wenn der Backup-Server Zugriff auf denselben Inhalt hat. Die Backup-Maschine verfügt über zwei zusätzliche CARP-Schnittstellen, eine für jede IP-Adresse des Master-Content-Servers. Sobald ein Fehler auftritt, übernimmt der Backup-Server die IP-Adresse des ausgefallenen Master-Servers. Die Benutzer werden einen Dienstausfall überhaupt nicht bemerken.

Dieses Beispiel benutzt zwei verschiedene Master namens `hosta.example.org` und `hostb.example.org` mit einem gemeinsamen Backup namens `hostc.example.org`.

Jede virtuelle IP-Adresse hat eine eindeutige Identifikationsnummer, die als *Virtual Host Identification* (VHID) bekannt ist. Alle Maschinen, die sich eine IP-Adresse teilen, verwenden die gleiche VHID. Die VHID für jede einzelne IP-Adresse muss, entsprechend der Broadcast-Domäne der Netzwerkschnittstelle, eindeutig sein.

31.10.2. CARP mit FreeBSD 10 (und neuer) benutzen

Unterstützung für CARP erhalten Sie durch das Laden des Kernelmoduls `carp.ko` in `/boot/loader.conf` :

```
carp_load="YES"
```

Das CARP-Modul kann auch, wie in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben, direkt in den FreeBSD Kernel eingebunden werden:

```
device  carp
```

Hostname, IP-Management-Adresse, Subnetzmaske, gemeinsame IP-Adresse und VHID werden durch das Hinzufügen in `/etc/rc.conf` gesetzt. Dieses Beispiel ist für `hosta.example.org`:

```
hostname="hosta.example.org"
ifconfig_em0="inet 192.168.1.3  netmask 255.255.255.0"
ifconfig_em0_alias0="vhid 1 pass testpass alias 192.168.1.50 /32"
```

Beispiel für `hostb.example.org`:

```
hostname="hostb.example.org"
ifconfig_em0="inet 192.168.1.4  netmask 255.255.255.0"
```

```
ifconfig_em0_alias0="vhid 2 pass testpass alias 192.168.1.51 /32"
```

Anmerkung

Achten Sie unbedingt darauf, dass die durch die Option `pass` an ifconfig(8) übergebenen Passwörter auf beiden Systemen identisch sind, da `carp`-Geräte nur mit Systemen kommunizieren können, die über ein korrektes Passwort verfügen.

Die dritte Maschine, `hostc.example.org` ist so konfiguriert, das sie aktiviert wird, wenn eines der beiden zuvor konfigurierten Systeme ausfällt. Diese Maschine ist mit zwei CARP VHIDs konfiguriert, eine für jede virtuelle IP-Adresse der beiden Master-Server. Die Option `advskew` (CARP advertising skew) wird gesetzt, um sicherzustellen, dass sich der Backup-Server später ankündigt wie der Master-Server. `advskew` steuert die Rangfolge für den Fall das mehrere Backup-Server zur Verfügung stehen. Passen Sie die Konfiguration in `/etc/rc.conf` an:

```
hostname="hostc.example.org"
ifconfig_em0="inet 192.168.1.5  netmask 255.255.255.0"
ifconfig_em0_alias0="vhid 1 advskew 100 pass testpass alias 192.168.1.50 /32"
ifconfig_em1_alias0="vhid 2 advskew 100 pass testpass alias 192.168.1.51 /32"
```

Durch die zwei konfigurierten CARP VHIDs ist `hostc.example.org` in der Lage festzustellen, wenn einer der Master-Server nicht mehr reagiert. Wenn der Master-Server sich später ankündigt als der Backup-Server, übernimmt der Backup-Server die gemeinsame IP-Adresse, bis der Master-Server erneut verfügbar ist.

Preemption ist standardmäßig deaktiviert. Wird *Preemption* aktiviert, kann es vorkommen, dass `hostc.example.org` die virtuelle IP-Adresse nicht wieder an den Master-Server zurückgibt. Der Administrator kann jedoch den Backup-Server dazu zwingen, die übernommene IP-Adresse wieder an den Master-Server zurückzugeben:

```
# ifconfig em0 vhid 1 state backup
```

An dieser Stelle muss entweder das Netzwerk neu gestartet, oder die Maschine neu gebootet werden, um CARP zu aktivieren.

Die Funktionalität von CARP kann, wie in der Manualpage carp(4) beschrieben, über verschiedene sysctl(8) Parameter kontrolliert werden. Mit dem Einsatz von devd(8) können weitere Aktionen zu CARP-Ereignissen ausgelöst werden.

31.10.3. CARP mit FreeBSD 9 (und älter) benutzen

Unterstützung für CARP erhalten Sie durch das Laden des Kernelmoduls `carp.ko` in `/boot/loader.conf`:

```
if_carp_load="YES"
```

CARP kann auch direkt in den Kernel eingebunden werden. Diese Prozedur wird in Kapitel 9, *Konfiguration des FreeBSD-Kernels* beschrieben:

```
device    carp
```

Die CARP-Schnittstellen selbst können mittels ifconfig(8) erstellt werden:

```
# ifconfig carp0 create
```

Konfigurieren Sie Hostnamen, IP-Management-Adresse, die gemeinsam genutzte IP-Adresse und die VHID, indem Sie die erforderlichen Zeilen in `/etc/rc.conf` hinzufügen. Hierzu ein Beispiel für `hosta.example.org`:

```
hostname="hosta.example.org"
ifconfig_fxp0="inet 192.168.1.3  netmask 255.255.255.0"
cloned_interfaces="carp0"
ifconfig_carp0="vhid 1 pass testpass 192.168.1.50 /24"
```

Beispiel für `hostb.example.org`:

```
hostname="hostb.example.org"
ifconfig_fxp0="inet 192.168.1.4  netmask 255.255.255.0"
cloned_interfaces="carp0"
ifconfig_carp0="vhid 2 pass testpass 192.168.1.51 /24"
```

Anmerkung

Preemption ist im GENERIC FreeBSD Kernel deaktiviert. Haben Sie jedoch *Preemption* in einem angepassten Kernel aktiviert, dass `hostc.example.org` die virtuelle IP-Adresse nicht wieder an den Master-Server zurückgibt. Der Administrator kann jedoch den Backup-Server dazu zwingen, die übernommene IP-Adresse wieder an den Master-Server zurückzugeben:

```
# ifconfig carp0 down && ifconfig carp0 up
```

Dieser Befehl muss auf dem `carp`-Gerät ausgeführt werden, dass dem betroffenen System zugeordnet ist.

An dieser Stelle muss entweder das Netzwerk neu gestartet, oder die Maschine neu gebootet werden, um CARP zu aktivieren.

Die Funktionalität von CARP kann, wie in der Manualpage carp(4) beschrieben, über verschiedene sysctl(8) Parameter kontrolliert werden. Mit dem Einsatz von devd(8) können weitere Aktionen zu CARP-Ereignissen ausgelöst werden.

Teil V. Anhang

Inhaltsverzeichnis

Anhang A. Bezugsquellen für FreeBSD

A.1. CD-ROM und DVD Verleger

A.1.1. FreeBSD-CDs und -DVDs

Die FreeBSD-CDs und -DVDs werden von vielen Online-Händlern angeboten:

- FreeBSD Mall, Inc.
  2420 Sand Creek Rd C-1 #347
  Brentwood, CA 94513
  USA
  Telefon: +1 925 240-6652
  Fax: +1 925 674-0821
  E-Mail: <info@freebsdmall.com >
  WWW: http://www.freebsdmall.com/

- Dr. Hinner EDV
  Kochelseestr. 11
  D-81371 München
  Germany
  Telefon: (0177) 428 419 0
  WWW: http://www.hinner.de/linux/freebsd.html

- Linux Distro UK
  42 Wharfedale Road
  Margate
  CT9 2TB
  United Kingdom
  WWW: https://linux-distro.co.uk

- The Linux Emporium
  The Techno Centre, Puma Way
  Parkside
  CV1 2TT
  United Kingdom
  Telefon: +44 (0)247 615 8121
  Fax: +44 1491 837016
  WWW: http://linuxemporium.co.uk

- LinuxCenter.Ru
  Galernaya Street, 55
  Saint-Petersburg

190000
Russia
Telefon: +7-812-3125208
E-Mail: <info@linuxcenter.ru >
WWW: http://linuxcenter.ru/shop/freebsd

A.2. FTP-Server

Die offiziellen Quellen von FreeBSD sind mit anonymous FTP über ein weltweites Netz von FTP-Spiegeln erhältlich. Obwohl `ftp://ftp.FreeBSD.org/pub/FreeBSD/` über eine gute Anbindung verfügt, sollten Sie einen Spiegel in Ihrer Nähe verwenden (insbesondere, wenn Sie selber einen Spiegel einrichten wollen).

Sie können FreeBSD auch über anonymous FTP von den folgenden Spiegeln beziehen. Wenn Sie FreeBSD über anonymous FTP beziehen wollen, wählen Sie bitte einen Spiegel in Ihrer Nähe. Die unter „Haupt-Spiegel“ aufgeführten Spiegel stellen normalerweise das komplette FreeBSD-Archiv (alle momentan erhältlichen Versionen für jede unterstützte Architektur) zur Verfügung. Wahrscheinlich geht es aber schneller, wenn Sie einen Spiegel in Ihrer Nähe benutzen. Die Länder-Spiegel stellen die neusten Versionen für die beliebtesten Architekturen bereit, sie stellen aber unter Umständen nicht das komplette FreeBSD-Archiv bereit. Auf alle Server kann mit anonymous FTP zugegriffen werden, einige Server bieten auch andere Zugriffsmethoden an. Die zur Verfügung stehenden Zugriffsmethoden sind bei jedem Server in Klammern angegeben.

Hauptserver, Hauptspiegel, Armenien, Australien, Brasilien, China, Dänemark, Deutschland, Estland, Finnland, Frankreich, Griechenland, Großbritannien, Hong Kong, Irland, Japan, Kanada, Korea, Lettland, Litauen, Neuseeland, Niederlande, Norwegen, Österreich, Polen, Russland, Saudi Arabien, Schweden, Schweiz, Slowenien, Spanien, Südafrika, Taiwan, Tschechische Republik, Ukraine, USA.

(aktualisiert am: UTC)

Hauptserver

- ftp://ftp.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp.FreeBSD.org/pub/FreeBSD/ / http://ftp.FreeBSD.org/pub/FreeBSD/)

Hauptspiegel
Bei Problemen wenden Sie sich bitte an den Betreuer <mirror-admin@FreeBSD.org> dieser Domain.

- ftp://ftp1.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp4.FreeBSD.org/pub/FreeBSD/ / http://ftp4.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp5.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp7.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp10.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp10.FreeBSD.org/pub/FreeBSD/ / http://ftp10.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp11.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp13.FreeBSD.org/pub/FreeBSD/ (ftp)

- ftp://ftp14.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp14.FreeBSD.org/pub/FreeBSD/)

Armenien
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@am.FreeBSD.org> dieser Domain.

- ftp://ftp1.am.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp1.am.FreeBSD.org/pub/FreeBSD/ / rsync)

Australien
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@au.FreeBSD.org> dieser Domain.

- ftp://ftp.au.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.au.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.au.FreeBSD.org/pub/FreeBSD/ (ftp)

Brasilien
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@br.FreeBSD.org> dieser Domain.

- ftp://ftp2.br.FreeBSD.org/FreeBSD/ (ftp / http://ftp2.br.FreeBSD.org/)
- ftp://ftp3.br.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)
- ftp://ftp4.br.FreeBSD.org/pub/FreeBSD/ (ftp)

China
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@cn.FreeBSD.org> dieser Domain.

- ftp://ftp.cn.FreeBSD.org/pub/FreeBSD/ (ftp)

Dänemark
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@dk.FreeBSD.org> dieser Domain.

- ftp://ftp.dk.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp.dk.FreeBSD.org/pub/FreeBSD/ / http://ftp.dk.FreeBSD.org/pub/FreeBSD/)

Deutschland
Bei Problemen wenden Sie sich bitte an den Betreuer <de-bsd-hubs@de.FreeBSD.org> dieser Domain.

- ftp://ftp.de.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp1.de.FreeBSD.org/freebsd/ (ftp / http://www1.de.FreeBSD.org/freebsd/ / rsync://rsync3.de.FreeBSD.org/freebsd/)
- ftp://ftp2.de.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.de.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp4.de.FreeBSD.org/FreeBSD/ (ftp / http://ftp4.de.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp5.de.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp7.de.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp7.de.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp8.de.FreeBSD.org/pub/FreeBSD/ (ftp)

Estland
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@ee.FreeBSD.org> dieser Domain.

- ftp://ftp.ee.FreeBSD.org/pub/FreeBSD/ (ftp)

Finnland
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@fi.FreeBSD.org> dieser Domain.

- ftp://ftp.fi.FreeBSD.org/pub/FreeBSD/ (ftp)

Frankreich
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@fr.FreeBSD.org> dieser Domain.

- ftp://ftp.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp1.fr.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp1.fr.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp3.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.fr.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)
- ftp://ftp7.fr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.fr.FreeBSD.org/pub/FreeBSD/ (ftp)

Griechenland
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@gr.FreeBSD.org> dieser Domain.

- ftp://ftp.gr.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.gr.FreeBSD.org/pub/FreeBSD/ (ftp)

Großbritannien
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@uk.FreeBSD.org> dieser Domain.

- ftp://ftp.uk.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.uk.FreeBSD.org/pub/FreeBSD/ (ftp / rsync://ftp2.uk.FreeBSD.org/ftp.freebsd.org/pub/FreeBSD/)
- ftp://ftp3.uk.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.uk.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.uk.FreeBSD.org/pub/FreeBSD/ (ftp)

Hong Kong

- ftp://ftp.hk.FreeBSD.org/pub/FreeBSD/ (ftp)

Irland
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@ie.FreeBSD.org> dieser Domain.

- ftp://ftp3.ie.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)

Japan
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@jp.FreeBSD.org> dieser Domain.

- ftp://ftp.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.jp.FreeBSD.org/pub/FreeBSD/ (ftp)

- ftp://ftp7.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.jp.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp9.jp.FreeBSD.org/pub/FreeBSD/ (ftp)

Kanada
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@ca.FreeBSD.org> dieser Domain.

- ftp://ftp.ca.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.ca.FreeBSD.org/pub/FreeBSD/ (ftp)

Korea
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@kr.FreeBSD.org> dieser Domain.

- ftp://ftp.kr.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)
- ftp://ftp2.kr.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.kr.FreeBSD.org/pub/FreeBSD/)

Lettland
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@lv.FreeBSD.org> dieser Domain.

- ftp://ftp.lv.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.lv.FreeBSD.org/pub/FreeBSD/)

Litauen
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@lt.FreeBSD.org> dieser Domain.

- ftp://ftp.lt.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.lt.FreeBSD.org/pub/FreeBSD/)

Neuseeland

- ftp://ftp.nz.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.nz.FreeBSD.org/pub/FreeBSD/)

Niederlande
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@nl.FreeBSD.org> dieser Domain.

- ftp://ftp.nl.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.nl.FreeBSD.org/os/FreeBSD/ / rsync)
- ftp://ftp2.nl.FreeBSD.org/pub/FreeBSD/ (ftp)

Norwegen
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@no.FreeBSD.org> dieser Domain.

- ftp://ftp.no.FreeBSD.org/pub/FreeBSD/ (ftp / rsync)

Österreich
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@at.FreeBSD.org> dieser Domain.

- ftp://ftp.at.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp.at.FreeBSD.org/pub/FreeBSD/ / http://ftp.at.FreeBSD.org/pub/FreeBSD/)

Polen
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@pl.FreeBSD.org> dieser Domain.

- ftp://ftp.pl.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp2.pl.FreeBSD.org

Russland
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@ru.FreeBSD.org> dieser Domain.

- ftp://ftp.ru.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.ru.FreeBSD.org/FreeBSD/ / rsync)
- ftp://ftp2.ru.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.ru.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp4.ru.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.ru.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp5.ru.FreeBSD.org/pub/FreeBSD/ / rsync)
- ftp://ftp6.ru.FreeBSD.org/pub/FreeBSD/ (ftp)

Saudi Arabien
Bei Problemen wenden Sie sich bitte an den Betreuer <ftpadmin@isu.net.sa> dieser Domain.

- ftp://ftp.isu.net.sa/pub/ftp.freebsd.org/ (ftp)

Schweden
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@se.FreeBSD.org> dieser Domain.

- ftp://ftp.se.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.se.FreeBSD.org/pub/FreeBSD/ (ftp / rsync://ftp2.se.FreeBSD.org/)
- ftp://ftp3.se.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.se.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp4.se.FreeBSD.org/pub/FreeBSD/ / http://ftp4.se.FreeBSD.org/pub/FreeBSD/ / http://ftp4.se.FreeBSD.org/pub/FreeBSD/ / rsync://ftp4.se.FreeBSD.org/pub/FreeBSD/ / rsync://ftp4.se.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp6.se.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp6.se.FreeBSD.org/pub/FreeBSD/)

Schweiz
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@ch.FreeBSD.org> dieser Domain.

- ftp://ftp.ch.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.ch.FreeBSD.org/pub/FreeBSD/)

Slowenien
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@si.FreeBSD.org> dieser Domain.

- ftp://ftp.si.FreeBSD.org/pub/FreeBSD/ (ftp)

Spanien
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@es.FreeBSD.org> dieser Domain.

- ftp://ftp.es.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.es.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp3.es.FreeBSD.org/pub/FreeBSD/ (ftp)

Südafrika
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@za.FreeBSD.org> dieser Domain.

- ftp://ftp.za.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.za.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.za.FreeBSD.org/pub/FreeBSD/ (ftp)

Taiwan
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@tw.FreeBSD.org> dieser Domain.

- ftp://ftp.tw.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp.tw.FreeBSD.org/pub/FreeBSD/ / rsync / rsyncv6)

- ftp://ftp2.tw.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp2.tw.FreeBSD.org/pub/FreeBSD/ / http://ftp2.tw.FreeBSD.org/pub/FreeBSD/ / http://ftp2.tw.FreeBSD.org/pub/FreeBSD/ / rsync / rsyncv6)
- ftp://ftp4.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp5.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.tw.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp6.tw.FreeBSD.org/ / rsync)
- ftp://ftp7.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp11.tw.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp11.tw.FreeBSD.org/FreeBSD/)
- ftp://ftp12.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp13.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp14.tw.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp15.tw.FreeBSD.org/pub/FreeBSD/ (ftp)

Tschechische Republik
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@cz.FreeBSD.org> dieser Domain.

- ftp://ftp.cz.FreeBSD.org/pub/FreeBSD/ (ftp / ftp://ftp.cz.FreeBSD.org/pub/FreeBSD/ / http://ftp.cz.FreeBSD.org/pub/FreeBSD/ / http://ftp.cz.FreeBSD.org/pub/FreeBSD/ / rsync / rsyncv6)
- ftp://ftp2.cz.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp2.cz.FreeBSD.org/pub/FreeBSD/)

Ukraine

- ftp://ftp.ua.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp.ua.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp6.ua.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp6.ua.FreeBSD.org/pub/FreeBSD / rsync://ftp6.ua.FreeBSD.org/FreeBSD/)
- ftp://ftp7.ua.FreeBSD.org/pub/FreeBSD/ (ftp)

USA
Bei Problemen wenden Sie sich bitte an den Betreuer <hostmaster@us.FreeBSD.org> dieser Domain.

- ftp://ftp1.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp2.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp3.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp4.us.FreeBSD.org/pub/FreeBSD/ (ftp / ftpv6 / http://ftp4.us.FreeBSD.org/pub/FreeBSD/ / http://ftp4.us.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp5.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp6.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp8.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp10.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp11.us.FreeBSD.org/pub/FreeBSD/ (ftp)
- ftp://ftp13.us.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp13.us.FreeBSD.org/pub/FreeBSD/ / rsync)

- ftp://ftp14.us.FreeBSD.org/pub/FreeBSD/ (ftp / http://ftp14.us.FreeBSD.org/pub/FreeBSD/)
- ftp://ftp15.us.FreeBSD.org/pub/FreeBSD/ (ftp)

A.3. Anonymous CVS (veraltet)

A.3.1. Warnung

Warnung

CVS wurde vom Projekt als veraltet eingestuft. Die Benutzung wird nicht weiter empfohlen. Nutzen Sie stattdessen Subversion.

A.4. CTM

Mit CTM[1] können Sie einen entfernten Verzeichnisbaum mit einem zentralen Baum synchronisieren. Es wurde extra zum Synchronisieren der FreeBSD Quellen entwickelt, obwohl es mit der Zeit vielleicht auch andere Anwendungen geben wird. Zurzeit existiert leider so gut wie keine Dokumentation zum Erstellen der Deltas. Wenn Sie Hilfe benötigen oder CTM für andere Zwecke einsetzen wollen, wenden Sie sich bitte an die Mailingliste ctm-users.

A.4.1. Warum soll ich CTM benutzen?

Mit CTM erhalten Sie eine lokale Kopie des FreeBSD-Quellbaums, den es in mehreren „Varianten“ gibt. Sie können das ganze Repository oder nur einen Zweig spiegeln. Wenn Sie ein aktiver FreeBSD-Entwickler mit einer schlechten oder gar keiner TCP/IP Verbindung sind, oder die Änderungen einfach automatisch zugesandt bekommen wollen, dann ist CTM das Richtige für Sie. Für die Zweige mit der meisten Aktivität müssen Sie sich täglich bis zu drei Deltas beschaffen, Sie sollten allerdings erwägen, die Deltas automatisch über E-Mail zu beziehen. Die Größe der Updates wird so klein wie möglich gehalten. Normalerweise sind sie kleiner als 5 kB, manchmal sind sie 10-50 kB groß (etwa jedes 10. Update) und ab und an werden Sie auch einmal ein Update mit 100 kB oder mehr erhalten.

Sie sollten sich über die Vorbehalte gegen die Verwendung der Quellen anstelle eines offiziellen Releases bewusst sein. Das trifft besonders auf FreeBSD-CURRENT zu, lesen Sie dazu bitte den Abschnitt FreeBSD-CURRENT.

A.4.2. Was brauche ich, um CTM zu benutzen?

Zwei Sachen: Das CTM Programm und die initialen Deltas, von denen aus Sie auf die „aktuellen“ Stände kommen.

CTM ist schon seit der Version 2.0 Teil des FreeBSD-Basissystems. Sie finden es in `/usr/src/usr.sbin/ctm` , wenn Sie eine Kopie der Quellen besitzen.

Die Deltas, die CTM verarbeitet, können Sie über FTP oder E-Mail beziehen. Wenn Sie über einen FTP Zugang zum Internet verfügen, erhalten Sie die Deltas unter der folgenden URL:

```
ftp://ftp.FreeBSD.org/pub/FreeBSD/CTM/
```

Die Deltas werden auch von CTM Spiegeln bereitgehalten.

Wechseln Sie in das passende Verzeichnisse zum Beispiel `src-cur` für FreeBSD-CURRENT und laden Sie sich von dort die Deltas herunter.

Sie können die Deltas auch über E-Mail beziehen.

[1] Abkürzung für „CVS Through eMail“

Abonnieren Sie dazu eine der CTM-Verteilerlisten. Über ctm-src-cur erhalten Sie den kompletten Subversion-Baum, über ctm-src-cur erhalten Sie FreeBSD-CURRENT und über ctm-src-9 erhalten Sie den FreeBSD 9.X-Zweig. Wenn Sie nicht wissen, wie Sie eine der Mailinglisten abonnieren, folgen Sie einem der Verweise von oben oder besuchen Sie die Seite http://lists.FreeBSD.org/mailman/listinfo. Weitere Informationen erhalten Sie, wenn Sie dort auf die gewünschte Liste klicken.

Benutzen Sie `ctm_rmail`, um die CTM Updates, die Sie per E-Mail empfangen, auszupacken und anzuwenden. Wenn Sie diesen Prozess automatisiert ablaufen lassen möchten, können Sie dazu einen Eintrag in `/etc/aliases` verwenden. Genauere Informationen finden Sie in der Manualpage von `ctm_rmail`.

Anmerkung

Sie sollten die Mailingliste ctm-announce abonnieren, egal wie Sie die CTM-Deltas erhalten. Ankündigungen, die den Betrieb des CTM-Systems betreffen, werden nur auf dieser Liste bekannt gegeben. Klicken Sie auf den Namen der Liste oder besuchen Sie die Seite http://lists.FreeBSD.org/mailman/listinfo, um diese Liste zu abonnieren.

A.4.3. Initialisieren von CTM

Bevor Sie die CTM Deltas benutzen können, brauchen Sie einen Startpunkt, auf den die nachfolgenden Deltas angewendet werden.

Sie können natürlich mit einem leeren Verzeichnis beginnen. In diesem Fall benötigen Sie ein `XEmpty`-Delta, mit dem Sie den CTM-Verzeichnisbaum initialisieren. Wenn Sie Glück haben, finden Sie ein `XEmpty`-Delta, mit dem sie beginnen können, auf einer der CDs Ihrer Distribution.

Da die Verzeichnisbäume mehrere Megabyte groß sind, sollten Sie nach Möglichkeit etwas schon vorhandenes benutzen. Wenn Sie eine -RELEASE CD besitzen, können Sie die Quellen von dieser CD benutzen. Sie ersparen sich damit das Übertragen großer Datenmengen.

Die Deltas, mit denen Sie beginnen können, enthalten ein `X` in ihrem Namen, wie in `src-cur.3210XEmpty.gz`. Hinter dem `X` wird der Startpunkt der Deltas angegeben, in diesem Fall steht `Empty` für ein leeres Verzeichnis. Nach etwa 100 Deltas wird ein neues `XEmpty`-Delta erstellt. Mit ungefähr 75 Megabyte komprimierter Daten sind diese `XEmpty`-Deltas übrigens sehr groß.

Nachdem Sie Ihren Startpunkt festgelegt haben, benötigen Sie alle Deltas mit einer höheren Nummer.

A.4.4. Benutzen von CTM

Um ein Delta einzuspielen, benutzen Sie das folgende Kommando:

```
# cd /Pfad/zu/den/Quellen
# ctm -v -v /Pfad/zu/den/Deltas/src-xxx.*
```

CTM kann mit Deltas arbeiten, die mit `gzip` komprimiert wurden. Sie brauchen die Deltas vorher nicht mit `gunzip` zu dekomprimieren und sparen damit Plattenplatz.

Ihr Quellbaum wird erst dann verändert, wenn CTM die Deltas sauber verarbeiten kann. Die Integrität der Deltas und ihre Anwendbarkeit auf den Quellbaum lassen sich durch die Angabe des Schalters `-c` überprüfen, CTM ändert in diesem Fall Ihren Quellbaum nicht.

CTM verfügt über weitere Kommandozeilenoptionen, Informationen dazu finden Sie in der Manualpage oder dem Quellcode.

Das war schon alles. Um Ihre Quellen aktuell zu halten, verwenden Sie CTM jedes Mal, wenn Sie neue Deltas bekommen.

Löschen Sie die Deltas nicht, wenn Sie diese nur schwer wieder beschaffen können. Behalten Sie sie für den Fall, das etwas passiert. Auch wenn Sie nur Disketten besitzen, sollten Sie erwägen, die Deltas mit `fdwrite` zu sichern.

A.4.5. Umgang mit lokalen Änderungen

Entwickler wollen mit den Dateien im Quellbaum experimentieren und diese verändern. In beschränkter Weise werden lokale Änderungen von CTM unterstützt. Wenn CTM die Datei `foo` bearbeiten will, überprüft es zuerst ob die Datei `foo.ctm` existiert. Wenn diese Datei existiert, werden Änderungen in ihr anstatt in `foo` vorgenommen.

Mit diesem Verfahren ist eine leichte Handhabung lokaler Änderungen möglich. Kopieren Sie die Dateien, die Sie ändern möchten, in Dateien, die das Suffix `.ctm` tragen. Sie können dann ungestört mit dem Quellcode arbeiten, während `CTM` die `.ctm` Dateien aktualisiert.

A.4.6. Weitere CTM-Optionen

A.4.6.1. Was wird aktualisiert?

Eine Liste der Änderungen, die CTM an Ihrem Quellbaum vornehmen wird, erhalten Sie, wenn Sie die Option `-l` angeben.

Das ist nützlich, wenn Sie Logs über die Änderungen führen wollen, geänderte Dateien vor- oder nachbearbeiten wollen, oder einfach ein bisschen paranoid sind.

A.4.6.2. Sicherungen vor einer Aktualisierung erstellen

Sie wollen vielleicht die Dateien, die durch eine CTM Aktualisierung verändert werden, sichern.

Mit `-B` *`backup-file`* weisen Sie CTM an, alle Dateien, die durch ein CTM Delta verändert würden, nach `backup-file` zu sichern.

A.4.6.3. Dateien ausschließen

Manchmal wollen Sie nur bestimmte Teile aktualisieren oder nur bestimmte Dateien aus einer Folge von Deltas extrahieren.

Sie können die Liste der Dateien, mit denen CTM arbeitet, einschränken, indem Sie reguläre Ausdrücke mit den Optionen `-e` und `-x` angeben.

Wenn Sie eine aktuelle Kopie von `lib/libc/Makefile` aus den gesicherten CTM Deltas erhalten wollen, setzen Sie das folgende Kommando ab:

```
# cd /wo/Sie/es/auspacken/wollen/
# ctm -e '^lib/libc/Makefile' ~ctm/src-xxx.*
```

Die Optionen `-e` und `-x` werden in der Reihenfolge angewandt, in der sie auf der Kommandozeile angegeben wurden. Eine Datei wird nur dann von CTM verarbeitet, wenn dies nach der Anwendung der Optionen `-e` und `-x` noch erlaubt ist.

A.4.7. Pläne für CTM

Mehrere:

- Hinzufügen eines Authentifizierungsmechanismus, damit gefälschte CTM-Deltas erkannt werden können.
- Aufräumen der CTM-Optionen, die mit der Zeit unübersichtlich und irreführend wurden.

A.4.8. Verschiedenes

Es gibt Deltas für die Ports-Sammlung, die aber nicht intensiv genutzt werden.

A.4.9. CTM-Spiegel

Die CTM-Deltas können Sie mit anonymous FTP von den folgenden Spiegeln beziehen. Versuchen Sie bitte einen Spiegel in Ihrer Nähe zu benutzen.

Bei Problemen wenden Sie sich bitte an die Mailingliste ctm-users.

Kalifornien, Bay Area, Offizieller Server

- `ftp://ftp.FreeBSD.org/pub/FreeBSD/development/CTM/`

Südafrika, Backup-Server für alte Deltas

- `ftp://ftp.za.FreeBSD.org/pub/FreeBSD/CTM/`

Taiwan/R.O.C.

- `ftp://ctm.tw.FreeBSD.org/pub/FreeBSD/development/CTM/`
- `ftp://ctm2.tw.FreeBSD.org/pub/FreeBSD/development/CTM/`
- `ftp://ctm3.tw.FreeBSD.org/pub/FreeBSD/development/CTM/`

Wenn die Liste keinen Spiegel in Ihrer Nähe enthält oder Sie Probleme mit dem ausgewählten Spiegel haben, versuchen Sie einen Spiegel mit einer Suchmaschine, wie alltheweb, zu finden.

A.5. Benutzen von Subversion

A.5.1. Einführung

Seit Juli 2012 nutzt FreeBSD Subversion (*svn*) als primäres Versionskontrollsystem zur Speicherung des gesamten FreeBSD Quellcodes, der Dokumentation und der Ports-Sammlung.

Anmerkung

Subversion ist hauptsächlich ein Werkzeug für Entwickler. Die meisten Benutzer sollten FreeBSD Update benutzen um ihr FreeBSD zu aktualisieren, und Portsnap um ihre Ports-Sammlung aktuell zu halten.

In Subversion werden URLs in der Form von *`protocol://hostname/path`* verwendet, um ein Repository zu kennzeichnen. Die Spiegel können, wie unten angegeben, verschiedene Protokolle unterstützen. Die erste Komponente des Pfades ist das FreeBSD Repository auf welches zugegriffen wird. Es gibt drei verschiedene Repositories. `base` für den Quellcode des FreeBSD Basissystems, `ports` für die Ports-Sammlung und `doc` für die Dokumentation. Als Beispiel spezifiziert die URL `svn://svn0.us-east.FreeBSD.org/ports/head/` den Hauptzweig des Port-Repositories auf dem Mirror `svn0.us-east.FreeBSD.org` , über das `svn`-Protokoll.

A.5.2. Installation

Subversion muss installiert werden, bevor Sie damit die Inhalte eines der Repositories auschecken können. Wenn eine Kopie der Ports-Sammlung bereits vorhanden ist, kann Subversion wie folgt installiert werden:

```
# cd /usr/ports/devel/subversion
#make install clean
```

Ist die Ports-Sammlung nicht vorhanden, kann Subversion als Paket installiert werden:

```
# pkg_add -r subversion
```

Wenn Sie pkgng verwenden, um Pakete zu verwalten, können Sie Subversion stattdessen so installieren:

```
# pkg install devel/subversion
```

A.5.3. Ausführen von Subversion

Der `svn` Befehl wird verwendet, eine Kopie der Quellen in ein lokales Verzeichnis zu holen. Die Dateien in diesem Verzeichnis werden *lokale Arbeitskopie* genannt.

Warnung

Wenn das lokale Verzeichnis bereits vorhanden ist, aber nicht von `svn` erstellt wurde, benennen Sie das Verzeichnis um oder löschen Sie es, bevor Sie Inhalte auschecken. In ein bestehendes nicht-`svn` Verzeichnis auszuchecken kann zu Konflikten zwischen den vorhandenen Dateien und denen aus dem Respository führen.

Das Auschecken aus einem bestimmten Repository kann wie folgt durchgeführt werden:

```
# svn checkout  svn-mirror /repository /branch  lcwdir
```

wobei:

- *`svn-mirror`* eine URL für einen Mirror aus Subversion Mirror Sites ist.
- *`repository`* eines der Projekt-Repositories ist, z. B. `base`, `ports` oder `doc`.
- *`branch`* vom verwendeten Repository abhängt. `ports` und `doc` werden meist im `head` Zweig aktualisiert, während `base` die neueste Version von -CURRENT unter `head` und die jeweilige neueste Version des -STABLE Zweiges unter `stable/8` (für 8.*x*), `stable/9` (9.*x*) und `stable/10` (10.*x*) verwaltet wird.
- *`lcwdir`* das Zielverzeichnis ist, in dem die Inhalte des angegebenen Zweiges plaziert werden sollen. Dies ist üblicherweise `/usr/ports` für `ports`, `/usr/src` für `base`, und `/usr/doc` für `doc`.

Dieses Beispiel checkt die Ports-Sammlung aus dem Repositroy im Westen der USA über das HTTPS Protokoll aus, und speichert die Arbeitskopie unter `/usr/ports` . Wenn `/usr/ports` bereits vorhanden ist, aber nicht von `svn` erstellt wurde, denken Sie vor dem Auschecken daran, das Verzeichnis umzubenennen oder zu löschen.

```
# svn checkout  https://svn0.us-west.FreeBSD.org  /ports/head /usr/ports
```

Dies kann eine Weile dauern, da beim ersten Auschecken der komplette Zweig vom entfernten Repository heruntergeladen werden muss. Bitte haben Sie Geduld.

Nach dem ersten Auschecken können Sie Ihre lokale Arbeitskopie wie folgt aktualisieren:

```
# svn update  lcwdir
```

Um `/usr/ports` aus dem oben erstellten Beispiel zu aktualisieren, benutzen Sie:

```
# svn update /usr/ports
```

Das Update ist viel schneller als ein Auschecken, da nur die Dateien übertragen werden müssen, die sich auch geändert haben.

Eine alternative Möglichkeit zur Aktualisierung Ihrer Arbeitskopie nach dem Auschecken ist es, das bestehende `Makefile` in den Verzeichnissen `/usr/ports` , `/usr/src` , und `/usr/doc` zu nutzen. Setzen Sie dazu `SVN_UPDATE` und benutzen Sie das `update` Ziel. Zum Beispiel, um `/usr/src` zu aktualisieren:

```
# cd /usr/src
# make update SVN_UPDATE=yes
```

A.5.4. Weiterführende Informationen

Weitere Informationen über die Verwendung von Subversion finden Sie im „Subversion Buch“ mit dem Namen Versionskontrolle mit Subversion, oder in der Subversion Dokumentation.

A.6. Subversion Mirror Sites

Alle Spiegel führen alle Repositories.

Der Master FreeBSD Subversion Server, `svn.FreeBSD.org` ist öffentlich zugänglich. Auf ihn kann allerdings nur lesend zugegriffen werden. Dies kann sich in Zukunft ändern, solange jedoch werden die Nutzer dazu aufgefordert, einen der offiziellen Spiegel zu verwenden. Um das FreeBSD Subversion Repository über einen Browser anzuzeigen, verwenden Sie http://svnweb.FreeBSD.org/.

Anmerkung

Das FreeBSD svn Mirror Netzwerk befindet sich noch in den Anfängen, und Veränderungen werden stattfinden. Verlassen Sie sich also nicht darauf, dass diese Liste statisch ist. Insbesondere werden sich die SSL-Zertifikate irgendwann ändern.

Name	Protokolle	Standort	SSL-Fingerabdruck
`svn.us-west.FreeBSD.org`	svn, http, https	USA, Kalifornien	SHA1 `1C:BD:85:95:11:9F:EB:75:A5:4B:C8:A3:FE:08:E4:02:73:06:1E:61`
`svn0.us-east.FreeBSD.org`	svn, http, https, rsync	USA, New Jersey	SHA1 `1C:BD:85:95:11:9F:EB:75:A5:4B:C8:A3:FE:08:E4:02:73:06:1E:61`
`svn0.eu.FreeBSD.org`	svn, http https, rsync	Europa, UK	SHA1 `39:B0:53:35:CE:60:C7:BB:00:54:96:96:71:10:94:BB:CE:1C:07:A7`

HTTPS ist das bevorzugte Protokoll, es schützt Sie vor anderen Computern, die vortäuschen, der FreeBSD-Mirror zu sein (gemeinhin bekannt als „man in the middle-Angriff“), oder anderweitig versuchen schlechte Daten an den Endnutzer zu senden.

Bei der ersten Verbindung zu einem HTTPS Mirror, wird der Benutzer aufgefordert, den Fingerabdruck des Servers zu überprüfen:

```
Error validating server certificate for 'https://svn0.us-west.freebsd.org:443':
 - The certificate is not issued by a trusted authority. Use the
   fingerprint to validate the certificate manually!
 - The certificate hostname does not match.
Certificate information:
 - Hostname: svnmir.ysv.FreeBSD.org
 - Valid: from Jul 29 22:01:21 2013 GMT until Dec 13 22:01:21 2040 GMT
```

```
 - Issuer: clusteradm, FreeBSD.org, (null), CA, US (clusteradm@FreeBSD.org)
 - Fingerprint: 1C:BD:85:95:11:9F:EB:75:A5:4B:C8:A3:FE:08:E4:02:73:06:1E:61
(R)eject, accept (t)emporarily or accept (p)ermanently?
```

Vergleichen Sie den Fingerabdruck mit dem in der obigen Tabelle. Wenn der Fingerabdruck übereinstimmt, kann das Sicherheitszertifikat des Server zeitweise oder dauerhaft akzeptiert werden. Ein temporäres Zertifikat wird nach einer einzigen Sitzung mit dem Server ablaufen, und die Überprüfung wird bei der nächsten Verbindung wiederholt werden. Akzeptieren Sie das Zertifikat dauerhaft, werden die Authentifizierungsinformationen in `~/.subversion/auth` gespeichert, und der Benutzer wird nicht wieder gefragt den Fingerabdruck zu prüfen, solange bis das Zertifikat abgelaufen ist.

Wenn HTTPS aufgrund von Firewall- oder anderen Problemen nicht verwendet werden kann, dann ist `SVN` die nächste Wahl. Sollte beides nicht verfügbar sein, nutzen Sie HTTP

A.7. Benutzen von CVSup (veraltet)

A.7.1. Einführung

CVSup ist eine Anwendung, die Verzeichnisbäume von einem entfernten CVS-Server bereitstellt und aktualisiert. Die Quellen von FreeBSD werden in einem CVS-Repository auf einer Entwicklungsmaschine in Kalifornien gepflegt. Mit CVSup können sich FreeBSD-Benutzer den eigenen Quellbaum auf aktuellem Stand halten.

Zum Aktualisieren benutzt CVSup die Pull-Methode, bei der die Aktualisierungen vom Client angefragt werden. Der Server wartet dabei passiv auf Anfragen von Clients, das heißt er verschickt nicht unaufgefordert Aktualisierungen. Somit gehen alle Anfragen vom Client aus und die Benutzer müssen CVSup entweder manuell starten oder einen `cron` Job einrichten, um regelmäßig Aktualisierungen zu erhalten.

CVSup in genau dieser Schreibweise bezeichnet die Anwendung, die aus dem Client `cvsup` und dem Server `cvsupd` besteht. `cvsup` läuft auf den Maschinen der Benutzer, `cvsupd` läuft auf jedem der FreeBSD-Spiegel.

Anmerkung

Mit csup gibt es in inzwischen auch eine in C geschriebene Neuimplementierung von CVSup. Der größte Vorteil dieser neuen Version ist neben einer höheren Geschwindigkeit der, dass dieses Programm nicht von der Sprache Modula-3 abhängig ist und Sie daher dieses Paket nicht mitinstallieren müssen. csup ist bereits im Basissystem enthalten und kann sofort verwendet werden. Wollen Sie künftig csup einsetzen, überspringen Sie in den folgenden Ausführungen einfach den Abschnitt zur Installation von CVSup und ersetzen alle Vorkommen von CVSup durch csup.

A.7.2. Installation von CVSup

CVSup können Sie leicht installieren, wenn Sie das vorkompilierte Paket net/cvsup aus der Ports-Sammlung benutzen. Alternativ können Sie net/cvsup auch ausgehend von den Quellen bauen, doch seien Sie gewarnt: net/cvsup hängt vom Modula-3 System ab, das viel Zeit und Platz zum Herunterladen und Bauen braucht.

Anmerkung

Wenn Sie CVSup auf einer Maschine ohne Xorg (also beispielsweise auf einem Server), benutzen, stellen Sie bitte sicher, dass Sie den Port ohne das CVSup-GUI, (net/cvsup-without-gui) verwenden.

A.7.3. Konfiguration von CVSup

Das Verhalten von CVSup wird mit einer Konfigurationsdatei gesteuert, die `supfile` genannt wird. Beispiele für Konfigurationsdateien finden Sie in dem Verzeichnis file://localhost/usr/share/examples/cvsup/.

Ein `supfile` enthält die folgenden Informationen:

- Welche Dateien Sie erhalten wollen.
- Welche Versionen der Dateien Sie benötigen.
- Woher Sie die Dateien beziehen wollen.
- Wo Sie die erhaltenen Dateien speichern.
- Wo Sie die Status-Dateien aufbewahren wollen.

In den folgenden Abschnitten erstellen wir ein typisches `supfile` indem wir nach und nach diese Punkte klären. Zuerst beschreiben wir aber den Aufbau dieser Konfigurationsdatei.

Ein `supfile` ist eine Textdatei. Kommentare beginnen mit einem # und gelten bis zum Zeilenende. Leerzeilen und Zeilen, die nur Kommentare enthalten, werden ignoriert.

Die anderen Zeilen legen die Dateien fest, die ein Benutzer erhalten will. Der Server organisiert verschiedene Dateien in einer „Sammlung“, deren Name auf einer Zeile angegeben wird. Nach dem Namen der Sammlung können mehrere durch Leerzeichen getrennte Felder folgen, die die oben angesprochenen Informationen festlegen. Es gibt zwei Arten von Feldern: Felder, die Optionen festlegen und Felder mit Parametern. Optionen bestehen aus einem Schlüsselwort, wie `delete` oder `compress` und stehen alleine. Ein Parameterfeld beginnt mit einem Schlüsselwort, dem = und ein Parameter, wie in `release=cvs` , folgt. Dieses Feld darf keine Leerzeichen enthalten.

In einem `supfile` werden normalerweise mehrere Sammlungen angefordert. Die erforderlichen Felder können explizit für jede Sammlung angegeben werden, dann werden jedoch die Zeilen ziemlich lang. Außerdem ist dieses Vorgehen sehr unhandlich, da die meisten Felder für alle Sammlungen gleich sind. CVSup bietet die Möglichkeit, Vorgaben für die Felder der Sammlungen festzulegen. Zeilen, die mit der Pseudo-Sammlung `*default` beginnen, legen Optionen und Parameter für nachfolgende Sammlungen im `supfile` fest. Der Vorgabewert kann in der Zeile einer bestimmten Sammlung überschrieben werden. Durch Hinzufügen weiterer `*default` Zeilen können die Vorgaben auch mitten im `supfile` überschrieben oder erweitert werden.

Mit diesem Wissen können wir nun ein `supfile` erstellen, das den Quellbaum von FreeBSD-CURRENT anfordert und aktualisiert.

- Welche Dateien wollen Sie empfangen?

  Dateien werden von CVSup in „Sammlungen“ organisiert. Die erhältlichen Sammlungen werden später beschrieben. Wir wollen den Quellbaum von FreeBSD empfangen, der in der Sammlung `src-all` enthalten ist. Das `supfile` enthält pro Zeile eine Sammlung, in diesem Fall also nur eine einzige Zeile:

  ```
  src-all
  ```

- Welche Versionen der Dateien werden benötigt?

  Mit CVSup können Sie jede Version der Quellen bekommen, da der cvsupd-Server seine Daten direkt aus dem CVS-Repository bezieht. Sie können die benötigten Versionen in den Parameterfeldern `tag=` und `date=` angeben.

Warnung

Achten Sie darauf, dass Sie das richtige `tag=`-Feld angeben. Einige Tags sind nur für spezielle Sammlungen gültig. Wenn Sie ein falsches Tag angeben oder sich verschreiben, wird

CVSup Dateien löschen, die Sie wahrscheinlich gar nicht löschen wollten. Achten Sie insbesondere bei den `ports-*` -Sammlungen darauf, *ausschließlich* `tag=.` zu verwenden.

Mit `tag=` wird ein symbolischer Name aus dem Repository angegeben. Es gibt zwei verschiedene Tags: Tags, die Revisionen bezeichnen und Tags, die Zweige bezeichnen. Die ersteren sind statisch und fest an eine Revision gebunden. Ein Tag, das einen Zweig bezeichnet, bezieht sich dagegen zu einem gegebenen Zeitpunkt immer auf die aktuellste Revision. Da ein Tag eines Zweiges nicht an eine bestimmte Revision gebunden ist, kann sich dessen Bedeutung von heute auf morgen ändern.

Abschnitt A.8, „CVS-Tags“ zählt für Benutzer relevante Tags auf. Wenn Sie in der Konfigurationsdatei ein Tag, wie `RELENG_8` , angeben, müssen Sie diesem `tag=` vorstellen: `tag=RELENG_8`. Denken Sie daran, dass es für die Ports-Sammlung nur `tag=.` gibt.

Warnung

Achten Sie darauf, dass Sie den Namen eines Tags richtig angeben. CVSup kann nicht zwischen richtigen und falschen Tags unterscheiden. Wenn Sie sich bei der Angabe eines Tags vertippen, nimmt CVSup an, Sie hätten ein gültiges Tag angegeben, dem nur keine Dateien zugeordnet sind. Die Folge davon ist, dass Ihre vorhandenen Quellen gelöscht werden.

Wenn Sie ein Tag angeben, das sich auf einen Zweig bezieht, erhalten Sie die aktuellsten Revisionen der Dateien auf diesem Zweig. Wenn Sie eine frühere Revision erhalten möchten, können Sie diese im `date=` Feld angeben. Einzelheiten dazu finden Sie in der Manualpage von `cvsup`.

Wir möchten gerne FreeBSD-CURRENT beziehen und fügen die folgende Zeile *am Anfang* der Konfigurationsdatei ein:

```
*default tag=.
```

Eine wichtige Ausnahme ist wenn Sie weder ein `tag=`-Feld noch ein `date=`-Feld angeben. In diesem Fall erhalten Sie anstelle einer speziellen Revision die wirklichen RCS-Dateien aus dem CVS-Repository des Servers. Diese Vorgehensweise wird von Entwicklern bevorzugt, da sie mit einem eigenen Repository leicht die Entwicklungsgeschichte und Veränderungen von Dateien verfolgen können. Dieser Vorteil muss allerdings mit sehr viel Plattenplatz bezahlt werden.

- Woher sollen die Dateien bezogen werden?

  Im `host=`-Feld wird angegeben, woher `cvsup` die Dateien holen soll. Sie können hier jeden der CVSup-Spiegel angeben, doch sollten Sie einen Server in Ihrer Nähe auswählen. Für dieses Beispiel wollen wir den erfundenen Server `cvsup99.FreeBSD.org` verwenden:

  ```
  *default host=cvsup99.FreeBSD.org
  ```

  Bevor Sie CVSup laufen lassen, sollten Sie hier einen existierenden Server einsetzen. Den zu verwendenden Server können Sie auf der Kommandozeile mit `-h` *`hostname`* überschreiben.

- Wo sollen die Dateien gespeichert werden?

  Im `prefix=`-Feld teilen Sie `cvsup` mit, wo die Dateien gespeichert werden sollen. In diesem Beispiel werden wir die Quelldateien direkt im Verzeichnisbaum für Quellen `/usr/src` ablegen. Das Verzeichnis `src` ist schon in der Sammlung, die wir beziehen enthalten, so dass wir die folgende Zeile angeben:

  ```
  *default prefix=/usr
  ```

- Wo sollen die Statusinformationen von `cvsup` gespeichert werden?

`cvsup` legt in einem Verzeichnis Statusinformationen ab, die festhalten, welche Versionen schon empfangen wurden. Wir verwenden das Verzeichnis `/var/db`:

```
*default base=/var/db
```

Wenn das Verzeichnis für die Statusinformationen nicht existiert, sollten Sie es jetzt anlegen, da `cvsup` ohne dieses Verzeichnis nicht startet.

- Verschiedene Einstellungen:

  Eine weitere Zeile sollte normalerweise in jedem `supfile` sein:

  ```
  *default release=cvs delete use-rel-suffix compress
  ```

  Mit `release=cvs` wird angegeben, dass der Server das FreeBSD-Haupt-Repository abfragen soll, was praktisch immer der Fall ist (die Ausnahmen werden in diesem Text nicht diskutiert).

  `delete` erlaubt es CVSup, Dateien zu löschen. Diese Option sollten Sie immer angeben, damit CVSup Ihren Quellbaum auch wirklich aktuell halten kann. CVSup löscht nur Dateien für die es auch verantwortlich ist. Andere Dateien, die sich in einem Baum unter Kontrolle von CVSup befinden, werden nicht verändert.

  Wenn Sie wirklich etwas über das obskure `use-rel-suffix` erfahren wollen, lesen Sie bitte in der Manualpage nach, ansonsten geben Sie es einfach an und vergessen es.

  Wenn Sie `compress` angeben, werden Daten auf dem Kommunikationskanal komprimiert. Wenn Sie über eine T1-Leitung oder eine schnellere Netzanbindung verfügen, brauchen Sie diese Option vielleicht nicht. In allen anderen Fällen beschleunigt sie aber den Ablauf.

- Zusammenfassung:

  Das vollständige `supfile` unseres Beispiels sieht nun so aus:

  ```
  *default tag=.
  *default host=cvsup99.FreeBSD.org
  *default prefix=/usr
  *default base=/var/db
  *default release=cvs delete use-rel-suffix compress

  src-all
  ```

A.7.3.1. Die `refuse` Datei

CVSup benutzt die Pull-Methode, das heißt wenn sich ein Client mit einem Server verbindet, erhält er eine Liste der verfügbaren Sammlungen und wählt aus diesen die herunterzuladenden Dateien aus. In der Voreinstellung wählt der Client alle Dateien aus, die zu einer gegebenen Sammlung und zu einem gegebenen Tag passen. Um nur einen Teil des Baumes herunterzuladen, benutzen Sie die `refuse` Datei.

Mit einer `refuse` Datei können Sie bestimmte Dateien einer Sammlung von der Übertragung ausschließen. Der Ort der `refuse` ist `base/sup/refuse` , wobei *`base`* in Ihrem `supfile` festgelegt wurde. Wir verwenden das Verzeichnis `/var/db` , der Ort der `refuse` Datei ist daher `/var/db/sup/refuse` .

Das Format der `refuse` Datei ist einfach: Sie enthält eine Liste der Dateien und Verzeichnisse, die Sie nicht herunterladen wollen. Zum Beispiel:

```
bin/
   usr.bin/
```

Die `refuse` Datei spart Anwendern von CVSup, die über eine langsame Internetanbindung verfügen oder deren Internetverbindung zeitlich abgerechnet wird, Zeit, da sie Dateien, die sie nicht benötigen, nicht mehr herunterladen müssen. Weitere Informationen zu `refuse` Dateien und anderen Eigenschaften von CVSup entnehmen Sie bitte der Manualpage.

A.7.4. Ausführen von CVSup

Wir können nun eine Aktualisierung mit der folgenden Kommandozeile starten:

```
# cvsup supfile
```

`supfile` gibt dabei das eben erstelle `supfile` an. Wenn Sie X11 benutzen, wird `cvsup` ein GUI starten. Drücken Sie go und schauen Sie zu.

Das Beispiel aktualisiert die Dateien im Verzeichnisbaum `/usr/src`. Sie müssen `cvsup` als `root` starten, damit Sie die nötigen Rechte haben, die Dateien zu aktualisieren. Sie sind vielleicht ein bisschen nervös weil Sie das Programm zum ersten Mal anwenden und möchten zuerst einmal einen Testlauf durchführen. Legen Sie dazu ein temporäres Verzeichnis an und übergeben es auf der Kommandozeile von `cvsup`:

```
# mkdir /var/tmp/dest
# cvsup supfile /var/tmp/dest
```

Aktualisierungen werden dann nur in dem angegebenen Verzeichnis vorgenommen. CVSup untersucht die Dateien in `/usr/src`, wird aber keine dieser Dateien verändern. Die veränderten Dateien finden Sie stattdessen in `/var/tmp/dest/usr/src`. Die Statusdateien von CVSup werden ebenfalls nicht geändert, sondern in dem angegebenen Verzeichnis abgelegt. Wenn Sie Leseberechtigung in `/usr/src` haben, brauchen Sie das Programm noch nicht einmal unter `root` laufen zu lassen.

Wenn Sie X11 nicht benutzen wollen oder keine GUIs mögen, sollten Sie `cvsup` wie folgt aufrufen:

```
# cvsup -g -L 2 supfile
```

`-g` verhindert den Start des GUIs. Wenn Sie kein X11 laufen haben, passiert das automatisch, ansonsten müssen Sie diesen Schalter angeben.

Mit `-L 2` gibt CVSup Einzelheiten zu jeder Aktualisierung aus. Die Wortfülle der Meldungen können Sie von `-L 0` bis `-L 2` einstellen. In der Voreinstellung `-L 0` werden nur Fehlermeldungen ausgegeben.

Eine Zusammenfassung der Optionen von CVSup erhalten Sie mit `cvsup -H`. Genauere Informationen finden Sie in der Manualpage von CVSup.

Wenn Sie mit dem Ablauf der Aktualisierung zufrieden sind, können Sie CVSup regelmäßig aus cron(8) ausführen. In diesem Fall sollten Sie natürlich nicht das GUI benutzen.

A.7.5. CVSup Sammlungen

Die CVSup Sammlungen sind hierarchisch organisiert. Es gibt wenige große Sammlungen, die in kleinere Teilsammlungen unterteilt sind. Wenn Sie eine große Sammlung beziehen, entspricht das dem Beziehen aller Teilsammlungen. Der Hierarchie der Sammlung wird in der folgenden Aufzählung durch Einrückungen dargestellt.

Die am häufigsten benutzte Sammlung ist `src-all`.

`cvs-all release=cvs`
: Das FreeBSD-Haupt-Repository einschließlich der Kryptographie-Module.

  `distrib release=cvs`
  : Dateien, die zum Verteilen und Spiegeln von FreeBSD benötigt werden.

  `projects-all release=cvs`
  : Quelltexte der verschiedenen FreeBSD-Projekte.

  `src-all release=cvs`
  : Die FreeBSD-Quellen einschließlich der Kryptographie-Module.

    `src-base release=cvs`
    : Verschiedene Dateien unter `/usr/src`.

`src-bin release=cvs`
: Benutzer-Werkzeuge die im Einzelbenutzermodus gebraucht werden (`/usr/src/bin`).

`src-cddl release=cvs`
: Werkzeuge und Bibliotheken, die der CDDL-Lizenz unterliegen (`/usr/src/cddl`).

`src-contrib release=cvs`
: Werkzeuge und Bibliotheken, die nicht aus dem FreeBSD Project stammen und wenig verändert übernommen werden. (`/usr/src/contrib`).

`src-crypto release=cvs`
: Kryptographische Werkzeuge und Bibliotheken, die nicht aus dem FreeBSD Project stammen und wenig verändert übernommen werden. (`/usr/src/crypto`).

`src-eBones release=cvs`
: Kerberos und DES (`/usr/src/eBones`). Wird in aktuellen Releases von FreeBSD nicht benutzt.

`src-etc release=cvs`
: Konfigurationsdateien des Systems (`/usr/src/etc`).

`src-games release=cvs`
: Spiele (`/usr/src/games`).

`src-gnu release=cvs`
: Werkzeuge, die unter der GNU Public License stehen (`/usr/src/gnu`).

`src-include release=cvs`
: Header Dateien (`/usr/src/include`).

`src-kerberos5 release=cvs`
: Kerberos5 (`/usr/src/kerberos5`).

`src-kerberosIV release=cvs`
: KerberosIV (`/usr/src/kerberosIV`).

`src-lib release=cvs`
: Bibliotheken (`/usr/src/lib`).

`src-libexec release=cvs`
: Systemprogramme, die von anderen Programmen ausgeführt werden (`/usr/src/libexec`).

`src-release release=cvs`
: Dateien, die zum Erstellen eines FreeBSD Releases notwendig sind (`/usr/src/release`).

`src-rescue release=cvs`
: Statisch gelinkte Programme zur Wiederherstellung eines defekten Systems. Lesen Sie dazu auch die Manualpage rescue(8) (`/usr/src/rescue`).

`src-sbin release=cvs`
: Werkzeuge für den Einzelbenutzermodus (`/usr/src/sbin`).

`src-secure release=cvs`
: Kryptographische Bibliotheken und Befehle (`/usr/src/secure`).

`src-share release=cvs`
: Dateien, die von mehreren Systemen gemeinsam benutzt werden können (`/usr/src/share`).

`src-sys release=cvs`
: Der Kernel (`/usr/src/sys`).

`src-sys-crypto release=cvs`
: Kryptographie Quellen des Kernels (`/usr/src/sys/crypto`).

`src-tools release=cvs`
: Verschiedene Werkzeuge zur Pflege von FreeBSD (`/usr/src/tools`).

`src-usrbin release=cvs`
: Benutzer-Werkzeuge (`/usr/src/usr.bin`).

`src-usrsbin release=cvs`
: System-Werkzeuge (`/usr/src/usr.sbin`).

`distrib release=self`
: Die Konfigurationsdateien des CVSup Servers. Diese werden von den CVSup benutzt.

`gnats release=current`
: Die GNATS Datenbank, in der Problemberichte verwaltet werden.

`mail-archive release=current`
: Das Archiv der FreeBSD-Mailinglisten.

A.7.6. Weiterführende Informationen

Die CVSup FAQ und weitere Informationen über CVSup finden Sie auf The CVSup Home Page.

FreeBSD spezifische Diskussionen über CVSup finden auf der Mailingliste FreeBSD technical discussions statt. Dort und auf der Liste FreeBSD announcements werden neue Versionen von CVSup angekündigt.

Bei Fragen und Problemberichten zu CVSup lesen Sie bitte die CVSup FAQ.

A.7.7. CVSup-Server

Die folgende Aufzählung enthält CVSup Server für FreeBSD:

Hauptserver,

(aktualisiert am: UTC)

Hauptserver

- cvsup.FreeBSD.org

A.8. CVS-Tags

Wenn Sie Quellen mit CVS oder CVSup erhalten oder aktualisieren wollen, müssen Sie ein Tag angeben. Ein Tag kann einen bestimmten FreeBSD-Zweig oder einen bestimmten Zeitpunkt (Release-Tag) bestimmen.

A.8.1. Tags für Zweige

Mit Ausnahme von `HEAD` (das immer ein gültiges Tag ist), können die folgenden Tags nur im `src/`-Quellbaum verwendet werden. Die Quellbäume `ports/` , `doc/` und `www/` sind nicht verzweigt.

HEAD
: Symbolischer Name für den Hauptzweig, auch FreeBSD-CURRENT genannt. Dies ist die Vorgabe, wenn keine Revision angegeben wird.

  In CVSup wird dieses Tag mit einem `.` (Punkt) bezeichnet.

Anmerkung

In CVS ist das die Vorgabe, wenn Sie kein Tag oder eine Revision angeben. Außer Sie wollen einen -STABLE Rechner auf -CURRENT aktualisieren, ist es *nicht* ratsam, die -CURRENT Quellen auf einem -STABLE Rechner einzuspielen.

RELENG_9
: Der Entwicklungszweig für FreeBSD-9.X, auch bekannt als FreeBSD 9-STABLE.

RELENG_9_1
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 9.1 durchgeführt werden.

RELENG_9_0
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 9.0 durchgeführt werden.

RELENG_8
: Der Entwicklungszweig für FreeBSD-8.X, auch bekannt als FreeBSD 8-STABLE.

RELENG_8_3
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 8.3 durchgeführt werden.

RELENG_8_2
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 8.2 durchgeführt werden.

RELENG_8_1
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 8.1 durchgeführt werden.

RELENG_8_0
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 8.0 durchgeführt werden.

RELENG_7
: Der Entwicklungszweig für FreeBSD-7.X, auch als FreeBSD 7-STABLE bekannt.

RELENG_7_4
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 7.4 durchgeführt werden.

RELENG_7_3
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 7.3 durchgeführt werden.

RELENG_7_2
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 7.2 durchgeführt werden.

RELENG_7_1
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 7.1 durchgeführt werden.

RELENG_7_0
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 7.0 durchgeführt werden.

RELENG_6
: Der Entwicklungszweig für FreeBSD-6.X, auch als FreeBSD 6-STABLE bekannt.

RELENG_6_4
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 6.4 durchgeführt werden.

RELENG_6_3
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 6.3 durchgeführt werden.

RELENG_6_2
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 6.2 durchgeführt werden.

RELENG_6_1
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 6.1 durchgeführt werden.

RELENG_6_0
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 6.0 durchgeführt werden.

RELENG_5
: Der FreeBSD 5.X Entwicklungszweig, der auch FreeBSD 5-STABLE genannt wird.

RELENG_5_5
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 5.5 durchgeführt werden.

RELENG_5_4
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 5.4 durchgeführt werden.

RELENG_5_3
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 5.3 durchgeführt werden.

RELENG_5_2
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 5.2 und FreeBSD 5.2.1 durchgeführt werden.

RELENG_5_1
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 5.1 durchgeführt werden.

RELENG_5_0
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 5.0 durchgeführt werden.

RELENG_4
: Der FreeBSD 4.X Entwicklungszweig, der auch FreeBSD 4-STABLE genannt wird.

RELENG_4_11
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.11 durchgeführt werden.

RELENG_4_10
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.10 durchgeführt werden.

RELENG_4_9
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.9 durchgeführt werden.

RELENG_4_8
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.8 durchgeführt werden.

RELENG_4_7
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.7 durchgeführt werden.

RELENG_4_6
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.6 und FreeBSD 4.6.2 durchgeführt werden.

RELENG_4_5
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.5 durchgeführt werden.

RELENG_4_4
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.4 durchgeführt werden.

RELENG_4_3
: Der Zweig, auf dem sicherheitsrelevante oder kritische Fehlerbehebungen für FreeBSD 4.3 durchgeführt werden.

RELENG_3
: Der FreeBSD-3.X Entwicklungszweig, der auch 3.X-STABLE genannt wird.

RELENG_2_2
: Der FreeBSD-2.2.X Entwicklungszweig, der auch 2.2-STABLE genannt wird.

A.8.2. Release-Tags

Diese Tags geben den Zeitpunkt an, an dem eine bestimme FreeBSD-Version veröffentlicht wurde. Das Erstellen einer Release ist in den Dokumenten `Release Engineering Information` und `Release Process` beschrieben. Der `src`-Baum benutzt Tags, deren Namen mit `RELENG_` anfangen. Die Bäume `ports` und `doc` benutzen Tags, deren Namen mit `RELEASE` anfangen. Im Baum `www` werden keine Release-Tags verwendet.

RELENG_9_2_0_RELEASE
: FreeBSD 9.2

RELENG_9_1_0_RELEASE
: FreeBSD 9.1

RELENG_9_0_0_RELEASE
: FreeBSD 9.0

RELENG_8_3_0_RELEASE
: FreeBSD 8.3

RELENG_8_2_0_RELEASE
: FreeBSD 8.2

RELENG_8_1_0_RELEASE
: FreeBSD 8.1

RELENG_8_0_0_RELEASE
: FreeBSD 8.0

RELENG_7_4_0_RELEASE
: FreeBSD 7.4

RELENG_7_3_0_RELEASE
: FreeBSD 7.3

RELENG_7_2_0_RELEASE
: FreeBSD 7.2

RELENG_7_1_0_RELEASE
: FreeBSD 7.1

RELENG_7_0_0_RELEASE
: FreeBSD 7.0

RELENG_6_4_0_RELEASE
: FreeBSD 6.4

RELENG_6_3_0_RELEASE
: FreeBSD 6.3

RELENG_6_2_0_RELEASE
: FreeBSD 6.2

RELENG_6_1_0_RELEASE
: FreeBSD 6.1

RELENG_6_0_0_RELEASE
: FreeBSD 6.0

RELENG_5_5_0_RELEASE
: FreeBSD 5.5

RELENG_5_4_0_RELEASE
: FreeBSD 5.4

RELENG_4_11_0_RELEASE
: FreeBSD 4.11

RELENG_5_3_0_RELEASE
: FreeBSD 5.3

RELENG_4_10_0_RELEASE
: FreeBSD 4.10

RELENG_5_2_1_RELEASE
: FreeBSD 5.2.1

RELENG_5_2_0_RELEASE
: FreeBSD 5.2

RELENG_4_9_0_RELEASE
: FreeBSD 4.9

RELENG_5_1_0_RELEASE
: FreeBSD 5.1

RELENG_4_8_0_RELEASE
: FreeBSD 4.8

RELENG_5_0_0_RELEASE
: FreeBSD 5.0

RELENG_4_7_0_RELEASE
: FreeBSD 4.7

RELENG_4_6_2_RELEASE
: FreeBSD 4.6.2

RELENG_4_6_1_RELEASE
: FreeBSD 4.6.1

RELENG_4_6_0_RELEASE
: FreeBSD 4.6

RELENG_4_5_0_RELEASE
: FreeBSD 4.5

RELENG_4_4_0_RELEASE
: FreeBSD 4.4

RELENG_4_3_0_RELEASE
: FreeBSD 4.3

RELENG_4_2_0_RELEASE
: FreeBSD 4.2

RELENG_4_1_1_RELEASE
: FreeBSD 4.1.1

RELENG_4_1_0_RELEASE
: FreeBSD 4.1

RELENG_4_0_0_RELEASE
: FreeBSD 4.0

RELENG_3_5_0_RELEASE
: FreeBSD-3.5

RELENG_3_4_0_RELEASE
: FreeBSD-3.4

RELENG_3_3_0_RELEASE
: FreeBSD-3.3

RELENG_3_2_0_RELEASE
: FreeBSD-3.2

RELENG_3_1_0_RELEASE
: FreeBSD-3.1

RELENG_3_0_0_RELEASE
: FreeBSD-3.0

RELENG_2_2_8_RELEASE
FreeBSD-2.2.8

RELENG_2_2_7_RELEASE
FreeBSD-2.2.7

RELENG_2_2_6_RELEASE
FreeBSD-2.2.6

RELENG_2_2_5_RELEASE
FreeBSD-2.2.5

RELENG_2_2_2_RELEASE
FreeBSD-2.2.2

RELENG_2_2_1_RELEASE
FreeBSD-2.2.1

RELENG_2_2_0_RELEASE
FreeBSD-2.2.0

A.9. rsync-Server

rsync wird ähnlich wie rcp(1) verwendet, besitzt aber mehr Optionen und verwendet das „rsync remote-update“ Protokoll, das nur geänderte Dateien überträgt und damit viel schneller als ein normaler Kopiervorgang ist. rsync ist sehr nützlich, wenn Sie einen FreeBSD-FTP-Spiegel oder einen CVS-Spiegel betreiben. Das Programm ist für viele Betriebssysteme erhältlich, mit FreeBSD können Sie den Port net/rsync oder das fertige Paket benutzen. Die folgenden Server stellen FreeBSD über das rsync Protokoll zur Verfügung:

Großbritannien
rsync://rsync.mirrorservice.org/

Verfügbare Sammlungen:

- ftp.freebsd.org: Kompletter Spiegel des FreeBSD-FTP-Servers.

Niederlande
rsync://ftp.nl.FreeBSD.org/

Verfügbare Sammlungen:

- FreeBSD: Kompletter Spiegel des FreeBSD-FTP-Servers.

Russland
rsync://ftp.mtu.ru/

Verfügbare Sammlungen:

- FreeBSD: Kompletter Spiegel des FreeBSD-FTP-Servers.
- FreeBSD-gnats: Die GNATS-Datenbank zur Verwaltung von Problemberichten.
- FreeBSD-Archive: Ein Spiegel des FreeBSD-Archive-FTP-Servers.

Schweden
rsync://ftp4.se.freebsd.org/

Verfügbare Sammlungen:

- FreeBSD: Kompletter Spiegel des FreeBSD-FTP-Servers.

Taiwan
rsync://ftp.tw.FreeBSD.org/

rsync://ftp2.tw.FreeBSD.org/

rsync://ftp6.tw.FreeBSD.org/

Verfügbare Sammlungen:

- FreeBSD: Kompletter Spiegel des FreeBSD-FTP-Servers.

Tschechische Republik
rsync://ftp.cz.FreeBSD.org/

Verfügbare Sammlungen:

- ftp: Unvollständiger Spiegel des FreeBSD-FTP-Servers.
- FreeBSD: Vollständiger Spiegel des FreeBSD-FTP-Servers.

USA
rsync://ftp-master.FreeBSD.org/

Dieser Server darf nur von primären Spiegeln benutzt werden.

Verfügbare Sammlungen:

- FreeBSD: Das Hauptarchiv des FreeBSD FTP Servers.
- acl: Die primäre ACL-Liste.

rsync://ftp13.FreeBSD.org/

Verfügbare Sammlungen:

- FreeBSD: Kompletter Spiegel des FreeBSD-FTP-Servers.

Anhang B. Bibliografie

Übersetzt von Frank Gründer <elwood@mc5sys.in-berlin.de>

Während die Manualpages eine definitive Referenz über bestimmte Teile des FreeBSD-Betriebssystems bieten, so können sie jedoch selten veranschaulichen, wie man die einzelnen Teile zusammenfügt, um ein vollständig laufendes Betriebssystem herzustellen. Daher gibt es keinen Ersatz für ein gutes Buch oder Benutzerhandbuch über die Administration von UNIX®-Systemen.

In der Regel handelt es sich im folgenden Kapitel um englische Ausgaben der genannten Werke. Übersetzungen oder Ausgaben in anderen Sprachen sind mit entsprechenden Hinweisen versehen.

B.1. Bücher speziell für FreeBSD

Internationale Bücher:

- Using FreeBSD, herausgegeben von Drmaster, 1997 (in traditionellem Chinesisch). ISBN 9-578-39435-7.
- FreeBSD Unleashed (in vereinfachtem Chinesisch), herausgegeben von China Press. ISBN 7-111-10201-0.
- FreeBSD From Scratch Second Edition (in vereinfachtem Chinesisch), herausgegeben von China Press. ISBN 7-111-10286-X.
- FreeBSD Handbook Second Edition (in vereinfachtem Chinesisch), herausgegeben von Posts & Telecom Press. ISBN 7-115-10541-3.
- FreeBSD & Windows (in vereinfachtem Chinesisch), herausgegeben von China Railway Publishing House. ISBN 7-113-03845-X.
- FreeBSD Internet Services HOWTO (in vereinfachtem Chinesisch), herausgegeben von China Railway Publishing House. ISBN 7-113-03423-3.
- FreeBSD (in japanischer Sprache), herausgegeben von CUTT. ISBN 4-906391-22-2 C3055 P2400E.
- Complete Introduction to FreeBSD (in Japanese), published by Shoeisha Co., Ltd. ISBN 4-88135-473-6 P3600E.
- Personal UNIX Starter Kit FreeBSD (in japanischer Sprache), herausgegeben von ASCII. ISBN 4-7561-1733-3 P3000E.
- FreeBSD Handbook (japanische Übersetzung), herausgegeben von ASCII. ISBN 4-7561-1580-2 P3800E.
- FreeBSD mit Methode (in deutscher Sprache), herausgegeben von Computer und Literatur Verlag /Vertrieb Hanser, 1998. ISBN 3-932311-31-0.
- FreeBSD de Luxe (in German), published by Verlag Modere Industrie, 2003. ISBN 3-8266-1343-0.
- FreeBSD Install and Utilization Manual (in japanischer Sprache), herausgegeben von Mainichi Communications Inc., 1998. ISBN 4-8399-0112-0.
- Onno W Purbo, Dodi Maryanto, Syahrial Hubbany, Widjil Widodo *Building Internet Server with FreeBSD* (in indonesischer Sprache), herausgegeben von Elex Media Komputindo.
- Absolute BSD: The Ultimate Guide to FreeBSD (in traditionellem Chinesisch), herausgegeben von GrandTech Press, 2003. ISBN 986-7944-92-5.
- The FreeBSD 6.0 Book (in traditionellem Chinesisch, herausgegeben von Drmaster, 2006. ISBN 9-575-27878-X.

Englischsprachige Bücher:

- Absolute FreeBSD, 2nd Edition: The Complete Guide to FreeBSD, herausgegeben von No Starch Press, 2007. ISBN: 978-1-59327-151-0
- The Complete FreeBSD, herausgegeben von O'Reilly, 2003. ISBN: 0596005164
- The FreeBSD Corporate Networker's Guide, herausgegeben von Addison-Wesley, 2002. ISBN: 0201704811
- FreeBSD: An Open-Source Operating System for Your Personal Computer, herausgegeben von The Bit Tree Press, 2001. ISBN: 0971204500
- Teach Yourself FreeBSD in 24 Hours, herausgegeben von Sams, 2002. ISBN: 0672324245
- FreeBSD6 Unleashed, herausgegeben von Sams, 2006. ISBN: 0672328755
- FreeBSD: The Complete Reference, herausgegeben von McGrawHill, 2003. ISBN: 0072224096

B.2. Handbücher

- Die Ohio State University hat ein UNIX Introductory Course veröffentlicht, welcher auch online im HTML- und PostScriptformat verfügbar ist.

  Eine italienische Übersetzung ist Teil des FreeBSD Italian Documentation Projects.

- Jpman Project, Japan FreeBSD Users Group. FreeBSD User's Reference Manual (japanische Übersetzung). Mainichi Communications Inc., 1998. ISBN4-8399-0088-4 P3800E.
- Edinburgh University hat einen Online Guide für Anfänger in Sachen UNIX geschrieben.

B.3. Administrations-Anleitungen

- Jpman Project, Japan FreeBSD Users Group. FreeBSD System Administrator's Manual (japanische Übersetzung). Mainichi Communications Inc., 1998. ISBN4-8399-0109-0 P3300E.
- Dreyfus, Emmanuel. Cahiers de l'Admin: BSD 2nd Ed. (in French), Eyrolles, 2004. ISBN 2-212-11463-X.

B.4. Programmierhandbücher

- Computer Systems Research Group, UC Berkeley. *4.4BSD Programmer's Reference Manual*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-078-3
- Computer Systems Research Group, UC Berkeley. *4.4BSD Programmer's Supplementary Documents*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-079-1
- Harbison, Samuel P. and Steele, Guy L. Jr. *C: A Reference Manual*. 4th Ed. Prentice Hall, 1995. ISBN 0-13-326224-3
- Kernighan, Brian and Dennis M. Ritchie. *The C Programming Language*. 2nd Ed., PTR Prentice Hall, 1988. ISBN 0-13-110362-9
- Lehey, Greg. *Porting UNIX Software*. O'Reilly & Associates, Inc., 1995. ISBN 1-56592-126-7
- Plauger, P. J. *The Standard C Library*. Prentice Hall, 1992. ISBN 0-13-131509-9
- Spinellis, Diomidis. *Code Reading: The Open Source Perspective*. Addison-Wesley, 2003. ISBN 0-201-79940-5
- Spinellis, Diomidis. *Code Quality: The Open Source Perspective*. Addison-Wesley, 2006. ISBN 0-321-16607-8

- Stevens, W. Richard and Stephen A. Rago. *Advanced Programming in the UNIX Environment*. 2nd Ed. Reading, Mass. : Addison-Wesley, 2005. ISBN 0-201-43307-9
- Stevens, W. Richard. *UNIX Network Programming*. 2nd Ed, PTR Prentice Hall, 1998. ISBN 0-13-490012-X

B.5. Betriebssystem-Interna

- Andleigh, Prabhat K. *UNIX System Architecture*. Prentice-Hall, Inc., 1990. ISBN 0-13-949843-5
- Jolitz, William. „Porting UNIX to the 386“. *Dr. Dobb's Journal*. January 1991-July 1992.
- Leffler, Samuel J., Marshall Kirk McKusick, Michael J Karels and John Quarterman *The Design and Implementation of the 4.3BSD UNIX Operating System*. Reading, Mass. : Addison-Wesley, 1989. ISBN 0-201-06196-1

  Kapitel 2 dieses Buchs ist Teil des FreeBSD Documentation Projects und online erhältlich.
- Leffler, Samuel J., Marshall Kirk McKusick, *The Design and Implementation of the 4.3BSD UNIX Operating System: Answer Book*. Reading, Mass. : Addison-Wesley, 1991. ISBN 0-201-54629-9
- McKusick, Marshall Kirk, Keith Bostic, Michael J Karels, and John Quarterman. *The Design and Implementation of the 4.4BSD Operating System*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-54979-4
- Marshall Kirk McKusick, George V. Neville-Neil. *The Design and Implementation of the FreeBSD Operating System*. Boston, Mass. : Addison-Wesley, 2004. ISBN 0-201-70245-2
- Marshall Kirk McKusick, George V. Neville-Neil, Robert N. M. Watson *The Design and Implementation of the FreeBSD Operating System, 2nd Ed.*. Westford, Mass: Pearson Education, Ind., 2014. ISBN 0-321-96897-2
- Stevens, W. Richard. *TCP/IP Illustrated, Volume 1: The Protocols*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-63346-9
- Schimmel, Curt. *Unix Systems for Modern Architectures*. Reading, Mass. : Addison-Wesley, 1994. ISBN 0-201-63338-8
- Stevens, W. Richard. *TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP and the UNIX Domain Protocols*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-63495-3
- Vahalia, Uresh. *UNIX Internals -- The New Frontiers*. Prentice Hall, 1996. ISBN 0-13-101908-2
- Wright, Gary R. and W. Richard Stevens. *TCP/IP Illustrated, Volume 2: The Implementation*. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-63354-X

B.6. Sicherheits-Anleitung

- Cheswick, William R. and Steven M. Bellovin. *Firewalls and Internet Security: Repelling the Wily Hacker*. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-63357-4
- Garfinkel, Simson. *PGP Pretty Good Privacy* O'Reilly & Associates, Inc., 1995. ISBN 1-56592-098-8

B.7. Hardware-Anleitung

- Anderson, Don and Tom Shanley. *Pentium Processor System Architecture*. 2nd Ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40992-5
- Ferraro, Richard F. *Programmer's Guide to the EGA, VGA, and Super VGA Cards*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-62490-7

- Die Intel Corporation veröffentlicht Dokumentationen Ihrer CPUs, Chipsets und Standards auf ihrer developer web site, normalerweise als PDF-Dateien.
- Shanley, Tom. *80486 System Architecture*. 3rd Ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40994-1
- Shanley, Tom. *ISA System Architecture*. 3rd Ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40996-8
- Shanley, Tom. *PCI System Architecture*. 4th Ed. Reading, Mass. : Addison-Wesley, 1999. ISBN 0-201-30974-2
- Van Gilluwe, Frank. *The Undocumented PC*, 2nd Ed. Reading, Mass: Addison-Wesley Pub. Co., 1996. ISBN 0-201-47950-8
- Messmer, Hans-Peter. *The Indispensable PC Hardware Book*, 4th Ed. Reading, Mass: Addison-Wesley Pub. Co., 2002. ISBN 0-201-59616-4

B.8. UNIX® Geschichte

- Lion, John *Lion's Commentary on UNIX, 6th Ed. With Source Code*. ITP Media Group, 1996. ISBN 1573980137
- Raymond, Eric S. *The New Hacker's Dictionary, 3rd edition*. MIT Press, 1996. ISBN 0-262-68092-0. Auch bekannt als das Jargon File
- Salus, Peter H. *A quarter century of UNIX*. Addison-Wesley Publishing Company, Inc., 1994. ISBN 0-201-54777-5
- Simon Garfinkel, Daniel Weise, Steven Strassmann. *The UNIX-HATERS Handbook*. IDG Books Worldwide, Inc., 1994. ISBN 1-56884-203-1. Online verfügbar.
- Don Libes, Sandy Ressler *Life with UNIX* — special edition. Prentice-Hall, Inc., 1989. ISBN 0-13-536657-7
- *The BSD family tree*. `https://svnweb.freebsd.org/base/head/share/misc/bsd-family-tree=view=co` oder unter `/usr/share/misc/bsd-family-tree` auf einem FreeBSD-System.
- *Networked Computer Science Technical Reports Library*. `http://www.ncstrl.org/`
- *Old BSD releases from the Computer Systems Research group (CSRG)*. `http://www.mckusick.com/csrg/` : Das Paket mit 4 CD-ROMs enthält alle BSD-Versionen von 1BSD bis 4.4BSD und 4.4BSD-Lite2 (nicht aber 2.11BSD). Die letzte CD beinhaltet auch die finalen Sourcen inklusive den SCCS Dateien.

B.9. Zeitschriften, Magazine und Journale

- `Admin Magazin` (in deutscher Sprache), herausgegeben von Medialinx AG. ISSN: 2190-1066
- `BSD Magazine`, herausgegeben von Software Press Sp. z o.o. SK. ISSN: 1898-9144
- `BSD Now – Video Podcast` , herausgegeben von Jupiter Broadcasting LLC
- `BSD Talk Podcast` , von Will Backman
- `FreeBSD Journal`, herausgegeben von S&W Publishing, gefördert durch The FreeBSD Foundation. ISBN: 978-0-615-88479-0

Anhang C. Ressourcen im Internet

Gedruckte Medien können mit der schnellen Entwicklung von FreeBSD nicht Schritt halten. Elektronische Medien sind häufig die einzige Möglichkeit, über aktuelle Entwicklungen informiert zu sein. Da FreeBSD ein Projekt von Freiwilligen ist, gibt die Benutzergemeinde selbst auch technische Unterstützung. Die Benutzergemeinde erreichen Sie am besten über E-Mail, Internetforen oder Usenet-News.

Die wichtigsten Wege, auf denen Sie die FreeBSD-Benutzergemeinde erreichen können, sind unten dargestellt. Wenn Sie weitere Ressourcen kennen, die hier fehlen, schicken Sie diese bitte an die Mailingliste des FreeBSD documentation project, damit sie hier aufgenommen werden können.

C.1. Mailinglisten

Die Mailinglisten sind der direkteste Weg, um Fragen an das gesamte FreeBSD Publikum zu stellen oder eine technische Diskussion zu beginnen. Es existiert eine grosse Vielfalt von Listen mit einer Reihe von verschiedenen FreeBSD Themen. Wenn Sie ihre Fragen an die richtige Mailingliste richten können Sie viel eher mit einer passenden Antwort darauf rechnen.

Die Chartas der verschiedenen Listen sind unten wiedergegeben. *Bevor Sie sich einer Mailingliste anschließen oder E-Mails an eine Liste senden, lesen Sie bitte die Charta der Liste.* Die meisten Mitglieder unserer Mailinglisten erhalten Hunderte E-Mails zum Thema FreeBSD pro Tag. Die Chartas und Regeln, die den Gebrauch der Listen beschreiben, garantieren die hohe Qualität der Listen. Die Listen würden ihren hohen Wert für das Projekt verlieren, wenn wir weniger Regeln aufstellen würden.

Anmerkung

Um zu testen, ob Sie eine Nachricht an eine FreeBSD-Liste senden können, verwenden Sie bitte Die Liste freebsd-test. Schicken Sie derartige Nachrichten bitte nicht an eine der anderen Listen.

Wenn Sie Sich nicht sicher sind, auf welcher Liste Sie Ihre Frage stellen sollen, sollten Sie den Artikel How to get best results from the FreeBSD-questions mailing list lesen.

Bevor Sie eine Nachricht an eine Mailingliste senden, sollten Sie die korrekte Nutzung der Mailinglisten erlernen. Dazu gehört auch das Vermeiden von sich häufig wiederholenden Diskussionen (lesen Sie deshalb zuerst die Mailing List Frequently Asked Questions).

Alle Mailinglisten werden archiviert und können auf dem FreeBSD World Wide Web Server durchsucht werden. Das nach Schlüsselwörtern durchsuchbare Archiv bietet die hervorragende Möglichkeit, Antworten auf häufig gestellte Fragen zu finden. Nutzen Sie bitte diese Möglichkeit, bevor Sie Fragen auf einer Liste stellen. Beachten Sie auch, dass das zur Folge hat, dass die Nachrichten an die FreeBSD Mailinglisten für die Ewigkeit erhalten bleiben. Wenn Sie am Schutz ihrer Privatsphähre interessiert sind, ziehen Sie die Verwendung einer Wegwerf-E-Mail-Adresse in Betracht und schreiben Sie nur solche Nachrichten, die für die Öffentlichkeit bestimmt sind.

C.1.1. Beschreibung der Mailinglisten

Allgemeine Listen: Jeder kann die folgenden allgemeinen Listen abonnieren (und ist dazu aufgefordert):

Mailingliste	Zweck
freebsd-advocacy	Verbreitung von FreeBSD
freebsd-announce	Wichtige Ereignisse und Meilensteine des Projekts
freebsd-arch	Architektur und Design von FreeBSD

Mailingliste	Zweck
freebsd-bugbusters	Diskussionen über die Pflege der FreeBSD Fehlerberichte-Datenbank und die dazu benutzten Werkzeuge
freebsd-bugs	Fehlerberichte
freebsd-chat	Nicht technische Themen, die die FreeBSD-Gemeinschaft betreffen
freebsd-chromium	Diskussionen zum Einsatz von Chromium unter FreeBSD
freebsd-current	Gebrauch von FreeBSD-CURRENT
freebsd-isp	Für Internet-Service-Provider, die FreeBSD benutzen
freebsd-jobs	Anstellung und Beratung im FreeBSD-Umfeld
freebsd-questions	Benutzerfragen und technische Unterstützung
freebsd-security-notifications	Ankündigungen zum Thema Sicherheit
freebsd-stable	Gebrauch von FreeBSD-STABLE
freebsd-test	Schicken Sie Testnachrichten an diese Liste anstelle der wirklichen Listen

Technische Listen: Auf den folgenden Listen werden technische Diskussionen geführt. Bevor Sie eine der Listen abonnieren oder Nachrichten an sie schicken, lesen Sie sich bitte die Charta der Liste durch, da der Inhalt und Zweck dieser Listen genau festgelegt ist.

Mailingliste	Zweck
freebsd-acpi	Entwicklung von ACPI
freebsd-afs	Portierung von AFS nach FreeBSD
freebsd-aic7xxx	Entwicklung von Adaptec® AIC 7xxx Treibern
freebsd-amd64	Portierung von FreeBSD auf AMD64-Systeme
freebsd-apache	Diskussion über Ports, die mit Apache zusammenhängen.
freebsd-arm	Portierung von FreeBSD auf ARM®-Prozessoren
freebsd-atm	Benutzung von ATM-Netzen mit FreeBSD
freebsd-binup	Design und Entwicklung eines Systems, das es erlaubt, ein FreeBSD-System mit binären Paketen zu aktualisieren
freebsd-bluetooth	Bluetooth® unter FreeBSD verwenden
freebsd-cluster	Benutzung von FreeBSD in einem Cluster
freebsd-cvsweb	Pflege von CVSweb
freebsd-database	Diskussion über Datenbanken und Datenbankprogrammierung unter FreeBSD
freebsd-desktop	FreeBSD als Desktop verwenden und verbessern
freebsd-doc	Erstellen der FreeBSD-Dokumentation
freebsd-drivers	Gerätetreiber für FreeBSD schreiben
freebsd-eclipse	Für FreeBSD-Anwender, die die Eclipse IDE, deren Werkzeuge, Anwendungen und Ports einsetzen
freebsd-embedded	FreeBSD in eingebetteten Anwendungen einsetzen
freebsd-emulation	Emulation anderer Systeme wie Linux, MS-DOS® oder Windows®

Mailingliste	Zweck
freebsd-eol	Support für FreeBSD-bezogene Software, die vom FreeBSD Project offiziell nicht mehr unterstützt wird.
freebsd-firewire	Technische Diskussion über FireWire® (iLink, IEEE 1394)
freebsd-fs	Dateisysteme
freebsd-gecko	Angelegenheiten zur Gecko Rendering Engine
freebsd-geom	Diskussion über GEOM
freebsd-gnome	Portierung von GNOME und GNOME-Anwendungen
freebsd-hackers	Allgemeine technische Diskussionen
freebsd-hardware	Allgemeine Diskussion über Hardware, auf der FreeBSD läuft
freebsd-i18n	Internationalisierung von FreeBSD
freebsd-ia32	FreeBSD für die IA-32 (Intel® x86) Plattform
freebsd-ia64	Portierung von FreeBSD auf Intel®s neue IA64-Systeme
freebsd-ipfw	Technische Diskussion über die Neubearbeitung der IP-Firewall Quellen
freebsd-isdn	Für Entwickler des ISDN-Systems
freebsd-java	Für Java™ Entwickler und Leute, die JDK™s nach FreeBSD portieren
freebsd-kde	Portierung von KDE und KDE-Anwendungen
freebsd-lfs	Portierung von LFS nach FreeBSD
freebsd-mips	Portierung von FreeBSD zu MIPS®
freebsd-mobile	Diskussionen über mobiles Rechnen
freebsd-mono	Mono und C# Anwendungen auf FreeBSD
freebsd-mozilla	Portierung von Mozilla nach FreeBSD
freebsd-multimedia	Multimedia Anwendungen
freebsd-new-bus	Technische Diskussionen über die Architektur von Bussen
freebsd-net	Diskussion über Netzwerke und den TCP/IP Quellcode
freebsd-numerics	Diskussionen über die Implementation hochwertiger Funktionen in libm
freebsd-office	Office-Anwendungen für FreeBSD
freebsd-performance	Fragen zur Optimierung der Leistung stark ausgelasteter Systeme
freebsd-perl	Pflege der portierten Perl-Anwendungen.
freebsd-pf	Diskussionen und Fragen zu *packet filter* als Firewallsystem.
freebsd-platforms	Portierungen von FreeBSD auf nicht-Intel® Architekturen
freebsd-ports	Diskussion über die Ports-Sammlung
freebsd-ports-announce	Wichtige Neuigkeiten und Anweisungen zur Ports-Sammlung

Mailingliste	Zweck
freebsd-ports-bugs	Diskussion über Fehler und PRs der Ports
freebsd-ppc	Portierung von FreeBSD auf den PowerPC®
freebsd-proliant	Technische Diskussionen zum Einsatz von FreeBSD auf der ProLiant-Serverplattform von HP.
freebsd-python	FreeBSD-spezifische Diskussionen zu Python
freebsd-rc	Diskussion über das `rc.d`-System sowie dessen Weiterentwicklung
freebsd-realtime	Entwicklung von Echtzeiterweiterungen für FreeBSD
freebsd-ruby	FreeBSD-spezifische Diskussionen zu Ruby
freebsd-scsi	Diskussion über das SCSI-Subsystem
freebsd-security	Sicherheitsthemen
freebsd-small	Gebrauch von FreeBSD in eingebetteten Systemen (obsolet; verwenden Sie stattdessen freebsd-embedded)
freebsd-sparc64	Portierung von FreeBSD auf SPARC® Systeme
freebsd-standards	Konformität von FreeBSD mit den C99- und POSIX-Standards
freebsd-sysinstall	sysinstall(8) Entwicklung
freebsd-threads	Leichgewichtige Prozesse (*Threads*) in FreeBSD
freebsd-tilera	Diskussionen zur Portierung von FreeBSD auf die Tilera-CPU-Familie.
freebsd-tokenring	Token-Ring Unterstützung in FreeBSD
freebsd-toolchain	Wartung der FreeBSD-Toolchain
freebsd-usb	USB-Unterstützung in FreeBSD
freebsd-virtualization	Diskussion über verschiedene Virtualisierungsverfahren, die von FreeBSD unterstützt werden
freebsd-vuxml	Diskussion über die Infratruktur von VuXML
freebsd-x11	Wartung und Unterstützung von X11 auf FreeBSD
freebsd-xen	Diskussionen über die FreeBSD Portierung auf Xen™ - Implementierung und Verwendung
freebsd-xfce	Portierung und Wartung von XFCE
freebsd-zope	Zope für FreeBSD - Portierung und Wartung

Eingeschränkte Listen: Die folgenden Listen wenden sich an Zielgruppen mit speziellen Anforderungen und sind nicht für die Öffentlichkeit gedacht. Bevor Sie eine dieser Listen abonnieren, sollten Sie einige der technischen Listen abonniert haben, um mit den Umgangsformen vertraut zu sein.

Mailingliste	Zweck
freebsd-hubs	Betrieb von FreeBSD-Spiegeln
freebsd-user-groups	Koordination von Benutzergruppen
freebsd-vendors	Koordination von Händlern vor einem Release
freebsd-wip-status	Status von in Arbeit befindlichen FreeBSD-Tätigkeiten
freebsd-wireless	Diskussionen zum 802.11-Stack sowie zur Entwicklung von Tools und Gerätetreibern

Mailingliste	Zweck
freebsd-www	Betreuer von www.FreeBSD.org

Zusammenfassungen: Alle eben aufgezählten Listen sind auch in zusammengefasster Form (*digest*) erhältlich. In den Einstellungen Ihres Accounts legen Sie fest, in welcher Form Sie die Listen empfangen.

SVN Listen: Die folgenden Listen versenden die Log-Einträge zu Änderungen an verschiedenen Teilen des Quellbaums. Diese Listen sollen *nur gelesen* werden, schicken Sie bitte keine Nachrichten an eine der Listen.

Mailingliste	Teil des Quellbaums	Beschreibung
svn-doc-all	`/usr/doc`	Änderungen im doc Subversion Repository (mit Ausnahme von `user`, `projects` und `translations`)
svn-doc-head	`/usr/doc`	Änderungen im „head“-Zweig des doc Subversion Repository
svn-doc-projects	`/usr/doc/projects`	Änderungen im `projects`-Bereich des doc Subversion Repository
svn-doc-svnadmin	`/usr/doc`	Änderungen an den administrativen Skripten, Hooks und anderen Konfigurationsdateien des doc Subversion Repository
svn-ports-all	`/usr/ports`	Alle Änderungen des ports Subverison Repository
svn-ports-head	`/usr/ports`	Änderungen im „head“-Zweig des ports Subversion Repository
svn-ports-svnadmin	`/usr/ports`	Änderungen an den administrativen Skripten, Hooks und anderen Konfigurationsdateien des ports Subversion Repository
svn-src-all	`/usr/src`	Änderungen im src Subversion Repository (ausser für `user` und `projects`)
svn-src-head	`/usr/src`	Änderungen im „head“ Zweig des src Subversion Repository (der FreeBSD-CURRENT Zweig)
svn-src-projects	`/usr/projects`	Änderungen im `projects` Bereich des src Subversion Repository
svn-src-release	`/usr/src`	Änderungen im `releases` Bereich des src Subversion Repository
svn-src-releng	`/usr/src`	Änderungen im `releng` Zweig des src Subversion Repository (der security / release engineering Zweige)
svn-src-stable	`/usr/src`	Änderungen an allen stable Zweigen des src Subversion Repository
svn-src-stable-6	`/usr/src`	Änderungen im `stable/6` Zweig des src Subversion Repository
svn-src-stable-7	`/usr/src`	Änderungen im `stable/7` Zweig des src Subversion Repository

Mailingliste	Teil des Quellbaums	Beschreibung
svn-src stable-8	`/usr/src`	Änderungen im `stable/8` Zweig des src Subversion Repository
svn-src-stable-9	`/usr/src`	Änderungen im `stable/8` Zweig des src Subversion Repository
svn-src-stable-other	`/usr/src`	Änderungen an älteren `stable` Zweigen des src Subversion Repository
svn-src-svnadmin	`/usr/src`	Änderungen an den administrativen Skripten, hooks, and anderen Daten zur Konfiguration des src Subversion Repository
svn-src-user	`/usr/src`	Änderungen am experimentellen `user` Bereich des src Subversion Repository
svn-src-vendor	`/usr/src`	Änderungen am Herstellerbereich des src Subversion Repository

C.1.2. Mailinglisten abonnieren

Um eine Liste zu abonnieren, folgen Sie dem oben angegebenen Hyperlink der Liste oder Sie besuchen die Webseite http://lists.FreeBSD.org/mailman/listinfo und klicken dort auf die Liste, die Sie abonnieren wollen. Sie gelangen dann auf die Webseite der Liste, die weitere Anweisungen enthält.

Um eine Nachricht an eine Mailingliste zu schicken, schreiben Sie einfach eine E-Mail an <*Liste*@FreeBSD.org>. Die E-Mail wird dann an alle Mitglieder der Mailingliste verteilt.

Wenn Sie das Abonnement aufheben wollen, folgen Sie der URL, die am Ende jeder Mail der Liste angegeben ist. Sie können das Abonnement auch mit einer E-Mail an <*Liste*-unsubscribe@FreeBSD.org> aufheben.

Verwenden Sie bitte die technischen Listen ausschließlich für technische Diskussionen. Wenn Sie nur an wichtigen Ankündigungen interessiert sind, abonnieren Sie die Mailingliste FreeBSD announcements, auf der nur wenige Nachrichten versendet werden.

C.1.3. Chartas der Mailinglisten

Alle FreeBSD-Mailinglisten besitzen Grundregeln, die von jedem beachtet werden müssen. Für die ersten beiden Male, in denen ein Absender gegen diese Regeln verstößt, erhält er jeweils eine Warnung vom FreeBSD-Postmaster <postmaster@FreeBSD.org>. Ein dritter Verstoß gegen die Regeln führt dazu, dass der Absender in allen FreeBSD-Mailinglisten gesperrt wird und weitere Nachrichten von ihm nicht mehr angenommen werden. Wir bedauern sehr, dass wir solche Maßnahmen ergreifen müssen, aber heutzutage ist das Internet eine recht rauhe Umgebung, in der immer weniger Leute Rücksicht aufeinander nehmen.

Die Regeln:

- Das Thema einer Nachricht soll der Charta der Liste, an die sie gesendet wird, entsprechen. Wenn Sie eine Nachricht an eine technische Liste schicken, sollte die Nachricht auch technische Inhalte haben. Fortwährendes Geschwätz oder Streit mindern den Wert der Liste für alle Mitglieder und wird nicht toleriert. Benutzen Sie FreeBSD chat für allgemeine Diskussionen über FreeBSD.

- Eine Nachricht sollte an nicht mehr als zwei Mailinglisten gesendet werden. Schicken Sie eine Nachricht nur dann an zwei Listen, wenn das wirklich notwendig ist. Viele Leute haben mehrere Mailinglisten abonniert und Nachrichten sollten nur zu ungewöhnlichen Kombinationen der Listen, wie „-stable“ und „-scsi“, gesendet werden. Wenn Sie eine Nachricht erhalten, die im `Cc`-Feld mehrere Listen enthält, sollten Sie das Feld kürzen, bevor Sie eine Antwort darauf verschicken. *Unabhängig von dem ursprünglichen Verteiler sind Sie für Ihre eigenen Mehrfach-Sendungen selbst verantwortlich.*

- Persönliche Angriffe und Beschimpfungen sind in einer Diskussion nicht erlaubt. Dies gilt gleichermaßen für Benutzer wie Entwickler. Grobe Verletzungen der Netiquette, wie das Verschicken oder Zitieren von privater E-Mail ohne eine entsprechende Genehmigung, werden nicht gebilligt. Die Nachrichten werden aber nicht besonders auf Verletzungen der Netiquette untersucht. Es kann sein, dass eine Verletzung der Netiquette durchaus zu der Charta einer Liste passt, aber der Absender aufgrund der Verletzung eine Warnung erhält oder gesperrt wird.

- Werbung für Produkte oder Dienstleistungen, die nichts mit FreeBSD zu tun haben, sind verboten. Ist die Werbung als Spam verschickt worden, wird der Absender sofort gesperrt.

Chartas einzelner Listen:

freebsd-acpi
: Die Entwicklung von ACPI und Energieverwaltungsfunktionen.

freebsd-afs
: *Andrew File System*

  Auf dieser Liste wird die Portierung des AFS von CMU/Transarc diskutiert.

freebsd-announce
: *Wichtige Ereignisse und Meilensteine*

  Diese Liste ist für Personen, die nur an den wenigen Ankündigungen wichtiger Ereignisse interessiert sind. Die Ankündigungen betreffen Schnappschüsse und Releases, neue Merkmale von FreeBSD und die Suche nach freiwilligen Mitarbeitern. Auf der Liste herrscht wenig Verkehr und sie wird streng moderiert.

freebsd-arch
: *Architektur und Design von FreeBSD*

  Auf dieser technischen Liste wird die FreeBSD-Architektur diskutiert. Beispiele für angemessene Themen sind:

  - Wie das Bausystem zu verändern ist, damit verschiedene Läufe gleichzeitig möglich sind.
  - Was am VFS geändert werden muss, damit Heidemann Schichten eingesetzt werden können.
  - Wie die Schnittstelle der Gerätetreiber angepasst werden muss, damit derselbe Treiber auf verschiedenen Bussen und Architekturen eingesetzt werden kann.
  - Wie ein Netzwerktreiber geschrieben wird.

freebsd-binup
: *FreeBSD Binary Update Project*

  Auf dieser Liste wird das Design und die Implementierung von binup diskutiert. Weitere Themen sind Fehlerbehebungen, Fehlerberichte und Anfragen nach Neuerungen. Die CVS-Logmeldungen des Projekts werden ebenfalls auf diese Liste gesendet.

freebsd-bluetooth
: *Bluetooth® unter FreeBSD*

  Diese Liste diskutiert Probleme der Verwendung von Bluetooth® unter FreeBSD. Designprobleme, Implementierungsdetails, Patches, Fehler- und Statusberichte, Verbesserungsvorschläge sowie alle anderen mit Bluetooth® zusammenhängenden Themen werden hier behandelt.

freebsd-bugbusters
: *Bearbeitung der Fehlerberichte*

  Auf dieser Liste wird die Bearbeitung der Fehlerberichte (PR, engl. *problem report*) koordiniert. Sie dient dem „Bugmeister“ und allen Leuten, die ein Interesse an der Datenbank der Fehlerberichte haben, als Diskussionsforum. Auf dieser Liste werden keine spezifischen Fehler, Fehlerbehebungen oder PRs diskutiert.

freebsd-bugs
Fehlerberichte

Auf dieser Liste werden Fehlerberichte gesammelt. Fehlerberichte sollten immer mit send-pr(1) oder dem Web Formular erstellt werden.

freebsd-chat
Nicht technische Themen, die die FreeBSD Gemeinschaft betreffen

Auf dieser Liste werden nicht-technische soziale Themen diskutiert, die nicht auf die anderen Listen passen. Hier kann diskutiert werden, ob Jordan wie ein Frettchen aus einem Zeichentrickfilm aussieht oder nicht, ob grundsätzlich in Großbuchstaben geschrieben werden soll, wer zuviel Kaffee trinkt, wo das beste Bier gebraut wird und wer Bier in seinem Keller braut. Gelegentlich können auf den technischen Listen wichtige Ereignisse wie Feste, Hochzeiten oder Geburten angekündigt werden, aber nachfolgende Nachrichten sollten auf die Liste FreeBSD chat gesendet werden.

freebsd-chromium
Diskussionen zum Einsatz von Chromium unter FreeBSD

Auf dieser technischen Liste werden Fragen zur Entwicklung, zur Installation sowie zum Einsatz von Chromium unter FreeBSD diskutiert.

freebsd-core
FreeBSD Core Team

Dies ist eine interne Mailingliste des FreeBSD Core Teams. Wenn in einer wichtigen Angelegenheit, die FreeBSD betrifft, entschieden werden muss oder die Angelegenheit einer genauen Prüfung unterzogen werden muss, können Nachrichten an diese Liste gesendet werden.

freebsd-current
Gebrauch von FreeBSD-CURRENT

Diese Mailingliste ist für die Benutzer von FreeBSD-CURRENT eingerichtet. Auf ihr finden sich Ankündigungen über Besonderheiten von -CURRENT, von denen Benutzer betroffen sind. Sie enthält weiterhin Anweisungen, wie man ein System auf -CURRENT hält. Jeder, der ein -CURRENT System besitzt, muss diese Liste lesen. Die Liste ist nur für technische Inhalte bestimmt.

freebsd-cvsweb
FreeBSD CVSweb Project

Technische Diskussion über den Gebrauch, die Entwicklung und die Pflege von FreeBSD-CVSweb.

freebsd-desktop
FreeBSD als Desktop verwenden und verbessern

Dies ist ein Forum für Diskussionen um FreeBSD auf dem Desktop. Es wird primär von Desktop-Portierern und Nutzern verwendet, um Probleme und Verbesserungen zu FreeBSDs Einsatz auf dem Desktop zu besprechen.

freebsd-doc
Documentation Project

Auf dieser Mailingliste werden Themen und Projekte diskutiert, die im Zusammenhang mit der Erstellung der FreeBSD Dokumentation stehen. „The FreeBSD Documentation Project“ besteht aus den Mitgliedern dieser Liste. Diese Liste steht jedem offen, Sie sind herzlich eingeladen teilzunehmen und mitzuhelfen.

freebsd-drivers
Gerätetreiber für FreeBSD schreiben

Ein Forum für technische Diskussionen über das Schreiben von Gerätetreibern für FreeBSD. Daher werden hier vor allem Fragen behandelt, die sich um das Schreiben von Treibern, die die APIs des Kernels nutzen, drehen.

freebsd-eclipse
Für FreeBSD-Anwender, die die Eclipse IDE deren Werkzeuge, Anwendungen und Ports einsetzen

Das Ziel dieser Liste ist es, Unterstützung für all jene zu bieten, die mit der Installation, Verwendung, Entwicklung und Wartung der Eclipse-IDE sowie deren Werkzeugen und Anwendungen unter FreeBSD zu tun haben. Außerdem wird Hilfe bei der Portierung der IDE und deren Plugins auf FreeBSD geboten.

Zusätzlich soll diese Liste einen Informationsaustausch zwischen der Eclipse- und der FreeBSD-Gemeinde ermöglichen, von dem beide Seiten profitieren können.

Obwohl sich diese Liste auf die Anforderungen von Anwendern konzentriert, möchte sie auch Entwickler unterstützen, die an der Entwicklung von FreeBSD-spezifischen Anwendungen unter Nutzung des Eclipse-Frameworks arbeiten.

freebsd-embedded
FreeBSD in eingebetteten Anwendungen einsetzen

Diese Liste diskutiert Themen im Zusammenhang mit dem Einsatz von ungewöhnlich kleinen und eingebetteten FreeBSD-Installationen. Auf dieser Liste werden ausschließlich technische Diskussionen geführt. Unter eingebetteten Systemen versteht diese Liste Systeme, bei denen es sich nicht um Desktopsysteme handelt, und die in der Regel nur einem einzigen Zweck dienen (im Gegensatz zu Desktopsystemen, die für die Bewältigung verschiedenster Aufgaben geeignet sind). In die Gruppe der eingebetteten Systeme gehören beispielsweise Telephone, Netzwerkgeräte wie Router, Switche oder PBX-Systeme, PDAs, Verkaufsautomaten und andere mehr.

freebsd-emulation
Emulation anderer Systeme wie Linux, MS-DOS® oder Windows®

Hier werden technische Diskussionen zum Einsatz von Programmen, die für andere Betriebssysteme geschrieben wurden, geführt.

freebsd-eol
Support für FreeBSD-bezogene Software, die vom FreeBSD Project offiziell nicht mehr unterstützt wird.

Diese Liste ist für all jene interessant, die Unterstützung für vom FreeBSD Project offiziell nicht mehr (in Form von Security Advisories oder Patches) unterstützte Programme benötigen oder anbieten wollen.

freebsd-firewire
FireWire® (iLink, IEEE 1394)

Auf dieser Liste wird das Design und die Implementierung eines FireWire®-Subsystems (auch IEEE 1394 oder iLink) für FreeBSD diskutiert. Relevante Themen sind die Standards, Busse und ihre Protokolle, sowie Adapter, Karten und Chipsätze. Des Weiteren die Architektur und der Quellcode, die nötig sind, diese Geräte zu unterstützen.

freebsd-fs
Dateisysteme

Diskussionen über FreeBSD-Dateisysteme. Dies ist eine technische Liste, in der nur technische Inhalte erwartet werden.

freebsd-gecko
Angelegenheiten zur *Gecko Rendering Engine*

Dies ist ein Forum über Gecko-Anwendungen, die FreeBSD verwenden.

Die Diskussion dreht sich um die Portierung von Gecko-Anwendungen, deren Installation, die Entwicklung sowie deren Unterstützung innerhalb von FreeBSD.

freebsd-geom
GEOM

Diskussion über GEOM und verwandte Implementierungen. Dies ist eine technische Liste, in der nur technische Inhalte erwartet werden.

freebsd-gnome
GNOME

Diskussionen über die grafische Benutzeroberfläche GNOME. Dies ist eine technische Liste, in der nur technische Inhalte erwartet werden.

freebsd-ipfw
IP Firewall

Diskussionen über eine Neubearbeitung des IP-Firewall Quelltexts in FreeBSD. Dies ist eine technische Liste, in der nur technische Inhalte erwartet werden.

freebsd-ia64
Portierung von FreeBSD auf die IA64-Plattform

Dies ist eine technische Liste für diejenigen, die FreeBSD auf die IA-64 Plattform von Intel® portieren. Themen sind die Probleme bei der Portierung und deren Lösung. Interessierte, die der Diskussion folgen wollen, sind ebenfalls willkommen.

freebsd-isdn
ISDN Subsystem

Mailingliste für die Entwickler des ISDN Subsystems von FreeBSD.

freebsd-java
Java™ Entwicklung

Mailingliste, auf der die Entwicklung von Java™ Anwendungen für FreeBSD sowie die Portierung und Pflege von JDK™s diskutiert wird.

freebsd-jobs
Stellenangebote und Stellengesuche

In diesem Forum können Sie Stellenangebote und Stellengesuche, die mit FreeBSD zu tun haben, aufgeben. Wenn Sie beispielsweise eine Beschäftigung im FreeBSD-Umfeld suchen oder eine freie Stelle haben, die mit FreeBSD zu tun hat, ist dies der richtige Ort. Diese Mailingliste ist *nicht* der Ort, um über allgemeine Beschäftigungsprobleme zu diskutieren; dazu gibt es anderswo geeignete Foren.

Beachten Sie bitte, dass diese Liste, wie die anderen `FreeBSD.org`-Listen, weltweit gelesen wird. Geben Sie daher bitte den Arbeitsort genau an. Geben Sie bitte auch an, ob Telearbeit möglich ist und ob Hilfen für einen Umzug angeboten werden.

Benutzen Sie in der E-Mail bitte nur offene Formate – vorzugsweise nur das Textformat. Andere Formate, wie PDF oder HTML, werden von den Lesern akzeptiert. Nicht offene Formate wie Microsoft® Word (`.doc`) werden vom Server der Liste abgelehnt.

freebsd-hackers
Technische Diskussionen

Dies ist ein Forum für technische Diskussionen über FreeBSD. Leute, die aktiv an FreeBSD arbeiten, können hier Probleme und deren Lösungen diskutieren. Interessierte, die den Diskussionen folgen wollen, steht die Liste ebenfalls offen. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-hardware
Allgemeine Diskussionen über Hardware

Allgemeine Diskussionen über die Hardware, auf der FreeBSD läuft: Probleme und Ratschläge welche Hardware man kaufen sollte und welche nicht.

freebsd-hubs
FreeBSD-Spiegel

Ankündigungen und Diskussionsforum für Leute, die FreeBSD-Spiegel betreiben.

freebsd-isp
Themen für Internet Service Provider

Diese Liste ist für Internet Service Provider (ISP), die FreeBSD benutzen. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-mono
Mono und C# Anwendungen auf FreeBSD

Diese Liste beinhaltet Diskussionen über das Mono Entwicklungsframework auf FreeBSD. Dies ist eine technische Mailingliste. Es ist für Personen gedacht, die aktiv an der Portierung von Mono oder C# Anwendungen auf FreeBSD sind, um Probleme oder alternative Lösungen zu beratschlagen. Personen die der technischen Diskussion folgen möchten sind ebenso willkommen.

freebsd-kde
KDE

Diskussionen über KDE auf FreeBSD-Systemen. Dies ist eine technische Liste, in der nur technische Inhalte erwartet werden.

freebsd-performance
Diskussionsforum mit dem Ziel, die Leistung von FreeBSD zu verbessern.

Auf dieser Liste diskutieren Hacker, Systemadministratoren und andere Interessierte die Leistung von FreeBSD. Zulässige Themen sind beispielsweise Systeme unter hoher Last, Systeme mit Leistungsproblemen oder Systeme, die Leistungsgrenzen von FreeBSD überwinden. Jeder, der mithelfen will, die Leistung von FreeBSD zu verbessern, sollte diese Liste abonnieren. Die Liste ist technisch anspruchsvoll und geeignet für erfahrene FreeBSD-Benutzer, Hacker oder Administratoren, die FreeBSD schnell, robust und skalierbar halten wollen. Auf der Liste werden Beiträge gesammelt oder Fragen nach ungelösten Problemen beantwortet. Sie ist kein Ersatz für das gründliche Studium der Dokumentation.

freebsd-pf
Diskussionen und Fragen zu packet filter als Firewallsystem.

FreeBSD-spezische Diskussionen zur Benutzung von *packet filter* (pf) als Firewallsystem. Sowohl technische Diskussionen als auch Anwenderfragen sind auf dieser Liste willkommen. Fragen zum ALTQ QoS Framework können ebenfalls gestellt werden.

freebsd-platforms
Portierung auf nicht-Intel® Plattformen

Plattformübergreifende Themen und Vorschläge für die Portierung auf nicht-Intel® Plattformen. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-ports
Diskussion über die Ports-Sammlung

Diskussionen über die FreeBSD-Ports-Sammlung und die Infrastruktur der Sammlung. Die Liste dient auch der allgemeinen Koordination der Dinge, die die Ports-Sammlung betreffen. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-ports-bugs
Diskussion über Fehler in den Ports

Diskussion über Fehler in der Ports-Sammlung (`/usr/ports`), neue Ports oder Änderungen an bestehenden Ports. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-proliant
Technische Diskussionen zum Einsatz von FreeBSD auf der ProLiant-Serverplattform von HP

Diese Mailingliste bietet technische Diskussionen zum Einsatz von FreeBSD auf der ProLiant-Serverplattform von HP, darunter Fragen zu ProLiant-spezifischen Treibern, Konfigurationswerkzeugen sowie BIOS-Aktualisierungen. Daher ist sie die erste Anlaufstelle, um die Module hpasmd, hpasmcli, sowie hpacucli zu diskutieren.

freebsd-python
Python unter FreeBSD

Diese technische Liste dient der Verbesserung der Python-Unterstützung unter FreeBSD. Sie wird von Personen gelesen, die an der Portierung von Python, von Python-Modulen Dritter und von Zope nach FreeBSD arbeiten. Personen, die diese technischen Diskussion verfolgen wollen, sind ebenfalls willkommen.

freebsd-questions
Benutzerfragen

Auf dieser Mailingliste können Fragen zu FreeBSD gestellt werden. Fragen Sie bitte nicht nach Anleitungen, wenn Sie nicht sicher sind, dass Ihre Frage wirklich technischer Natur ist.

freebsd-ruby
Ruby unter FreeBSD

Diese technische Liste dient der Verbesserung der Ruby-Unterstützung unter FreeBSD. Sie wird von Personen gelesen, die an der Portierung von Ruby, von Bibliotheken Dritter und Frameworks arbeiten. Personen, die diese technischen Diskussionen verfolgen wollen, sind ebenfalls willkommen.

freebsd-scsi
SCSI Subsystem

Diese Mailingliste ist für die Entwickler des SCSI Subsystems von FreeBSD. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-security
Sicherheitsthemen

Sicherheitsthemen, die FreeBSD betreffen, wie DES, Kerberos, bekannte Sicherheitslöcher und Fehlerbehebungen. Stellen Sie bitte auf dieser Liste keine allgemeinen Fragen zum Thema Sicherheit. Willkommen sind allerdings Beiträge zur FAQ, das heißt eine Frage mit der passenden Antwort. Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-security-notifications
Ankündigungen zum Thema Sicherheit

Ankündigungen über Sicherheitsprobleme von FreeBSD und deren Behebungen. Diese Liste ist kein Diskussionsforum, benutzen Sie FreeBSD security, um Sicherheitsthemen zu diskutieren.

freebsd-small
Gebrauch von FreeBSD in eingebetteten Systemen.

Diese Liste für ungewöhnlich kleine FreeBSD Installation oder den Einsatz von FreeBSD in eingebetteten Systemen gedacht. Auf dieser Liste finden nur technische Diskussionen statt.

Anmerkung

Diese Liste wurde durch freebsd-embedded ersetzt.

freebsd-stable
Gebrauch von FreeBSD-STABLE.

Diese Mailingliste ist für die Benutzer von FreeBSD-STABLE eingerichtet. Auf ihr finden sich Ankündigungen über Besonderheiten von -STABLE, von denen Benutzer betroffen sind. Sie enthält weiterhin Anweisungen, wie man ein System auf -STABLE hält. Jeder, der ein -STABLE System besitzt, muss diese Liste lesen. Die Liste ist nur für technische Inhalte bestimmt.

freebsd-standards
Konformität von FreeBSD mit den C99- und POSIX® Standards

Dieses Forum ist für technische Diskussionen über die Konformität von FreeBSD mit den C99- und POSIX-Standards.

freebsd-toolchain
Wartung der FreeBSD-Toolchain

Auf dieser Mailingliste werden alle Themen rund um die FreeBSD-Toolchain diskutiert. Dazu gehören der Status von Clang und GCC, aber auch Fragen zu Programmen wie Assemblern, Linkern und Debuggern.

freebsd-usb
USB-Unterstützung in FreeBSD.

Auf dieser Liste finden nur technische Diskussionen statt.

freebsd-user-groups
Koordination von Benutzergruppen

Diese Liste ist für Koordinatoren lokaler Benutzergruppen und einem ausgesuchten Mitglied des Core Teams eingerichtet worden. Der Inhalt sollte Inhalte von Treffen und die Koordination von Projekten mehrerer Benutzergruppen beschränkt sein.

freebsd-vendors
Koordination von Händlern

Koordination zwischen dem FreeBSD Project und Händlern, die Soft- und Hardware für FreeBSD verkaufen.

freebsd-virtualization
Diskussion über verschiedene Virtualisierungsverfahren, die von FreeBSD unterstützt werden

Eine Liste, auf der die verschiedenen Virtualisierungsverfahren, die von FreeBSD unterstützt werden, diskutiert werden. Auf der einen Seite liegt der Fokus auf der Implementierung der zugrundeliegenden Funktionalitäten, ebenso wie das Hinzufügen neuer Eigenschaften. Auf der anderen Seite haben die Benutzer ein Forum, um Fragen bei Problemen zu stellen oder um ihre Anwendungsfälle zu besprechen.

freebsd-wip-status
Status von in Arbeit befindlichen FreeBSD-Tätigkeiten

Diese Mailingliste kann dazu verwendet werden, eigene Kreationen und deren Fortschritt von FreeBSD-verwandten Tätigkeiten anzukündigen. Die Nachrichten werden moderiert. Es wird vorgeschlagen, die Nachricht "An:" eine mehr themenverwandte FreeBSD-Liste zu senden und diese Liste nur in Blindkopie zu setzen. Auf

diese Weise kann ihre in Arbeit befindliche Tätigkeit auch auf der themenverwandten Liste diskutiert werden, da auf dieser Liste keine Diskussionen erlaubt sind.

Sehen Sie sich das Archiv der Liste für passende Nachrichten an.

Redaktionelle Auszüge der Nachrichten an diese Liste werden eventuell alle paar Monate auf die FreeBSD Webseite als Teil der Statusberichte [1] gestellt. Weitere Beispiele und zurückliegende Berichte können Sie auch dort finden.

freebsd-wireless
Diskussionen zum 802.11-Stack sowie zur Entwicklung von Tools und Gerätetreibern

Die Mailingliste freebsd-wireless diskutiert Themen rund um den 802.11-Stack (sys/net80211). Besprochen werden die Entwicklung von Tools und Gerätetreibern sowie auftretende Probleme, neue Funktionen sowie die Wartung der existierenden Werkzeuge und Treiber.

freebsd-xen
Diskussionen über die FreeBSD Portierung auf Xen™ - Implementierung und Verwendung

Eine Liste, die die FreeBSD Portierung auf Xen™ behandelt. Das erwartete Nachrichtenaufkommen ist klein genug, so dass es als Forum für sowohl technische Diskussionen über die Implementierung und Entwurfsdetails, als auch administrative Verteilaspekte ausgelegt ist.

freebsd-xfce
XFCE

Eine Liste, auf der Fragen zum Einsatz von XFCE unter FreeBSD diskutiert werden. Es handelt sich um eine technische Mailingliste, die sich primär an Entwickler richtet, die aktiv an der Portierung von XFCE nach FreeBSD arbeiten. Aber auch Nutzer, die einfach nur die technischen Diskussionen verfolgen wollen, sind willkommen. Diskutiert werden vor allem bei der Portierung auftretende Probleme und mögliche Lösungswege.

freebsd-zope
Zope

Ein Forum für Diskussionen darüber, wie man die Zope-Umgebung auf FreeBSD portieren kann. Dies ist eine technische Mailingliste. Sie ist für Leute gedacht, die aktiv an der Portierung von Zope auf FreeBSD arbeiten, um aufkommende Probleme oder alternative Lösungsansätze zu besprechen. Personen, die der technischen Diskussion folgen möchten, sind ebenfalls willkommen.

C.1.4. Filter der Mailinglisten

Um die Verbreitung von Spam, Viren und anderen nicht erwünschten E-Mails zu verhindern, werden auf den FreeBSD-Mailinglisten Filter eingesetzt. Dieser Abschnitt beschreibt nur einen Teil der zum Schutz der Listen eingesetzten Filter.

Auf den Mailinglisten sind nur die unten aufgeführten Anhänge erlaubt. Anhänge mit einem anderen MIME-Typ werden entfernt, bevor eine E-Mail an eine Liste verteilt wird.

- application/octet-stream
- application/pdf
- application/pgp-signature
- application/x-pkcs7-signature
- message/rfc822

[1] `http://www.freebsd.org/news/status/`

- multipart/alternative
- multipart/related
- multipart/signed
- text/html
- text/plain
- text/x-diff
- text/x-patch

Anmerkung

Einige Mailinglisten erlauben vielleicht Anhänge mit anderem MIME-Typ. Für die meisten Mailinglisten sollte die obige Aufzählung aber richtig sein.

Wenn eine E-Mail sowohl aus einer HTML-Version wie auch aus einer Text-Version besteht, wird die HTML-Version entfernt. Wenn eine E-Mail nur im HTML-Format versendet wurde, wird sie in reinen Text umgewandelt.

C.2. Usenet-News

Neben den Gruppen, die sich ausschließlich mit BSD beschäftigen, gibt es viele weitere in denen über FreeBSD diskutiert wird, oder die für FreeBSD-Benutzer wichtig sind.

C.2.1. BSD spezifische Gruppen

- comp.unix.bsd.freebsd.announce
- comp.unix.bsd.freebsd.misc
- de.comp.os.unix.bsd (deutsch)
- fr.comp.os.bsd (französisch)
- it.comp.os.bsd (italienisch)
- tw.bbs.comp.386bsd (Traditionelles Chinesisch)

C.2.2. Weitere UNIX Gruppen

- comp.unix
- comp.unix.questions
- comp.unix.admin
- comp.unix.programmer
- comp.unix.shell
- comp.unix.user-friendly
- comp.security.unix

- comp.sources.unix
- comp.unix.advocacy
- comp.unix.misc
- comp.bugs.4bsd
- comp.bugs.4bsd.ucb-fixes
- comp.unix.bsd

C.2.3. X Window System

- comp.windows.x.i386unix
- comp.windows.x
- comp.windows.x.apps
- comp.windows.x.announce
- comp.windows.x.intrinsics
- comp.windows.x.motif
- comp.windows.x.pex
- comp.emulators.ms-windows.wine

C.3. World Wide Web Server

C.3.1. Foren, Blogs und soziale Netzwerke

- Die FreeBSD Foren dienen als webbasiertes Diskussionsforum für Fragen und technische Diskussionen zu FreeBSD.
- Planet FreeBSD bietet einen gesammelten Feed aus dutzenden von Blogs, die von den FreeBSD Entwicklern geschrieben werden. Viele Entwickler nutzen dies, um schnell Aufzeichnungen darüber zu veröffentlichen, woran sie gerade arbeiten, welche neuen Erweiterungen es gibt und andere Arbeiten, die gerade im Gange sind.
- Der BSDConferences YouTube-Kanal beinhaltet eine Sammlung von qualitativ hochwertigen Videos von BSD Konferenzen aus der ganzen Welt. Dies ist eine ausgezeichnete Art und Weise, den Entwicklern beim Präsentieren von neuen Arbeiten an FreeBSD zuzuschauen.

C.3.2. Official Mirrors

Hauptserver, Armenien, Australien, Dänemark, Deutschland, Finnland, Frankreich, Großbritannien, Hong Kong, Irland, Japan, Kanada, Lettland, Litauen, Niederlande, Norwegen, Österreich, Russland, Schweden, Schweiz, Slowenien, Spanien, Südafrika, Taiwan, Tschechische Republik, USA.

(aktualisiert am: UTC)

- Hauptserver
  - http://www.FreeBSD.org/
-

Armenien

- http://www1.am.FreeBSD.org/ (IPv6)

- Australien
  - http://www.au.FreeBSD.org/
  - http://www2.au.FreeBSD.org/
- Dänemark
  - http://www.dk.FreeBSD.org/ (IPv6)
- Deutschland
  - http://www.de.FreeBSD.org/
- Finnland
  - http://www.fi.FreeBSD.org/
- Frankreich
  - http://www1.fr.FreeBSD.org/
- Großbritannien
  - http://www1.uk.FreeBSD.org/
  - http://www3.uk.FreeBSD.org/
- Hong Kong
  - http://www.hk.FreeBSD.org/
- Irland
  - http://www.ie.FreeBSD.org/
- Japan
  - http://www.jp.FreeBSD.org/www.FreeBSD.org/ (IPv6)
- Kanada
  - http://www.ca.FreeBSD.org/
  - http://www2.ca.FreeBSD.org/
- Lettland

- http://www.lv.FreeBSD.org/

- Litauen
  - http://www.lt.FreeBSD.org/

- Niederlande
  - http://www.nl.FreeBSD.org/

- Norwegen
  - http://www.no.FreeBSD.org/

- Österreich
  - http://www.at.FreeBSD.org/ (IPv6)

- Russland
  - http://www.ru.FreeBSD.org/ (IPv6)

- Schweden
  - http://www.se.FreeBSD.org/

- Schweiz
  - http://www.ch.FreeBSD.org/ (IPv6)
  - http://www2.ch.FreeBSD.org/ (IPv6)

- Slowenien
  - http://www.si.FreeBSD.org/

- Spanien
  - http://www.es.FreeBSD.org/
  - http://www2.es.FreeBSD.org/

- Südafrika
  - http://www.za.FreeBSD.org/

- Taiwan
  - http://www.tw.FreeBSD.org/
  - http://www2.tw.FreeBSD.org/

- http://www4.tw.FreeBSD.org/
- http://www5.tw.FreeBSD.org/ (IPv6)

- Tschechische Republik
  - http://www.cz.FreeBSD.org/ (IPv6)

- USA
  - http://www5.us.FreeBSD.org/ (IPv6)

C.4. E-Mail Adressen

Die folgenden Benutzergruppen stellen ihren Mitgliedern für die Arbeit an FreeBSD E-Mail-Adressen zur Verfügung. Der aufgeführte Administrator behält sich das Recht vor, die Adresse zu sperren, wenn sie missbraucht wird.

Domain	Angebot	Benutzergruppe	Administrator
ukug.uk.FreeBSD.org	nur zum Weiterleiten	<ukfreebsd@uk.FreeBSD.org>	Lee Johnston <lee@uk.FreeBSD.org>

Anhang D. OpenPGP-Schlüssel

Verwenden Sie die nachstehenden Schlüssel, wenn Sie eine Signatur überprüfen oder eine verschlüsselte E-Mail an einen Ansprechpartner oder einen Entwickler schicken wollen. Eine vollständige Liste der FreeBSD OpenPGP-Schlüssel finden Sie im Artikel PGP Keys. Den vollständigen Schlüsselring der Entwickler von FreeBSD finden Sie unter https://www.FreeBSD.org/doc/pgpkeyring.txt.

D.1. Ansprechpartner

D.1.1. Security Officer Team <security-officer@FreeBSD.org>

```
pub   rsa4096/ED67ECD65DCF6AE7 2013-09-24 [expires: 2018-01-01]
      Key fingerprint = 1CF7 FF6F ADF5 CA9F BE1B  8CB2 ED67 ECD6 5DCF 6AE7
uid                           FreeBSD Security Officer <security-officer@FreeBSD.org>
sub   rsa4096/B64357A343D9CBAE 2013-09-24 [expires: 2018-01-01]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBFJBjOYBEADuKnefrbTVFTZf9mITVx1lFAqwDHPRHZeWBr2Vq1B/Y1eKKsen
BKbK/O/CXaLuGFRn/6Ptvi9eLuWnho88qzaPU1Aa7BFRRiZlN+WrTmaDwdONJnJQ
p1LTPjqHmLVAkD7mFZe/H8Glxot62zEqY7LrEs+ZuxQ8oI51YKjhGaACvkrFMinO
09+TDey1fupVH1+yskVKQZo1zp//Hl/IrPbZKfGCxIGePQowZF7YLvl8DKPo4jI5
KO4tZ1kOPcPL2CqwhuCDy0fpUhrQZBswp6tsGx5mRJxDxfgePRBYDK4tMK+BSVsR
putIKOZ4zoBf12hYFiJ8Yd7e9cqxTiPa7AhxPbAjppiH7qJ3NJKCXOOp9DcSvrfb
ymu9cbDIPNwh/LQ1wt3T+U8QkD6a1a2kJL5+mdg03Ny+8Ej8hUyuJOEx+sxLs+JX
4TS1KRreLzxN7Ak21dNMr8361lB+Uprgi9lOBNLO31TWPABtJhIzwBOhohSqstB9
w6I2ZsPpLqUp/p9BrWlw6+UfOqNDFILZ0CqL1CyFIyrkjutXrUshqniSc/u1VbTU
RlIcufZhN3FtW1P6ktUq5ss4dqEh/QZfR1WxBYRMbKXXAN61XO8M2t44I+44DHi7
jOs1q6jrbfAli1ZGYam/5wjOJkvQ3xemP6SaDKnCKOnPHC45EAt2SEVGywARAQAB
tDdGcmVlQlNEIFNlY3VyaXR5IE9mZmljZXIgPHNlY3VyaXR5LW9mZmljZXJARnJl
ZUJTRC5vcmc+iQI9BBMBCgAnBQJSQYzmAhsDBQkIB+1BBQsJCAcDBRUKCQgLBRYC
AwEAAh4BAheAAAoJEO1n7NZdz2rnKEkQAJWJ2ctNY7vg2pqrabavfRZ4UOWrLi4A
gOMnKrsm4ozZ1mc7NVMRj0Ve8jLLHrySW5QaSmp8TcaI6twxKD8FfTOFYjBU35DU
liyRlcbZmsBk7aG561TPwaK0XnF47RyPZWKbHrO7WgiDveGx52AmBdm2VRyMBwnu
e3b5RlKnNVMMSm4RLmrolkL0SAZNAWZGG4FqFtaxPRZo7LR9fEv/NydQN91b2cR8
SnLc2F2yiVc5mq/1f/t8dMBEbNx2+NoFaqP1O+1JeGYgmA/vE9fk1oDnn1pHej8O
hoJJ9SsQEuaITvzKP9bU+5/o/UqYzAX+y8QbTthjhzpkRwjqwjuMVmp6/f/o8ivl
nzD5K1lQOP/OJAki63h5LDUC/JHYkT/XN/bbgoSNveFSGV7cdocdSpCoBaZUJ9pf
zZpqRxypRB57f7bKBCI36E42KJKJ3wo873MJeElAeo31tXi2pBvTN/Idmrl6sDCN
PWwgsIOmu4Xd2FG5lanbTsXHKebCDPh/KK51mWra5judWWFVxChsNSwRHJACBXVa
2fPsahfz4GAEVp0/VbC114m8CHrgm3nh/ZAyNjgJQN5jJ37gQjx2LFsAhW5WKK8U
0Es5YXffjLEiNOnmJ+q8IZj6Mj5lWXkbCvrqjfNTOKnzzZGws+6y4gRQkgkSY3BP
p+mpCQPjORc/iEYEEBEKAAYFAlJBjuoACgkQFdaIBMps37Jv6QCeJjxijseWZzn/
z7Cv3zSwSFMAWPwAnig7ZgzoqKqwpvnwAXsQpGSnE8K5iQIcBBABCgAGBQJSQZHe
AAoJEJLIQ0VtpqZu8r8P/jHm+xi5yMz3DVj6emMazJdXLtnnGrKTNw5xL1X10a1R
vmo+sj4J1gmL+Cy2hM6fl6r054E/BYt9GVGaIC4eYiF6DUzlcPWkwniDKfi1lNJz
NIja4qhanuGrK7EJtZXACRhUuNr2EzEm4dd3nXNaBQZv9FlIn79tk4vVho7wK7ui
IT7nseUMWDh7T0h4IVSs2LWdvP71WDx8acoyfspI35C2pKXB5GRWxnzN+wOl+V0k
Dn2fGd+nL7ZEb/c/01h6AfyYJGetCXY1omkXSzgD9KKu/RqZuxL8TMMjNN6z4SAy
MTthOHW0lTK/5h55dJYSquBQwuEAX0Z8RT8S4Nva5LKGr25IpIJuP/TxaHIgdncr
in4D0FtuG0JMOxjuzNdo2lOiMZ/lqZ75l61C68GuKAhU2Rn1toqc/NReL1yLhHoM
1o3EvovAfZmzX3sOugU2N8L+oiTnFFXezpY5Huup5KUkrX+C5EErBIVfvKjNyhhK
Fru6Jwy9z3qiGhxNUFAAzftVYhNT1lDkMNqa4jPjOrcWS6+gwVfQAo9k0p5uwPNb
Iw59RA2q/wwhZuRoai4nqN9WkgnwmWn0sS9XO87jwN3uvK0IF97MGPSXNcmAGXlx
zF3GBFHYf/bpagrvT4v+DE+gLpgfplo86oZbjDPsXGhVNu1iffC64R+vecw7r3Di
iQEcBBABCAAGBQJSRqY/AAoJEFF75hSlwe7HvwsIAJUnlLFMOBLvlBrRuxVeAO6X
8DhytdD5YlRzt866cXq6A/dw57O9qwyyDy3upJIGRy6hYlL18ngGZXv5djcw7Rch
QmvBJ9ROkmkCHLe3+fYn668nkxtgQJHWADd90MGFHkLDWa4Pbu5yJKqkTy3tqx2N
mBDEz317F6mMtyTP56QI8PVnh1p6w0McQIVctS3LOC3u4Wjbw7l3Hwof9Pl3u4BZ
```

```
L/gJz5KAozUa5TqNV4SLwtUqXBg7kipwfshXVuQekG9XfMC84GaFMqEKTExscHoF
VdSzrBKHn6VlEl1sdhcdS9aKSOsqMXB25xhBe0hOl4Ddw63j7b47XCqcyqAE5eiJ
AhwEEAEIAAYFAlJHAsIACgkQ8cUWs8g1l1OXkhAAvXUR237vXF/sZCZgG0748Dp0
eOhish/c4ODgW3JRehVWAyAlTAit/+xK6oI5xkQA+z3KO6+/bAtnDQgikAkykgpt
VeVW/6v4GGBarUTc/CTcofEpC3rsrEm1ZwPLyva3YuFFnYHATq/2Qi1a5PnSfj5C
O3fZrOgJTXsm6eNt21bH7RYF4DYi4kDNQHxtBOaEcUhcIkS1MsMz5F+/YeqOd12/
FrcIPDq8c0G3Ol+QsHFx+Y6b5Fp/HgkQem9Pzu7XkNcf7nj5UFJw+qx+BivaVYhJ
8Ugq3pXYkNkhYSy/AP/YYp7moOgpo2tY5e+fqho4pVlrHoPqWTNKJJrfYg2Mg/vP
e0nPxiCU3anmFXhfeZy87QLrA2BrO0I45StbU3uBhzT1dfNW2BIgxg+LqUZyTrZ2
qHq8TOPsnplu5Xn/UjEDQ5soTq1zDpslEjCX36R8wL3eai74HUTjstF4xq+kiXmK
bX7HhGKD9TILRjU+toOPXY0ffbS7FOUijLqOJqWEW1nBpoYoHbGfMHn2g2rNFGzz
wiLZgbL2HZsC+kDoog33s60b//A9E3yFIIiPtk668kQmiobs9Iel3RC+eOdHP8lD
gcMN/Rc/5B1S9a+wYC8VTf6KInUTq5YwC0veKbg1s+Ow7tB9ejqgxtHT7iFjR5NB
oOpVkI4UtHDpewRAW9SJAhwEEAEIAAYFAlJIEEoACgkQi+h5sChzHhzyGQ//e6o3
y+pnFTS4UWjUxFTKCtqJeqtS84jvcbXhXFGKfnXX15atLYkVoD2LcO5yvrFRNvY6
PjRkxJmLo2Lb/MpoDupRMfR1PxotFYuNYodmoHxVUun+1eIFQ5XUSiQSsIsjcUYd
EcOoZFzMfWIHZUOA1cGAtb8WL/Ql6cLcZT3fhPjEO253O8XcxKmU7sJ1sCCh3tyL
CY0dvLffA0jgxEXUYmf3DpC6p+MNkPU3EDk60OUzy4/C2HT26Lt4NR6TNcEZg6O/
lPvmD1/ATO9fAHCb4uEIkqR3VLdeg31EHND32gO/2HXc4Xp2dbV8qs+ts13w5L26
D+94PSsTwYF+85mfgu8nBhPOOn7lqWxIO/1MnOrEIVNu+K/fwh4lu8v/6PJYEYIn
LtYkDH3/LcKTsK6N/2KLbtROlHXeNKXyt0UliINteDlV9xYkn6TtzUcTrZ4Xa3HM
yN5mi+a0vptJFBPxyonMMHDAXRkLR8BexxUJqdk2aupIs0Y0Cet6Vk+8Q9bn04gl
pKjTjnnarJJsTlhrdmVobkDhbEGYB3KyrjZp2JmdYYzAbHXbdp3T7yJ4R3/7aQRg
XJIQgEHjmgFf0Wwzxs1JIN2URDZS8k2pyuI6M8ndPtJiYbwqy1Wcflz57aWYAOVf
b/G4IEsicSd1mHjYjsaMV/kp1kGrWihB/Dt79nWJAhwEEwECAAYFAlJJfnUACgkQ
cTWO1j93QHkxbA//SKb0a0wo5dTJpMp7pUL4pkCx1gR3YCZMyiJHAGnC0vHoTmxI
+6+YAU9DBFWjQk2uqqn+GW+3AxLEN08s2xYvNoxJHUB1bF43HI9lXscGmzfjDR62
cIptcWtggeMw6W66UStdFWUudwDM6WV8BTxg2LYD3upeY69GnN92HinMj90D6PMc
iQjfUdZxZAYLKEhic12dKHpWRC0PH9NIAS0EchARkZQmjyPc4trWevAyhmpqdw+H
gxh9EBH2I194SvIXVuU5Gyl/l3a/6ntEUZnitBijU3uUjRnkS5XkJfqy1MjdrJ0o
ymo8mlxOVFKV879ez10KBnE1BLe9ioylOeGQRNcyYehFE7GmzkZHbOk+Pqd1Meaf
AjNIgQxrqgh8pJ2F8Zd8pGDrYspjICGbbdR0WRNcoN4kckJruTWFQ1xr//Kfwp1b
kCQWRwYcRL/RNVVZuHGgvTiTa2wZNbWfZk3tF9cXaYHIqhYU8l7Lc1zK0Fhv2E1t
Phw4pu495RbGRAFOE14S+QmknIy+DgIkTzQ1s36vnI4SVw9zs0D4Np6d1mF1p4gi
VVrgTQnlF3poZNppCUK9Rih8s5kMnyuRruGm/Lod4jL3wcbBz4sxBkCgrc2pyU1M
SNAjM2V8c7cGLgPOqX0eVqgXJoTnlNItF07aIZyFEA6e7YeiTeXxPfU10Q2ISgQQ
EQoACgUCUk3NEAMFAXgACgkQOfuToMruuMAgxQCfScnmgUcnT0J07KNsLKLMGW/6
ffAAn2J50o8KV/wu8auCY1o6EkjpiJt/iEYEEBECAAYFAlJKlYkACgkQ20zMSyow
1ymmfwCeLqsUDHBH8JnuaJjEUYqACGWZo88An0wcNy95yGdSJtgBFXNPZQJL2gSu
iF4EEBEIAAYFAlJNSA0ACgkQUYUJaGx+XoKvBAD/bUBqzL0oZtaF7WUDXchb4yki
f0ko+zh832R2Ad0KfygBAKNEUUKOnZFLJ8GZqAXmIWktgMiWFOMSxAXDLsyionoh
iQIcBBABCAAGBQJSTYUGAAoJECC3DeE/HR5PCH4P/ic8LWEp8aJLLlOR+DSB9H3I
cES36ulQLHKmmWMc/ysR/bLhGhBqF8TM3hzvdTqj6p7zMZKThhKKVLLBxjlV2MLc
OVwhCzQow/D8EpUqQw3ufpWDYzCI7SF4nohremXjjv9FZVV80QhxLSqDfeopIBGs
ZD6v5mZn0CtT0hBXD1rowcZVo2Zdgx7/HgL4BRH19ZMiKMVdp365ZQzGlRVNTbww
fs13UTINcchA4ggbJXX5h5oUo8pbp3yXso6cMnuuawFRDu15JjQctkpaDyB0QohS
z3i5LqA912kRR1rEQjgXH8GcudfQ671FKZ+SJ7lwd+s7vdUMIfAXflCUCKMLAaFP
QB/J/ZT7FEwlO3ZeFKrWcYmkx0Af9/ieKO/ptdiOf2OX7VvE6AkReRBiqAeK9M4a
dgS1hnvs+QdPB40dTXEFRuk7+hcEqqan+ZuMhWohJlAhTHxTF8Vxl0oyNyXiXiJi
mJMTsGmvF2x+uQ/S4+7Mg8+A0oGYjwvnFC+0jWO92Ix9M3y+upxkc8KOM1/U9nq5
p7wje5MNdcCHyVTpSxvg/bDaQYopKTD6aVu94u4OlbhUXki4JnTQlwqFVkGHnpW+
BPbpQyqhY+t1QoaUWgRL+n8+WBVCqlFQF8vIoqbYGP4WxeVfylZTfSVWDoJUPKKv
bEsyhpVFj5XT70vJ866EiQIcBBABAgAGBQJSRaaeAAoJECZJ5ijF000F4jIP+weC
FBeCkY7sprDa61kp10GNF4YujiZ1QKQDgrQA9ipgv3pN+5ovC/ClzZm5baVGi+j5
zWD/blG9YZAApM/kkpAIvCPYIuQ9b+/crOUjuxyywuE2HSbaFuh66lW7Eox3NT8N
NMEl6Zry6m8RDHqTZIpwJPBiCgEcNqr/dcbtE0XgzJj94NOWSuq1URpP4wIT9aAV
Bqdj+0KQDkDk6Sqvmf59Cjt8hihvXAhOqcguKo8y262ABEO8kxwfqvRYECCE+eDE
APUEyOi/6uI0dQjQMytTWKogPIYg4wQjpG+Pa7wl7AnxOTBp4WvoS0BuCgjSYaxn
wVKHBMvxSCuDHBurLN0wqOaKSg9ib6m/Vy2vfi9ak8crXJFZ6eLrIxt73gyiozfK
Efvd6LBOJ9AeXstnubEs7ltNq9qKyW4+vR9eABmn/wABxCsHNjW+mmi8xAVhhc1K
qZC/D4vm6r8ZwrVAsmTADqcTr6A48J15FmIwcaQRQWQ4oytxTGA7rHRFVjrt3YIj
/WP62byp8s59HOKJE+mA9q7ksAvnToLfrMiNA8/18Zm4CADKUny6GLzpuKgcYwTu
cqE/zBWUszI2NrJNtaKWafdXyEAwgBxNIl1FiYF9+ntoMWlqDQROPZLYChRThJvR
nNNsT+WwcuSHSFexLl14yrPJ3MBEe7e+2Vpj9HR2iQIcBBABAgAGBQJSSFmrAAoJ
EDpFFvNRg85IHx8P/3exX3fATzNwqfININlvYjxMzuGIHdV03w2pHrOllmPX28/U
UHSQL9yRRNhzimm/9v3dvu5XHzjUzCEozoAa74DnICe8wUfju8sGmN5FKolbvSz7
```

```
VvcW4mAC5RY85zk+7luTg2wHZIIdgirTDrgPSirtYkm+qpuX/k5LAkwmYtH6gghq
v7rnYNKUChh+Ga+4yNbsdD7blWYr52UwnfT3evbgI5GqBMZEbghmqNiR2fcII6tr
NnuawH646UcucwogxPtLxLuZnslEpWiHQlAVvHlrCMoEkYqS+NRXOwZF04zTwRpL
CUlj0PxlRInvTrEpBd1KVejbkNWKK7wfyL/bF3rR9pMGWuDC32/9BfjtGgNDXJhQ
MDGntyAeQfiI3Ml5b5SA8bT5DsR/FIQDg0UDe5jjeVIEGZKunmRT/IqOLFMpZoMH
qNqWW8YrHlpN2o2c0/VqWSLzPKmocgqLwlkx5oqvn/F12xUzazGhFTFp6IXpqQVT
lkSPdDsVJuidj9ZJLMRoKfFD9tISqTocGw3suLqp8u5KZf43THWspBi4tD4IoN5r
lrLWtPnkteffyO62NZOOyg7rPUGJYlpgAMIDkXmsp58CyXqrL1/art0Ymcy5z8ea
1eUCnq/ZJJxrj+HrXuwko4fXTewf+nzSbJ2GEL/fMBkzAOKl9j5bOPAKwiD9iQIc
BBABAgAGBQJSTTdGAAoJEE2hFOXEouV/uSQP/i/yJbvVkxXlWZhk2JFhDpZaewdL
TUCkgsDeS9M7fde1Y/NbnVwSm/TtzysI6XPa5lIeXUTTlbwGiI/ZqFPDaDptUmL5
1b3cgMReW2o5zfLtnDZZHYPn8wosMFMhj2wkOXpQv7DOJBQf5MNnPHublBwY05o4
dfDBKi0GKVWl8ZkHInGvREJw7wF6ukYtnWQOIaW//qmVwokv36I2EJoooFdl7oFh
a+Pq1n3DhQAgiln6/Mz/96fn7NvYvdbQlMGluPRANVuKjfP9zQroF8BmhWQbEHZG
aLT+FsD06A/CjWlKkb3Ys/N0wDi9kQ2ez/DZhjXgBMXhJrdPmeTEHrnX701Am+2D
CSpz7bbkOayILC5gx8DWq4hjGu5JtGcpJE4AsN69dXn4r/w8IUecoGZG/CjVQyAc
RxsIc9nOJmzbJkQGrP8A26Io0/xrwOjU2gGkYR+EaR3o9Qa8tY/uZpYb3t3yh+b0
Pqn8pLOMnpo16uJni3/tIY/kiqBnGF53yVLjlekwfORXBRFZ3GNroe21OXrfbHQ7
9BytMjTBsQahfaMdFZF1QINvENdJ+PQhhx7R2g80yxj67oa0F/W0zdqYDbYnM2bt
Mw89mv/q1f0xmdtaTJXz6ZpLPY3MtDWCJ/LcKDKUQqnyS7XilD95HdFnck9GPKQe
F/mgs5YlQeQZg8cSiQEcBBABAgAGBQJTMetBAAoJEEE5xLeoRUEkcGgIAL9ZRsk/
BMWQf4tK9RTY82bihv5T5XL5ybqnXuuPMC+E2IHDR1hGE9WcFr237nyfVxdnlBKn
IUbPrghdeGAWg6ki2IWOjgy1Q46M+P69yroc6KCa3V6LdM5L/CCk5Sr7L1LbvZ9g
Mj4AkN0xGhy3NNZGsomiXZWmBOiOQ4EJwlIwtFgMCKc4KmRD/h+f+/opMW9782bN
L6txp3tk5MOUXa+Xk1gy8MzGtowL2Q+P4zxa94NSVYQ6picYFvjWGtzUJ5izdyb/
se9wLIT8p0iyPrADP+P93EjkUrH4Im4OuY9ieKc3hFsnLhnI5VLpPSy29xXCi3C5
t72Nl5dU+/JJrtyJAZwEEwECAAYFAlQUczoACgkQjw7rxHtHFslqFAwAka5jXdrV
IGHT/n2YWzGTfgy5+bJfMZXUa6fuo+zzvB4hS3MH3YMpHRjwUrpkAjTh3dFkziVU
Ns7j4+7x5uEOE9Y1Baj6DTzEAxZNwtSeCYzCA0FZ/ufuUxGfZElcrU7AN6/ep/lm
gsE3+5tak8VYJxDjgu56uEIz449Lscj4G3FO6eXhCiiWIb7+y0a9m6cZ3yE7k8fo
TVObr8xdhGzw4+YEx3/4usD89GIKwLN3LZFjndqdPnYidneJ9NCrGH9g4+DRlOkw
8LGlSFxcNLqeVBDBS2bw1G1ZSsd0NH+8deeAf9rEsmOT4CQKOWdgTnkK8O9erwvi
dUvsANlOypecGbHMo+NoS6kjR/CwPF8vDnwhEpy3N+VRZGhSD77D4LUWKBLQDisv
6HuyALmE02Lq9v5cK8fWy5cehS8hvAdn/FU0GOvPg6JowBZkyvqbOQDHIO3buAr6
NtnPdhhOd/eCOOkCPgcTAdwqWrX+l7D4SImMYjFAe9GWONAkkcRVMo6jiQIcBBAB
AgAGBQJUdXNxAAoJECZwmtY/E3EPJ80P/1AuTYo48UmvVkLd443cvaUpItzLUfrW
4q24KjiCTT63ETf1+V8RZCRreqt3mFJnZI0n8X+hSLAIPdJrJ1xtIKDoEWbQ1U8j
ClFq4FtUaqSHkQIWvW2VzGgVz2MvPTWK0EbWHdD9vhtotnYrq4H+T5cBuSyrW9Zu
Gct6zsZbC/O/iyiKQg3Kz6PtCiSPP3AHNH3ok1Nh0QsSQl1ggGp1J4gr9A0/Kcf7
lQ+/X0G7kHVxQnKzzuYI7XsV25Mp3oBsioQB/9aHt/JVfjrKpH0FtdTUEUcMfJqe
TMxW6xXHvslOIj3iXj8frSMYuJaQXVjTwu2yhY2oZfnI+JG0Gc9TA20lijhfyOW7
2wE/qdFW3I7CY/3hBYa63IwNGUK/t052OmOZmhrzKADvWc6lCGG02M7fY/Q+IgOT
PS4+5A1fs708Ds7qHj/TkltVmJftaCkBZCTWqvQ2XxStzYnHVojNxsTCqhIOZLM8
+/SSUMzox4G2d+z4WTlok+HLwcf4h5iAOQg2HAzGO84bamwOE/r+hB19YV07dGND
h/7TI25Slhk46CbuLajnAiIfg4UnbMpUZt+ZC+tdCuKsFQcRl7cUXqkJ3gKAiFOl
8Dly72t5gYwYUEZDkuKisAztRMCvdI1bg8j8ALFjbtd5cYbrtyLYVbg5Nm+mawXh
/U5lqcrjWdbFiQIcBBABAgAGBQJUdg4iAAoJEGJ6sNnqQ9eZRyIP/1geWFuerAtS
jO9ew9bhqC6oCVBi7R/DNT9WLNXvV5h3DYzGXnlhoEHdBzF3G4RmC6RaxZcjTQaI
lYF0qGtvZXWsQG4W6UIT58E6vyNyOj1Ugl4Siqvo1L7IxCt0Bdti0sQ082P44B2K
bSWqN8zMww3lMjnnmrRIb+PcC8PQgLZU5twosEOl/MuuDXRTAgPa4jXuKY17V/6K
NPND4d/rnsENr3+YXlG7/pdAgT9CAdNfFBKSRAUHQ454QzGaJln54FAs5INXf6iv
phfbRQp5on7MyShFNQr5AIeF5SNg/ktBlvrHItCdwM/FnOk5fg8EM3eqZHNC8HE2
SxGKa95o8QcbI2E/0iPJqzLWlmLAxaaV53Ei4RzVkGzPfS0hWFuP/NVg2FYqUTZK
Ie5Btsqd9rPvCTqywjGMKcQUIVK/aiqcDV1J7SewjxuIG4+4eaTNiQgVZspqGCbH
FHdssU/oedCIURRV2vMCCWaFEKR94vIK4IbXF07AVAZPsO1itZj6PWaFZ1zwLbWt
+VmgkM8Pj5L7xy+vX/bGQRO75JYrLYP1a9h/iG/Am0ezZQrtjTPtV07hLBQDOmpt
s3BhEsl9VCH2GkqBhsHjxhYM9cnQqMCMSp5fERRqphxyCoNcBdLHurikt34XMuq1
otgC2RQoYGsIdQlYX2dxIQhDbij7OGrRiQIcBBMBAgAGBQJUFHNAAAoJEDk/yxUg
Q+mJ4NwP/1gH4LefqQu+pbXAD6zezvM7r4dLca2TeFMCWSIRpRdtMqiavsrBtubp
kInXup616EcEY1nKi+mNiHYZx7TiUxdlLljrkr0HVtp3MD/AgBoal+J1muESe5Yb
0frp+NwJOLikUBG1v2cY2mZgIAkFvbfwVFCtJmwGL8nLyZrG1QRy242I179lNFCA
1xZu+9vKbakwnn4rqwT8ihft8o2POY2cq/MHs0XNmalUhF9Emc6sNR0vXkDeBKAA
gk+3lcbabqSni6IOpruX1XwtfnlIEqZVU49unNYbvylh4NTl2vjawsXAec+tbVQp
aFWvwru+07kC8BwOKb83iBHDLQC+oE6c1CdkbyfmQ+aH/OJs0cyGqJGeh4Q0Pfk
RSd44Ew3l/rzuHWjw++/JpfznK5mhVObpmWd3HH77gwm+FNeo5C60tkPtMfVfPqK
PbBTrzzdN1l26VloqFcRzXAnIKMqEP4J1Jd4l5awopqeBfRwVx4+XVV0y2qfvP/6
```

```
DyKWk6D97p7jrB6yuMoYbKvJKoxx75SxiGMv4gubj22iqIp8tJarrbBONdnhZCcx
LfDMcVJDSzI2LmDk5SXvNycNyxLCVvda6upMDB9O4wDE1EJJnsvkhudbwAdYEYCW
8CnF3toHcP1bGRiJGJ6Qrl11NPCdCjOmbq9KSxfkadBQ93uXo56QiQIiBBMBCgAM
BQJTd5xqBYMHhh+AAAoJEGwc0Sh9sBEAfBgP/ieZTSvyMwnOZOPNlQYnhkhaZRHP
i5fzOMzbwd+hC/3mi2U8mZOYXvTeN6+JiWJ7s+4UB0+JoOwwMkkNGYWygMF0OUl+
03FJB9cDIxFW5n3rjjbwX2RLcbx2ATQnNHRSsZdXWg1jTbzoRpOAL9ZhoyWJtRYz
fCd+r5JZrd59zGgc70aDAjF77PVA5L6lZXzTH4U4hlQzF8ugAmtNqTEfEhKRo5pt
ecu6Slf36OLc1L6Coc4amU3fMCPXP6IK5aMBPwzfXahAylITvxjbujIkh/y3KifL
cgsgc6a5y24+OBo02RzCnB1QB+alr6312b3FMrixsev2RfyPzWxfN8eE8JElobbz
4sPd5SgQ3P+iF+g9E4fTnXhk5f4u+wU5PtiWXxzWy6EYz0hGgE4Dz/uQ2fcRBASO
xMJQvPAeFM59SVTJGiFRzeNY6H/zWeC8DTE9jKbzhZ8kIzxyr9iTd7XJhp5pCVez
zkG7R9xDALq3ySMO0s7cWNB1V8Ne0YwNPZlStCpW6kW8nfC4qmNorukPcVk8tRYf
Gy+ebrWdXphdhLRZB5NpQ4ECG8kOIP/1bSRNvMs4WHU1C+hkOn8vcfOZDMO8zPro
SCNnHB0MUbXENfv4+ZXM6I39fAHohQlHw2LlqibeJHr5lbIukGQ6v8qdo5xdaoel
JnXUSVN4XvroE+uZiQJFBBABCgAvBQJTd6AZKBpodHRwczovL3BhZXBzLmN4L3Bn
cC9zaWduaW5nLXBvbGljeS5hc2MACgkQu14sRioPqLA7cQ/+NvSnh6fW7Gf89uy9
l4+/8hjGm0REFQf0LlYdiqf1pJ9N6Vf4MdhEFZs/2bvOgitSZzyScvxkAuv0LXE4
xPxOnwVYQ/VuxLSOBdTjU2srdnHrHaQxos16WLq85C2NsCSZNLOCXalMZk3XD0FH
HrcyWGfyiX4vr0tn+4G70FWbsfrK1Epmx3v+nCpCPmgBjdLRy9iU6uUjWB0w/ZVE
eD5MNAWyWumLJz32gpEQFSpELcviBoYxec8pIzlfV0db5ydJGZlsWM5W/KOy1ZFm
dpCfsl/hPGbBEtEeEf0mszchZDGtwaSpo0oiZj0LX6kSUTsp5GhjeTtntu2Hk9oq
b+u4TtAJbKHaYovJn2cySmWyE7Hqvvh2Lo+uxwm9RjKRNbtYBylZnV6QFaeMejen
RFwLdtdiil1UmXhV6MUHNIIZ0oJZ1zo+GkZKWQdoFpzayrWpfkAC+x2ovV106RYM
BAEcGg12ZZ1RcCytM/67efGG9KxjukARycsv1pU6Cf6l0yjqOikM3pnxWfdLvybU
9E4U7THfJ0sfXfs2U7d7LAX8WfWru1I90ZmFBLO5Fm3WMAOplJmdaSoNJ22IJrZb
StCDb7GynBD9x/qUGrRfIxKtzxGZghor5xHWxTtn6hLgxvoF5cDmN8g6dI0snbXK
DljubF4feV9MBwiGwpjeG/71PcKJARwEEAEIAAYFAlWTlaIACgkQogW5MOpw+irQ
+gf/TRWhT+XFhokH4E3v+J9lTqhQ+IBjVfYVZm1nzBxTkvrEt48i3VBuJjp2Q5H+
cnRr2VE76IVNsvt8liUe9GF+1tylVA7qDGDMOlqkGjPVfD1viGRgtrGBJFg3oVr6
uyjKUyhzELQPkGU+lfbhXLVE3oMyhLxf3xUd/TvGXEeaqMoPgNFGiwfjtSX6oxas
HEu3HFDO2EqM45dtjbHoj98gvDSb4ReA2ZknD/gYnNt8cMdHnv/VieeCpSDxiD6D
qljpYSPUXjz0kh/LwcVZaMm+nD5BzKcXu2lD8A0fy6hVKSm2tGyKzfWr0oZw47e2
eVMBPG/l4YB2H1Im8PPsuSAeQbkCDQRSQYzmARAAtqDVVjJvadVMDJipe7K1POK4
QtcFswiKYAwc0JOkOeM0tDirorP869gdHtkuKr3fEuW0rtId5OeAjSCI9NIihX0r
0A2iJ1PrdhhleoV7CF0u0DxTVAo/Z9HSmuQWoTz3zr01XXyb7pSzJb8oGMLGFuQ4
X1yNRZw/0daGN3jYmxRWLJj+/vlUazvzt8L24JdJJ4PEZ3TNTOOophZDjnzxGMR2
6d9Fb3MV9kCcBIpIv3e1I8IKJyigmlOUqGRCmuv6CMADM0NWEGRBkAUg+YjP7C25
QR7DtFLSyorpJ2QXUax2Bwg8F2b1+54pfXbQO28nYRszy0/ySirYjahrT+LiKOgK
N5HLTn6vIb/KZgmaPR4F1cVJYPjlxDibu26kRIlHBIZuYJ9diZSpm/ump4ZXy5R8
41NjoGZBpIFsfm4or02nLuxEof6khy2K1l9WO8U9AjG05azNQhDGijv4GB4KXfnL
xDW51q8PZWmPxwBEi7mQbh/d2DyqzVLnIQiUWMk4OOCB3VEHWzS0sMOf9NBHWOC8
jhlHak9zzB3qNsJ6x1DXIHC3f4D8owFAqy6z2BVkKQys7mxXzciTvyYIqrb5ynRt
sLL2GMdn1NeMFziB51yMak9BDMGGymXObY3Gmg22gFwnb+ZBx+rNAGt4R3ngk+/C
0jeXHyGQNns6wwoCyuMAEQEAAYkCJQQYAQoADwUCUkGM5gIbDAUJCAftQQAKCRDt
Z+zWXc9q5+HLD/94Jdl4HSb2bA6N1k+Snajvy7C2xCS6GpOMIkDaIo+Aowe85ixc
JeqiNM4lBr8OGMqAe2z0cMs4BvPHudNmN/4ceBsxwUnmcCr1hJiEaQr4eAR/lDC/
pz6gvkCndDKSeOvg5FiiXIqf6sTXpMu4euabQO4485obSBaF7/3zOUfO3Rtadw0h
zEcv/XaiCToA8fwGFNjGhAml/O7uaPmAI3rel4HMEHNxIgqARMiYNmFP8nVgJi7O
qz6rv65/E9shtNVQiHBgEXGZf/lsrxBsQJsgG2Vj+ggkDZPX5AfT7KCV4UrHtM+d
X3yKbwiWwosPwKmgzYxR5qX6JZyAr+72Zs5/eb56NGEGUJYKLmWJd85zCQHTZirF
sDIuZs87oYdrDYmFrwoWmoxZqSLeBCNbURu/BbO4nnJWzr3fwlqWO6LPO7rCafUK
6/mdYPJt4CpmDe3oGntdP/UVA10ZQ7qWYHTzNK2heBm4NHY0QMBXYoAE5bHMCDnN
YVN4QRVxUyjsXKfgjOyi0jZHX+9/CHihIXaCqOI4MdLwRx7dUwhoKOt519/Wlh71
w0qvg9kVt63A7Etyr7xj/IPpEGURDfD/EhXXn0offWL8+BKcLYp1Tbp9gJe3Ab/F
3VOWuafMlpON+Ii4YEem56Al/Ei8sDA+BN7cpw7o5Xf+HAG70CdcRDn7Vg==
=mLLk
-----END PGP PUBLIC KEY BLOCK-----
```

D.1.2. Security Team Secretary <secteam-secretary@FreeBSD.org>

```
pub   4096R/3CB2EAFCC3D6C666 2013-09-24 [expires: 2018-01-01]
      Key fingerprint = FA97 AA04 4DF9 0969 D5EF  4ADA 3CB2 EAFC C3D6 C666
uid                          FreeBSD Security Team Secretary <secteam-secretary@FreeBSD.↺
org>
sub   4096R/509B26612335EB65 2013-09-24 [expires: 2018-01-01]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBFJBjIIBEADadvvpXSkdnBOGV2xcsFwBBcSwAdryWuLk6v2VxjwsPcY6Lwqz
NAZr2Ox1BaSgX7106Psa6v9si8nxoOtMc5BCM/ps/fmedFU48YtqOTGF+utxvACg
Ou6SKintEMUa1eoPcww1jzDZ3mxx49bQaNAJLjVxeiAZoYHe9loTe1fxsprCONnx
Era1hrI+YA2KjMWDORcwa0sSXRCI3V+b4PUnbMUOQa3fFVUriM4QjjUBU6hW0Ub0
GDPcZq45nd7PoPPtb3/EauaYfk/zdx8Xt0OmuKTi9/vMkvB09AEUyShbyzoebaKH
dKtXlzyAPCZoH9dihFM67rhUg4umckFLc8vc5P2tNblwYrnhgL8ymUaOIjZB/fOi
Z2OZLVCiDeHNjjK3VZ6jLAiPyiYTG1Hrk9E8NaZDeUgIb9X/K06JXVBQIKNSGfX5
LLp/j2wr+Kbg3QtEBkcStlUGBOzfcbhKpE2nySnuIyspfDb/6JbhD/qYqMJerX0T
d5ekkJ1tXtM6aX2iTXgZ8cqv+5gyouEF5akrkLi1ySgZetQfjm+zhy/1x/NjGd0u
35QbUye7sTbfSimwzCXKIIpy06zIO4iNA0P/vgG4v7ydjMvXsW8FRULSecDT19Gq
xOZGfSPVrSRSAhgNxHzwUivxJbr05NNdwhJSbx9m57naXouLfvVPAMeJYwARAQAB
tD9GcmVlQlNEIFNlY3VyaXR5IFRlYW0gU2VjcmV0YXJ5IDxzZWN0ZWFtLXNlY3Jl
dGFyeUBGcmVlQlNELm9yZz6JAj0EEwEKACcFAlJBjIICGwMFCQgH7b8FCwkIBwMF
FQoJCAsFFgIDAQACHgECF4AACgkQPLLq/MPWxmYt8Q/+IfFhPIbqglh4rwFzgR58
8YonMZcq+5Op3qiUBh6tE6yRz6VEqBqTahyCQGIk4xGzrHSIOIj2e6gEk5a4zYtf
0jNJprk3pxu2Og05USJmd8lPSbyBF20FVm5W0dhWMKHagL5dGS8zInlwRYxr6mMi
UuJjj+2Hm3PoUNGAwL1SH2BVOeAeudtzu80vAlbRlujYVmjIDn/dWVjqnWgEBNHT
SD+WpA3yW4mBJyxWil0sAJQbTlt5EM/XPORVZ2tvETxJIrXea/Sda9mFwvJ02pJn
gHi6TGyOYydmbu0ob9Ma9AvUrRlxv8V9eN7eZUtvNa6n+IT8WEJj2+snJlO4SpHL
D3Z+l7zwfYeM8FOdzGZdVFgxeyBU7t3AnPjYfHmoeqgLcCO0nJDKq/98ohz5T9i
FbNR/vtLaEiYFBeX3C9Ee96pP6BU26BXhw+dRSnFeyIhD+4g+/AZ0XJ1CPF19D+5
z0ojanJkh7lZn4JL+V6+mF1eOExiGrydIiiSXDA/p5FhavMMu8Om4S0sn5iaQ2aX
wRUv2SUKhbHDqhIILLeQKlB3X26obx1Vg0nRhy47qNQn/xc9oSWLAQSVOgsShQeC
6DSzrKIBdKB3V8uWOmuM7lWAoCP53bDRW+XIOu9wfpSaXN2VTyqzU7zpTq5BHX1a
+XRw8KNHZGnCSAOCofZWnKyJAhwEEAEKAAYFAlJBjYgACgkQ7Wfs1l3PaudFcQ//
UiM7EXsIHLwHxez32TzA/0uNMPWFHQN4Ezzg4PKB6Cc4amva5qbgbhoeCPuP+XPI
2ELfRviAHbmyZ/zIgqplDC4nmyisMoKlpK0Yo1w4qbix9EVVZr2ztL8F43qN3Xe/
NUSMTBgt/Jio7l5lYyhuVS3JQCfDlYGbq6NPk0xfYoYOMOZASoPhEquCxM5D4D0Z
3J3CBeAjyVzdF37HUw9rVQe2IRlxGn1YAyMb5EpR2Ij612GFad8c/5ikzDh5q6JD
tB9ApdvLkr0czTBucDljChSpFJ7ENPjAgZuH9N5Dmx2rRUj2mdBmi7HKqxAN9Kdm
+pg/6vZ3vM18rBlXmw1poQdc3srAL+6MHmIfHHrq49oksLyHwyeL8T6BO4d4nTZU
xObP7PLAeWrdrd1Sb3EWlZJ9HB/m2UL9w9Om1c6cb6X2DoCzQAStVypAE6SQCMBK
pxkWRj90L41BS62snja+BlZTELuuLTHULRkWqS3fFkUxlDSMUn96QksWlwZLcxCv
hKxJXOX+pHAiUuMIImaPQ0TBDBWWf5d8zOQlNPsyhSGFR5Skwzlg+m9ErQ+jy7Uz
UmNCNztlYgRKeckXuvr73seoKoNXHrn7vWQ6qB1IRURj2bfphsqlmYuITmcBhfFS
Dw0fdYXSDXrmG9wad98g49g4HwCJhPAl0j55f93gHLGIRgQQEQoABgUCUkGO5gAK
CRAV1ogEymzfsol4AKCI7rOnptuoXgwYx2Z9HkUKuugSRwCgkyW9pxa5EovDijEF
j1jG/cdxTOaJAhwEEAEKAAYFAlJBkdUACgkQkshDRW2mpm6aLxAAzpWNHMZVFt7e
wQnCJnf/FMLTjduGTEhVFnVCkEtI+YKarveE6pclqKJfSRFDxruZ6PHGG2CDfMig
J6mdDdmXCkN//TbIlRGowVgsxpIRg4jQVh4S3D0Nz50h+Zb7CHbjp6WAPVoWZz7b
Myp+pN7qx/miJJwEiw22Eet4Hjj1QymKwjWyY146V928BV/wDBS/xiwfg3xIVPZr
RqtiOGN/AGpMGeGQKKplkeITY7AXiAd+mL4H/eNf8b+o0Ce2Z9oSxSsGPF3DzMTL
kIX7sWD3rjy3Xe2BM20stIDrJS2a1fbnIwFvqszS3Z3sF5bLc6W0iyPJdtbQ0pt6
nekRl9nboAdUs0R+n/6QNYBkj4AcSh3jpZKe82NwnD/6WyzHWtC0SDRTVkcQWXPW
EaWLmv8VqfzdBiw6aLcxlmXQSAr0cUA6zo6/bMQZosKwiCfGl3tR4Pbwgvbyjoii
pF+ZXfz7rWWUqZ2C79hy3YTytwIlVMOnp3MyOV+9ubOsFhLuRDxAksIMaRTsO7ii
5J4z1d+jzWMW4g1B50CoQ8W+FyAfVp/8qGwzvGN7wxN8P1iR+DZjtpCt7J+Xb9Pt
L+lRKSO/aOgOfDksyt2fEKY4yEWdzq9A3VkRo1HCdUQY6SJ/qt7IyQHumxvL90F6
vbB3edrR/fVGeJsz4vE10hzy7kI1QT65Ag0EUkGMggEQAMTsvyKEdUsgEehymKz9
MRn9wiwfHEX5CLmpJAvnX9MITgcsTX8MKiPyrTBnyY/QzA0rh+yyhzkY/y55yxMP
INdpL5xgJCS1SHyJK85HOdN77uKDCkwHfphlWYGlBPuaXyxkiWYXJTVUggSjuO4b
jeKwDqFl/4Xc0XeZNgWVjqHtKF91wwgdXXgAzUL1/nwN3IglxiIR31y10GQdOQEG
4T3ufx6gv73+qbFc0RzgZUQiJykQ3tZK1+Gw6aDirgjQYOc90o2Je0RJHjdObyZQ
aQc4PTZ2DC7CElFEt2EHJCXLyP/taeLq+IdpKe6sLPckwakqtbqwunWVoPTbgkxo
Q1eCMzgrkRu23B2TJaY9zbZAFP3cpL65vQAVJVQISqJvDL8K5hvAWJ3vi92qfBcz
jqydAcbhjkzJUI9t44v63cIXTI0+QyqTQhqkvEJhHZkbb8MYoimebDVxFVtQ3I1p
EynOYPfn4IMvaItLFbkgZpR/zjHYau5snErR9NC4AOIfNFpxM+fFFJQ7W88JP3cG
JLl9dcRGERq28PDU/CTDH9rlk1kZ0xzpRDkJijKDnFIxT2ajijVOZx7l2jPL1njx
s4xa1jK0/39kh6XnrCgK49WQsJM5IflVR2JAi8BLi2q/e0NQG2pgn0QL695Sqbbp
NbrrJGRcRJD9sUkQTpMsLlQTABEBAAGJAiUEGAEKAA8FAlJBjIICGwwFCQgH7b8A
CgkQPLLq/MPWxmZAew//et/LToMVR3q6/qP/pf9ob/QwQ3MgejkC0DY3Md7JBRl/
6GWfySYnO0Vm5IoJofcv1hbhc/y3OeZTvK4s+BOQsNokYe34mCxZG4dypNaepkQi
x0mLujeU/n4Y0p0LTLjhGLVdKina2dM9HmllgYr4KumT58g6eGjxs2oZD6z5ty0L
```

```
viU5tx3lz3o0c3I9soH2RN2zNHVjXNW0EvWJwFLxFeLJbk/Y3UY1/kXCtcyMzLua
S5L5012eUOEvaZr5iYDKjy+wOxY4SUCNYf0GPmSej8CBbwHOF2XCwXytSzm6hNb3
5TRgCGbOSFTIy9MxfV5lpddQcdzijmuFSl8LySkL2yuJxjlI7uKNDN+NlfODIPMg
rdH0hBSyKci6Uz7Nz/Up3qdE+aISq68k+Hk1fiKJG1UcBRJidheds29FCzj3hoyZ
VDmf6OL60hL0YI1/4GjIkJyetlPzjMp8J7K3GweOUkfHcFihYZlbiMe7z+oIWEc7
0fNScrAGF/+JN3L6mjXKB6Pv+ER5ztzpfuhBJ/j7AV5BaNMmDXAVO4aTphWl7Dje
iecENuGTpkK8Ugv5cMJc4QJaWDkj/9sACc0EFgigPo68KjegvKg5R8jUPwb8E7T6
lIjBtlclVhaUrE2uLx/yTz2Apbm+GAmD8M0dQ7IYsOFlZNBW9zjgLLCtWDW+p1A=
=5gJ7
-----END PGP PUBLIC KEY BLOCK-----
```

D.1.3. Core Team Secretary <core-secretary@FreeBSD.org>

```
pub   rsa4096/36A7C05FE1ECF9BB 2014-07-09 [expires: 2016-07-08]
      Key fingerprint = C07B F5E3 10AE 64BF 6120  B0F6 36A7 C05F E1EC F9BB
uid                           Core Secretary <core-secretary@freebsd.org>
uid                           FreeBSD Core Team Secretary <core-secretary@freebsd.org>
sub   rsa4096/7B5150C8D7CE5D02 2014-07-09 [expires: 2016-07-08]
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBFO9HvEBEADRfuWeoNUwib7ZjNmhg0Kt1kjiGEEosf3O2yMDfYuAXt4De6qK
S4KECe5+vZH2T8g+zmNLl/7JxdqHiWj9cnoZ6T3bqKh7w7pW7QzC/Q2k4mZsQkGl
xzhStHvaHSPKw5808TME0d3ewAfs0dQkDuA0eari0HipCbOVzqHUMTIROr/syPXs
jHxb2bj0KVzzq7wgy+vF4Cv25VzaAPBVgPv3HAoO/gLOr4SnXqBCw2vgprWx335t
QX1JslWlsUDmwwq40q4+eMnSFPZ0ing1DgfhMb+Dnrl6Rbxhb0pwPhbwubppUKfe
W6owOrTuUbATVoAhsfNySmUWQKc2p9w/8uFV/jJj9HOSgIMKrNONvqekPrjWOQn9
/lcQtGhldWmtPbMogOfaQisBEn1XjMZ3VEOagQxIe/6LDjU7GGoYvSdwf8Z0wXUY
/qDntPwudjJA4wQid1Tzf53gpUjr0tYq7aclpiBGs3F5EOs4HMXq5/xlwRGtBDHY
i9RNAlbRSfSD2s1nGsfsImPowlpjtLa+3PqYs/cRLGDu51DsgV/p/CqtAyebG+9O
WsF0Ydt4Q62jEuU8HY7SOj+AuKJVdUkyAZGk5vkPvsKzjdZUqRslurme7d3LqKai
FjBGj8UyId/IomDCjth3baGc/Y4e+JKyx1XDXgFY2HoQ2KzEoANrizjy5QARAQAB
tDhGcmVlQlNEIENvcmUgVGVhbSBTZWNyZXRhcnkgPGNvcmUtc2VjcmV0YXJ5QGZy
ZWVic2Qub3JnPokCPQQTAQoAJwUCU70sxwIbAwUJAeEzgAULCQgHAwUVCgkICwUW
AwIBAAIeAQIXgAAKCRA2p8Bf4ez5u2R4D/4o7Qb/hFz3XFRiBLtcsr+v5CS1fYgk
VN2Xp40dOTtt+Xo257+1oQDsS6McNXT7XSAFOTrFW+XID8GBnY6+ZpRYzontMCJd
25qGHzGBs2aFA/ROfhdvpOkhngxsYG89+IOUjS6SRNqG85Iro6eI7ZLi8sznEhZd
lH1jEWHiJhEubOvcedWS1zSHpOZwNIpyQV5d2O3/EhAOJRWgy8VmegeQLGg0zqdL
r9MKfg8/OwNQlysjrsA5MRnZMZAm6kEO381zYg8+Rwe+HtspzL/1cu4M7k5aBw0m
a8nShgXTqfgX+pwW2zBBL8c42p5D3Jh4Oe7KQ1KEvysZJGWO5ZjR9p4KLu49X2Yl
9cuD3Ii6+Nz9/cUlEGaMm4I8N3jpSNH+GCj7k1Wa9KWMh2ZxmVO5AML8h4g+A7aT
qNasKpG4XLXpcZ38sOw2/h6G2kwnrCel1ViCe0Z4pxeW95bxREqs6pkkym7Wz4sy
F5Mkba0uwFhWg5zBANBteXfOS/WU3fLEWeHXvZu9t44vyvP84qqZfNzBhz9puDVi
kPAY4uiZK4BTDkYZZr3YNKIIwEQ/rjcJSDKZKGJ6dsHKBSi0oGxFZpCE+YMH/Zqq
hTQHgXbMKWLgJ7xxRYJS/apvgnTd1Dc7zc5J7IT7XCdML0+Qj8eD48DwyMu+uAAq
/048iyhu/R8RP4kCHAQQAQoABgUCU70tAgAKCRADb2ye5/Oevxk0D/4wEJarg3nO
/B8ks+s7DadvpmNZrNSGeeR4qSEeZlH1ye30xD1HDE+ekJXC97Td4mKWKTrbRVEX
F1y1aBCe8tL5BbsRDmwzKwpv3OqQ1rhowlAKZD8hdBEyEn15ciaoRisFKpWAv+mm
mZeTnG5sYOo9BF1v36WrX+9AOvy7yduVcZE3kLNoivS+8bXe5ULj6uMoyo/eC5uk
q35f1p1qBzrplMc+vs7z8Npc7YFP2ORhR/mVg0rYFLkPVdPV3uFkrkb/Z2SVu02B
IFWEeJojEBEbFNfziHUaQ8Z7ZdtkZ9OIS1Z0Wuy64xjgORWVcTlRe4AwZKVC8DJs
Fs9Gk9gnsPmfLgM46pcKaac445w9uUQzKLxdCBKLPxMV/JHBRrMghyahevwnpsuB
Ok+ldsCNqjFaeAnVAmv3YtTbZ1JdorupeFa65JyjQon7l1JwWD67P7N0xrn1gxJ+
9uGUeHwGAvDF3IRr5ECISKAW92NfLYuNpyv6re/bt6A+FxaSA+E8P7Z+I4Gl5l37
czz+9ciYipiusweDuxectOlEnh13/WKXuUb68v6ZzbwGjkW8wunavhAN9fWxqqgF
W0tKtmpAOHFzmNOw5WGPfGdKOzc+g5ZqRt3CREVHj/uizNJsHzfadkTqYum+kQTa
y4tSgGH0S3EAiHGsEy4YZ4M/nxdIyakmsIkBHAQQAQIABgUCU701xQAKCRBNWP3N
LKSXdmOGCADEySzz4Q6wKsx/gLiAyhYNbEJbiv1MirxhjIYGP9MqNpxxI1+Q3kuj
01K6ELIMuAhehoQOgU4AssJQxu7q78+hz2O7s+V0Syl+pvEOl2zUCgAmOYfle+BQ
75ZEEiINBuh6SOXBVLhfNp90FZ55KUSW4EeyoT+A4nRGHRgCTEfZ5WHi3lGlaLQd
Z9viLfNKA/DxrLWww+joTPIEhc3eU1mgDrcmfxo/L95EmTyUa5BtE0WuLwQEaY8H
J3eBgA9Y130ubuzzY4jGl4SCNedMzeIroHw2Bogd3V+E5aFtGd8gZUjXXr8rM6yX
PpttP2Hc8Bie2YXI2NffqwVqpL0dxo3uiQIcBBABAgAGBQJTvTYRAAoJEMATMJ1t
```

```
fkRccM4P/Rbg0W6l4KPFUvyKcUE6odRwoXExRGHdG9qW8Vf6xtW5eXUX/AZoCnXD
f3yWWttxgzN1e8iNRh0aYfuNSFTuHcHut/xw6GZ1yqASbuDmGWQ6uTb0yHYQcwQ5
ioaRaZZo5cpnSs0qZUpnrSzdUzyVmlKsD+1ut0/Z8yM8WGRyhplWX0dfXKnUUxJG
yh4GQc2dQon1vrsiuDTD2hr3EVues7le4WU+csegZTGPgPjhTSH6ZNFdDs4Y5KPi
unjXx+X6avPKPSJCnC9YlPMkI0RcokVLJW+K3+4QnbqU8m2MpZWVaaOo5s9PCx1I
208EHQ77A7EAFYNFrPZmtSV5X3BhU2rYNoRu0fpsNqJC34Oi7JdZdplPoO7FHRAC
AyQJyv8KUG8VVzK7m6Kt/Okq7LBc8RuvLQpUHSv1Z19fQvFgTegM5Pcpp3/ful/H
QIIc7XRElM57e+t8kbsoRpOlaKa3Okl3KisXdksWB4Fu6XdVArY/jIIQGs6dCpYa
jhRZcjkjHUAPvY/OqD2mBSwj0YwP0RoMVvFHMP1cgB3gjaB37A+DJeiKEXTWzGe1
fKC1TxCcOUZsrcqXnUyy23lKV9CXC7za3eB23dPIfWzJnD9BsVgYsemRVJx8r3Sv
QIL5zjKVDRAuV3M/HbCtSOgO68MExC0TFEl/8LPIMW8oVCY4+iqliQIcBBABCgAG
BQJTvUdSAAoJEO1n7NZdz2rnXxsP/R8WHku1nxjELqdM9M72JLD8UBlaAIwLStDy
hnTvLa0GO6eN0r2eJ1+tG8mKB+PZKOvNt8eZcSO/kjUvTIBILt7fAtN1BhsWpjQz
n+tuVWs4GVoPQssM4N09AYUzx8ni4byADY6n9l4zof2HsPsjXvuw/bzYXctKTQxg
gD3nswtLGY3q6unYewIChyaG8DStihFLcXIhXbwc6EOqdPN3VWwNInG/602UTO2L
eXoEM+tTaXkE51P5otACVH37AW0Vqqh1GxklYlMYLrKKn/YIBRvLVS5G+95iKs3g
MJhnaeFND2s9dmOTXyKyfTUffr/XTL/PVJSCbdqwiuXZQp8J77MtYyJn1262H8ko
59OlPtqvpBNuywco0/F8BOFvSTw5sS8CmU0EHvyunKaofS3mxCFd2B0DzX89+AoY
ZY7CKU8OYt/VqhsfsL0C+DL4+XschBOUoTg6HrqG4F69+gerkK4Ps2984vOeTXe3
IqlYN/Bn92m9rGy9PKkpqG5C6w5X58BgvfeWTAkM4X32rZHKOmyYSCdRJQw7MGSR
24OaWKPmgKvMaH8MQaJx9oSaAgF1y1892+ykVI9ntCVwywkmxNg1lfMuVFM4Vh9j
+C7OngnbQbhYtbFG90z9zfoMln39z8KT2yDP4A7Hklw0xmmT6t8KduGD2tfmaW4+
oz9attTyiQIcBBABCgAGBQJTvUdkAAoJEJLIQ0VtpqZuLNQP/RazXTtk1mixmLfu
nrScgh/1Gtb6XPVDTP5SGb/8HVdY1a+dDBClUIAFbEAUBIcBO4/NVlW9un4IHyri
xdD/ijE/Cr/BMLnSFU5EmHKn8yOc3Bv6eTTRbJ8EYru0Cj5MdSIAoq8JKF8Lbxba
hBFw9ZSIpREPGlxnKI3TEuHJQreSZR07/GPk50suTK2CXxzEsk4VZesSTnwt/1m+
hrdtyNke3+wy8R0Use+KKgmTzpQ9phc8wq3uhHFzLJ5HbE21VRHgFd5+osZuSDuH
jw5o/zU5o0Bq9DDY3TWXPD9lgqKQ2PgH5DG6od9gjpjJdvXpXpck08HJDu2V+u7M
zW5lsnthDsW39YVLRD4ZwZJQaw+0wBuGDDxV+8x4fYhYJnXD8ZtUHCQ56FlcMyzL
m00WRLNxBSJjRvH12geg5xK3JWZ8V6Ce+XhvIAduDkajuUKQoWvTMzYaviqQhfV1
zXNNt5rxDNT+jMiwnAtDeCTZPMfGvzOPw9sYBz5RUa4liVPrGiEW/snAhcMB9JDK
jMAsKHvJwIvB9QrRC9sz6JIgEqv+jlsdkslEVU2AALlcJ3QuXlW0j9Z26q0sNGt8
5FVqhc4DOyXtZhKR5Ru8Lyc3swYRi2ofu9Roycq3L0swcMoGnOikRbj2PuuRdQiU
ozJ7mT6JmNF8ynlx/1+uzniVneEliQIcBBIBCAAGBQJUGDc+AAoJEOqwPFi/3EeP
EpUP/i2p7BLtyrujFmwr6wxru8G2HBWySFeRZ7YC5iH1ZIiGPPi7nuinWEv6FaW6
dW8fzrwmJy0iUpstKM8CUdR10PZ66Un6A9yfj9MuYSWBDQwzkd3wSJ2+HPLeaTsc
hb545CYnIJBaUAWyfufyoeB2+u3DNQd4oHv5ceI781D6J0h9MUz1bNgV2w3prubD
9o8ycaUvXqHrADqDZiUA1zblm19AmbJIJbgeUaXD8iwkyk2hRHScVe9aLzd2J4TX
PIQt9Dj/VnW3TLvTtMPEls1SM+pQ9xYF3IBz5BWbrMuk3o3mpsIijs+vfjJdv77Q
UeyT0Ur0fJ5xa7zWxCs5AMpCFDdLGRW1jzFYipJBBhDVkylIjpr3eoIIU/RxVC6f
5dTaW8Geasplly JT8BKe54R9u/uf7QPgIkuXdIdaZ4qbbftLxvyBJk6A2gSM/sLY
qeoo+zRxKi+lMZ/Zw8MUZ/ON+yE1ccQJAk1GDqAhUs5f7zCPU6xBLFeXhz9d6bFP
BGWZ3dy7mZaHgALDGqSd240hIoknwzqHaWZK5lWTmzIVUUQPV1Vn68stcaVsuaFd
tnGHv/JMvAj1F9ZlDDRwaPx0ATTXQbBI6JlvDQ8u2tTKIYEswtez/enkjX1dALFl
tV5FNartIWGiHWYxkGVRlh/QtAb0NsBU4/5uZkWpWzG0GtOniQI9BBMBCgAnAhsD
BQsJCAcDBRUKCQgLBRYDAgEAAh4BAheABQJVntrCBQkDwu9RAAoJEDanwF/h7Pm7
aw8QAIJadQ8+oJgvc9yMS+kW+EzkdmGQUDk/7Ozkkj8gk3Fyh2gTI6AenQaDm3QD
/qvShJts/JX9+zVREwGU7WAvXm2EGIB2OB6pVQOGhNc0V8BihykQ27BtetvMivgp
KTYCwdh/CWdF3hAGaPFcc/GkrQThTBZARISUC5zce/FvCSugvRsMdGYjhaCA/LYp
YBGRzXZIrfCaT19tvfW89WdFamT5dVcMVs0uVHw1qYzz9EyH8cj3KwVanZ/KF7zA
pENdsmx2/+EPmVL/oGqctcx5vXtzMYTdr54aRay67oDbtWZNqYoIksShaw5rsNkI
OnqMXCqQkDd0lf46CmqAqqwH0bNlnxTpm8cEcfGIee1yNUc0BGs7qgmtm/7+96My
WPKHixQR48OPiTAdT6/n8msm/+MeL8aRWvhLUkhi3zDWoilScPW121JbE0QytpWP
v+2QCRtvnJritKrzyfjO2CRHOoe5zOY/yRd91CVBde7Tpd0UdQP3vw3B0MznahYQ
+qdt1bMD6NLgADNDP0B2yWMnUS6xku6iBANwZcdLSav4gE7gVFbePKNJnGW6rOIp
ljl0Ivm//KVufLPNX0DAokfvHIvDNNuVuk82UpvVSVxrIaig4HhbWDOnYiDhtE0Q
CDNP7v5e9u/iC3VwBVI73iOsDpPGqEFp1E6UZF6Uvfcxmsp5tCtDb3JlIFNlY3Jl
dGFyeSA8Y29yZS1zZWNyZXRhcnlAZnJlZWJzZC5vcmc+iQIcBBABCgAGBQJTvR9B
AAoJEANvbJ7n856/QGAQANf7Qn3AvTB1Co9oCtKobbtLxOx/FFw6/jnfurJxQ2Y1
8N9zTNJ1KCzI8pYbanicWQFqUfC1wu6FrnSlNGQvW464NqcERElbFE41pvqX+Tb6
/dOX07mMBZYK8wgLDcHEjl4i7NHurx1AKA2ro/5utRvfIqmhPxcHwhNiP1He4MD1
NgkyrxmRWtO4VM99mhXdm+pl/8XwuFJrdg4v36pEws6tYJgPwDc86/XrmeJT6GOC
RFREdwXn6osSvvVYnx4Pyto/xTG5Fm4sa7S4bxgvvSzp2/L+eO4JpOGXuhiIGhfE
wISta1yf14GKTa4a0Qd+gquml4yd1DBybNoa0zcz/sJOBULe/CLKzSs5IuGkfdH0
os1WEjdqQ7JPct3Yizb7Iw/j1YfvDmnM+tt3EMU1DJ1ttY9+XB6pZvtjSHNApaDP
fSeizstpoLle3kvECBJyEIr5u/hL72dYEZtFiYFlHcvWIq6KqWJlIJrOa7vG7r58
```

```
6qstiG270tCeaVOfZT5grKNcDf4vYEoxL+2NKcHVA0rogRWPMwSWZbWEAaIilK/6
AVzc8xmefZJEHHxH7PprcPsH3MPpOwmWjfheHBKfIfEu1UqWAD+cRQmE+jEz1vc6
DzVUfA4c27j9/GXT9/NQsBTamC6rT3YUZKWlFulCC3ncRwf9ZTGSsiT5qCuV1ECn
iQIcBBABCgAGBQJTvUdSAAoJEO1n7NZdz2rnVCcP/3zh4y7MrLnV536rB0tDOM+l
sP3UYDmclWZmTENZ+r0ESM4YJzDjKO6ltXhh+MdYqDddY3vqLnsKTNYuMjKiu6jd
0ETy4ThzHxVhcyrgl1yWxyaSsdi5gMOnwNVClHfOD5gaOF0jdRJnLTLMueAEM3fy
NzSUjBOHJpk+RcIV3r/u8LvPFV2qwLWa937vYwfLRJ9jaQur5MnEVOWBz7CB0g1F
06JAcLv16FyWiLCOBxXZJd974lKXHd2yEMkSlF3Of1qX5FI6FK3HQU+c6eJcrWc5
4++zvgWHZTM3SwY9g5rl57Wz9Vpi13Ev6ArIIIEQ9P1ViwK4zyW78rFoM09juqHk
N4uUCWuk0f57XCFkrDA/n6YCSfAxSYXc1I+MKpAm/6yBYiBNpyS3Jz5HGO2S0QGs
PsBcUHGEmO6k/Z6boJLwaCAGx2dSQ84R4DQeFAd3NjBPab2xTNlitcOi+xnidCJZ
oDWQqx5dSLwLeORsw25ikOWroUS1hqfta2HDnnou4zuyppov0Q+50JGFJIril8sv
oUmLfuSk3XUrlxPHgL57Wp5K8NOIU4u/DStX4UaRuHQ+Uu9GV+c6rr3F46+MooqX
ISAw5cm/kDMwuOfQ4GOo8J/ADUfLQa6a/JnWFg3hb/rgt/lHJxjZliOZy8G08HyB
ddNfKCTBauqEyPYWTHzgiQIcBBABCgAGBQJTvUdkAAoJEJLIQ0VtpqZu/g0QAMXE
q8sNraENb3zO8wisdE0UZXOGuQduXDSrwpe26L9mCR/usjeWeGqbG9b6mP+fAwXx
m/BovdkF3bWguo1GCzztEHy+aTB4voxI6lEnyDKB8GG3mlkVjNAbDjVi/jCZfe6T
bJ6xDhX7633ees1An8tvizMHRr+z8zQ3xF4MNjlxLzawPE7/As5uHaT6Q3NhGTGy
G1oGsVlO8pYp97p2E/d44m6IlY5XEzO2A2fIq+0N4dcy8omTX8P4eUZFlUezRbbZ
tNP8Av77hESXO79gpmQir9fC5/qMBgJN+3iB9O+VcB0SeLm0TvUWtSFULqEdDkKA
rLOgZf0HNnsu7/rb1tR9zqSYN8gsF3MvF0RNHUdbyEh189LZTmapwSxcaoUYPcoO
Bfwo0MqMuEuyCkMWSD53BvsaSbcs2OWKYAp+oluM0TrnLup37O2G/EbxmMRHZVVy
uX60pIQDXO4DjLo9tqbM6OUNCG+1tKEX7Bs9GIzUL2mxZ072qE8x1A+eidSzy5Tx
2nE7DOurziiuv8G3JPFDtLkUVtPx9gqyyG3wmfThkMCl1jnutYDjetpeC8LcI5S9
mFE8BXBka7qEEY19GI/1LJcfMI8lMn16OITYv4/cwqWPMbjSMg6JpWBCFdsxRmIW
iggKodt6LfnuEciChejk8ewTf6/47z7aVhdBkYaOiQIcBBIBCAAGBQJUGDc+AAoJ
EOqwPFi/3EePCM0QAKFrkt0wW+am/O8ZzIejSCY+htWilGAIa6REk5gv00k2dKPC
Wf5rNPAXeQRAX4qItmd35hz7czElm2EVbrylDD+F9uN8wbkCMLdIe88caWfoj12l
JACAd0NiBSwJPgrajvER92fr173I31cKT6hwXP6bgjU3J4HICc1h7h5j7g+/YSeH
UacPSiY4MuXAQao6e2BtFI77LOwFvIFFdCEMdZDwoH+7lIF9I+Krm7ojMF5fauaS
K4e3kL029QugIFYlgb7HeDGLlonBSn40YXPenafAin0lNGWMWVv3SKN7tweNkKEh
MVoVReropjYpRg+khKaMumwJ9bdGkYP8jq7DwCKXy/J2rfCUzsyVX5Ga7keT8Ztn
y4RO6YqFtTryraiSPrxDZQ0Gt6kc1m3u+4vh93qJk+foUDRSLWfWjmX9aRf+7+4z
dsYBOrpt3tab6FqXrW7IcI+p8PCyBW4c/WHkU3YWreEba21BXGiMDoxfhQ9OyvSH
t6G8kgO9+k8sRY/78oGYxR4Aait8/Y54DmHkyZKSewu96So0+TDmcbkeatatDhra
sbjfQLWx8363tnQvmhWpw/bpWGmDQDLVTHn9OcXlckGU1fiUM7721g4s2UdijTMp
yYWfsLTaxOujHyxkwkBBtGV3DCas5Ep2KPMfSOgf3YVtPQH9IaotJSw/A6FdiQI9
BBMBCgAnBQJTvR7xAhsDBQkB4TOABQsJCAcDBRUKCQgLBRYDAgEAAh4BAheAAAoJ
EDanwF/h7Pm7QvMQAKE3pM3e7LrDH6+xsdafxb/RxnVwUI6FaoN3dIZRjIIh7Dyd
6WypD43+f4c4AeIX+b78RuCuu+oZMMkHk4/Y4PIRv6jw1wuGa67iHopFXy9KPYjE
QOtLptZUAorqC62CzoVJxwbpIPw1AkKBag7FFKtiymQKbxSAkEkCOTa64RF+FFDJ
zUqbRQPJMMhKR35lJ/W3TfNQQViF/nydDdNmSY+gYAPU8kqhx4K7K9al9DUwVa/P
dL0l549BLOHzmFcEtw4FQOGMYt4Gkma5+6OIMJOuoM/ADAUz7qdcWYYdsFl42HzC
73u7MGLcfGkElcZKkH8sn2zuKsTTtTKD5rhLfIiu132vK7vqoONdJLd7U1X2Bwif
/ub1we7x4eGonZjhKajENpD3o/1Y072gLy8rlZ1r6/J+GQ9TEwUBNV8NNOfDPv0p
xTP6OCFPHEFA4toG0rRBm7OIxmQXFWmfxMT3NnwBqPCufWlOm20JhaU/pefPCqHJ
Vc8Ap+k6/bct3iNuAg1buggFVDWg89uBqF9vfdELiCDF3nRYm4bQ6S1cWxvnu5aq
9MZdt4Dc1WnTSNfY9/zjKJWmG3miv1D1eo3fSyVJNYVfVzQ23KMOPwR/jdr47Gle
8/5OM38zPhZ+vC+XD//Lq0/c8iMO39B4pwQ0Bb8FAhk/6Ug0cYbap+lPWkY+iQI9
BBMBCgAnAhsDBQsJCAcDBRUKCQgLBRYDAgEAAh4BAheABQJVntrOBQkDwu9RAAoJ
EDanwF/h7Pm77qQQALNcLFh7Eqc6gztkIySQtqDgieH1Frj2YGMrTxdzP7fnl4sT
wWQF2H0rsLgvOTZszACwLm9Rg2GaRi/bFZ9S8AIM1klQEQmu5LoQDSPrfX0pNxDl
1WhESECgEAF7++HDk3kHUxr5bCr+AN6kG3mY2nuKtw33G5ZbPtM5Nwgk/GT8dB0w
EazZVAN103RpdDmqGgWhYP/dYjGZJjLQCjGyM4VoQcEu+dPrUlAxhFbm13wRyfgi
fBkOug9pB0lpblEXAbB3slDw+4A1++lUjTLzGHayRZIys3d4uM4l1C1b8Z5SeU4r
mpDcqvRbDkhxwVh6e7YdEbf1VTWjQ2WgvEDQXG74N/nN3npbjPfnyFF79udHTZvb
qW+uqyozxXdtLvIbcEHWVDdQGqqTeeRinRu7sK8PRuJxbI7YaECGFw/1QrXdIUzi
1jxRWgtfWihCArXiEuo1LBwdto8CaSxkKV+QTsaH/t5mUiA+fHzsMlexWe55K/7i
OZ5wViUcffNYwbQ7DgcRxwuzc+IYaGg0GYGzJfWeKnrpwm8uZSccH1LLXdIyf5vp
Q9UojD0qz1+byBPtl5/DBwFi5st248YMnXCItIYkjTfLeXQp4oSi3AHH0TEAxhUt
MLoI7CRnIiuCLmssfGI3sNH55wJQQeD7h8c/DaBI8yDuJ28dNY6H5zjPc82suQIN
BFO9HvEBEACynbl7EgcRIGWP7O6h1O6mrYXNZ2JpJBgYosqizdDHyru2nQSrNfgi
wAM1feB2NLJC0coQzRO1sDK2JP770+eK3ZhbWSP5BWN2toSFVEGlVpGWLBGoefae
ZnZA22IDzpOIjIi7iC92JBsTXESsBoV8iG1rylQ15pcE03IQEuuDu9r7H8RJ3vTf
X1c+a+B8MUHn56kn3QkdG2blV0/3gjFqqavZeOxZpAmyn9n9Vc3yCCPkagtNQwle
NyZOSOLjjVpBjncE6dATdLOj85phfOU6eO/0bMXAgTr7mY41EIqYqdPQYrY93ySG
```

```
gBvBkyNaH5AlDNZZwJ4ddtDMFoP8nUhBoRrf5ApYyHcEmSXahLfW3a2qrPm/w5VL
EGLt53/6GZvEetpP+TtBLAxX6XaC2SXAOrzfSZENdYt/Ew6F/dTCZ622m0eW65iV
wSi1sNZD2hNFPs/12a2tem7DAWqD2bi8BltKRbO+8T7BARwIl5hXGq5+YnO+DgTI
f4SYkSt8aiPYwDAF3YSkzpiUmZoBSRt6Sb9sZ3zIxpfnrtLFmSeujzinyCVNzFdn
+HKxZvI9Mc3Tv/LqPruVuWHt1Aj+eygH5bRZw4PTsMNX1FxM/K8hRY91A6Fyp3GC
kb5RzqdEGuSONBseaZirC0d+EYZ4smy1jydpzwT1O8VjY4wi5BdgwQARAQABiQIl
BBgBCgAPBQJTvR7xAhsMBQkB4TOAAAoJEDanwF/h7Pm7gzUP/Auq4I876RUqAP2D
gPNArjuc7jqvIIsgqBvwS2Vdvxg0pFQmsL13nvlmFHR1dp+yOhiz1WmXIVGa9UoB
rVdmDrzqCifHUVlFct99DLJlM+hNmQnHgTABTsClR4idkekOvyEgAW/gsgddYfqm
N8nRKSrokFZvwoR7HlCcSXEuATbqgHjJC8IuJuIgDWjshy3yVLIo5v/g/Yuio4bx
tRTJm72DtUIdDH3tDejc37wd945Rzk4hKzet64inF6XfRhqBsRob1m6/wtP62xPc
Gbb1GlzxYg9WllRJ/ZOTktB4BemVjRn4/0KU93YyxFxHOtXsoZpRGDlovQBUhRkd
b2JEyHaLnFVKa7w7JFKW9d+OdGjV93l//6139QEQlcuocV4QciXZOXTZkWsaKQOb
6WkrSwkQwvNFS2yeUd0x6HD0xebztmSGoXL3uCKyeIxP/8MJjJUkWHh/+DLue4kh
lHwva7Wym0cvYoS3RzfYd26t9OgM7rflE0tOeJovaoaKP6Aant/EkQj2Wb1MssO/
Ia44NyMTrLARAPlAEGW/HI5JL9DZDnmUtByf29yqaHDuJCZS3nvR3T/4q+pblTVd
xQgwopUWjiKkg72yCXMhr/6FcOr9Xgqn++GuMZzh/TFdEoFoJn9stUaScYsoMoih
ym2NFbd1l2Pw/ifTQk5/KpQD/HkwiQIlBBgBCgAPAhsMBQJVnts0BQkDwu/DAAoJ
EDanwF/h7Pm7fV4P/Rb4aYTZmpajgk26vnXIJ2L/Xorx1lV7+a1bDxct71qWSLv3
aCDGDXe1ZLYMnyaNCJHNn4SP2biF7QrRhRmGdETA2VXXCFYLDS7s4OQKltF3p1DS
1UPozfIyXkCFGbOasbaDtJ9SOSv04q1E6nXfCcF8xgjxMHbbrCsGiiHSNet+5A9f
AeZPWKWeUDgc3U7tho0FWyHZWgCjt5x1pYlCsUPgV3vlyo8HqRiWxLYF1WQ7SwV/
1536r5DD5vFbzhdxQ03mPDqLh0VZwPlQ6nb+bKwDX/75Q6tBNBkUQGM7XqI5tZ07
CjjI1/sTYMJz+GNnkIgrSiBr3vihtJgUzbJFfJqoIPkSdpayCP17k2Mdk9eCW6mc
tmtfhU91lzJUvAl3Og1TQP/wiyPQhTrQOlonwkrzY/Bve9Ilh2ZDt/9Q+XzEtSdG
59phFT4s7Yt5f9gVGx2hUhE5mmLN27rQJ2JnyIkA9IyNiycFPRo4ZURXI0yzCwXH
bwLEmA2FVnnyotURN7Z1k2PzUOL6lb7j2fMJZBbaFDR1G7J2pKNihJO3IY80z7Le
tpRpWKSdHTpSW1UXJFQBFT4W7fja03aEXP3bK+RV1MkAp/TCWlVCsqnyJae7BnEu
g9NgaY6PRozzy5ZAdWM36ylnwFyz48fgUWHFhDc7fMCHjwqDtP62K1TbRvFM
=cQn3
-----END PGP PUBLIC KEY BLOCK-----
```

D.1.4. Ports Management Team Secretary <portmgr-secretary@FreeBSD.org>

```
pub   2048R/D8294EC3BBC4D7D5 2012-07-24
      Key fingerprint = FB37 45C8 6F15 E8ED AC81  32FC D829 4EC3 BBC4 D7D5
uid                          FreeBSD Ports Management Team Secretary <portmgr-
secretary@FreeBSD.org>
sub   2048R/5CC117965F65CFE7 2012-07-24
sub   4096R/CA20328577064EB7 2013-10-05
sub   4096R/8B114B3613867E00 2013-10-05
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQENBFAOzqYBCACYd+KGv0/DduIRpSEKWZG2yfDILStzWfdaQMD+8zdWihB0x7dd
JDBUpV0o0Ixzt9mvu5CHybx+9lOHeFRhZshFXc+bIJOPyi+JrSs100o7Lo6jg6+c
Si2vME0ixG4x9YjCi8DisXIGJ1kZiDXhmVWwCvL+vLInpeXrtJnK8yFkmszCOr4Y
Q3GXuvdU0BF2tL/Wo/eCbSf+3U9syopVS2L2wKcP76bbYU0ioO35Y503rJEK6R5G
TchwYvYjSXuhv4ec7N1/j3thrMC9GNpoqjVninTynOk2kn+YZuMpO3c6b/pfoNcq
MxoizGlTu8VT4OO/SF1y52OkKjpAsENbFaNTABEBAAG0R0ZyZWVCU0QgUG9ydHMg
TWFuYWdlbWVudCBUZWFtIFNlY3JldGFyeSA8cG9ydG1nci1zZWNyZXRhcnlARnJl
ZUJTRC5vcmc+iQE4BBMBAgAiBQJQDs6mAhsDBgsJCAcDAgYVCAIJCgsEFgIDAQIe
AQIXgAAKCRDYKU7Du8TX1QW2B/0coHe8utbTfGKpeM4BY9IyC+PFgkE58Hq50o8d
shoB9gfommcUaK9PNwJPxTEJNlwiKPZy+VoKs/+dO8gahovchbRdSyP1ejn3CFy+
H8pol0hDDU4n7Ldc50q54GLuZijdcJZqlgOloZqWOYtXFklKPZjdUvYN8KHAntgf
u361rwM4DZ40HngYY9fdGc4SbXurGA5m+vLAURLzPv+QRQqHfaI1DZF6gzMgY49x
qS1JBF4kPoicpgvs3o6CuX8MD9ewGFSAMM3EdzV6ZdC8pnpXC8+8Q+p6FjNqmtjk
GpW39Zq/p8SJVg1RortCH6qWLe7dW7TaFYov7gF1V/DYwDN5iQEcBBABAgAGBQJQ
DuVrAAoJENk3EJekc8mQ3KwIAImNDMXAF8ajPwCZFpM6KDi3F/jpwyBPISGY1oWu
YPEi1zN94k5jS90aZb3W8Y8x4JTh35Ewb6XODi3uGLSLCmnlqu2a80yPfXf5IuWm
IQdFNQxvosj9UHrg+icZGFmm+f0hPJxMTsZREv3AvivQfnb/N3xIICxW4SjKSYXQ
cq4hr4ObhUx7GKnjayq+ofU2cRlujr87uOH0fO3xhOJG4+cX5mI1HGK38k0Csc1z
qYa/66Qe5dnIZz+sNXpEPMLAHIt1a45UB967igJdZSDFN33bPl1QWmf3aUXU3d1V
```

```
ttiSyHkpm4kb9KgsDkUk1IJ5nUe9OXydWtoqNW5afDa5N0aIRgQQEQIABgUCUA7l
wwAKCRB59uBxdBRinNh2AJ41+zfsaQSRHWvSkqOXGcP/fgOduwCfUJDT+M1eXe2u
dmKof/9yzGYMirKJASIEEAECAAwFAlAaIT8FAwASdQAACgkQlxC4m8pXrXwCHAf+
J7l+L7AvRpqlQcezjnjFS/zG1098qkDflThHZlpVnrBMJZaXdvL6LzVgiIYVWZC5
CSSazW9EWFjp9VjM7FBHdWFZNMV7GAuUt0jzx6gGXOWwi+/v/hs1P11RyDZN5hIC
HdPNmyZVupciDxe+sIEP9aEbVxcaiccqzM/pFzIVIMMP5tCiA42q6Mz3h0hy6hnt
UKptS8Uon6sje5cDVcVlKAUj1wO2cphCqkYlwMQfZV5J9f/hcW5ODriD3cBwK8So
cA2Cq5JYF8kYDL1+pXnUutGnvAHUYt87RWvQdKmfXjzBcMFJ2LlPUB1+IFvwQ13V
9R8j9B/EdLmSWQYT9qRA2rkBDQRQDs6mAQgAzNxJYpf5PrqV8pdRXkn36Fe45q67
1YtbZ2WrT7D0CVZ8Z+AZsxnP/tiY1SrM2MepCeA2xBAhKGsWBWo1aRk5mfZOksKs
iXsi2XeBVhdZlCkrOMKBTVian7I1lH59ZnNIMX0Nl0tlj3L1IjeWWNvfej43URV8
1S9EmSwpjaWboatr2A+1oJku5m7nPD9JIOckE1TzBsyhx7zIUN9w6MKr7gFw8DCz
ypwUKyYgKYToVm8QlkT/L3B0fuQHWhT6ROGk4o8SC71ia5tc1TzUzGEZ1AQO8bbn
bmJLBDKveWHCoaeAkRzINzoD9wAn9z4pnilze59QtKC1cOqUksTvBSDh6wARAQAB
iQEfBBgBAgAJBQJQDs6mAhsMAAoJENgpTsO7xNfVOHoH/i5VyggVdwpqPX8YBmN5
mXQziYZNQoiON8IhOsxpX4W2nXCj5m6MACV6nJDVV6wyUH8/VvDQC9nHarCe1oaN
sHXJz0HamYt5gHJ0G1bYuBcuJp/FEjLa48XFI7nXQjJHn8rlwZMjK/PWj1lw2WZi
ekviuzTEDH8c3YStGJSa+gYe8Eyq3XJVAe2VQOhImoWgGDR3tWfgrya/IdEFb/jm
jHSG5XUfbI0vNwqlf832BqSQKPG/Zix4MmBJgvAz4R71PH8WBmbmNFjDelxVyfz8
0+iMgEb9aL91MfeBNC2KB1pFmg91mQTsiq7ajwVLVJK8NplHAkdLmkBCO8MgMjzG
hlG5Ag0EUk+ViAEQALkCj95JmvmfgytOxnR1w8xnQBuUxtYxf+BWwqU0IfOMBxAm
XDRfbfe9Bc3DDZygmjrZ0RCDcVJ4OmDNRc/vvoTst870so49dM1h2i6aWUhhD4Hy
cNjkx0wAIyZ9jXidT3LyAhCTSf/GjtpcORCvmAsXcJHzhhbJ5eM2SBkXmNdn4MeC
/hCbfUfzN64fmsGR4tKKIlPPayyQaQv/dOP3ofEh4SWRc8KtjEA6uIyHztWMWEJQ
KN/7nSeVuwDpVJL9l4yUGB0TqlK0p/necoQkOolnHUX5oK7Emin9TjPYCGqYAEWu
M1BbWXiYIdivlEm3ZU+vqq6CbSw/SKDZ2ZuK4UzDGQnW7WycsbXqZ18aewh1mdf+
5+YBxeXazRJvfFapB/WvtVqi84LOWior1IBMMg3PCyBiCjhHwXvoV5V6M3lthrnV
gyiCTreLXvTLNgIKQQFUvwW2Jscuwa3HhBgkhkJBSCPmpWAIZ9eA1RugvFQ893Xh
t8vy+z33WqxgUzhD8ZonKCBEZa29Zj7SeIYXcZmVmxCbRt4PYkgFoq6VC+93qtXu
OaOvC+0b9qkmNj5EqvozRt6SNv0sDe844T0LacR6QS9Pq/YMjAs8ao3gw0iwOAlV
vpiZfoEb6purf0Tz0h/URRHUZngEBsFmhciZxuCg9g/GaD8o/+PX0nzOWH4PABEB
AAGJAz4EGAECAAkFAlJPlYgCGwICKQkQ2ClOw7vE19XBXSAEGQECAAYFAlJPlYgA
CgkQyiAyhXcGTrfF9g/8C6JhufyxnYO0DmPURCFV0em3WjXcxQkhQa03u/sU8OFN
yp2UOB8jxI3XAsGe4RnclPF273K5G5OfUAJ2hjWFGrMqXGNLlsuJvDJCUmdHnI6c
sw1RvOn2myMzUSTqxPQynkvntughqiyXbGQaoxQJVWjruhogcIkVYvOT4nMGXCyk
9Q+uooLNw8xpzsZBw83znWhjjzemu/BbOfjXbf3qdK9NWMkBfcaR9sDhRA+YY4bc
o7oTKXFfh3gMN1pqTuAM8dbGc0/bS558yNQuFaJ+cX3lbSLXz5oibJi6kudyL9E/
CkVvMFWTLcQQWCPS5rK0dctodW1x8JnwVnaG80trFIxmQ+wk4S156CdYwTv/1LbW
GKcVz9Og1e9wIL2LsVaLD9OIRnx4G8dLqZJz4pmGEES7CX6rrJuSpXxNOg4PthiP
s0z/q9deohKJkUgj6tln0OCgI8203GSSLq3FTAFer2VS+m1XMkA2mqk1Wv6tZeyY
MnYAZCBzc16F2zMg5dVKmV114bHmRTX0b5QQNB7JU3C34kdTagjLr4dz/5BhXgof
zjP9HgVQKCp7JvTdUT7N0y/k+mRMmnK8vnWdbOZH48IOELDhMkU5QIiKHoJaXsw7
wVo68LtSQItfIoA/m6EORGZxUWCi/8G+kBK6NEExR1dlAW+M+fRhzNgqvqoLrcfR
4wgAkQGq3M2/hGdU/Z1j6CDKEvA1/iSRpnBnfVM8KUR661QrEctmIo1YwSU7x5+g
H5lfDUjU2cIlD+HgfxERKwl+hb2KY9OyLq2AUhhf8rAvG2dU9djDp7TWIvF3wexL
liu4C6EWcwlnEecCpkPUYv7/PKb0h0xAx32Umb6dzkfd24miWZTI/Gg7R1Qyl1DC
wBM9kgPRgEhnV3ummsD3KPXf5UwzHPslhqWTO1p2iML4exnXlknL7mJKjj4d3gyi
5LxVVSzt2xi5MyCkFhiHFkSfItihcRpHdzxsBW+7YWw1ELlcKMeOkZQcyBm3VpE/
ZD4DQK6zwGfk4y8WS7A5SJEjJrkCDQRST5W/ARAAwbT4loEoK7ZY8fzt8hyhdBgS
bpXFWAB6yeyGDYp9ucG4ySjlfZIAew/EOzIR/68pKiGzIE0+kxKjlvVDcJoqWfrr
gomondGK2oyK35qOsYOlae6tsZm73InJuGSC+fZ1Vv0HBIn8JZpJXFV7z4FotaG5
iaKDdZCjxawVEWDl7z0zDSLKgPLtK4uSsEVsQIhPo5YLSqT7mCnleCqdNu3S4cun
tXaZmmSZnfv3qnkiPNNNSagZrMUK349fVCdvnJ4hKDbMAEgbxZVwEeB4HFicjwAN
UQTO1qSc+h8bwkrN/RgaT0Zz3nak+DJOOrKRV5VWWemx4Oy2oT08SqMRbhip/veI
LvrV+rKH+lb9uxp0vYWnRvWodOlEla0NeBzfz6Sv8ZI2D+xLJ260mIB4f4BUYCBi
SDu8UKTDffYba+50I9RgV9+umJ2WTcp/PA5/59M9e6R2F3dcM0Qey8hVDcf+rPY2
thXymS85IHcPauDjvjpHpx1xvhoEu2iX0yJwBDUX/xVWwH/lei7DT0cd1dB1pKiP
zNvWSoPp6zhqcB6tUvE69CLRzotmb2OiAkFOReEVkcATL+bG9PBN0zWtSXJuCSL5
WAb+syRzBGwLA7+iQOY0yXCZ+Etl6BG8rsyQkBRwpTG5jXzDKUTl2hmKGFmU72xc
cPeNxhbq1tGMGfDVVmkAEQEAAYkBHwQYAQIACQUCUk+VvwIbDAAKCRDYKU7Du8TX
1cEOB/0WwX3yF/MC8upI0YAbjht+KG4cLgU6qo1ydZgQyHpAf8cttzq/uCDu4wLE
g0CDmreTXoiNR9W0ULhkn1LFkZQI6Z2uSH/wqQUJrE4P9Oo61TXClIzvjBoEufkS
3bg3wuAKiQ4cD1XxyThUG3qa4nbGVvKi4eWYuubpzJof7QD75LudHmLneY4mtCNA
ZgmgEWA2Utn0GWN1QNfoy1NGeLt0kza0VFjm6l6KeMc28ULZZztg0KbuFmQIreN9
JHQbJrrqX8ev57SqEtanoPyX4IjxVJFEmTn4xVSlRXY2uFZZtlB1SuyEPYmrdnjb
pDN9ZUgVOk/c+0+5u+G3JBRYu6P1
```

```
=fyjO
-----END PGP PUBLIC KEY BLOCK-----
```

FreeBSD Glossar

Dieser Abschnitt enthält Begriffe und Abkürzungen, die innerhalb des FreeBSD-Projekts sowie der zugehörigen Dokumentation verwendet werden.

A

ACL	Siehe Access Control List.
ACPI	Siehe Advanced Configuration and Power Interface.
AMD	Siehe Automatic Mount Daemon.
AML	Siehe ACPI Machine Language.
API	Siehe Application Programming Interface.
APIC	Siehe Advanced Programmable Interrupt Controller.
APM	Siehe Advanced Power Management.
APOP	Siehe Authenticated Post Office Protocol.
ASL	Siehe ACPI Source Language.
ATA	Siehe Advanced Technology Attachment.
ATM	Siehe Asynchronous Transfer Mode.
ACPI Machine Language	Pseudocode, der von einer virtuellen Maschine innerhalb eines ACPI-konformen Betriebssystems ausgeführt wird. Bietet eine Verbindungsschicht (*Layer*) zwischen der verwendeten Hardware und der dokumentierten Schnittstelle, auf die das Betriebssystem zugreift.
ACPI Source Language	Die Programmiersprache, in der die AML geschrieben ist.
Access Control List	Eine Liste von Zugriffsrechten, die einem Objekt, normalerweise eine Datei oder ein Gerät im Netzwerk, angehängt ist.
Advanced Configuration and Power Interface	Eine Spezifikation, die eine Abstrahierung der Schnittstelle darstellt, die Hardware und Betriebssystem verbindet. Dadurch benötigt das Betriebssystem keine Informationen über die vorhandene Hardware, um diese einsetzen zu können. ACPI ist eine Weiterentwicklung von APM, PNPBIOS und anderen Technologien und bietet Funktionen zur Kontrolle des Energieverbrauchs, zur Versetzung von Rechnern in den Ruhezustand, zur Aktivierung und Deaktivierung von Geräten und andere mehr.
Application Programming Interface	Eine Sammlung von Prozeduren, Protokollen und Werkzeugen, die das Zusammenspiel von verschiedenen Programmteilen festlegt. Wie, wann und warum arbeiten sie zusammen, welche Daten werden zwischen ihnen ausgetauscht und anderes mehr.
Advanced Power Management	Eine API, die es dem Betriebssystem ermöglicht, zusammen mit dem BIOS die Stromversorgung zu verwalten. APM wurde für die meisten Anwendungen durch die allgemeinere und leistungsfähigere ACPI Spezifikation abgelöst.
Advanced Programmable Interrupt Controller	

Advanced Technology Attachment

Asynchronous Transfer Mode

Authenticated Post Office Protocol

Automatic Mount Daemon — Ein Daemon, der ein Dateisystem automatisch einhängt, wenn auf eine Datei oder ein Verzeichnis dieses Dateisystems zugegriffen wird.

B

BAR — Siehe Base Address Register.

BIND — Siehe Berkeley Internet Name Domain.

BIOS — Siehe Basic Input/Output System.

BSD — Siehe Berkeley Software Distribution.

Base Address Register — Register, die den zu einem PCI-Gerät gehörenden Adressbereich festlegen.

Basic Input/Output System — Die Bedeutung des Begriffs BIOS hängt vom Kontext ab, in dem es verwendet wird. Einmal wird damit der ROM-Chip bezeichnet, der über einen Basisbefehlssatz eine Schnittstelle zwischen Hard- und Software schafft. Aber auch die Routinen, die in diesen Chip implementiert wurden, und die dabei helfen, Ihr System zu starten, werden als BIOS bezeichnet. Und nicht zuletzt wird manchmal die Bildschirmmaske, über die der Bootprozess konfiguriert werden kann, ebenfalls als BIOS bezeichnet. Der Begriff BIOS ist zwar PC-spezifisch, andere Systeme verfügen aber über ähnliche Systeme.

Berkeley Internet Name Domain — Eine Implementierung des DNS-Protokolls.

Berkeley Software Distribution — Diesen Namen gab die Computer Systems Research Group (CSRG) der The University of California at Berkeley den Verbesserungen und Änderungen an AT&Ts 32V UNIX®. FreeBSD beruht auf der Arbeit der CSRG.

Bikeshed Building — Die Beobachtung, dass viele Leute Meinungen zu unkomplizierten Themen äußern, während gleichzeitig über ein kompliziertes Thema gar nicht oder nur wenig diskutiert wird. Die Herkunft des Ausdrucks wird in den FAQ erläutert.

C

CD — Siehe Carrier Detect.

CHAP — Siehe Challenge Handshake Authentication Protocol.

CLIP — Siehe Classical IP over ATM.

COFF — Siehe Common Object File Format.

CPU — Siehe Central Processing Unit.

CTS — Siehe Clear To Send.

CVS — Siehe Concurrent Versions System.

Carrier Detect — Ein RS232C-Signal. Notwendig, um eine serielle Verbindung aufbauen zu können.

Central Processing Unit

Auch als Prozessor bekannt. Dieser stellt das Gehirn eines Computers dar, in dem alle Berechnungen erfolgen. Es gibt verschiedene Prozessor-Architekturen, die über verschiedene Befehlssätze verfügen, beispielsweise Intel-x86-, Sun SPARC-, PowerPC- und Alpha-Systeme.

Challenge Handshake Authentication Protocol

Eine Vorgehensweise, einen Benutzer anhand eines Geheimnisses zu authentisieren, dass zwischen Client und Server ausgetauscht wird.

Classical IP over ATM

Clear To Send

Ein RS232C-Signal. Das entfernte System erhält durch dieses Signal die Erlaubnis, Daten zu senden.
Siehe auch Request To Send.

Common Object File Format

Concurrent Versions System

Ein Versionskontrollsystem, das es erlaubt, mit vielen verschiedenen Versionen einer Datei zu arbeiten und die an diesen Dateien durchgeführten Änderungen zu verfolgen. CVS ermöglicht es, individuelle Änderungen durchzuführen, in ein Repository einzubringen und auch wieder rückgängig zu machen. Außerdem ist es möglich, nachzuvollziehen, welche Änderungen wann, von wem und warum erfolgten.

D

DAC

Siehe Discretionary Access Control.

DDB

Siehe Debugger.

DES

Siehe Data Encryption Standard.

DHCP

Siehe Dynamic Host Configuration Protocol.

DNS

Siehe Domain Name System.

DSDT

Siehe Differentiated System Description Table.

DSR

Siehe Data Set Ready.

DTR

Siehe Data Terminal Ready.

DVMRP

Siehe Distance-Vector Multicast Routing Protocol.

Discretionary Access Control

Data Encryption Standard

Eine Methode zur Verschlüsselung von Informationen. Wird traditionellerweise zur Verschlüsselung von UNIX®-Passwörtern und von crypt(3) verwendet.

Data Set Ready

Ein RS232C-Signal, das von einem Modem an einen Computer oder ein Terminal geschickt wird und die Sende- und Empfangsbereitschaft des Modems meldet.
Siehe auch Data Terminal Ready.

Data Terminal Ready

Ein RS232C-Signal, das von einem Computer oder einem Terminal an das Modem geschickt wird und die Sende- und Empfangsbereitschaft des Computers oder des Terminals meldet.

Debugger

Eine interaktive, in den Kernel eingebaute Funktion, um den Status eines Systems zu untersuchen. Wird in der Regel nach einem Systemabsturz eingesetzt, um die Ursache für den Absturz zu finden.

Differentiated System Description Table

Eine ACPI-Tabelle, die Informationen über die Konfiguration des Basissystems enthält.

Distance-Vector Multicast Routing Protocol

Domain Name System

Das System, dass Klartext-Rechnernamen (wie mail.example.net) in Internet-IP-Adressen (oder umgekehrt) konvertiert.

Dynamic Host Configuration Protocol

Ein Protokoll, das auf Anforderung dynamisch eine IP-Adresse an einen Rechner vergibt. Diese Adresszuweisung wird als „Lease“ bezeichnet.

E

ECOFF

Siehe Extended COFF.

ELF

Siehe Executable and Linking Format.

ESP

Siehe Encapsulated Security Payload.

Encapsulated Security Payload

Executable and Linking Format

Extended COFF

F

FADT

Siehe Fixed ACPI Description Table.

FAT

Siehe File Allocation Table.

FAT16

Siehe File Allocation Table (16-bit).

FTP

Siehe File Transfer Protocol.

File Allocation Table

File Allocation Table (16-bit)

File Transfer Protocol

Ein auf TCP aufsetzendes Protokoll, das zum Transfer von Daten über ein TCP/IP-Netzwerk verwendet wird.

Fixed ACPI Description Table

G

GUI

Siehe Graphical User Interface.

Giant

Der Name für einen wechselseitigen Ausschluss (*mutual exclusion*), der einen großen Teil der Kernel-Ressourcen schützt. Zu Zeiten, als auf einer Maschine nur ein paar Prozesse liefen und die Maschine nur eine Netzwerkkarte und insbesondere nur einen Prozessor besaß, war dieser einfache Mechanismus zum Verriegeln (*lock*) einer Ressource völlig ausreichend. Heutzutage entstehen durch den wechselseitigen Ausschluss Geschwindigkeitsengpässe. Die FreeBSD-Entwickler arbeiten daran, Giant durch Locks zu ersetzten, die einzelne Ressourcen schützen. Auf Einprozessor- und Mehrprozessor-Maschinen können dadurch mehr Prozesse parallel ausgeführt werden.

Graphical User Interface — Eine grafische Oberfläche, über die der Anwender mit dem System interagiert.

H

HTML — Siehe HyperText Markup Language.

HUP — Siehe HangUp.

HangUp

HyperText Markup Language — Die Auszeichnungssprache, mit der Internetseite erstellt werden können.

I

I/O — Siehe Input/Output.

IASL — Siehe Intel's ASL-Compiler.

IMAP — Siehe Internet Message Access Protocol.

IP — Siehe Internet Protocol.

IPFW — Siehe IP Firewall.

IPP — Siehe Internet Printing Protocol.

IPv4 — Siehe IP Version 4.

IPv6 — Siehe IP Version 6.

ISP — Siehe Internet Service Provider.

IP Firewall

IP Version 4 — Die IP-Protokollversion 4, die 32-Bit-Adressen einsetzt. Diese Version stellt derzeit noch den in der Praxis am meisten verwendeten Standard dar, sollt aber sukzessive durch IPv6 ersetzt werden.
Siehe auch IP Version 6.

IP Version 6 — Das neue IP-Protokoll. Es wurde entwickelt, weil der Adressraum von IPv4 nicht mehr ausreichend ist. IPv6 verwendet 128-Bit-Adressen.

Input/Output

Intel's ASL-Compiler — Intel's Compiler zur Konvertierung von ASL nach AML.

Internet Message Access Protocol — Ein Protokoll für den Zugriff auf einen E-Mail-Server. Charakteristisch für dieses Protokoll ist, dass die Nachrichten in der Regel auf dem Server verbleiben und nicht vom E-Mail-Client heruntergeladen werden.
Siehe auch Post Office Protocol Version 3.

Internet Printing Protocol

Internet Protocol — Das Standardprotokoll zur Paketübertragung im Internet. Wurde ursprünglich vom U.S. Department of Defense entwickelt, und ist ein zentraler Bestandteile des TCP/IP-Stacks. Ohne das Internet Protocol wäre das Internet in der heutigen Form nicht möglich. Das Internet Protocol ist im RFC 791 definiert.

Internet Service Provider — Ein Unternehmen, das anderen den Zugang zum Internet ermöglicht.

K

KAME

Japanisch für „Schildkröte“. Der Begriff KAME wird in Computerkreisen für das KAME Project verwendet, das an einer IPv6-Implementierung arbeitet.

KDC

Siehe Key Distribution Center.

KLD

Siehe Kernel ld(1).

KSE

Siehe Kernel Scheduler Entities.

KVA

Siehe Kernel Virtual Address.

Kbps

Siehe Kilo Bits Per Second.

Kernel ld(1)

Eine Methode, um den Kernel dynamisch um zusätzliche Funktionen zu erweitern, ohne das System neu zu starten.

Kernel Scheduler Entities

Threads, die im Kernel laufen. Näheres entnehmen Sie der Home-Page des Projekts.

Kernel Virtual Address

Key Distribution Center

Kilo Bits Per Second

Maßeinheit, in der die Bandbreite (also die Menge der Daten, die in einer bestimmten Zeit übertragen werden kann) angegeben wird. Statt Kilo können auch Mega, Giga, Tera und weitere Präfixe verwendet werden.

L

LAN

Siehe Local Area Network.

LOR

Siehe Lock Order Reversal.

LPD

Siehe Line Printer Daemon.

Line Printer Daemon

Local Area Network

Ein Netzwerk, das nur in einem lokalen Bereich, wie einem Büro, einen Unternehmen oder einem Haus, eingesetzt wird.

Lock Order Reversal

Der FreeBSD-Kernel benutzt eine Reihe von Ressource-Locks, um den Zugriff auf Ressourcen zu regeln. In FreeBSD-CURRENT-Kerneln (nicht in Release-Kerneln) befindet sich das Diagnose-System witness(4), das Verklemmungen (*deadlock*) zur Laufzeit erkennt. witness(4) ist vorsichtig: daher gibt es schon mal Falschmeldungen aus. Eine richtig erkannte Verklemmung bedeutet soviel wie „Wenn Sie Pech gehabt hätten, wäre es jetzt zu einer Verklemmung gekommen“.

Richtig erkannte Verklemmungen (LOR) werden schnell behoben. Prüfen Sie daher http://lists.FreeBSD.org/mailman/listinfo/freebsd-current und die Seite LORs Seen bevor Sie die Mailinglisten kontaktieren.

M

MAC

Siehe Mandatory Access Control.

MADT

Siehe Multiple APIC Description Table.

MFC — Siehe Merge From Current.

MFP4 — Siehe Merge From Perforce.

MFS — Siehe Merge From Stable.

MIT — Siehe Massachusetts Institute of Technology.

MLS — Siehe Multi-Level Security.

MOTD — Siehe Message Of The Day.

MTA — Siehe Mail Transfer Agent.

MUA — Siehe Mail User Agent.

Mail Transfer Agent — Eine Anwendung zum Transfer von E-Mails. Ein MTA war von jeher im BSD-Basissystem enthalten. Aktuell handelt es sich dabei um Sendmail. Es exisitieren aber auch zahlreiche andere MTAs, darunter postfix, qmail und Exim.

Mail User Agent — Ein Programm zur Anzeige und zum Verfassen von E-Mails.

Mandatory Access Control

Massachusetts Institute of Technology

Merge From Current — Das Einbringen von Funktionen oder Fehlerbehebungen aus dem -CURRENT-Zweig in einen anderen Zweig, meist -STABLE.

Merge From Perforce — Das Einbringen von Funktionen oder Fehlerbehebungen aus dem Perforce-Repository des -CURRENT-Zweigs.
Siehe auch Perforce.

Merge From Stable — Normalerweise werden Änderungen an FreeBSD zuerst im -CURRENT-Zweig getestet und dann in den -STABLE-Zweig übernommen. Selten kommt es vor, dass eine Änderung zuerst im -STABLE-Zweig vorgenommen wird und anschließend im -CURRENT-Zweig übernommen wird.

Dieser Ausdruck wird auch benutzt, wenn eine Fehlerbehebung von -STABLE in einem der Sicherheitszweige übernommen wird.
Siehe auch Merge From Current.

Message Of The Day — Eine Nachricht, die in der Regel beim Anmelden an einem System angezeigt wird. Enthält häufig Informationen für die Benutzer des Systems.

Multi-Level Security

Multiple APIC Description Table

N

NAT — Siehe Network Address Translation.

NDISulator — Siehe Project Evil.

NFS — Siehe Network File System.

NTFS — Siehe New Technology File System.

NTP — Siehe Network Time Protocol.

Network Address Translation
: Eine Technik, bei der IP-Pakete auf dem Weg durch ein Gateway umgeschrieben werden. Dadurch wird es möglich, dass sich mehrere Rechner hinter dem Gateway eine einzige IP-Addresse teilen.

Network File System

New Technology File System
: Ein von Microsoft entwickeltes Dateisystem, das in dessen „New Technology“-Betriebssystemen, wie Windows® 2000, Windows NT® und Windows® XP, eingesetzt wird.

Network Time Protocol
: Ein Protokoll, um die Systemzeit über ein Netzwerk zu synchronisieren.

O

OBE
: Siehe Overtaken By Events.

ODMR
: Siehe On-Demand Mail Relay.

OS
: Siehe Operating System.

On-Demand Mail Relay

Operating System
: Eine Sammlung von Programmen, Bibliotheken und Werkzeugen, die den Zugriff auf die Hardware eines Computers erlauben. Die Bandbreite aktueller Betriebssysteme reicht von einfachen Designs, die lediglich die Ausführung eines einzigen Programms und die Nutzung eines einzigen Geräts zur gleichen Zeit erlauben bis hin zu Multitasking- und Multiprozess-Systemen, die gleichzeitig Tausende Benutzer bedienen können, von denen jeder wiederum Dutzende Programme laufen lassen kann.

Overtaken By Events
: Zeigt an, dass eine gewünschte Änderung (aus einem Fehlerbericht oder einer Anforderung) überholt ist. Die Ursache können beispielsweise spätere Änderungen in FreeBSD, geänderte Netzwerk-Standards oder jetzt veraltete Hardware sein.

P

p4
: Siehe Perforce.

PAE
: Siehe Physical Address Extensions.

PAM
: Siehe Pluggable Authentication Modules.

PAP
: Siehe Password Authentication Protocol.

PC
: Siehe Personal Computer.

PCNSFD
: Siehe Personal Computer Network File System Daemon.

PDF
: Siehe Portable Document Format.

PID
: Siehe Process ID.

POLA
: Siehe Principle Of Least Astonishment.

POP
: Siehe Post Office Protocol.

POP3
: Siehe Post Office Protocol Version 3.

PPD
: Siehe PostScript Printer Description.

PPP — Siehe Point-to-Point Protocol.

PPPoA — Siehe PPP over ATM.

PPPoE — Siehe PPP over Ethernet.

PPP over ATM

PPP over Ethernet

PR — Siehe Problem Report.

PXE — Siehe Preboot eXecution Environment.

Password Authentication Protocol

Perforce — Ein von Perforce Software entwickeltes Versionskontrollsystem, das mehr Funktionen als CVS aufweist. Obwohl es sich dabei nicht um Open-Source handelt, dürfen Open-Source-Projekte wie FreeBSD die Software kostenlos einsetzen.

Einige FreeBSD-Entwickler verwenden ein Perforce-Repository, um Quellcode zu verwalten, der selbst für den -CURRENT-Zweig zu experimentell ist.

Personal Computer

Personal Computer Network File System Daemon

Physical Address Extensions — Eine Möglichkeit, um auf Systemen, die physikalisch nur über einen 32-Bit-Adressraum verfügen, bis zu 64 GB RAM ansprechen zu können. Ohne PAE wären diese Systeme auf maximal 4 GB Hauptspeicher beschränkt.

Pluggable Authentication Modules

Point-to-Point Protocol

Pointy Hat — Ein Kopfschmuck, ähnlich den Eselsohren, der FreeBSD-Committern gereicht wird, wenn sie den Bau kaputtmachen, Revisionsnummern verkleinern oder sonstigen Schaden im Quellbaum anrichten. Jeder Committer, der etwas taugt, besitzt schnell eine stattliche Sammlung. Der Begriff wird (meist?) scherzhaft verwendet.

Portable Document Format

Post Office Protocol — Siehe auch Post Office Protocol Version 3.

Post Office Protocol Version 3 — Ein Protokoll für den Zugriff auf einen E-Mail-Server. Dadurch gekennzeichnet, dass neue Nachrichten vom E-Mail-Client heruntergeladen und nicht auf dem Server verbleiben.
Siehe auch Internet Message Access Protocol.

PostScript Printer Description

Preboot eXecution Environment

Principle Of Least Astonishment — Prinzip der kleinsten Überraschung

Änderungen an FreeBSD sollten nach Möglichkeit für den Benutzer nachvollziehbar sein. Das willkürliche Umordnen der Variablen in `/etc/defaults/`

`rc.conf` verletzt zum Beispiel dieses Prinzip. Entwickler beachten das Prinzip, wenn Sie über für Benutzer sichtbare Änderungen nachdenken.

Problem Report

Die Beschreibung eines Problems, das im FreeBSD-Quellcode oder in der Dokumentation gefunden wurde. Lesen Sie dazu auch den Artikel Writing FreeBSD Problem Reports.

Process ID

Eine eindeutige Zahl, die einem Prozess zugewiesen ist. Identifiziert den Prozess und erlaubt es, diesen Prozess zu bearbeiten.

Project Evil

Der Arbeitstitel des von Bill Paul geschriebenen NDISulator. Der Name bezieht sich darauf, dass es (philosophisch gesehen) schlimm ist, einen solchen Treiber überhaupt schreiben zu müssen. Der NDISulator ist ein Kompatibilitätsmodul, das es erlaubt, Microsoft Windows™ NDIS-Miniport-Netzwerktreiber mit FreeBSD/i386 zu benutzen. Für gewöhnlich ist dies die einzige Möglichkeit, Karten mit einem Treiber, dessen Quellen verschlossen sind, zu benutzen. Siehe `src/sys/compat/ndis/subr_ndis.c` .

R

RA

Siehe Router Advertisement.

RAID

Siehe Redundant Array of Inexpensive Disks.

RAM

Siehe Random Access Memory.

RD

Siehe Received Data.

RFC

Siehe Request For Comments.

RISC

Siehe Reduced Instruction Set Computer.

RPC

Siehe Remote Procedure Call.

RS232C

Siehe Recommended Standard 232C.

RTS

Siehe Request To Send.

Random Access Memory

Revision Control System

Das *Revision Control System* (RCS) ist eines der ältesten „Versionsverwaltungssysteme“ für reine Textdateien. Es erlaubt das Speichern, Laden, Archivieren, Protokollieren, Identifizieren sowie das Zusammenführen von verschiedenen Revisionen einer Datei. Bei RCS handelt es sich um eine Sammlung von vielen kleinen zusammenarbeitenden Werkzeugen. Zwar fehlen im Vergleich zu CVS oder Subversion einige Funktionen, allerdings ist RCS sehr einfach zu installieren, zu konfigurieren und zu benutzen, solange die Anzahl der zu verwaltenden Dateien überschaubar bleibt. RCS ist dabei für praktisch alle wichtigen UNIX-artigen Betriebssysteme verfügbar.
Siehe auch Concurrent Versions System, Subversion.

Received Data

Ein RS232C-Pin oder -Draht, über den neue Daten ankommen.
Siehe auch Transmitted Data.

Recommended Standard 232C

Ein Standard für die Kommunikation zwischen seriellen Geräten.

Reduced Instruction Set Computer

Ein Ansatz im Prozessordesign, bei dem die von der Hardware durchzuführenden Operationen so weit als möglichst vereinfacht und verallgemeinert werden. Vorteile dieses Design sind ein geringerer Energieverbrauch, eine geringere Transistoranzahl und übersichtlicherer Code. Zu den RISC-Plattformen gehören Alpha, SPARC®, ARM® sowie PowerPC®.

Redundant Array of Inexpensive Disks

Remote Procedure Call

repocopy — Siehe Repository Copy.

Repository Copy — Eine direkte Kopie von Dateien innerhalb eines Repositories.

Ohne eine Repocopy müsste ein Committer eine Datei mit `cvs add` an der neuen Position einfügen und mit `cvs rm` an der alten Position löschen.

Der Nachteil dieser Methode wäre allerdings, dass dabei die Datei-Historie (also die CVS-Logs) nicht an die neue Position kopiert werden würde. Da das FreeBSD-Project diese Informationen als äßerst nützlich ansieht, wird stattdessen häfig eine Repocopy durchgeführt. Bei diesem Prozess kopiert ein Repositoy Meister die Datei direkt innerhalb des Repository an die neue Position, statt cvs(1) einzusetzen.

Request For Comments — Eine Sammlung von Dokumenten, die wichtige Internetstandards, Protokolle und so weiter definieren und die unter www.rfc-editor.org zu finden sind.

Kann aber auch allgemein verwendet werden, wenn jemand eine Änderung vorschlägt und dazu Feedback möchte.

Request To Send — Ein RS232C-Signal, das der Gegenstelle signalisiert, dass sie mit dem Senden der Daten beginnen kann.
Siehe auch Clear To Send.

Router Advertisement

S

SCI — Siehe System Control Interrupt.

SCSI — Siehe Small Computer System Interface.

SG — Siehe Signal Ground.

SMB — Siehe Server Message Block.

SMP — Siehe Symmetric MultiProcessor.

SMTP — Siehe Simple Mail Transfer Protocol.

SMTP AUTH — Siehe SMTP Authentication.

SSH — Siehe Secure Shell.

STR — Siehe Suspend To RAM.

SVN — Siehe Subversion.

SMTP Authentication

Server Message Block

Signal Ground — Ein RS232-Pin oder -Draht, der als Untergrundreferenz für das Signal verwendet wird.

Simple Mail Transfer Protocol

Secure Shell

Small Computer System Interface

Subversion
: Subversion ist ein Versionskontrollsystem, ähnlich wie CVS, aber mit einer grösseren Liste von Eigenschaften.
Siehe auch Concurrent Versions System.

Suspend To RAM

Symmetric MultiProcessor

System Control Interrupt

T

TCP
: Siehe Transmission Control Protocol.

TCP/IP
: Siehe Transmission Control Protocol/Internet Protocol.

TD
: Siehe Transmitted Data.

TFTP
: Siehe Trivial FTP.

TGT
: Siehe Ticket-Granting Ticket.

TSC
: Siehe Time Stamp Counter.

Ticket-Granting Ticket

Time Stamp Counter
: Ein interner Zähler bei modernen Pentium®-Prozessoren, der die Ticks der *core frequency clock* bestimmt.

Transmission Control Protocol
: Ein Protokoll, das auf dem IP-Protokoll aufsetzt. Es garantiert, dass Datenpakete zuverlässig und geordnet transportiert werden.

Transmission Control Protocol/Internet Protocol
: Die Kombination aus TCP- und IP-Protokoll. Ein Großteil des Internets basiert auf TCP/IP.

Transmitted Data
: Ein RS232C-Pin oder -Draht, über den Daten verschickt werden.
Siehe auch Received Data.

Trivial FTP

U

UDP
: Siehe User Datagram Protocol.

UFS1
: Siehe Unix File System Version 1.

UFS2
: Siehe Unix File System Version 2.

UID
: Siehe User ID.

URL
: Siehe Uniform Resource Locator.

USB
: Siehe Universal Serial Bus.

Uniform Resource Locator
: Eine Methode um eine Ressource, z.B. ein Dokument im Internet, zu lokalisieren und eine Art, diese Ressource zu identifizieren.

Unix File System Version 1
: Das Original UNIX® Dateisystem, manchmal auch das Berkeley Fast File System genannt.

Unix File System Version 2
: Eine Erweiterung für UFS1, eingeführt in FreeBSD5-CURRENT. UFS2 enthält 64-bit Blockzeiger (durchbricht dadurch die 1T Grenze), Unterstützung für extended file storage und andere Merkmale.

Universal Serial Bus
: Ein Hardware-Standard, der verwendet wird um eine grosse Vielfalt von Computerperipherie an eine einheitliche Schnittstelle anzuschliessen.

User ID
: Eine eindeutige Nummer, die einem Benutzer eines Computers zugewiesen wird. Kann zur Identifizierung von zugewiesenen Ressourcen und Berechtigungen verwendet werden.

User Datagram Protocol
: Ein einfaches, nicht-zuverlässiges Protokoll für Datagramme, das beim Datenaustausch in einem TCP/IP Netzwerk benutzt wird. UDP enthält keine Fehlerüberprüfung und -korrektur wie TCP.

V

VPN
: Siehe Virtual Private Network.

Virtual Private Network
: Eine Methode ein öffentliches Netzwerk wie das Internet zu nutzen, um einen entfernten Zugriff auf ein lokales Netz, wie etwa ein Unternehmens-LAN, zu ermöglichen.

Stichwortverzeichnis

Symbole

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

Kolophon

Dieses Buch ist aus den Beiträgen vieler Freiwilliger zum „FreeBSD Documentation Project“ entstanden. Der Text ist in SGML entsprechend der Docbook DTD verfasst. Mit Hilfe von Jade, einem Open Source DSSSL-Prozessor, wird er in verschiedene Formate umgewandelt. Die Umwandlung wird von Norm Walsh's DSSSL Stylesheets und eigens entwickelten Stylesheets gesteuert. Die gedruckte Ausgabe des Buchs wäre ohne die Satzbeschreibungssprache TeX von Donald Knuth, LaTeX von Leslie Lamport oder den JadeTeX-Makros von Sebastian Rahtz nicht möglich.