

# >> Gentoo Linux Frequently Asked Questions

[Bitte Kapitel auswählen] ▀

## 1. Übersicht

### 1.1 Gentoo allgemein

- Wie wird Gentoo ausgesprochen, und was ist damit gemeint?
- Was ist an Gentoo anders??

### 1.2 Installation

- Wo soll ich Bugs melden?
- Wo ist der Unterschied zwischen .iso und .tbz2 Dateien?
- Warum werden .iso und .tbz2 Dateien mit unterschiedlichen -r (Revisionen) erstellt/veröffentlicht?
- Mir erscheint das ganze sehr instabil und ich benutze "-O9 -ffast-math -fomit-frame-pointer" optimierungen. Warum?
- Wie lautet das Standard root Passwort nach der Installation?
- Wie kann ich bei root (oder einem anderen User) das Passwort ändern?
- Wie kann man einen normalen Benutzer hinzufügen?
- Warum kann ein Benutzer nicht mit su zu root werden?
- Wie kann ich devfs aktivieren?
- Wie kann ich devfs deaktivieren?
- Wie erhalte ich bei Benutzung von devfs ein /dev/mouse das nach dem Booten nicht wieder verloren geht?
- Grub kann die stage x.y nicht finden.
- Mein ASUS CUV4X-D will nicht booten und es friert bei manchen Kernel-Prozessen und der Hardware-Suche ein.
- Wenn ich ein Gentoo 1.4\_rc1 habe, kann ich mein System dann ohne Neuinstallation auf 1.4\_rc2, 1.4\_final/\_rc3 aktualisieren?
- Mein Kernel bootet nicht (richtig), was kann ich tun?
- Mein Proxy verlangt Authentifizierung, was habe ich zu tun?

### 1.3 Paketverwaltung

- In welchem Format werden die Pakete gespeichert?
- Warum ein neues Port System schreiben (Portage), wenn man die BSD Version verwenden könnte?
- Wo liegt der Unterschied zu Debian's apt oder den BSD Ports?
- Wie installiere und deinstalliere ich Pakete?
- Wie kann ich globale Einstellungen zum kompilieren der Pakete einstellen?
- Was ist mit /etc/make.defaults passiert?
- Gibt es eine Möglichkeit alle installierten Pakete zu aktualisieren wie z. B. apt-get upgrade oder make world?
- Wenn ich ein Paket mit emerge oder ebuild aktualisiere, wie kann ich verhindern, dass meine config Dateien sich anhäufen?
- Ich möchte den ./configure Schritt selber ausführen. Geht das?
- Was, wenn rsync nicht für mich funktioniert?
- Wie benutze ich emerge hinter einer Firewall?
- Kann ich von einem anderen Betriebssystem aus ein rsync machen?
- Ich habe zu Hause nur eine langsame Modem-Verbindung. Kann ich die Quellen woanders herunterladen und dann zu meinem System hinzufügen?
- Die ganzen .tar.gz Quelldateien sammeln sich in /usr/portage/distfiles/ an und verbrauchen wertvolle Plattenkapazitäten. Können diese Dateien gelöscht werden?
- Ich habe blackdown-jdk und blackdown-jre installiert, aber java-config --list-available-vms zeigt nur blackdown-jre an. OpenOffice.org lässt sich nicht installieren. Was soll ich tun?
- Was ist mit /var/tmp/portage? Ist es sicher die Dateien und Verzeichnisse in /var/tmp/portage zu löschen?

### 1.4 Gentoo Linux benutzen

- Ich habe openssh auf meiner Box installiert, aber ich kann mich nur als root und nicht als normaler Benutzer von außen einloggen.
- Ich kann X Anwendungen nur als root ausführen.

- Wie stelle ich ein internationales Tastatur-Layout ein?
- Die DNS Namensauflösung funktioniert nur für root.
- Warum liest KDE nicht /etc/profile?
- Warum können Benutzer nicht ihre crontab benutzen?
- Wie starte ich den Numlock beim booten?
- Wie lösche ich mein Terminal beim Logout?

## 1.5 Verwaltung

- ReiserFS und Dateisystem Fehler -- und wie man die behebt
- Metalogd logt nicht in Echtzeit!

## 1.6 Entwicklung

- Wie kann ich Fehler melden?
- Wie oft gibt es Neuerscheinungen?
- Ich möchte ein neues Paket in Portage einbringen? Wie gehe ich vor?
- Wie kann ich eine neue Frage zu diesem FAQ hinzufügen?
- make -f Makefile.cvs für eine KDE Anwendung ergibt folgende Fehlermeldung: "invalid unused variable"
- Mein eingebauter Lautsprecher piept wie verrückt während der kompilierung von Mozilla. Wie kann ich Piepstöne in der Konsole ausschalten?

## 1.7 Ressourcen

- Wo kann ich mehr über supervise, das bei Gentoo Linux 1.0\_rc5 und früher benutzt wurde, finden?
- Wo kann man mehr Informationen über Gentoo Linux finden?
- Kann man CDs von Gentoo Linux kaufen?
- Diese FAQ beantwortet nicht meine Frage(n). Was kann ich nun tun?

## 2. Gentoo allgemein

### 2.1 Wie wird Gentoo ausgesprochen, und was ist damit gemeint?

Gentoo ist eine Art von kleinen, schnellen Pinguinen, ausgesprochen "schen-tuu".

### 2.2 Was ist an Gentoo anders?

Gentoo Linux ist eine schnelle, moderne Distribution mit einem sauberen und flexiblen Design -- in dieser Hinsicht kann Gentoo [Slackware](#), [Linux From Scratch](#) oder [BSD](#) Benutzern gefallen. Anders als die meisten Linux Distributionen hat Gentoo ein Paket System, das an die "Ports" von BSD erinnert; damit können Sie sich sicher sein, dass Sie immer die aktuellsten Versionen der Pakete haben.

## 3. Installation

### 3.1 Wo soll ich Bugs melden?

#### Warnung

Berichten Sie alle Bugs an [bugs.gentoo.org](http://bugs.gentoo.org)! Berichten Sie die Bugs nicht Upstream (den originalen Autoren). Berichten Sie Bugs an Gentoo, wenn nötig leiten wir diese an die Autoren weiter.

Wir sind auch im [FreeNode](#) IRC Netzwerk unter `#gentoo-bugs` zu erreichen.

### 3.2 Wo ist der Unterschied zwischen .iso und .tbz2 Dateien?

Die fertigen **.tbz2** Dateien sind ein Minimal-Archiv von System Dateien die Ihnen den bootstrap und die Installation von Gentoo Linux erlauben. Die Fertige **.iso** ist ein komplettes, bootfähiges CD Image, das einen System Kernel, eine komplette Zusammenstellung von Kernel Modulen, wichtigen System Hilfsprogrammen wie *mkfs* und Netzwerk Unterstützung, sowie das Minimal-System des **.tbz2** Archivs enthält. In den meisten Fällen werden Sie das .iso Image auf eine CD brennen, von dieser CD booten und von der Gentoo Boot CD installieren. Sie können Gentoo jedoch auch direkt aus einer bereits vorhandenen Linux Distribution zu installieren. In diesem Fall müssen Sie nur eine .tbz2 Datei downloaden, den Inhalt auf eine Reserve Partitionen entpacken (stellen Sie sicher, dass Sie das *p*

Argument benutzen wenn Sie das Archiv entpacken), dann in das System einen chroot ausführen und über normale Art und Weise installieren.

### 3.3 Warum werden .iso und .tbz2 Dateien mit unterschiedlichen -r (Revisionen) erstellt/veröffentlicht?

Das Minimal-System aus dem .tbz2-Archiv braucht nur erneuert zu werden, wenn signifikante Änderungen am Kern von Gentoo Linux gemacht wurden (z.B. Änderungen am Basislayout oder ein neues Profil) - dadurch sind .tbz2 Updates relativ selten. Das .tbz2 Minimal-System Archiv benötigt nur versioniert zu sein wenn bedeutende Änderungen im Gentoo Linux Basissystem (wie baselayout Änderungen, oder ein neues Profil) gemacht wurden, und solche .tbz2 Updates sind relativ selten. Die .iso Datei muss nur aktualisiert werden, wenn jemand andere Hardware hat die nicht mit der .iso booten will. Seitdem Aktualisierungen der Kernel Module und Updates regelmässig gemacht werden kann es nur selten zu dieser Situation kommen.

### 3.4 Mir erscheint das ganze sehr instabil und ich benutze "-O9 -ffast-math -fomit-frame-pointer" optimierungen. Warum?

Bei Optimierungen über -O3 hinaus, ist das Risiko hoch, dass einzelne Pakete nicht mehr funktionieren. Sehr aggressive Optimierungen können den Assembler Code insoweit bis zu einem Punkt verändern, wo er nicht mehr das tut, was eigentlich von ihm erwartet wird. **Loc-Dog** (im IRC) benutzt -O3 -mcpu=i686 -march=i686 -fforce-addr -fomit-frame-pointer -funroll-loops -frerun-cse-after-loop -frerun-loop-opt -malign-functions=4, um sein System so weit es geht im Bereich des noch tragbaren zu optimieren. Darüber hinaus sollten ultra-hohe Optimierungen nur noch bei einzelnen Paketen angewendet werden, wo die Leistungssteigerung um 2% noch gerechtfertigt ist (Graphik und verschiedene Multimedia-Programme) auch, um die jeweiligen Pakete noch testen zu können und sicherzustellen, dass sie nicht bis in den Status des Vergessens optimiert wurden.

Bitte versuchen Sie zunächst mit folgenden CFLAGS -march= -O2 zu kompilieren, bevor Sie einen Bug melden.

### 3.5 Wie lautet das Standard root Passwort nach der Installation?

Das Standard Passwort ist leer, einfach Enter drücken.

### 3.6 Wie kann ich bei root (oder einem anderen User) das Passwort ändern?

Sie können *passwd* benutzen, um das Passwort für den User, mit dem Sie angemeldet sind, zu ändern. Für zusätzliche Optionen und Einstellungen, schauen Sie sich bitte *man passwd* an, nachdem sie Gentoo Linux fertig installiert haben.

### 3.7 Wie kann man einen normalen Benutzer hinzufügen?

Jeder weiß wohl, dass es Sinn macht nur die Dinge als **root** zu tun, die ein normaler User nicht darf. Wie kann man also andere Benutzer hinzufügen?

Der Befehl *adduser gentoo* würde einen Benutzer mit dem Usernamen gentoo erstellen. Der nächste Schritt ist, diesem User ein Passwort zu geben und *passwd* tut genau das. Anstelle von *adduser* können Sie auch folgendes benutzen:

**Befehlsauflistung 1:** Benutzer hinzufügen

```
# useradd gentoo -m -G users,audio,wheel -s /bin/bash
```

Dies fügt den Benutzer gentoo hinzu. Dieser hat die nötigen Rechte um auf Sound Devices zuzugreifen (*/dev/sound/\**), erlaubt ihm per (*su*) root zu werden und nutzt */bin/bash* als Login-Shell.

Sie können auch *superadduser* mittels *emerge superadduser* installieren und dann *superadduser gentoo* ausführen, um den Benutzer gentoo anzulegen. Folgen Sie einfach den Anweisungen von *superadduser*.

### 3.8 Warum kann ein Benutzer nicht mit su zu root werden?

Aus Sicherheitsgründen können Benutzer nur mit *su* zu root werden, wenn sie in der **wheel** Gruppe sind. Um einen **Username** zu der **wheel** Gruppe hinzuzufügen, benutzen sie den folgenden Befehl als root:

**Befehlsauflistung 2:** Benutzer der Gruppe wheel hinzufügen

```
# usermod -G users,wheel username
```

### 3.9 Wie kann ich devfs aktivieren?

Wenn Sie 1.0\_rc5 oder neuer benutzen, brauchen Sie nichts machen, um devfs zu aktivieren: es ist schon aktiviert (haben sie auch devfs in den Kernel eingebunden?). Wenn sie jedoch, eine Version von Gentoo Linux **vor** der Version 1.0\_rc5 benutzen, fügen sie bitte *devfs=mount* zu Ihrer GRUB Kernel Boot Option hinzu, sodass die Zeile in etwa so aussieht wie *kernel /boot/boot/bzImage devfs=mount foo=bar*. Der Kernel wird auf */dev* das **devfs** Dateisystem automatisch booten.

### 3.10 Wie kann ich devfs deaktivieren?

Bei Gentoo Linux 1.0\_rc6 und neuer kann man devfs durch das Kernelargument *gentoo=nODEVFS* deaktivieren.

### 3.11 Wie erhalte ich bei Benutzung von devfs ein /dev/mouse, das nach dem Booten nicht wieder verloren geht?

Ab Version 1.0\_rc6 oder später, genügt es einfach mit *ln -s* einen symbolischen link von */dev/mouse* zu erstellen, welcher auch bei Reboots erhalten bleibt.

Alle anderen Benutzer sollten Ihre */etc/devfsd.conf* editieren und folgende Zeilen hinzufügen:

**Befehlsauflistung 3:** devfsd.conf für /dev/mouse editieren

```
REGISTER          ^misc/psaux$      CFUNCTION GLOBAL symlink misc/psaux mouse
UNREGISTER        ^misc/psaux$      CFUNCTION GLOBAL unlink mouse
```

Benutzen Sie eine andere Maus, als die standardmäßig vorgegebene devfs PS/2 Maus unter */dev/misc/psaux*, sollten die *misc/psaux* Zeilen entsprechend abgeändert werden. Anschließend sollte noch *killall -HUP devfsd* ausgeführt werden, um die Pfadangaben in */etc/devfsd.conf* erneut einzulesen.

### 3.12 Grub kann die stage x.y nicht finden.

Während der Installation hat grub die boot Dateien nach */boot/grub* (*/boot/boot/grub* in Gentoo Linux 1.0\_rc5 und älter) kopiert. Grub sucht automatisch das Verzeichnis */boot/grub* auf der Boot Partition ab. (Wir empfehlen zur Sicherheit die boot Partition auf */boot* nicht automatisch zu mounten, da es so schwieriger ist aus Versehen den Kernel oder die Boot-Informationen zu zerstören.) Der obige Fehler tritt deshalb meistens auf, wenn (a) */boot* nicht auf einer separaten Partition installiert wird, oder (b) die Boot Partition auf */boot* vergessen wurde zu mounten bevor man das Minimal-System Archiv entpackt oder *emerge --usepkg system* ausgeführt hat, oder (c) vergessen des *notail* Argumentes wenn */boot* eine ReiserFS Partition ist. Sie können mehr Informationen über Grub in der Debug Anleitung finden, die in der Standard Installation vorhanden ist. Dazu geben Sie einfach **debug grub** von dem Grub Terminal oder lesen Sie das [IBM DeveloperWorks Grub Tutorial](#).

### 3.13 Mein ASUS CUV4X-D will nicht booten und es friert bei manchen Kernel-Prozessen und der Hardware-Suche ein.

Deaktivieren sie MPS 1.4 (multi-processor-system) im BIOS oder stellen sie diese Funktion auf 1.1. Beim benutzen dieser Option verändern sie einfach die MPS Version. Das Multi-Processor-System wird weiterhin richtig arbeiten. Stellen sie sicher das Gentoo Linux mit der Boot Option "noapic" bootet.

### 3.14 Wenn ich ein Gentoo 1.4\_rc1 habe, kann ich mein System dann ohne Neuinstallation auf 1.4\_rc2, 1.4\_final/\_rc3 aktualisieren?

**Nach der Installation** gibt es keinen Unterschied zwischen 1.4 Releases. Gentoo 1.4 und neuer sind *glibc-2.3.x* basierend. Zum Beispiel eine 1.4rc1 Maschine auf der Sie *emerge sync; emerge -u world* ausführen, ist **genauso aktuell** wie eine Maschine mit installiertem 1.4rc2, nachdem auch dort ein

*emerge sync*; *emerge -u world* ausgeführt wurde. Die wirklichen Unterschiede liegen in der Installation.

### 3.15 Mein Kernel bootet nicht (richtig), was kann ich tun?

Sie brauchen nicht jeden Schritt der Installation zu wiederholen, es genügen die Schritte zum Kernel Bau und davon abhängende. Vorausgesetzt Sie haben Gentoo auf `/dev/hda1` (/boot) und `/dev/hda3` (/) mit `/dev/hda2` als Swap installiert gilt folgendes:

**Befehlsauflistung 4:** Rekonfiguration des Kernel

Booten von der LiveCD, warten bis zum Login-Prompt

Zunächst mounten Sie alle Partitionen:

```
# mount /dev/hda3 /mnt/gentoo
# mount /dev/hda1 /mnt/gentoo/boot
# swapon /dev/hda2
# mount -t proc none /mnt/gentoo/proc
```

Dann chrooten Sie in ihre Gentoo Umgebung und konfigurieren den Kernel:

```
# chroot /mnt/gentoo /bin/bash
# env-update && source /etc/profile
# cd /usr/src/linux
# make menuconfig
```

Nun können sie alles an/abwählen, dass sie bei Ihrem vorherigen Versuch falsch ausgewählt hatten. Dann beenden Sie menuconfig und kompilieren den Kernel:

```
# make dep && make bzImage modules modules_install
```

Nun können Sie ihre bzImage Datei über den alten Kernel kopieren:

```
# cp arch/i386/boot/bzImage /boot
```

Wenn Sie LILO benutzen führen Sie lilo aus -- GRUB Benutzer können diesen Schritt überspringen:

```
# /sbin/lilo
```

Nun verlassen Sie die chroot Umgebung und starten neu.

```
# exit
# umount /mnt/gentoo/proc /mnt/gentoo/boot /mnt/gentoo
# reboot
```

Wenn Sie ein Problem mit der Konfiguration Ihres Bootloaders haben folgen Sie den gleichen Schritten, anstelle der Konfiguration/Kompilation des Kernel sollten Sie ihren Bootloader konfigurieren (neukompilieren ist nicht notwendig).

### 3.16 Mein Proxy verlangt Authentifizierung, was habe ich zu tun?

Wenn Sie etwas mit *wget* herunterladen möchten, folgen Sie dieser Syntax, um sich zu authentifizieren:

**Befehlsauflistung 5:** Proxy Authentifizierung mit wget

```
# wget --proxy-user=username --proxy-passwd=password <url>
```

Um Portage automatisch mit Proxy Authentifizierung zu benutzen editieren Sie die `/etc/make.conf` wie folgt:

**Befehlsauflistung 6:** /etc/make.conf für Proxy-Benutzung editieren

```
FETCHCOMMAND="wget --proxy-user=username --proxy-passwd=password -t 5 --passive-ftp -P \${DISTDIR}
RESUMECOMMAND="/usr/bin/wget --proxy-user=username --proxy-passwd=password -c -t 5 --passive-ftp"
```

Traurigerweise scheint *rsync* Proxy Authentifizierung nicht zu unterstützen. Schauen Sie in "[Was, wenn rsync nicht für mich nicht funktioniert?](#)" für weitere Informationen nach.

## 4. Paketverwaltung

### 4.1 In welchem Format werden die Pakete gespeichert?

Die Pakete existieren in unserem Portage "Baum/Stammbaum/Archiv" als **ebuild**, den Installations Skripten. Gentoo ist hauptsächlich eine Ports-basierende Distribution. Darunter versteht man, dass

Gentoo Skripte (*.ebuild* Dateien) und ein spezielles System (Portage) bereitstellt, um Programme bequem aus dem Quellcode bauen zu können.

## 4.2 Warum ein neues Port System schreiben (Portage), wenn man die BSD Version verwenden könnte?

In einem Satz: Weil Portage in vielen Dingen viel besser ist. Eine der Entwicklungsphilosophien von der *.ebuild* Syntax ist die Analogie zu dem, was Sie an der Kommandozeile eingeben würden, um die Pakete von Hand zu installieren. Somit sollte Portage sehr einfach zu erlernen sein und auch ohne großen Aufwand an die eigenen Bedürfnisse angepasst werden können. Wir haben auch "gefälschte" Installationen im OpenBSD-Stil, sicheres Entfernen, Systemanpassungen, Paketmaskierung, ein echtes System, welches Abhängigkeiten berücksichtigt und noch viele andere guten Sachen.

## 4.3 Wo liegt der Unterschied zu Debian's apt oder den BSD Ports?

Portage enthält die besten Funktionen von apt und Ports; zum Beispiel die USE Optionen, ein komplettes Abhängigkeitssystem, sicheres Installieren und Deinstallieren, eine Paket-Datenbank. Stellen Sie sich Portage als die beste beider Welten vor; ein Ports-System das die Sensibilität und Sicherheit eines Linux Paket Management Systems enthält.

## 4.4 Wie installiere und deinstalliere ich Pakete?

Der [Portage User Leitfaden](#) erklärt ausführlich, wie man Pakete installiert, deinstalliert und Portage aktualisiert.

## 4.5 Wie kann ich globale Einstellungen zum Kompilieren der Pakete einstellen?

In der */etc/make.conf* können Sie globale und/oder Profil-spezifische Standardeinstellungen anpassen, die beim Kompilieren und Installieren der Pakete verwendet werden. Mit den folgenden Optionen sollten Sie sich auf jeden Fall befassen:

| Flag           | Beschreibung   |
|----------------|--|
| CHOST          | Dies setzt die HOST Variable für Compiler, z.B. <i>i686-pc-linux-gnu</i>   |
| CFLAGS         | Die Optionen für den <i>gcc</i> zum Kompilieren von Programmen, die in C (*.c Dateien) geschrieben sind  |
| CXXFLAGS       | Die Optionen für den <i>gcc</i> zum Kompilieren von Programmen, die in C++ (*.c,*.cpp etc. Dateien) geschrieben sind   |
| USE            | Dies Erlaubt das Setzen von gewünschten Komponenten, die mit hinein kompiliert werden sollen, sofern diese vorhanden sind. Zum Beispiel, wenn Sie <i>gnome</i> innerhalb der USE Zeile haben, und Sie <i>xchat</i> kompilieren, wird dies die Gnome Unterstützung von xchat hinzufügen. Auch alle Paket-Abhängigkeiten machen von den Veränderungen in der USE-Zeile Gebrauch. Durch Leerzeichen getrennte Liste von URIs die Gentoo Pakete spiegeln. Portage versucht zunächst von einem <i>GENTOO_MIRROR</i> herunter zu laden, bevor es die offizielle <i>SRC_URI</i> ausprobiert. Um das Überspringen von Mirrors zu erzwingen, kann diese Variable auf "" gesetzt werden. |
| GENTOO_MIRRORS |  |

## 4.6 Was ist mit /etc/make.defaults passiert?

Seit Portage 1.5 und danach ist */etc/make.defaults* veraltet; wenn Sie portage-1.5-r1 oder neuer installiert haben, können Sie es ohne Probleme löschen. Die Datei wurde ersetzt durch */etc/make.profile/make.defaults* (*/etc/make.profile* sollte aktuell ein Symbolischer-Link nach */usr/portage/profiles/default* sein), welche System-Profil-Spezifische Standard Einstellungen enthält. Die Prioritätsreihenfolge der unterschiedlichen Konfigurations-Dateien sieht wie folgend aus: (höchste Priorität zuerst):

- Umgebungs-Variablen
- */etc/make.conf*, für Ihre Verwendung
- */etc/make.profile/make.defaults*, für Profil-Spezifische Standards
- */etc/make.globals*, für globale Standards (Einstellungen die nicht speziell an irgend einer anderen Stelle gesetzt wurden, kommen von hier)

## 4.7 Gibt es eine Möglichkeit alle installierten Pakete zu aktualisieren wie z. B. apt-get upgrade oder make World?

**JA!** Geben Sie `emerge --update` ein (erst mit `--pretend` aufrufen), um die Kern-System-Pakete zu aktualisieren, und `emerge --update world` (wieder erst mit `--pretend` aufrufen), um ein komplettes System Upgrade aller installierten Pakete zu machen.

#### 4.8 Wenn ich ein Paket mit emerge oder ebuild aktualisiere; wie kann ich verhindern, dass meine Config-Dateien sich anhäufen?

Portage enthält standardmäßig eine Unterstützung für Config-Dateien. `emerge --help config` gibt darüber mehr Informationen. Grund dafür ist, dass ein Paket namens `foo` irgendeine Config-Datei unter `/etc` ablegt und eine Config-Datei eines anderen `foo` Paketes bereits darin existiert, was zum Überschreiben dieser Datei führen würde. Die neue Config-Datei von `foo` wird deshalb als `._cfgxxxx_foo` in diesem Verzeichnis abgespeichert. Ein sinnvolles Werkzeug zum Untersuchen und Aktualisieren von geschützten config Dateien ist `etc-update`, das nun Teil von Portage ist.

#### 4.9 Ich möchte den ./configure Schritt selber ausführen. Geht das?

Ja, aber es ist nicht ganz einfach und die nachfolgende Methode funktioniert nur, wenn es um ein einfaches Ebuild geht. (Zum Beispiel nur `./configure` und `make && make install`). Schauen Sie zunächst ins Ebuild um zu sehen, was Gentoo macht.

Als erstes sollten Sie die Sourcen mittels `ebuild /usr/portage/<kategorie>/<paket>/<ebuild> unpack` entpacken.

Wechseln Sie nun nach `/var/tmp/portage/<paket>-<version>/work`. Hier finden Sie die entpackten Sourcen. Führen Sie die nötigen Schritte aus, um das Paket zu konfigurieren und kompilieren.

Wenn Sie fertig sind, führen Sie `touch /var/tmp/portage/<paket>-<version>/compiled` aus, um Portage glauben zu lassen, dass das Paket konfiguriert und kompiliert wurde. Schliessen Sie die Installation mit `ebuild /usr/portage/<kategorie>/<paket>/<ebuild> merge` ab.

#### 4.10 Was, wenn rsync nicht für mich funktioniert?

Wenn sich Ihre Maschine hinter einer Firewall befindet, die keinen rsync Verkehr erlaubt, können Sie `emerge-webrsync` nutzen, das einen Portage Snapshot über normales HTTP herunterlädt und installiert. `emerge-webrsync` benutzt `wget` zum Herunterladen, Proxys können also benutzt werden.

**Befehlsauflistung 7:** Benutzung von emerge-webrsync

```
~# emerge-webrsync
```

Wenn auch dies für Sie kein Weg ist, können Sie manuell einen Snapshot von <http://distro.ibiblio.org/pub/linux/distributions/gentoo/snapshots/> herunterladen. Um diesen Snapshot korrekt zu installieren müssen Sie zunächst ihr aktuelles `/usr/portage` löschen, damit keine veralten Ebuilds auf Ihrem System verbleiben. Sicherlich wollen Sie ihr Distfiles Verzeichnis `/usr/portage/distfiles` sichern, um nicht alle bisher heruntergeladenen Sourcecodes zu verlieren.

**Befehlsauflistung 8:** Manuelle Installation eines Portage Snapshot

```
// (Zunächst laden Sie einen Snapshot herunter und schieben in nach /usr)
~# cd /usr
~# mv /usr/portage/distfiles /usr/distfiles-temp
~# rm -rf /usr/portage
~# tar xvjf portage-foo.tbz2
~# mv /usr/distfiles-temp /usr/portage/distfiles
```

#### 4.11 Wie benutze ich emerge hinter einer Firewall?

Verändern Sie die PROXY Einstellungen in `/etc/make.conf`. Wenn das nicht funktioniert, bearbeiten Sie `/etc/wget/wgetrc` und passen Sie `http_proxy` und `ftp_proxy` entsprechend an.

#### 4.12 Kann ich von einem anderen Betriebssystem aus ein rsync machen?

Es gibt ein Programm Namens unison, das sowohl unter UNIX als auch unter Win32 läuft. Es ist verfügbar unter: <http://www.cis.upenn.edu/~bcpierce/unison/>.

#### 4.13 Ich habe zu Hause nur eine langsame Modem-Verbindung. Kann ich die Quellen woanders herunterladen und dann zu meinem System hinzufügen?

Natürlich. Mit dem Befehl `emerge --pretend <package>` kann man sehen, welche Pakete für die Installation benötigt werden. Die heruntergeladenen Quelldateien können nun über ein Medium nach Hause genommen werden und in das `/usr/portage/distfiles` Verzeichnis kopiert. Anschließend muss nur noch `emerge <package>` ausgeführt werden, welches nun die mitgebrachten Quelldateien verwendet.

#### 4.14 Die ganzen .tar.gz Quelldateien sammeln sich in /usr/portage/distfiles/ an und verbrauchen wertvolle Plattenkapazitäten. Können diese Dateien gelöscht werden?

Ja, das ist ohne Probleme möglich. Aber wenn Sie eine langsame Internetverbindung besitzen, sollten Sie die Archive möglichst behalten, da manchmal für ein einziges Stück Software mehrere verschiedene Ebuilds existieren -- nach dem Löschen des Archives muss das Paket neu aus dem Internet geladen werden.

#### 4.15 Ich habe blackdown-jdk und blackdown-jre installiert, aber java-config --list-available-vms zeigt nur blackdown-jre an. OpenOffice.org lässt sich nicht installieren. Was soll ich tun?

Lösung:

**Befehlsauflistung 9:** Lösung

```
# emerge unmerge blackdown-jre blackdown-jdk
# CONFIG_PROTECT="" emerge blackdown-jdk
```

#### 4.16 Was ist mit /var/tmp/portage? Ist es sicher die Dateien und Verzeichnisse in /var/tmp/portage zu löschen?

Während der Kompilation der Pakete speichert Gentoo die Quellen der Pakete in `/var/tmp/portage`. Es ist sicher diese Dateien und Verzeichnisse zu löschen.

### 5. Gentoo Linux benutzen

#### 5.1 Ich habe openssh auf meiner Box installiert, aber ich kann mich nur als root und nicht als normaler Benutzer von außen einloggen.

Dies kann passieren wenn der Benutzer Account nicht eine richtige Shell eingestellt hat. Überprüfen Sie den Benutzer-Eintrag in `/etc/passwd` und sehen Sie nach ob die Zeile mit `/bin/bash` (oder einer anderen Shell) endet. Wenn dies nicht der Fall ist, müssen Sie eine Shell für den User einstellen. Dies ist mit dem `usermod` Befehl möglich.

**Befehlsauflistung 10**

```
# usermod -s /bin/bash meinuser
```

#### 5.2 Ich kann X Anwendungen nur als root ausführen.

Ihr `/tmp` Verzeichnis hat die falschen Zugriffsrechte. Geben Sie folgendes als root ein:

**Befehlsauflistung 11**

```
# chmod 1777 /tmp
```

#### 5.3 Wie stelle ich ein internationales Tastatur-Layout ein?

Passen Sie die `KEYMAP` Variable in `/etc/rc.conf` an. Entweder booten Sie danach das System, oder starten Sie das Keymaps-Script neu: `/etc/init.d/keymaps restart`.

## 5.4 Die DNS Namensauflösung funktioniert nur für root.

`/etc/resolv.conf` hat die falschen Zugriffsrechte. `chmod` es zu folgendem:

### Befehlsauflistung 12

```
# chmod 0644 /etc/resolv.conf
```

## 5.5 Warum liest KDE nicht `/etc/profile`?

Man muss `--login` in der ersten Zeile in `/opt/kde2.1/bin/startkde` hinzufügen, damit es so aussieht:

### Befehlsauflistung 13

```
#!/bin/sh --login
```

Diese Änderung wurde in neueren KDE Versionen bereits vorgenommen.

## 5.6 Warum können Benutzer nicht ihre crontab benutzen?

Sie müssen die Benutzer der `cron` Gruppe hinzufügen.

## 5.7 Wie starte ich den numlock beim booten?

Wenn Sie sich grafisch einloggen oder den numlock aktivieren möchten, wenn Sie `startx` ausführen, müssen Sie `numlockx` installieren und `/usr/X11R6/bin/numlockx` in die `/etc/X11/xinit/xinitrc` (für `startx`) oder `/etc/X11/Sessions/` (für jeden grafischen Login-Manager) wie zum Beispiel `/etc/X11/Sessions/Gnome` für GDM eintragen.

Wenn Sie auf der Kommandozeile arbeiten, brauchen Sie nur numlock dem Default Runlevel hinzufügen (`rc-update add numlock default`) und numlock wird beim nächsten Neustart aktiviert.

## 5.8 Wie lösche ich mein Terminal beim Logout?

Um das Terminal zu Löschen, fügen Sie `clear` zu Ihrem `~/bash_logout` Script hinzu:

### Befehlsauflistung 14: Löschen des Terminals beim Logout

```
$ echo clear >> ~/.bash_logout
```

Wenn Sie diese Funktion für jeden neu angelegten Benutzer wünschen, tun Sie das selbe für die `/etc/skel/.bash_logout`:

### Befehlsauflistung 15: Löschen des Terminals beim Logout für neu angelegte Benutzer

```
# echo clear >> /etc/skel/.bash_logout
```

## 6. Verwaltung

### 6.1 ReiserFS und Dateisystem Fehler -- und wie man die behebt

Wenn Ihre ReiserFS Partition fehlerhaft ist, versuchen Sie die Gentoo Linux Boot CD zu booten und starten Sie `reiserfsck --rebuild-tree` auf dem fehlerhaften Dateisystem. Dies sollte das Dateisystem wieder stabil machen, manchmal verliert man aber manche Dateien und Verzeichnisse durch das fehlerhafte Dateisystem.

### 6.2 Metalogd logt nicht in Echtzeit!

Metalog speichert die Ausgaben in Blöcken auf die Festplatte. Wenn Sie versuchen einen Daemon zu debuggen, ist diese Performance-Verbesserung nicht gerade hilfreich. Wenn Ihr Gentoo Linux System gestartet ist und läuft, können Sie metalog ein `USR1` Signal schicken, um die Zwischenspeicherung temporär abzuschalten (Mit `tail -f /var/log/everything/current` können Sie nun in Echtzeit auf die Logs zugreifen). Mit einem `USR2` Signal schalten Sie die Zwischenspeicherung wieder an. Wenn Sie dieses

Feature permanent deaktivieren wollen, können Sie in der `/etc/conf.d/metalog` `METALOG_OPTS="-B"` in `METALOG_OPTS="-B -s"` ändern.

**Befehlsauflistung 16:** Zwischenspeicherung von Metalog an/ausschalten

```
// Um die Zwischenspeicherung auszuschalten:  
# killall -USR1 metalog  
// Um die Zwischenspeicherung anzuschalten:  
# killall -USR2 metalog
```

## 7. Entwicklung

### 7.1 Wie kann ich Fehler melden?

#### Warnung

Berichten Sie alle Bugs an [bugs.gentoo.org](http://bugs.gentoo.org)! Berichten Sie die Bugs nicht Upstream (den originalen Autoren). Berichten Sie Bugs an Gentoo, wenn nötig leiten wir diese an die Autoren weiter.

Wir sind auch im [FreeNode](#) IRC Netzwerk unter `#gentoo-bugs` zu erreichen.

### 7.2 Wie oft gibt es Neuerscheinungen?

Neuerscheinungen werden in der [gentoo-announce](#) Mailliste angekündigt. Im Normalfall werden die Pakete aktualisiert, wenn von den Autoren neuer Quellcode bereitgestellt wird. Neue CD Images werden in der Regel nur dann erstellt, wenn große Veränderungen am Basissystem vorgenommen wurden, oder neue Module hinzugefügt werden.

### 7.3 Ich möchte ein neues Paket in Portage einbringen? Wie gehe ich vor?

Unter <http://bugs.gentoo.org> sollte einfach ein Bug von der Sorte "ebuild" erstellt werden. Das ebuild sollte dann an diese Meldung angehängt werden.

### 7.4 Wie kann ich eine neue Frage zu diesem FAQ hinzufügen?

Ebenfalls unter <http://bugs.gentoo.org> einen neuen Bugreport erstellen und in der Kategorie "Docs-user" einreichen. Dies gilt allerdings nur für die englischen Originaldokumente. Für die deutschen Übersetzungen genügt eine Mail an [gentoo-deutsch-dev@lists.berlios.de](mailto:gentoo-deutsch-dev@lists.berlios.de)

### 7.5 `make -f Makefile.cvs` für eine KDE Anwendung ergibt folgende Fehlermeldung: "invalid unused variable"

Für alle KDE Projekte sollte vorher `WANT_AUTOMAKE_1_4=1` exportiert werden, bevor der Befehl `make -f Makefile.cvs` ausgeführt wird. Für KDE2 Anwendungen gilt analog: `WANT_AUTOCONF_2_1=1`, und für KDE3: `WANT_AUTOCONF_2_5=1`.

### 7.6 Mein eingebauter Lautsprecher piept wie verrückt während der Kompilierung von Mozilla. Wie kann ich Piepstöne in der Konsole ausschalten?

Piepstöne in der Konsole können mit `setterm` geändert werden:

**Befehlsauflistung 17**

```
# setterm -blength 0
```

Soll dies bei jedem Start ausgeführt werden, so muss dieser Befehl in `/etc/conf.d/local.start` eingefügt werden. Dies deaktiviert allerdings nur das aktuelle Terminal. Wenn alle Terminals stumm geschaltet werden sollen, sollte der Befehl auch auf die anderen Terminals geleitet werden:

**Befehlsauflistung 18**

```
# setterm -blength 0 >/dev/vc/1
```

Für `/dev/vc/1` muss das gewünschte Terminal eingetragen werden.

## 8. Ressourcen

### 8.1 Wo kann ich mehr über supervise, das bei Gentoo Linux 1.0\_rc5 und früher benutzt wurde, finden?

<http://cr.yip.to/daemontools.html>

### 8.2 Wo kann man mehr Informationen über Gentoo Linux finden?

Die Offizielle Gentoo Dokumentation kann bei <http://www.gentoo.org> gefunden werden. Die Webseite des Deutschen Übersetzungsteam finden Sie auf <http://www.gentoo.de>. Standard Linux Information sind unter <http://www.tldp.org> verfügbar.

### 8.3 Kann man CDs von Gentoo Linux kaufen?

In Deutschland kann man Gentoo Linux 1.4 CD's unter anderem bei <http://www.lin24.de> und <http://www.liniso.de> finden.

### 8.4 Diese FAQ beantwortet nicht meine Frage(n). Was kann ich nun tun?

Ein guter erster Schritt ist das durchschauen der wichtigen Dokumentationen - [Englisch](#) oder [Deutsch](#) (nicht vollständig), falls dies nichts bringt, durchsuchen Sie das [Forum](#), die vielen verschiedenen Gentoo Linux Mailinglisten unter [Google](#). Zum Durchsuchen der Gentoo Mailinglisten, geben Sie einfach "lists.gentoo.org foo" ein, um "foo" zu suchen. Falls alles andere nicht funktioniert, oder einfach nur um mit anderen Gentooists zu reden, besuchen Sie uns im irc: **#gentoo.de** auf **irc.freenode.net**. Desweiteren finden Sie die gentoo.de FAQ unter [gentoo.de](#). Sie enthält Erweiterungen zu dieser FAQ.



## >> Gentoo Linux Frequently Asked Questions 2

[Bitte Kapitel auswählen] ▣

### 1. Installation

#### 1.1 Installation mit Stage 1 über ein Proxy funktioniert nicht.

Stellen Sie sicher, dass in der `/etc/make.conf` der `RSYNC_PROXY` angegeben ist. Erlaubt der Proxy nur Verbindungen über die Ports 80, so sollte dies mit:

##### Befehlsauflistung 1

```
export http_proxy=123.123.123.123:8080
export ftp_proxy=123.123.123.123:8080
```

eingestellt werden.

#### 1.2 Wie installiere ich eine USB Maus unter Gentoo?

Stellen Sie sicher, dass im Kernel USB Unterstützung aktiviert ist. Folgende Optionen sollten direkt in den Kernel (X) oder als Modul (M) einkompiliert werden:

##### Befehlsauflistung 2

```
Input Core Support
(M) Input Core Support
(M) Mouse Support

USB support
(X) Support for USB
(X) Preliminary USB device filesystem
(M) UHCI (Intel PIIX4, VIA, ...) support oder "UHCI Alternate Driver (JE) Support
(M) USB Human Interface Device (full HID) support
(X) HID input layer support
```

Dies erstellt die Module `usb-uhci.o` (oder: `uhci.o`), `hid.o`, `input.o`, `mousedev.o`. Diese werden dann in `/etc/modules.autoload` noch eingetragen, damit sie auch beim Start verfügbar sind:

##### Befehlsauflistung 3

```
input
mousedev
hid
usb-uhci (oder uhci)
```

Nun muss dies in `/etc/X11/XFConfig` noch eingetragen werden:

##### Befehlsauflistung 4

```
Option "Device" "/dev/input/mice"
```

und die USB Maus sollte funktionieren.

#### 1.3 Wie installiere ich Gentoo auf einem Motherboard mit dem nforce Chipsatz?

Für den Onboard Netzwerkcontroller ist ein spezieller Treiber ist auf der nVidia [Webseite](#), zu haben. Falls Sie keine externe Netzwerkkarte während der Installation zur Hand haben, kopieren Sie diesen Treiber einfach auf ein Medium und verfahren Sie wie bei einer Stage 3 Installation bis zu der Stelle, nach dem Sie den Kernel kompiliert haben. Dann entpacken Sie das tar Archiv und wechseln Sie in das Unterverzeichnis `nforce/nvnet/` und führen Sie folgendes aus:

##### Befehlsauflistung 5

```
# make clean
# make
# make install
```

Dies muss bei jeder Kernel Neukompilierung erneut ausgeführt werden. Dann sollte noch in die `/etc/modules.conf` um folgende Zeilen erweitert werden:

#### **Befehlsauflistung 6**

```
alias eth0 nvnet
```

Anschliessend sollte wie gewohnt nach der Installationsanleitung weiter verfahren werden. ASUS Motherboards mit dem nforce Chipsatz besitzen ein modifizierten Netzwerkcontroller. Der Realtek 8139TOO ist hier die richtige Wahl.

## **2. Laufendes System**

### **2.1 Was genau hat es mit ACCEPT\_KEYWORDS auf sich ?**

Es gibt einen Portage-Tree mit allen nur erdenklichen Paketen. Welche Pakete für Ihren Rechner bestimmt sind, wird durch ein Schlüsselwort festgelegt. Für jede Architektur gibt es im Portage-Tree stabile Pakete und Pakete, die noch im Testbetrieb sind. Die Schlüsselwörter sind *arch* für stabile Ebuilds und *~arch* für Pakete im Testbetrieb (wobei arch hier mit einem gültigen Wert zu besetzen ist z.B. x86 oder ~x86. Nähere Informationen in `/etc/make.conf`).

Um nun ein noch im Testbetrieb befindliches Paket (also ein mit ~x86 maskiertes) zu installieren kann man auf zwei Weisen vorgehen. Zum einen können Sie die Variable nur für das aktuelle Paket setzen:

#### **Befehlsauflistung 7**

```
ACCEPT_KEYWORDS="~x86" emerge testingpaket
```

oder Sie können die Konfigurationsvariable global in `/etc/make.conf` setzen. In dieser Datei finden Sie auch weitere Erklärungen zur Maskierung.

### **2.2 Wenn ich meinen Kernel aktualisiere, habe ich keinen Sound mehr. Ausserdem bekomme ich Fehlermeldungen beim booten, dass die Module der Soundkarte nicht gefunden werden.**

Wenn ALSA Sound verwendet wird, muss nach jeder Kernel-Neukompilation auch die ALSA Treiber neu kompiliert werden. Stellen Sie sicher, dass `/usr/src/linux` auf die richtigen Quelldateien verweist und dann starten sie:

#### **Befehlsauflistung 8**

```
# emerge alsa-driver
```

Eine ausführliche ALSA-Installationsanleitung finden Sie im Desktop-Guide.

### **2.3 Wie kann ich aus meiner NVIDIA Karte noch mehr Leistung herausquetschen?**

Die Zauberwörter heissen hier SBA (Side Band Adressing) und AFW (AGP Fast Writes). Als erstes, sollte man schauen ob die NVIDIA Karte dies überhaupt unterstützt:

#### **Befehlsauflistung 9**

```
cat /proc/driver/nvidia/agp/card
Fast Writes:      Supported
SBA:              Supported
AGP Rates:       4x 2x 1x
Registers:       0x1f000217:0x1f000314
```

Nun muss evtl. im BIOS des Motherboards "Fast Writes" aktiviert werden und wir kontrollieren ob das Motherboard hier mitspielt:

#### Befehlsauflistung 10

```
$ cat /proc/driver/nvidia/agp/status
Status:      Enabled
Driver:      AGPGART
AGP Rate:    4x
Fast Writes: Disabled
SBA:        Disabled
```

Nun editieren wir `/etc/modules.d/nvidia` um folgende Zeilen:

#### Befehlsauflistung 11

```
alias char-major-195 NVdriver
options NVdriver NVreg_EnableAGPSBA=1 NVreg_EnableAGPFW=1
```

und rebooten unser System. Nun sollte es aktiviert sein:

#### Befehlsauflistung 12

```
$ cat /proc/driver/nvidia/agp/status
Status:      Enabled
Driver:      AGPGART
AGP Rate:    4x
Fast Writes: Enabled
SBA:        Enabled
```

und alle EGO Shooter noch etwas flüssiger laufen.

## 3. Netzwerk

### 3.1 Das Mounten von NFS Verzeichnissen dauert sehr lange (teilweise bis zu fünf Minuten). Auch der Bootvorgang ist davon betroffen.

Für das Mounten eines Verzeichnisses via NFS braucht das System bis zu fünf Minuten. Beim Einbinden der NFS Verzeichnisse aus der `/etc/fstab` beim Booten bleibt der Bootvorgang beim Punkt NFS Verzeichnisse stehen und wird, wenn überhaupt, erst nach Minuten fortgesetzt. Die Erklärung ist simpel, zum Zeitpunkt des Mountens fehlt Portmap. Wenn Sie regelmäßig mit NFS arbeiten, empfiehlt es sich, portmap zum standard Runlevel zu ergänzen:

#### Befehlsauflistung 13

```
# rc-update add portmap default
```

## 4. gentoo.de

### 4.1 Wie benutze ich die gentoo.de Ebuilds von BerliOS?

Seit Portage Version 2.0.21 existiert eine einfache Möglichkeit "inoffizielle" Ebuilds zu benutzen. Dies geschieht indem man die Environmentvariable **PORTDIR\_OVERLAY** entsprechend setzt. Z.B. in `/etc/profile` durch

#### Befehlsauflistung 14

```
export PORTDIR_OVERLAY=/usr/local/ebuilds
```

Um die Ebuilds von BerliOS dort abzulegen einfach folgende Befehle (als root) ausführen:

#### Befehlsauflistung 15

```
# cd /usr/local
# cvs -d:pserver:anonymous@cvs.gentoo-deutsch.berlios.de:/cvsroot/gentoo-deutsch
# cvs -z3 -d:pserver:anonymous@cvs.gentoo-deutsch.berlios.de:/cvsroot/gentoo-deutsch
```

Fertig. Diese Befehle müssen nur einmal ausgeführt werden. Zum Aktualisieren der Ebuilds ist später nur noch folgendes (wiederum als root) nötig:

**Befehlsauflistung 16**

```
# cd /usr/local/ebuilds
# cvs update -Pd
```

## 4.2 Wie wandle ich meine übersetzten XML Dokumente in HTML um?

Dazu braucht man xsltproc, welches mit emerge libxslt installiert werden kann. Nun wechselt man lokal in das Verzeichnis www-xml und wandelt die Datei wie folgt um:

**Befehlsauflistung 17**

```
"xsltproc ../www-xsl/guide-main-german.xsl datei.xml >
../www-berlios/htmlfromxsl/datei.html"
```

Gar nicht so schwer, oder?



## >> Gentoo Linux 1.4 Installationsanweisungen

[Bitte Kapitel auswählen]

### 1. Über die Installation

#### 1.1 Einführung

Willkommen zu Gentoo Linux! Gentoo Linux kann über verschiedene Wege installiert werden. Wenn Sie schnell installieren möchten, können Sie fertig erstellte Pakete installieren. Oder wünschen Sie andererseits eine komplette Anpassung, können Sie Gentoo Linux von Grund auf aus dem Quellcode kompilieren. Die Wahl der Methode liegt bei Ihnen.

Eine entscheidende Änderung in unserem offiziellen 1.4 Release ist unser neues 2-CD Installationsset, das in unserem [Gentoo Linux Store](#) geordert werden kann, als auch auf unseren [Mirror-Servern](#) bereitliegt. Wir haben derzeit 2-CD Installationssets für x86 (486 und neuer), i686 (Pentium Pro, Pentium II, Athlon/Duron und neuer), Pentium III, Pentium 4 und Athlon XP. Welches 2-CD-Set das richtige für Sie ist, können Sie in der detaillierten Beschreibung in unserem [Laden](#) nachlesen. Diese Beschreibungen beinhalten recht ausführliche Informationen über die CPU-Kompatibilitäten.

Die ISOs der Live CDs befinden sich auf vielen unserer [Mirror-Servern](#). Die Live CDs für die x86 Architektur sind im Unterverzeichnis [releases/x86/1.4/livecd/](#) abgelegt.

#### 1.2 Die Installations-CDs

Soweit zum 2-CD-Set -- hier ist beschrieben, was auf jeder CD ist. Die erste CD ("CD 1") heißt "Live CD Installation" und ist eine bootbare CD-ROM, das heißt, Sie können "CD 1" in Ihr CD-ROM-Laufwerk einlegen und Gentoo Linux davon laufen lassen. Sie können diese CD-basierte Version von Gentoo dafür verwenden, Gentoo Linux 1.4 auf Ihre Festplatte zu installieren. Neben der bootbaren Gentoo Linux Umgebung beinhaltet CD 1 alles, was Sie für eine schnelle Installation von Gentoo Linux benötigen, selbst ohne eine Anbindung an das Internet. Zusätzlich befinden sich einige vorkompilierte Pakete auf CD 1, wie den wichtigen XFree86 X Server. Haben Sie ein ISO CD-ROM Image von CD 1, ist im Dateinamen `-cd1` enthalten.

Im Gegensatz dazu ist die zweite CD ("CD 2") nicht bootbar und enthält viele vorkompilierte Pakete für Ihr System. Auf dieser CD sind optimierte Versionen von Paketen wie KDE, GNOME, OpenOffice, Mozilla, Evolution und andere verfügbar. CD 2 ist **optional** und für diejenigen gedacht, die Gentoo Linux sehr schnell installieren möchten. Die Pakete auf dieser CD brauchen auf einem üblichen, modernen Einzelprozessorsystem etwa 36 Stunden für eine Kompilierung aus dem Quellcode. Haben Sie ein ISO CD-ROM Image von CD 2, ist im Dateinamen `-cd2` enthalten.

#### Notiz

Ein komplettes Gentoo Linux 2-CD Set enthält die Gentoo Reference Platform ("GRP"), das ein komplettes vorkompiliertes Gentoo Linux System mit GNOME, KDE, Mozilla und OpenOffice ist. Die Gentoo Reference Platform wurde erstellt, um eine schnelle Gentoo Linux paketbasierte Installation bereitzustellen. Die Möglichkeit der "Kompilierung vom Quellcode", die der Grundstein von Gentoo Linux ist, wird immer eine komplett unterstützte Installationsoption bleiben. Der Grund für die GRP ist, Gentoo Linux angenehmer für einige Benutzer zu machen, ohne Gentoos mächtigen "Kompilierung vom Quellcode" Installationsprozess in irgendeiner Weise zu beeinträchtigen.

Neben dem 2-CD Set haben wir auch eine sehr kleine "basic" Live CD, die Sie zum Booten Ihres Systems verwenden können. Sobald Ihr System gebootet ist, können Sie den Zugang zum Internet einrichten und dann Gentoo Linux über das Netzwerk installieren. Der Vorteil dieser "basic" CD liegt in ihrer Größe, sodass das ISO CD-ROM Image schnell heruntergeladen werden kann. Wenn Sie ein erfahrener Benutzer sind, die allerneueste Version von Gentoo Linux installieren möchten und eine schnelle Internetanbindung haben, dann sollten Sie diese Möglichkeit bevorzugen. Haben Sie ein ISO CD-ROM Image von unserer "basic" Live CD ist im Dateinamen `-basic` enthalten.

#### 1.3 Anforderungen

Für jede der CD-basierten Gentoo Linux Installationsmethoden benötigen Sie einen 486+ Prozessor und idealerweise mindestens 64 Megabyte RAM. (Gentoo Linux wurde schon erfolgreich mit 64MB RAM und 64MB Swap installiert, jedoch ist der Build Process furchtbar langsam unter diesen Umständen.)

#### 1.4 Eine Installationsmethode wählen

Nachdem Sie von einer unserer Live CDs gebootet haben, gibt es mehrere Optionen. Gentoo Linux kann über drei verschiedene "stage" tar-Archiven installiert werden. Ausgehend von dem gewählten tar-Archiv, entscheiden Sie, wie viel Ihres Systems selbst kompiliert werden soll. Wählen Sie das **stage1** tar-Archiv, wenn Sie den bootstrap-Prozess durchführen und das komplette System von Grund auf erzeugen wollen. Mit dem **stage2** tar-Archiv kompilieren Sie ihr gesamtes System mit einem vorab durchgeführten bootstrap-Prozess. Das **stage3** tar-Archiv enthält ein bereits kompiliertes Gentoo Linux Basissystem. Für die "GRP" Installationsmethode müssen Sie das stage3 tar-Archiv verwenden.

### **Sie möchten keine Installation mit der GRP machen -- mit welchem stage1, stage2 oder stage3 tar-Archiv sollen Sie dann beginnen?**

Hier gibt es einige Informationen, die Ihnen helfen werden, die Entscheidung zu fällen:

Ausgehend von stage1 haben Sie die totale Kontrolle über die Optimierungseinstellungen und optionalen Funktionen während des Kompilierens, die von Beginn an für Ihr System aktiviert sind. Dadurch ist stage1 für "Power-User" gut geeignet, die wissen, was sie tun. Es ist auch eine gute Installationsmethode für diejenigen, die mehr über die innere Funktionsweise von Gentoo Linux erfahren möchten.

Die Installation von stage2 überspringt den bootstrap-Prozess; das ist recht hilfreich, wenn Sie mit den von uns vorgenommenen Optimierungen für Ihr stage2 tar-Archiv zufrieden sind.

Eine Installation mit stage3 ist die schnellste Möglichkeit, Gentoo Linux zu installieren. Jedoch wird dadurch das Basissystem mit den Optimierungen, die wir für Sie gewählt haben (die, um ehrlich zu sein, gute Einstellungen sind und sorgfältig ausgewählt wurden, um die Performanz bei gleichzeitiger Gewährleistung der Stabilität zu steigern), installiert. Da jedes größere Release von Gentoo Linux stage3 tar-Archive mit Optimierungen für die gängigen Prozessoren bereitstellt, ist die Installation von stage3 die Methode der Wahl -- eine schnelle Installation und ein gut optimiertes System.

### **Sollten Sie zum ersten Mal Gentoo Linux installieren, benutzen Sie ein stage3 tar-Archiv oder ein stage3 mit GRP.**

#### **Notiz**

**Erfahrene Benutzer:** Wenn Sie ein stage3 zur Installation verwenden, ändern Sie nicht die CHOST Einstellung in `/etc/make.conf`. Wenn Sie diese Änderung benötigen, sollten Sie mit einem stage1 tar-Archiv beginnen und Ihr System mit der nötigen CHOST-Einstellung bauen. Die CHOST Einstellung ist üblicherweise sowas wie `i686-pc-linux-gnu`

#### **Wichtig**

Fällt Ihnen ein Problem während der Installation auf und möchten Sie uns über diesen "Bug" informieren, füllen Sie bitte einen Bug-Report unter <http://bugs.gentoo.org> aus. Soll der Bug die ursprünglichen Entwickler einer Software erreichen (wie z.B. das KDE Team), dann werden die **Gentoo Linux Developer** das für Sie übernehmen. Sie helfen mit einem Bug-Report auch anderen Gentoo Linux Benutzern, die möglicherweise ebenso Ihr Problem haben. Mit einer Behebung des Problems helfen Sie von daher der gesamten Gentoo Linux Gemeinschaft!

#### **Notiz**

Die Installationsanweisungen auf der Live CD sind möglicherweise nicht auf den neuesten Stand. Bitte verwenden Sie <http://www.gentoo.de/inhalte/doku/gentoo-x86-install.xml> für die aktuellen Installationsanweisungen (oder die wirklich immer aktuellste Version der Englischen Anweisungen unter <http://www.gentoo.org/doc/en/gentoo-x86-install.xml>).

## **1.5 Der Installationsprozess**

Jetzt lassen Sie uns kurz den Installationsprozess skizzieren. Zuerst laden Sie die CD(s) herunter, brennen diese und booten eine Live CD. Nachdem Sie am root-Prompt angelangt sind, erstellen Sie die Partitionen, erzeugen die Dateisysteme und entpacken entweder das stage1, stage2 oder stage3 tar-Archiv. Wenn Sie das stage1 oder stage2 tar-Archiv verwenden, müssen Sie die nötigen Schritte vornehmen, um Ihr System auf den Stand des stage3 Archivs zu bringen. Wenn Sie stage3 erreicht haben, können Sie das System konfigurieren (die Konfigurationsdateien anpassen, einen Bootloader installieren, usw.) und in ein voll funktionsfähiges Gentoo Linux booten. Nachdem Ihr Gentoo Linux Basissystem läuft, können Sie optional die "CD 2" unseres 2-CD Set verwenden und weitere fertig erstellte Pakete wie KDE, GNOME, OpenOffice, Mozilla oder was Sie sonst auf Ihrem System wünschen, installieren.

Abhängig von der gewählten Installationsstufe, von der Sie starten, hier die benötigten Schritte:

| <b>stage tar-Archiv</b> | <b>Internetzugang erforderlich</b> | <b>Benötigtes Medium</b> | <b>Schritte</b> |
|-------------------------|------------------------------------|--------------------------|-----------------|
|-------------------------|------------------------------------|--------------------------|-----------------|

|       |   |                               |  |
|-------|---|-------------------------------|--|
| 1     | Ja  | <b>basic</b> oder <b>CD 1</b> | Partitions/Dateisystem Einrichtung, emerge sync, bootstrap, emerge system, emerge kernel sources, abschließende Konfiguration  |
| 2     | Ja  | <b>basic</b> oder <b>CD 1</b> | Partitions/Dateisystem Einrichtung, emerge sync, emerge system, abschließende Konfiguration  |
| 3     | Nein, wenn <b>CD 1</b> verwendet wird, ansonsten Ja | <b>basic</b> oder <b>CD 1</b> | Partitions/Dateisystem Einrichtung, emerge sync (nicht nötig, wenn <b>CD 1</b> verwendet wird), abschließende Konfiguration  |
| 3+GRP | Nein  | <b>CD 1, CD 2</b> optional    | Partitions/Dateisystem Einrichtung, abschließende Konfiguration, Installation der vorkompilierte Pakete von CD 1 (optional), reboot, Installation von zusätzlichen vorkompilierten Paketen wie KDE und GNOME (bei Verwendung von "CD 2") |

### Notiz

Hardware ATA RAID Benutzer sollten den Abschnitt über ATA RAID am Ende des Dokuments lesen, bevor sie fortfahren.

## 2. Booten

### Warnung

Lesen Sie diesen kompletten Abschnitt bevor Sie fortfahren, besonders die verfügbaren Boot Optionen. Andernfalls kann dieses zu falschen Tastatureinstellungen, nicht gestarteten PCMCIA-Diensten usw. führen.

Booten Sie nun die von Ihnen gewählte Live CD. Sie sollten ein nettes Boot-Prompt mit dem Gentoo Linux Logo sehen. Drücken Sie hier Enter, um den Boot-Prozess zu starten oder geben Sie optionale Boot-Optionen an, indem Sie einen Kernel auswählen, gefolgt weiteren Optionen und mit Enter abschließen. Zum Beispiel *gentoo nousb nohotplug*. Installieren Sie Gentoo Linux auf einem System mit mehr als einem Prozessor ("SMP"), dann geben Sie *smp* anstelle von *gentoo* am Prompt ein. Dadurch erkennt die Live CD alle Prozessoren und nicht nur den ersten.

Für eine detaillierte Beschreibung der möglichen Boot-Optionen, schauen Sie in die folgende Tabelle der verfügbaren Kernel und Optionen oder drücken Sie F2 und F3 am Boot-Prompt für die Hilfe.

### Verfügbare Kernelbeschreibung

|         |  |
|---------|--|
| gentoo  | normaler Gentoo Kernel (default)                                       |
| nofb    | Framebuffer Modus abgeschaltet   |
| smp     | lädt einen SMP Kernel (für mehrere Prozessoren) ohne Framebuffer Modus |
| acpi    | Schaltet ACPI ein und lädt ACPI Module während des init-Prozesses      |
| memtest | Bootet das RAM Test Programm   |

### Verfügbare Boot Optionen

#### Beschreibung

|            |   |
|------------|---|
| doataraid  | Lädt IDE RAID Module in initrd  |
| dofirewire | Lädt FireWire Module in initrd (für FireWire CD-ROMs, usw.)   |
| dokeymap   | Lässt die Tastaturbelegung für non-us Tastaturen auswählen  |
| dopcmcia   | Startet PCMCIA Dienste  |
| doscsi     | Scannt nach SCSI Geräten (kann zu Komplikationen mit manchen Ethernet-Karten führen)  |
| noapm      | Verhindert das Laden vom APM (Powermanagement) Modul  |
| nodetect   | Führt nicht die automatische Hardwareerkennung mit hwsetup/kudzu und hotplug aus  |
| nodhcp     | DCHP wird nicht ausgeführt, wenn eine Netzwerkkarte erkannt wurde   |
| nohotplug  | Verhindert das Laden von hotplug Diensten   |
| noraid     | Verhindert das Laden von evms Modulen   |
| nousb      | Verhindert das Laden von USB Modulen, impliziert nohotplug  |
| ide=nodma  | Erzwingt das Abschalten von DMA für inkorrekt funktionierende IDE Geräte  |
| cdcache    | Lädt alle benötigten Komponenten der Live CD in den RAM (etwa 40MB werden benötigt), wodurch <code>/mnt/cdrom</code> ungemountet und eine andere CD eingelegt werden kann |

Sobald Sie Enter gedrückt haben, erscheint ein noch netterer Bootbildschirm und ein Statusbalken.

### Warnung

Bleibt Ihr Bildschirm dunkel anstatt einen netten Bootbildschirm zu zeigen, versuchen Sie den **nofb** Kernel zu starten. Es ist sehr wahrscheinlich, dass unser Kernel Ihre Grafikkarte für den Framebuffer-Modus nicht unterstützt (ein Grafikkmodus ohne X-Server).

Im "Live" Gentoo Linux werden Sie automatisch als "*root*", dem "super user", eingeloggt. Sie sollten einen root-Prompt ("*#*") auf der aktuellen Console haben. Sie können zu weiteren Consolen wechseln, indem Sie Alt-F2, Alt-F3 und Alt-F4 drücken. Um zu der ersten Console zurückzukehren, drücken Sie Alt-F1.

### Notiz

**Erfahrene Benutzer:** Wenn Sie die Live CD booten wird das Live CD root Passwort aus Sicherheitsgründen auf eine zufällige Zeichenkette gesetzt. Wenn Sie vorhaben *ssh* zu starten und einen Remote Login zur Live CD einrichten wollen, setzen Sie das Live CD root Passwort mit *passwd* und den folgenden Anweisungen. Ansonsten kennen Sie das entsprechende Passwort zum einloggen auf die Live CD über das Netzwerk nicht.

Sie haben möglicherweise schon den Hilfetext oberhalb des #-Prompts gesehen. Dieser erklärt, wie Sie ihr Netzwerk konfigurieren und wo Sie auf der CD die Gentoo Linux stage tar-Archive und Pakete finden.

### 3. Optionale Hardware Konfiguration

Wenn die Live CD bootet versucht sie Ihre Hardware zu erkennen und lädt die entsprechenden Kernel Module, damit Sie auf die Hardware zugreifen können. Nahezu immer klappt dieses einwandfrei. Jedoch wird manchmal das benötigte Kernel Modul nicht automatisch geladen. Falls die PCI-Autoerkennung Ihre Hardware nicht komplett erkannt hat, müssen Sie die entsprechenden Kernel Module von Hand nachladen. Für eine Liste aller vorhandenen Netzwerkkarten (NIC) Module, geben Sie `ls /lib/modules/`uname -r`/kernel/drivers/net/*` ein. Um ein spezielles Modul zu laden, schreiben Sie:

**Befehlsauflistung 1:** PCI Modul Konfiguration

```
# modprobe pcnet32
(Ersetzen Sie pcnet32 durch das Modul Ihrer NIC)
```

Wenn Sie noch auf SCSI Hardware zugreifen möchten, die nicht automatisch erkannt wurde, müssen Sie das passende Modul aus `/lib/modules` laden. Verwenden Sie dazu wieder *modprobe*:

**Befehlsauflistung 2:** SCSI Modul laden

```
# modprobe aic7xxx
(ersetzen Sie aic7xxx durch das richtige SCSI-Modul für Ihren Controller)
# modprobe sd_mod
(sc_mod ist das Modul für SCSI Festplatten Unterstützung)
```

#### Notiz

Unterstützung für SCSI CD-ROMs und Festplatten ist schon im Kernel eingebaut.

#### Notiz

**Erfahrene Benutzer:** Die Gentoo Live CD sollte DMA für den schnellstmöglichen Datentransfer an Ihren Festplatten aktiviert haben. Falls dem nicht so ist, können Sie mit *hdparm* DMA für Ihre Festplatten einschalten.

**Befehlsauflistung 3:** DMA einschalten

```
(Ersetzen Sie hdX durch die entsprechende Bezeichnung für Ihre
Festplatte)
# hdparm -d 1 /dev/hdX
(DMA einschalten)
# hdparm -d1 -A1 -m16 -u1 -a64 /dev/hdX
(DMA einschalten und andere sichere Performance-steigernde Optionen)
# hdparm -X66 /dev/hdX
(Zwinge die Verwendung von Ultra-DMA -- gefährlich! -- kann manche
Festplatten in Unordnung bringen)
```

### 4. Optionale Konfiguration der Netzwerkanbindung

#### 4.1 Vielleicht klappt es schon?

Wenn Ihr System an ein Ethernet Netzwerk angeschlossen ist, ist es möglich, dass Ihre Netzwerkanbindung schon automatisch konfiguriert wurde. Falls es so ist, können Sie die auf der Live CD beinhalteten Programme für Netzwerke verwenden, wie z.B. *ssh* (Sichere Shell Verbindung zu einem anderen Rechner), *scp* (Sicheres Kopieren von Dateien zu/von einem anderen Rechner), *ping* (ICMP Test-Pakete senden), *irssi* (IRC Chat Programm), *wget* (Dateien von http oder ftp herunterladen), *links* (Textbasierter Web-Browser), sowie weitere Programme.

Falls die Netzwerkanbindung schon konfiguriert wurde, sollte der Aufruf von `/sbin/ifconfig` die Netzwerkkarten neben *lo* anzeigen, wie z.B. *eth0*:

**Befehlsauflistung 4:** `/sbin/ifconfig` bei einer funktionierenden Netzwerkkarte

```
eth0      Link encap:Ethernet  HWaddr 00:50:BA:8F:61:7A
          inet addr:192.168.0.2  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::50:ba8f:617a/10 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1498792 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1284980 errors:0 dropped:0 overruns:0 carrier:0
          collisions:1984 txqueuelen:100
          RX bytes:485691215 (463.1 Mb)  TX bytes:123951388 (118.2 Mb)
          Interrupt:11 Base address:0xe800
```

Sie sollten außerdem versuchen, den DNS (Domain Name System) Server Ihres ISP (Internet Service Provider) anzu"ping"en (die IP-Adresse befindet sich in `/etc/resolv.conf`), als auch eine Web-Seite Ihrer Wahl, einfach nur um sicherzustellen, dass Ihre Pakete das Netz erreichen und die DNS Namensauflösung ordentlich funktioniert, usw.

**Befehlsauflistung 5:** Weiteres Netzwerktesten

```
# ping -c 3 www.yahoo.com
```

Können Sie Ihre Netzwerkanbindung verwenden? Falls dem so ist, können Sie den Rest dieses Kapitels überspringen.

## 4.2 PPPoE Konfiguration

Angenommen Sie benötigen PPPoE, um sich zum Internet zu verbinden, so macht es die Live CD (egal welche Version) für Sie mit `rp-pppoe` leicht. Benutzen Sie das angebotene `adsl-setup` Skript, um Ihre Verbindung einzurichten. Sie werden nach der Netzwerkkarte, die mit Ihrem ADSL-Modem verbunden ist, Ihrem Usernamen und dem Passwort, den IP-Adressen Ihres DNS Server und ob Sie eine einfache Firewall benötigen, gefragt.

**Befehlsauflistung 6:** PPPoE Konfigurieren

```
# adsl-setup
# adsl-start
```

Wenn es nicht auf Anhieb funktioniert, überprüfen Sie, ob Sie Ihren Usernamen und das Passwort korrekt angegeben haben, indem Sie in `/etc/ppp/pap-secrets` oder `/etc/ppp/chap-secrets` nachschauen. Außerdem seien Sie sicher, dass Sie die richtige Netzwerkkarte gewählt haben.

## 4.3 Automatische Netzwerkkonfiguration

Der einfachste Weg die Netzwerkanbindung zu konfigurieren, falls sie nicht automatisch eingerichtet wurde, ist mit dem `net-setup` Skript.

**Befehlsauflistung 7:** net-setup Skript

```
# net-setup eth0
```

Natürlich, falls Sie es lieber wünschen, können Sie auch Ihre Netzwerkanbindung manuell konfigurieren. Das wird als nächstes beschrieben.

## 4.4 Manuelle DHCP Konfiguration

Mit DHCP ist die Netzwerkkonfiguration sehr einfach. Wenn Ihr Provider kein DHCP anbietet, geht es für Sie bei der statischen Netzwerkkonfiguration weiter.

**Befehlsauflistung 8:** Netzerk Konfiguration via DHCP

```
# dhcpcd eth0
```

### Notiz

Einige Provider erwarten, dass Sie einen Hostnamen übergeben. Um dies zu tun, hängen Sie die Option `-h meinhostname` an das oben stehende Kommando.

Wenn Sie **dhcpConfig** Warnungen bekommen, keine Panik. Die meisten Meldungen sind harmlos. Jetzt geht es für Sie unten, mit dem Test der Netzwerkanbindung, weiter.

## 4.5 Manuelle statische Konfiguration

Sie müssen Ihre Netzwerkanbindung so weit konfigurieren, dass Sie die Quellen für die Programme herunterladen können. Außerdem wird das lokale (localhost) Interface benötigt. Die benötigten Informationen stehen in der folgenden Tabelle:

| Information       | Beschreibung   | Beispielwert  |
|-------------------|--|---------------|
| IP Adresse        | Die IP Adresse, die Sie Ihrer Netzwerkkarte zuweisen möchten   | 192.168.1.2   |
| Broadcast Adresse | Die IP Adresse, die Pakete an alle Hosts in einem Netzwerk broadcastet   | 192.168.1.255 |
| Netzwerk Maske    | Die Netzwerk Maske wird zusammen mit der IP Adresse dafür verwendet, um herauszufinden, welcher Teil der Adresse als Netzwerk- und welcher als Hostbereich definiert ist | 255.255.255.0 |
| Gateway           | Die IP Adresse des Rechners, der Pakete weiterleitet, die nicht für das lokale Netzwerk bestimmt sind (meistens der Rechner, der mit dem Internet verbunden ist)         | 192.168.1.1   |

Bitte geben Sie dazu folgende Befehle ein und ersetzen *\$IFACE* mit Ihrem Netzwerkinterface (typischerweise *eth0*), *\$IPNUM* mit Ihrer IP Adresse, *\$BCAST* mit Ihrer Broadcast Adresse und *\$NMASK* mit Ihrer Netzwerkmaske. Bei dem *route* Befehl ersetzen Sie *\$GTWAY* durch die IP Adresse Ihres default Gateway.

**Befehlsauflistung 9:** Statische IP Netzwerkkonfiguration

```
# /sbin/ifconfig $IFACE $IPNUM broadcast $BCAST netmask $NMASK
# /sbin/route add -net default gw $GTWAY netmask 0.0.0.0 metric 1 $IFACE
```

Jetzt müssen Sie die Datei */etc/resolv.conf* erstellen, damit die Namensauflösung (Web/FTP Server durch den Namen, anstatt der IP Adresse finden) funktioniert. Sie können *nano -w /etc/resolv.conf* verwenden, um die Datei zu erstellen. *nano* ist ein kleiner und einfach zu bedienender Editor.

Hier ein Beispiel der Datei */etc/resolv.conf*:

**Befehlsauflistung 10:** */etc/resolv.conf* Vorlage

```
domain mydomain.com
nameserver 10.0.0.1
nameserver 10.0.0.2
```

Ersetzen Sie *10.0.0.1* und *10.0.0.2* mit den IP Adressen Ihres primären und sekundären DNS Servers.

## 4.6 Proxy Konfiguration

Wenn Sie sich hinter einem Proxy befinden, kann es nötig sein, dass Sie Ihre Einstellungen entsprechend konfigurieren, bevor Sie fortfahren. Setzen Sie einige Umgebungsvariablen, um Ihren Proxy zu verwenden:

### Notiz

Benötigt Ihr Proxy eine Authentifikation verwenden Sie folgendes Konstrukt: *http://username:passwort@proxy.netzwerk.lan* (beachten Sie das hinzugefügte "username:passwort").

**Befehlsauflistung 11:** Proxy konfigurieren

```
Wenn der Proxy HTTP-Datenverkehr regelt:
# export http_proxy="proxy.netzwerk.lan:1234"
Wenn der Proxy FTP-Datenverkehr regelt:
# export ftp_proxy="$http_proxy"
Wenn der Proxy RSYNC-Datenverkehr regelt:
# export RSYNC_PROXY="$http_proxy"
```

## 4.7 Die Netzwerkanbindung funktioniert!

Die Netzwerkanbindung sollte nun konfiguriert und nutzbar sein. Es sollte möglich sein, dass sie mit den vorhandenen Programmen *ssh*, *scp*, *links*, *irssi* und *wget* andere Rechner in Ihrem LAN oder dem Internet erreichen können.

## 5. Einstellen Ihres Datums und der Zeit

Nun müssen Sie Ihr Datum und Ihre Zeit einstellen. Sie können dies mit dem *date* Kommando machen

**Befehlsauflistung 12:** Setzen des Datums

```
# date
Thu Feb 27 09:04:42 CST 2003
(Falls Ihr Datum falsch ist, setzen Sie es mit nächsten Kommando)
# date 022709042003
(date MMDDhhmmCCYY)
```

## 6. Dateisysteme, Partitionen und Block-Geräte

### 6.1 Einführung in Block-Geräte

In diesem Abschnitt werden die Festplatten-orientierten Aspekte von Gentoo Linux und Linux im Allgemeinen, sowie die Linux Dateisysteme, Partitionen und Block-Geräte, erläutert. Dann, sobald Sie damit vertraut sind, werden Sie durch den Prozess zur Einrichtung von Partitionen und Dateisystemen für Ihr Gentoo Linux geführt.

Zu Beginn werden die "Block-Geräte" eingeführt. Das bekannteste Block-Gerät ist vermutlich das, welches die erste IDE-Festplatte in einem Linux-System repräsentiert:

**Befehlsauflistung 13:** /dev/hda, das Block-Gerät, das die erste IDE-Festplatte repräsentiert

```
/dev/hda
```

Falls Sie in Ihrem System SCSI-Laufwerke verwenden, dann wird Ihre erste Festplatte diese sein:

**Befehlsauflistung 14:** /dev/sda, das Block-Gerät, das die erste logische SCSI-Festplatte repräsentiert

```
/dev/sda
```

Die oben genannten Block-Geräte repräsentieren eine **allgemeine** Schnittstelle zu der Festplatte. Programme können diese Block-Geräte verwenden, um auf die Festplatte zuzugreifen, ohne sich darum zu kümmern, ob sie IDE, SCSI oder irgendwas anderes ist. Das Programm kann einfach auf die gespeicherten Daten der Festplatte in Bündeln von zusammenhängenden, frei adressierbaren, 512-Byte-Blöcken zugreifen.

### 6.2 Partitionen und fdisk

Unter Linux erstellen Sie Dateisysteme mit dem speziellen Kommando *mkfs* (oder *mke2fs*, *mkreiserfs*, usw.) unter der Angabe eines bestimmten Block-Gerätes als Kommandozeilen-Argument.

Wie auch immer, obwohl es theoretisch möglich ist, eine "komplette Festplatte" als Block-Gerät (eines, das die **gesamte** Festplatte repräsentiert) zu benutzen, um ein einziges Dateisystem darauf unterzubringen, wird dieses in der Praxis so gut wie nie angewandt. Stattdessen wird eine Festplatte in mehrere kleine Block-Geräte, so genannte "Partitionen", aufgeteilt. Die Partitionen werden mit dem Programm *fdisk* erzeugt, das benötigt wird, um die auf jeder Festplatte gespeicherte Partitionstabelle zu erzeugen und zu ändern.

Sie können die Partitionstabelle ansehen, indem Sie *fdisk* mit dem Argument eines Block-Gerätes aufrufen:

#### Notiz

Alternative Programme, um die Partitionstabelle einer Festplatte anzusprechen, sind *cfdisk*, *parted* oder *partimage*. Die Empfehlung lautet jedoch *fdisk*, da es sehr funktionsreich und in der Unix/Linux-Welt gut bekannt ist.

**Befehlsauflistung 15:** fdisk starten

```
# fdisk /dev/hda
```

oder

**Befehlsauflistung 16:** fdisk starten und die Partitionstabelle auf /dev/sda ansehen

```
# fdisk /dev/sda
```

### Wichtig

Beachten Sie, dass Sie die Partitionstabelle **nicht** speichern oder Änderungen an ihr vornehmen, wenn irgendeine dieser Partitionen über Dateisysteme verfügen, die in Benutzung sind, oder wichtige Daten beinhalten. Dieses führt meistens zum Verlust von Daten auf dieser Festplatte.

Bei fdisk werden Sie mit einer Ausgabe wie der folgenden begrüßt:

**Befehlsauflistung 17:** Das fdisk Prompt

```
Command (m for help):
```

Geben Sie *p* ein, um Ihre aktuelle Partitionsaufteilung darzustellen:

**Befehlsauflistung 18:** Eine beispielhafte Partitionsaufteilung

```
Command (m for help): p
```

```
Disk /dev/hda: 240 heads, 63 sectors, 2184 cylinders  
Units = cylinders of 15120 * 512 bytes
```

| Device    | Boot | Start | End  | Blocks   | Id | System     |
|-----------|------|-------|------|----------|----|------------|
| /dev/hda1 | *    | 1     | 14   | 105808+  | 83 | Linux      |
| /dev/hda2 |      | 15    | 49   | 264600   | 82 | Linux swap |
| /dev/hda3 |      | 50    | 70   | 158760   | 83 | Linux      |
| /dev/hda4 |      | 71    | 2184 | 15981840 | 5  | Extended   |
| /dev/hda5 |      | 71    | 209  | 1050808+ | 83 | Linux      |
| /dev/hda6 |      | 210   | 348  | 1050808+ | 83 | Linux      |
| /dev/hda7 |      | 349   | 626  | 2101648+ | 83 | Linux      |
| /dev/hda8 |      | 627   | 904  | 2101648+ | 83 | Linux      |
| /dev/hda9 |      | 905   | 2184 | 9676768+ | 83 | Linux      |

```
Command (m for help):
```

Diese Festplatte ist so partitioniert, dass sie sieben Linux Dateisysteme (jede entsprechende Partition wird als "Linux" gelistet), sowie eine swap Partition (gelistet als "Linux swap") besitzt.

Beachten Sie die Namensgebung der Partitionen zugehörigen Block-Geräte auf der linken Seite, beginnend mit */dev/hda1* bis zu */dev/hda9*. In den frühen PC-Tagen erlaubte die Partitionierungs-Software maximal vier Partitionen (sogenannte "primäre Partitionen"). Das war zu begrenzt, sodass eine Erweiterung mit **erweiterten Partitionen** geschaffen wurde. Eine erweiterte Partition ist sehr ähnlich zu einer primären Partition und wird innerhalb der ersten vier primären Partitionen verwendet. Dennoch, erweiterte Partitionen können eine beliebige Anzahl an sogenannten **logischen Partitionen** beinhalten, wodurch eine effiziente Möglichkeit, das Limit von vier Partitionen zu umgehen, geboten wird.

Alle Partitionen von */dev/hda5* an und höher sind logische Partitionen. Die Nummern von 1 bis 4 sind reserviert für primäre oder erweiterte Partitionen.

Somit sind in dem oben genannten Beispiel */dev/hda1* bis */dev/hda3* primäre Partitionen. */dev/hda4* ist eine erweiterte Partition, die die logischen Partitionen */dev/hda5* bis */dev/hda9* beinhaltet. Sie werden niemals */dev/hda4* benutzen, um darin direkt Dateisysteme zu generieren -- es arbeitet einfach nur als ein Container für die Partitionen */dev/hda5* bis *hda9*.

Beachten Sie außerdem, dass jede Partition eine "ID" besitzt, den sogenannten "Partitionstyp". Immer, wenn Sie eine neue Partition erstellen, sollten Sie darauf achten, dass der Partitionstyp korrekt angegeben ist. '83' ist der korrekte Partitionstyp für eine Partition, die ein Linux Dateisystem beinhaltet, '82' ist der korrekte Partitionstyp für eine Linux swap Partition und 'fd' der korrekte Partitionstyp für Software RAID-Partitionen. Sie können den Partitionstypen mit der Option *t* in *fdisk*

setzen. Der Linux Kernel verwendet den Partitionstypen, um automatisch Dateisysteme und swap Partitionen beim Booten zu erkennen.

## 6.3 fdisk zur Partitionierung verwenden

Da Sie nun eine Einführung in die Festplattenpartitionierung unter Linux erhalten haben, ist es an der Zeit, den Prozess zur Erstellung von Partitionen für Ihre Gentoo Linux Installation zu durchschreiten. Dadurch werden Sie eine Partitionierung ähnlich zu dieser haben:

**Befehlsauflistung 19:** Die Partitionierung nach Durchführung der folgenden Schritte

```
Disk /dev/hda: 30.0 GB, 30005821440 bytes
240 heads, 63 sectors/track, 3876 cylinders
Units = cylinders of 15120 * 512 = 7741440 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1    *           1          14    105808+   83  Linux
/dev/hda2                15          81     506520   82  Linux swap
/dev/hda3                82         3876    28690200   83  Linux
```

Command (m for help):

In unserer vorgeschlagenen "Anfänger" Partitionierung haben wir drei Partitionen. Die erste (`/dev/hda1`) am Anfang der Festplatte ist eine kleine, sogenannte "boot Partition". Der Zweck der boot Partition ist, alle kritischen Daten, die zum Booten benötigt werden zu beinhalten -- die GRUB Bootmanager Informationen (falls Sie GRUB verwenden werden) als auch Ihre(n) Linux Kernel. Die boot Partition biete einen sicheren Ort, um alles wichtige zu speichern, was zum Booten von Linux benötigt wird. Während der alltäglichen Benutzung von Gentoo Linux sollte Ihre boot Partition zur Sicherheit **ungemountet** bleiben. Wenn Sie ein SCSI System aufsetzen, wird Ihre boot Partition `/dev/sda1` sein.

Es ist empfohlen, eine boot Partition am Anfang einer Festplatte einzurichten. Obwohl es inzwischen egal ist, wo sich die boot Partition befindet, ist es eine Tradition aus den Tagen, als der LILO Bootmanager noch keine Kernel von Dateisystemen oberhalb des Zylinders 1024 laden konnte.

Die zweite Partition (`/dev/hda2`) wird für den "swap Bereich" benötigt. Der Kernel verwendet swap Bereiche als einen virtuellen Speicher, wenn der vorhandene RAM zur Neige geht. Diese Partition ist, relativ gesehen, auch nicht sehr groß; üblicherweise etwa 512MB. Wenn Sie ein SCSI-System verwenden, wird die Partition `/dev/sda2` heißen.

Die dritte Partition (`/dev/hda3`) ist recht groß und nimmt den kompletten Rest der Festplatte ein. Diese Partition wird "root Partition" genannt und wird für das Haupt-Dateisystem, das Gentoo Linux selbst beinhaltet, verwendet. Bei einem SCSI-System wird diese Partition `/dev/sda3` sein.

Bevor wir die Festplatte partitionieren, hier ein kurzer technischer Überblick der empfohlenen Partitionen und zugehörigen Dateisysteme für die Installation von Gentoo Linux:

| Partition   | Größe  | Dateisystem  | beispielhaftes Gerät   |
|---|--|--|------------------------|
| boot Partition, beinhaltet Kernel und Boot Informationen  | 32 Megabyte  | ext2/3 stark empfohlen (am einfachsten); falls ReiserFS, dann mit <code>-o notail</code> mounten. Außerdem wird für ext3 oder ReiserFS Platz für das Journal benötigt; in diesem Fall sollte die Partition 64MB groß sein. | <code>/dev/hda1</code> |
| swap Partition (kein 128MB limit, nun 2GB)  | üblicherweise hat der swap Bereich eine Größe von etwa ein bis zwei Mal des physischen RAM | Linux swap   | <code>/dev/hda2</code> |
| root Partition, beinhaltet das Haupt-Dateisystem ( <code>/usr</code> , <code>/home</code> , usw.) | $\geq 1.5$ Gigabyte  | ReiserFS oder ext3 empfohlen; ext2 ist okay  | <code>/dev/hda3</code> |

Gut, nun erzeugen Sie die Partitionen wie in dem oben genannten Beispiel. Zuerst rufen Sie `fdisk` mit `fdisk /dev/hda` oder `fdisk /dev/sda`, abhängig ob Sie IDE oder SCSI verwenden, auf. Dann geben Sie `p` ein, um Ihre aktuelle Partitionierung anzusehen. Gibt es etwas auf der Festplatte, das Sie weiterhin benötigen? Dann **stoppen Sie jetzt**. Wenn Sie mit diesen Anweisungen fortfahren, werden **alle Ihre vorhandenen Daten auf der Festplatte gelöscht**.

## Wichtig

Eine weitere Befolgung der Anweisungen bewirkt ein **Löschen** aller ihrer bereits vorhandenen Daten auf Ihrer Festplatte! Falls sich etwas auf der Festplatte befindet, seien Sie sich sicher, dass es keine kritischen Daten sind und deren Verlust Sie verschmerzen können. Seien Sie sich außerdem sicher, dass Sie die **korrekte Festplatte gewählt haben**, sodass Sie nicht aus Versehen Daten auf der falschen Festplatte löschen.

Nun ist es an der Zeit alle existierenden Partitionen zu löschen. Dazu tippen Sie *d* ein und drücken Enter. Sie werden nach der Partitionsnummer gefragt, die Sie löschen möchten. Um eine existierende `/dev/hda1` Partition zu löschen, gehen Sie so vor:

### Befehlsauflistung 20: Löschen einer Partition

```
Command (m for help): d
Partition number (1-4): 1
```

Die Partition wird zum Löschen vorgesehen. Sie wird nicht mehr angezeigt, wenn Sie *p* eingeben, aber sie wird nicht eher gelöscht, bevor Sie die Änderungen speichern. Wenn Sie einen Fehler gemacht haben und ohne Speichern der Änderungen `fdisk` beenden wollen, geben Sie sofort *q* ein und drücken Sie Enter, sodass Ihre Partition nicht gelöscht wird.

Nun, angenommen Sie wollen wirklich alle vorhandenen Partitionen löschen, geben Sie wiederholt *p* ein, um die Partitionen zu listen und dann *d*, gefolgt von der Partitionsnummer, die Sie löschen möchten. Letzten Endes werden Sie eine leere Partitionstabelle erhalten:

### Befehlsauflistung 21: Eine leere Partitionstabelle

```
Disk /dev/hda: 30.0 GB, 30005821440 bytes
240 heads, 63 sectors/track, 3876 cylinders
Units = cylinders of 15120 * 512 = 7741440 bytes

   Device Boot      Start         End      Blocks   Id  System
Command (m for help):
```

Mit einer im Speicher befindlichen leeren Partitionstabelle, sind Sie bereit, eine boot Partition zu erstellen. Dazu geben Sie *n* ein, um eine neue Partition zu erstellen, dann *p*, um `fdisk` mitzuteilen, dass Sie eine primäre Partition wünschen. Dann drücken Sie *1* zur Generierung der ersten primären Partition. Wenn Sie nach dem ersten Zylinder gefragt werden, drücken Sie einfach Enter. Bei der Nachfrage zum letzten Zylinder geben Sie `+32M` ein, um eine 32MB große Partition zu erstellen. Die Ausgabe dieser Schritte sieht folgendermaßen aus:

## Notiz

Journalling Dateisysteme benötigen zusätzlichen Platz für das Journal. Die Standardeinstellungen benötigen etwa 33MB an Platz. Wenn Sie also ein Journalling Dateisystem für `/boot` verwenden, sollten Sie `+64M` eingeben, wenn Sie nach dem letzten Zylinder gefragt werden.

### Befehlsauflistung 22: Schritte zur Erstellung der boot Partition

```
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
P
Partition number (1-4): 1
First cylinder (1-3876, default 1): (Drücken Sie Enter)
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-3876, default 3876): +32M
```

Geben Sie nun *p* ein, sehen Sie die folgende Partitionierung:

### Befehlsauflistung 23: Die erste Partition wurde erzeugt

```
Command (m for help): p

Disk /dev/hda: 30.0 GB, 30005821440 bytes
240 heads, 63 sectors/track, 3876 cylinders
Units = cylinders of 15120 * 512 = 7741440 bytes

   Device Boot      Start         End      Blocks   Id  System
```

```
/dev/hda1          1          14      105808+  83  Linux
```

Als nächstes erzeugen Sie eine swap Partition. Dazu geben Sie *n* ein, um eine neue Partition zu erzeugen, dann *p*, um fdisk mitzuteilen, dass Sie eine primäre Partition wünschen. Darauf 2 zur Erzeugung der zweiten primären Partition, in diesem Fall also `/dev/hda2`. Werden Sie nach dem ersten Zylinder gefragt, drücken Sie Enter. Bei der Nachfrage zum letzten Zylinder, geben Sie `+512M` ein, um eine Partition von 512MB zu erstellen. Darauf drücken Sie *t*, um den Partitionstyp zu setzen, 2 um die soeben erzeugte Partition auszuwählen und geben 82 ein, entsprechend für einen "Linux Swap" Partitionstyp. Nachdem Sie diese Schritte durchgeführt haben, sollte *p* ähnlich zu dem Folgenden aussehen:

**Befehlsauflistung 24:** Die swap Partition wurde erzeugt

```
Command (m for help): p
```

```
Disk /dev/hda: 30.0 GB, 30005821440 bytes
240 heads, 63 sectors/track, 3876 cylinders
Units = cylinders of 15120 * 512 = 7741440 bytes
```

| Device    | Boot | Start | End | Blocks  | Id | System     |
|-----------|------|-------|-----|---------|----|------------|
| /dev/hda1 |      | 1     | 14  | 105808+ | 83 | Linux      |
| /dev/hda2 |      | 15    | 81  | 506520  | 82 | Linux swap |

Abschließend erzeugen Sie die root Partition. Dazu drücken Sie *n*, um eine neue Partition zu erzeugen, dann *p*, um fdisk mitzuteilen, dass Sie eine primäre Partition wünschen. Dann drücken Sie 3 für die Generierung der dritten primären Partition, also `/dev/hda3`. Wird nach dem ersten Zylinder gefragt, drücken Sie Enter. Bei der Nachfrage zum letzten Zylinder, drücken Sie ebenso Enter, um eine Partition, die den restlichen freien Platz einnimmt, zu erzeugen. Darauf sollte *p* eine Ausgabe ähnlich zu der folgenden darstellen:

**Befehlsauflistung 25:** Die root Partition wurde erzeugt

```
Command (m for help): p
```

```
Disk /dev/hda: 30.0 GB, 30005821440 bytes
240 heads, 63 sectors/track, 3876 cylinders
Units = cylinders of 15120 * 512 = 7741440 bytes
```

| Device    | Boot | Start | End  | Blocks   | Id | System     |
|-----------|------|-------|------|----------|----|------------|
| /dev/hda1 |      | 1     | 14   | 105808+  | 83 | Linux      |
| /dev/hda2 |      | 15    | 81   | 506520   | 82 | Linux swap |
| /dev/hda3 |      | 82    | 3876 | 28690200 | 83 | Linux      |

Abschließend muss noch das "bootable" Flag bei der boot Partition gesetzt und die Änderungen abgespeichert werden. Um `/dev/hda1` als eine bootbare Partition zu kennzeichnen, geben Sie im Menü *a* gefolgt von 1 für die Partitionsnummer ein. Schauen Sie sich nun die Partitionstabelle mit *p* an, sehen Sie, dass `/dev/hda1` ein \* in der Spalte "Boot" besitzt. Schreiben Sie nun die Änderungen auf die Festplatte. Dazu geben Sie *w* ein und drücken Enter. Ihre Festplatte ist nun ordentlich für ein Gentoo Linux konfiguriert.

### Notiz

Falls *fdisk* oder *cdisk* Sie anweist, neu zu starten, damit Ihr System die Änderungen an der Partitionstabelle akzeptiert, dann tun Sie dieses bitte.

## 6.4 Dateisysteme erzeugen

Da nun die Partionen erzeugt wurden, ist es an der Zeit, die Dateisysteme auf der boot und root Partition einzurichten, sodass sie zur Datenspeicherung in Ihr System eingebunden werden können. Außerdem wird die swap Partition konfiguriert, um als swap Bereich zu dienen.

Gentoo Linux unterstützt eine Auswahl von unterschiedlichen Dateisystemen; jedes hat seine Stärken und Schwächen und seine eigene Zusammenstellung an charakteristischer Leistung. Derzeit wird die Generierung von ext2, ext3, XFS, JFS und ReiserFS Dateisystemen unterstützt.

ext2 ist ein erprobtes Linux Dateisystem, unterstützt aber kein Journaling, was dazu führt, dass die routinemäßigen ext2 Dateisystem Prüfungen während des Startens recht zeitintensiv sind. Daher gibt es eine Auswahl von neueren **Journaling** Dateisystemen, deren Konsistenz sehr schnell überprüft

werden kann und von daher grundsätzlich den Nicht-Journaling Dateisystemen vorgezogen werden sollten. Journaling Dateisysteme beugen langen Wartezeiten beim Booten Ihres System mit **inkonsistenten** Dateisystemen vor.

ext3 ist das entsprechende Journaling Dateisystem zu ext2, welches Metadaten Journaling für eine schnelle Wiederherstellung nutzt, zusätzlich zu anderen verbesserten Journaling Modi, wie komplettes Daten und geordnetes Daten Journaling. ext3 ist ein sehr gutes und zuverlässiges Dateisystem. Es bietet grundsätzlich anständige Leistung während der meisten Einsatzumgebungen. Dadurch, dass es im internen Design nicht sehr stark von "trees" Gebrauch macht, skaliert es nicht sehr gut; sprich, es ist keine gute Wahl für ein sehr großes Dateisystem oder Situationen in denen Sie mit großen Dateien oder einer großen Anzahl von Dateien in einem einzigen Verzeichnis arbeiten werden. Solange Sie es innerhalb seiner Design Parameter verwenden, ist ext3 ein exzellentes Dateisystem.

ReiserFS ist ein B\*-tree basiertes Dateisystem, das im Großen und Ganzen eine gute Leistung bietet und ext2 und ext3 weit übertrifft, wenn Sie mit kleinen Dateien (Dateien kleiner als 4kB) arbeiten, meistens mit einem Faktor von 10- bis 15-fach. ReiserFS skaliert außerdem sehr gut und hat ein Metadata Journaling. Seit Kernel 2.4.18 ist ReiserFS sicher und die erste Wahl sowohl für ein Mehrzweck-Dateisystem als auch für Extremfälle, wie großen Dateisystemen, die Verwendung von vielen kleinen Dateien, sehr großen Dateien und Verzeichnissen mit tausenden von Dateien. ReiserFS ist das Dateisystem, das wir standardmäßig für alle nicht-boot Partitionen empfehlen.

XFS ist ein Dateisystem mit Metadata Journaling, das durchgängig mit dem Gentoo Linux *xfs-sources* Kernel kooperiert. Es beinhaltet ein robustes Feature-Set und ist für Skalierbarkeit optimiert. Wir empfehlen, dieses Dateisystem nur in high-end SCSI und/oder Glasfaserkanal-Speicherung mit einer ununterbrechbaren Stromversorgung zu betreiben. Dadurch, dass XFS sehr häufig die in Bewegung befindlichen Daten im RAM speichert, können unsachgemäß erstellte Programme (diese, die sich nicht um saubere Vorkehrungen beim Speichern auf Festplatte kümmern -- und davon gibt es einige) zu Datenverlust führen, wenn das Programm unvorhergesehen beendet wird.

JFS ist ein eigens von IBM erstelltes hochleistungsfähiges Journaling Dateisystem. Es wurde kürzlich erst produktionsreif, weshalb es noch keine Auflistung an Vor- oder Nachteilen zu seiner grundsätzlichen Stabilität gibt.

Möchten Sie das stabilste Journaling Filesystem verwenden, wählen Sie ext3. Für ein gutes Mehrzweck-Dateisystem mit hoher Leistung und Journaling, verwenden Sie ReiserFS. Beide, ext3 und ReiserFS, sind voll entwickelt, raffiniert und empfohlen für den allgemeinen Gebrauch.

Ausgehend von der oben genannten Beispielpartitionierung, geben Sie die folgenden Kommandos ein, um Ihre Dateisysteme zu generieren:

**Befehlsauflistung 26:** Generierung der Dateisysteme (Beispiel)

```
# mke2fs /dev/hda1
# mkswap /dev/hda2
# mkreiserfs /dev/hda3
```

Für die boot Partition */dev/hda1* wurde ext2 gewählt, da es ein robustes Journaling Dateisystem ist, das von allen Bootmanagern unterstützt wird. *mkswap* ist für die swap Partition */dev/hda2* -- die Wahl ist hier offensichtlich. Für die root Partition */dev/hda3* wurde ReiserFS gewählt, da es ein solides Journaling Dateisystem mit einer hervorragenden Leistung ist. Nun, fahren Sie fort und generieren Sie die Dateisysteme.

Als Referenz dient diese Auflistung der verschiedenen *mkfs* Kommandos, die während der Installation zur Verfügung stehen:

*mkswap* ist das Kommando, um eine swap Partition zu initialisieren:

**Befehlsauflistung 27:** swap Partition initialisieren

```
# mkswap /dev/hda2
```

Sie können *mke2fs* benutzen, um ein ext2-Dateisystem zu erstellen:

**Befehlsauflistung 28:** Ein ext2 Dateisystem erzeugen

```
# mke2fs /dev/hda1
```

Wenn Sie ext3 vorziehen, können Sie es mit *mke2fs -j* erzeugen:

**Befehlsauflistung 29:** Ein ext3 Dateisystem erzeugen

```
# mke2fs -j /dev/hda3
```

### Notiz

Mehr Informationen über ext3 mit Linux 2.4 finden Sie unter <http://www.zip.com.au/~akpm/linux/ext3/ext3-usage.html>.

Für die Erstellung eines ReiserFS Dateisystems, verwenden Sie das *mkreiserfs* Kommando:

**Befehlsauflistung 30:** Ein ReiserFS Dateisystem erzeugen

```
# mkreiserfs /dev/hda3
```

Zum Erstellen eines XFS Dateisystems, rufen Sie *mkfs.xfs* auf:

**Befehlsauflistung 31:** Ein XFS Dateisystem erzeugen

```
# mkfs.xfs /dev/hda3
```

### Notiz

Möglicherweise möchten Sie weitere Attribute an *mkfs.xfs* überreichen: *-d agcount=3 -l size=32m*. Das *-d agcount=3* Attribut verringert die Anzahl der Zuordnungsgruppen. XFS besteht auf mindestens eine Zuordnungsgruppe je vier Gigabyte Ihrer Partition, wodurch z.B. eine 20GB Partition mindestens ein *agcount* von 5 benötigt. Das *-l size=32m* Attribut erhöht die Journalgröße auf 32 MB und verbessert dadurch die Leistung.

Um ein JFS Dateisystem zu erstellen, verwenden Sie *mkfs.jfs*:

**Befehlsauflistung 32:** Ein JFS Dateisystem erzeugen

```
# mkfs.jfs /dev/hda3
```

## 7. Partitionen mounten

Aktivieren Sie nun Ihre Swap Partition, denn es könnte sein, dass Sie den virtuellen Speicher später noch benötigen:

**Befehlsauflistung 33:** swap Partition aktivieren

```
# swapon /dev/hda2
```

Als nächstes erstellen Sie das Verzeichnis */mnt/gentoo/boot*, anschließend mounten Sie Ihre Partitionen. Sobald diese gemountet sind, wird jede Datei, die in */mnt/gentoo* kopiert wird, in das neue Dateisystem geschrieben. Wenn Sie Gentoo Linux mit separaten */usr* oder */var* Partitionen installieren, müssen diese in die entsprechenden Verzeichnisse gemountet werden (*/mnt/gentoo/usr* und */mnt/gentoo/var*).

### Wichtig

Wenn Ihre */boot* Partition, dort wo der Kernel unterbracht ist, eine ReiserFS Partition ist, achten Sie darauf, dass diese mit der *-o notail* Option gemountet wird, sonst wird GRUB nicht richtig installiert. Stellen Sie auch sicher, dass die Option *notail* mit in die */etc/fstab* eingetragen wird, doch dazu später mehr. Wenn Sie LILO mit ReiserFS verwenden möchten, dann ist die *-o notail* Option nicht notwendig. Dennoch ist es immer sicher, die *-o notail* Option mit ReiserFS zu verwenden, wenn Sie nicht wissen, was zu tun ist.

**Befehlsauflistung 34:** Mount Verzeichnisse erzeugen

```
# mount /dev/hda3 /mnt/gentoo
# mkdir /mnt/gentoo/boot
# mount /dev/hda1 /mnt/gentoo/boot
```

### Wichtig

Sollten Sie Probleme haben, Ihre boot Partition als ext2 zu mounten, versuchen Sie es mit `mount /dev/hda1 /mnt/gentoo/boot -t ext2`.

## 8. Stage tar-Archive und chroot

### 8.1 Wahl des gewünschten stage tar-Archivs

Nun sollten Sie sich endgültig entscheiden, mit welchem stage Sie ihre Installation durchführen wollen. Die stage tar-Archive sind auf der Live CD im Verzeichnis `/mnt/cdrom/stages` verfügbar; Sie können `ls /mnt/cdrom/stages` eingeben, um eine Liste der verfügbaren stage tar-Archive zu erhalten.

**GRP Benutzer** mögen das `stage3-xx-yy.tar.bz2` tar-Archiv verwenden.

Wenn Sie eine Installation von einem stage tar-Archiv durchführen wollen, das **nicht** auf der CD ist (was der Fall sein wird, wenn Sie unsere "basic" Live CD verwenden), müssen Sie es mit den folgenden Anweisungen herunterladen. Falls Sie das stage tar-Archiv, das Sie benutzen wollen, bereits besitzen (bei den meisten Benutzern üblich), dann fahren Sie mit "Entpacken des stage tar-Archivs" fort.

#### Notiz

Um einen Proxy-Server (z.B. `proxy.server.tld:8080`) zu verwenden, rufen Sie `links` mit der Option `-http-proxy proxy.server.tld:8080` auf.

**Befehlsauflistung 35:** Herunterladen des gewünschten stage tar-Archivs

```
# cd /mnt/gentoo
```

Verwenden Sie `links`, um die URL des tar-Archivs zu erhalten:

```
# links http://gentoo.oregonstate.edu/releases/x86/1.4/
```

Verwenden Sie die `hoch` und `runter` Cursor-Tasten (oder benutzen Sie `TAB`), um das gewünschte Verzeichnis hervorzuheben.

Drücken Sie dann `d` wodurch der Download gestartet wird. Speichern Sie die Datei und beenden Sie den Browser mit `q`.

Oder verwenden Sie `wget` von der Kommandozeile:

```
# wget fügen Sie hier den URL zum gewünschten tar-Archiv ein
```

### 8.2 Entpacken des stage tar-Archivs

Nun ist es an der Zeit das gepackte stage tar-Archiv Ihrer Wahl in `/mnt/gentoo/` zu entpacken. Bedenken Sie, dass Sie nur **ein** stage tar-Archiv zu entpacken brauchen, also nur `stage1`, `stage2` oder `stage3`. Möchten Sie eine Installation von Gentoo mit `stage3` durchführen, dann packen Sie es folgendermaßen aus:

#### Wichtig

Stellen Sie sicher, dass Sie das Argument `p` mit `tar` verwenden. Vergessen Sie dieses, werden die Dateien mit falschen Rechten entpackt.

**Befehlsauflistung 36:** Entpacken des stage tar-Archivs

```
# cd /mnt/gentoo
```

Ändern Sie "stage3" in "stage2" oder "stage1", wenn Sie von diesen stages an installieren möchten.

Wenn Sie ein stage tar-Archiv heruntergeladen haben, ändern Sie den Pfad von `/mnt/cdrom/stages/` in `/mnt/gentoo/`.

```
# tar -xvjpgf /mnt/cdrom/stages/stage3-*.tar.bz2
```

Wenn sie Ihr stage tar-Archiv nach `/mnt/gentoo/` heruntergeladen haben, können Sie dieses mit `rm /mnt/gentoo/stage*.tar.bz2` löschen.

### 8.3 Schritte für ein GRP Portage Paket

#### Wichtig

Die folgenden Anweisungen sind nur für GRP Benutzer. Wenn Sie kein GRP nutzen, fahren Sie mit "Auswahl eines Mirror-Server (optional)" fort.

**GRP Benutzer:** Auf der CD existiert eine "Momentanaufnahme" von Portage. Benutzen Sie diese Momentanaufnahme, um das später durchzuführende `emerge sync` auszulassen, denn `emerge sync` benötigt eine Netzwerkverbindung. Entpacken Sie die die Momentanaufnahme folgendermaßen:

### **Befehlsauflistung 37:** Verwenden einer Portage Momentanaufnahme

Ersetzen Sie `yyyymmdd` mit dem Zeitstempel im Dateinamen.

```
# tar -xvjf /mnt/cdrom/snapshots/portage-yyyymmdd.tar.bz2 -C /mnt/gentoo/usr
```

Damit entpacken Sie die Momentanaufnahme des Portage Tree in Ihre frische Gentoo Installation. Nun brauchen Sie keine Internetanbindung mehr, um *emerge sync* durchzuführen und den Portage Tree herunterzuladen. Kopieren Sie jetzt die distfiles und Pakete von der Live CD an ihren zukünftigen Ort:

### **Befehlsauflistung 38:** Kopieren der GRP Dateien

```
# cp -R /mnt/cdrom/distfiles /mnt/gentoo/usr/portage/distfiles
# cp -a /mnt/cdrom/packages /mnt/gentoo/usr/portage/packages
```

Alle relevanten Dateien für GRP sind nun an ihrem benötigten Platz. Sie haben alles kopiert und entpackt, um Gentoo Linux zu installieren -- selbst ohne eine Netzwerkanbindung.

## **8.4 Auswahl eines Mirror-Server (optional)**

*mirrorselect* ist ein Programm, das dazu entwickelt wurde, automatisch den schnellsten Mirror-Server abhängig von Ihrem Ort zu wählen oder manuell einen Mirror-Server aus einer Liste auszuwählen. Unglücklicherweise läuft *mirrorselect* nicht hinter jedem Router einwandfrei.

### **Befehlsauflistung 39:** mirrorselect verwenden

Automatische Auswahl eines Mirror-Server:

```
# mirrorselect -a -s4 -o >> /mnt/gentoo/etc/make.conf
```

Interaktive Auswahl eines Mirror-Server:

```
# mirrorselect -i -o >> /mnt/gentoo/etc/make.conf
```

Falls *mirrorselect* aus irgendeinem Grund keinen Mirror-Server eintragen konnte, können Sie dennoch mit dieser Installationsanweisung fortfahren, da keine Änderungen vorgenommen wurden. Einer der Gründe, warum *mirrorselect* versagt, ist, dass es nicht auf allen Installationsmedien vorhanden ist.

## **8.5 chroot betreten**

Als nächstes werden Sie in Ihre neue Gentoo Linux Installationsumgebung *chrooten*, und dort Ihr neues Gentoo Linux System "betreten"

### **Notiz**

Eventuell erhalten Sie während des *env-update* eine Meldung, dass `/etc/make.profile/make.defaults` nicht existiert: Ignorieren Sie diese. Später in diesem Dokument wird *emerge sync* ausgeführt, das dieses Problem löst.

### **Befehlsauflistung 40:** Vorbereiten und betreten der chroot Umgebung

```
# mount -t proc proc /mnt/gentoo/proc
# cp /etc/resolv.conf /mnt/gentoo/etc/resolv.conf
# chroot /mnt/gentoo /bin/bash
# env-update
Regenerating /etc/ld.so.cache...
# source /etc/profile
(Obiges erneuert die Pfade Ihrer Shell und frischt die Binärdateien auf.)
```

Nachdem Sie diese Kommandos ausgeführt haben, befinden Sie sich "innerhalb" Ihrer neuen Gentoo Linux Umgebung in `/mnt/gentoo`. Sie können den Rest der Installation innerhalb dieser chroot Umgebung ausführen.

## **9. Den aktuellen Portage Tree mit sync herunterladen**

### **Wichtig**

Installieren Sie mittels GRP können Sie diesen Abschnitt über *emerge sync* überspringen.

Als nächstes ist es nötig, *emerge sync* aufzurufen. Mit diesem Kommando teilen Sie Portage mit, die aktuelle Kopie des Gentoo Linux Portage Tree herunterzuladen. Haben Sie schon vorher eine Momentanaufnahme des Portage Tree von **CD 1** entpackt, können Sie diesen Schritt überspringen. Der

Portage Tree beinhaltet alle Skripte (sogenannte ebuilds), die für das Kompilieren aller Pakete unter Gentoo Linux verwendet werden. Derzeit gibt es ebuild Skripte für knapp 4000 Pakete. Sobald *emerge sync* beendet ist, haben Sie einen kompletten Portage Tree in `/usr/portage`.

**Befehlsauflistung 41:** Update des Portage Tree

```
# emerge sync
```

## 10. Gentoo Optimierungen festlegen (make.conf)

Da Sie nun eine aktuelle Kopie des Portage Tree haben, ist es an der Zeit, die Optimierungen und optionalen Einstellungen zum Kompilieren für Ihr Gentoo Linux System vorzunehmen. Portage wird diese Einstellungen beim Kompilieren von jedem Programm verwenden. Dazu editieren Sie die Datei `/etc/make.conf`. Darin sollten Sie die USE Flags setzen, wodurch - falls vorhanden - optionale Funktionen in Ihre Programme einkompiliert werden; üblicherweise sind die Standardeinstellungen (eine **leere** oder ungesetzte USE Variable) okay. Mehr Informationen über USE Flags können im [Gentoo Handbuch für USE-Flags](#) gefunden werden. Weiterhin gibt es eine Liste über alle aktuellen [USE Flags](#).

Beginnen Sie mit einem stage1 tar-Archiv, sollten Sie auch entsprechende Einstellungen an CHOST, CFLAGS und CXXFLAGS für Ihr System (Beispiele befinden sich direkt in der `/etc/make.conf`) vornehmen.

### Warnung

Benutzen Sie ein stage2 oder stage3 tar-Archiv wurden diese Einstellungen schon von uns optimal vorgenommen und benötigen keiner weiteren Modifikation.

### Warnung

**Erfahrene Benutzer:** Wenn Sie planen ein Gentoo System mit `ACCEPT_KEYWORDS="~x86"` zu installieren, setzen Sie `ACCEPT_KEYWORDS` nicht bevor die Bootstrap Phase (stage1) beendet wurde.

### Wichtig

**Erfahrene Benutzer:** Die CFLAGS und CXXFLAGS Einstellungen werden benötigt, um dem C und C++ Compiler mitzuteilen, welche Optimierungen während des Kompilierens vorgenommen werden sollen. Es ist z.B. für Benutzer von Athlon XP Prozessoren üblich, eine `"-march=athlon-xp"` Einstellung in CFLAGS und CXXFLAGS vorzunehmen, sodass alle Pakete für ihre CPU entsprechend optimiert kompiliert werden. Die Datei `/etc/make.conf` enthält eine allgemeine Beschreibung für die richtigen Einstellungen Ihrer CFLAGS und CXXFLAGS.

Falls notwendig können Sie ebenso Informationen zum Proxy eintragen, z.B. wenn Sie sich hinter einer Firewall befinden. Benutzen Sie das folgende Kommando, um die `/etc/make.conf` mit *nano*, einem simplen Editor, zu bearbeiten.

**Befehlsauflistung 42:** Einstellen von make.conf Optionen

```
# nano -w /etc/make.conf
```

### Notiz

**Erfahrene Benutzer:** Möchten Sie den build Prozess im wesentlichen verändern, sollten Sie sich die Datei `/etc/make.globals` ansehen. Diese Datei umfasst die Gentoo Standardeinstellungen und darf niemals verändert werden. Wenn Ihnen die Default Einstellungen nicht genügen, können Sie neue Einträge in der `/etc/make.conf` vornehmen, da diese die Einträge in der `/etc/make.globals` **überschreiben**. Wenn Sie an den Anpassungen der USE Variablen interessiert sind, schauen Sie sich `/etc/make.profile/make.defaults` an. Möchten Sie dort aktivierte USE Variable abschalten, tragen Sie in der `/etc/make.conf` entsprechend `USE="-foo"` ein, um jede *foo* USE Variable abzuschalten, die standardmäßig in `/etc/make.globals` oder `/etc/make.profile/make.defaults` aktiviert ist.

### Warnung

Fügen Sie nicht vor dem Abschluss von stage1 *'static'* zu Ihrer USE Variable hinzu.

## 11. Beginnen von stage1

### Notiz

Wenn Sie nicht von einem stage1 tar-Archiv beginnen, können Sie diesen Abschnitt überspringen.

Das stage1 tar-Archiv ist für komplette Anpassungen und Optimierungen gedacht. Wenn Sie sich für dieses tar-Archiv entschieden haben, werden Sie auf ein über-optimiertes und aktuelles System blicken können. Haben Sie Spaß, denn Optimierung ist das, was Gentoo Linux ausmacht! Die Installation von stage1 benötigt eine Menge Zeit, aber das Ergebnis ist ein System, das von Grund auf für Ihren Computer und Ihren Anforderungen optimiert wurde.

Nun ist es an der Zeit den "bootstrap" Prozess zu starten. Dieser Prozess benötigt selbst auf einem 1200 MHz AMD Athlon System etwa zwei Stunden. Während dieser Zeit werden die GNU C Library, der Compiler und andere wichtige Systemprogramme gebaut. Starten Sie den bootstrap Prozess wie folgt:

**Befehlsauflistung 43:** Bootstrapping

```
# cd /usr/portage
# scripts/bootstrap.sh
```

Der bootstrap Prozess beginnt nun.

#### **Notiz**

*bootstrap.sh* bietet eine *--fetchonly* Option. Wenn Sie eine Dial-Up-Verbindung verwenden, kann diese nützlich sein. Sie wird alle für den bootstrap-Prozess benötigten Dateien für die spätere Kompilierung in einem Schritt herunterladen. Schauen Sie sich *bootstrap.sh -h* für weitere Informationen an.

#### **Notiz**

Portage benutzt standardmäßig das Verzeichnis */var/tmp* während der Kompilierung, oftmals unter Verwendung von mehreren hundert Megabyte an temporären Dateien. Wenn Sie Portage ein anderes Verzeichnis zum Speichern der temporären Dateien mitteilen wollen, setzen Sie die Variable *PORTAGE\_TMPDIR* **bevor** Sie mit dem bootstrap Prozess beginnen:

**Befehlsauflistung 44:** Ändern des Portage Pfades für temporäre Auslagerungsdateien

```
# export PORTAGE_TMPDIR="/anderes/Verzeichnis/tmp"
```

*bootstrap.sh* wird *binutils*, *gcc*, *gettext* und *glibc* bauen, daraufhin *gettext* mit der neuen *glibc* nochmals kompilieren. Es ist unnötig darauf hinzuweisen, dass dieser Prozess eine ganze Weile in Anspruch nimmt. Sobald der Prozess beendet ist, entspricht Ihr System einem "stage2" System, weshalb Sie mit den Anweisungen zu stage2 im kommenden Abschnitt fortfahren können.

## **12. Beginnen von stage2 und Fortsetzung von stage1**

#### **Notiz**

Dieser Abschnitt ist für jene, die eine Installation von stage1 an fortsetzen oder mit stage2 beginnen. Falls Sie nicht dazu gehören (z.B. da Sie stage3 verwenden möchten), können Sie diesen Abschnitt überspringen.

#### **Warnung**

Wenn Sie mit stage2 beginnen ändern Sie nicht die *CHOST*-Variable in */etc/make.conf*. Andernfalls gibt es komische und umfangreiche Fehler beim Kompilieren.

Das stage2 tar-Archiv hat den bootstrapping Prozess bereits hinter sich gebracht. Sie müssen lediglich den Rest des Basissystems installieren.

#### **Notiz**

Wenn Sie von einem vorbereiteten stage2 die Installation beginnen und sicher sein wollen, dass der Compiler auf dem aktuellen Stand ist, fügen Sie zu den unten genannten Kommandos *-u* hinzu. Wenn Sie nicht wissen, was diese Anmerkung bedeutet, können Sie sie ignorieren.

**Befehlsauflistung 45:** Den Rest des Systems installieren

```
# emerge -p system
(listet die zu installierenden Pakete auf)
# emerge system
```

Es braucht einige Zeit, um das Kompilieren des Basissystems zu beenden. Der Erfolg ist ein an Ihre Anpassungen optimiertes System. Der Nachteil ist, dass Sie sich während dieser Zeit irgendwie beschäftigen müssen. Der Autor empfiehlt "Star Wars - Super Bombad Racing" für die Playstation 2...

Das Basissystem ist nun kompiliert. Fahren Sie mit dem Abschnitt "Einstellen der Zeitzone" fort.

## **13. Beginnen von stage3**

#### **Notiz**

Dieser Abschnitt ist für jene, die mit stage3 **beginnen**. Wenn Sie mit stage1 oder stage2 begonnen haben, können Sie ihn überspringen. Auch Benutzer von GRP fahren erst im nächsten Abschnitt fort.

#### **Warnung**

Wenn Sie mit stage3 beginnen ändern Sie nicht die *CHOST*-Variable in */etc/make.conf*. Andernfalls gibt es komische und umfangreiche Fehler beim Kompilieren.

Das stage3 tar-Archiv bietet ein voll funktionsfähiges Gentoo System, sodass keine Kompilierung benötigt wird.

### Notiz

**Erfahrene Benutzer:** Wie auch immer, das stage3 tar-Archiv ist vorkompiliert und kann ein wenig veraltet sein. Falls Sie das beunruhigen sollte, können Sie automatisch Ihr existierendes stage3 updaten, um die aktuellsten Versionen des Basissystems zu erhalten, indem Sie eine Kopie von `/etc/make.conf` anlegen, `CONFIG_PROTECT="-*" emerge -u system` ausführen und die Kopie danach wieder einspielen. Bedenken Sie, dass dieses eine lange Zeit benötigen kann, wenn Ihr stage3 sehr alt ist; ansonsten wird dieser Prozess recht schnell mit dem Vorteil, die aktuellsten Gentoo Updates und Fehlerbehebungen zu beinhalten, beendet sein. Wie dem auch sei, wenn Sie möchten, können Sie diesen Befehl überspringen und mit dem nächsten Abschnitt fortfahren.

## 14. Einstellen der Zeitzone

Nun ist es erforderlich, dass Sie Ihre Zeitzone einstellen.

Suchen Sie Ihre Zeitzone (oder GMT falls Sie die Greenwich Mean Time verwendenden) in `/usr/share/zoneinfo`. Dann erzeugen Sie eine symbolische Verknüpfung zu `/etc/localtime` durch:

**Befehlsauflistung 46:** Erzeugen einer symbolischen Verknüpfung zu Ihrer Zeitzone

```
# ln -sf /usr/share/zoneinfo/Pfad/zur/Zeitzoneendatei /etc/localtime
```

## 15. Ändern der `/etc/fstab` für Ihre Maschine

Ihr Gentoo Linux System ist beinahe zur Benutzung fertig. Es müssen lediglich noch ein paar wichtige Systemdateien konfiguriert und der Bootloader installiert werden. Zuerst wird die Datei `/etc/fstab` editiert. Beachten Sie, dass die `notail` Option zu Ihrer boot Partition hinzugefügt werden muss, wenn Sie diese als ReiserFS angelegt haben. Weiterhin geben Sie die Dateisystemtypen `ext2`, `ext3` oder `reiserfs` entsprechend Ihrer Konfiguration an.

### Warnung

Verwenden Sie eine `/etc/fstab` ähnlich zu der unten gezeigten, jedoch tauschen Sie "BOOT", "ROOT" und "SWAP" durch die tatsächlichen Partitionsnummern (wie z.B. `hda1` usw.) als auch "ext2" und "ext3" durch die von Ihnen verwendeten Dateisysteme aus.

**Befehlsauflistung 47:** Editieren der `fstab`

```
# /etc/fstab: static file system information.
#
# noatime turns off atimes for increased performance (atimes normally aren't
# needed; notail increases performance of ReiserFS (at the expense of storage
# efficiency). It is safe to drop the noatime options if you want and to
# switch between notail and tail freely.

# <fs>          <mount point>  <type>  <opts>          <dump/pass>

# NOTE: If your BOOT partition is ReiserFS, add the notail option to opts.

/dev/BOOT        /boot          ext2      noauto,noatime  1 2
/dev/ROOT        /              reiserfs noatime         0 1
/dev/SWAP        none           swap      sw              0 0
/dev/cdroms/cdrom0 /mnt/cdrom    iso9660   noauto,ro,user  0 0
none             /proc          proc      defaults       0 0
```

### Warnung

Bitte beachten Sie, dass `/boot` **nicht** während des Bootens gemountet wird. Das geschieht aus dem Grunde, um die Daten in `/boot` zu schützen. Wenn Sie Zugriff auf `/boot` benötigen, müssen Sie es mounten.

## 16. Installieren des Kernels und des System Logger

### 16.1 Auswahl des Kernels

Es gibt zwei Möglichkeiten zur Installation des Kernels. Entweder konfigurieren Sie ihn von Hand oder benutzen das `genkernel` Hilfsprogramm, das automatisch Ihren Kernel konfiguriert und kompiliert.

Ob Sie nun Ihren Kernel von Hand konfigurieren oder *genkernel* verwenden, müssen Sie vorab die Linux Kernel Sourcen, die Sie verwenden möchten, mergen. Gentoo bietet Ihnen einige Kernel ebuilds; eine Liste mit Beschreibung gibt es im [Gentoo Kernel Handbuch](#). Wenn Sie sich in der Wahl nicht schlüssig sind, empfehlen wir die *gentoo-sources*. Benötigen Sie die Unterstützung für XFS, installieren Sie *xfs-sources* oder *gs-sources*. Die Gentoo Live CD verwendet *gs-sources* und *xfs-sources*. Es gibt außerdem die *gaming-sources*; ein Kernel optimiert für Spiele, der wunderbar mit der aktivierten "Preemptive Kernel" Option zusammenarbeitet.

Wählen Sie einen Kernel und mergen Sie ihn wie folgt:

**Befehlsauflistung 48:** Emergen der Kernel Quellen

```
# emerge -k sys-kernel/gentoo-sources
```

Die symbolische Verknüpfung `/usr/src/linux` zeigt auf den neu installierten Kernel Quellpfad. Portage verwendet die symbolische Verknüpfung `/usr/src/linux` für spezielle Zwecke. Jedes ebuild mit Kernel Modulen ist so konfiguriert, dass es den Kernel Quellcode in `/usr/src/linux` verwendet. `/usr/src/linux` wird erstellt, wenn Sie Ihr erstes Kernel Quellpaket installieren, jedoch modifiziert Portage diese symbolische Verknüpfung später nicht mehr.

## 16.2 Verwenden von *genkernel* zur Kernelkompilierung

Da Sie nun den Kernel Quellcode installiert haben, müssen Sie Ihren Kernel kompilieren. Dazu gibt es zwei Möglichkeiten. Die erste verwendet unser neues *genkernel* Skript, das automatisch einen Kernel für Sie erzeugt. *genkernel* funktioniert dabei in ähnlicher Weise wie unsere Live CD. Das heißt, wenn Sie *genkernel* zur Erzeugung des Kernels verwenden, wird die Hardware Ihres System während des Bootens erkannt, wie es auch unsere Live CD macht. Da *genkernel* keine manuelle Konfiguration benötigt, ist dieses eine ideale Lösung für Benutzer, die sich nicht mit dem Kompilieren eines eigenen Kernels auskennen.

Schauen wir uns nun *genkernel* an. Zuerst müssen Sie *genkernel* emergieren:

**Befehlsauflistung 49:** *genkernel* emergieren

```
# emerge -k genkernel
```

Kompilieren Sie nun Ihren Kernel mittels *genkernel*:

### Notiz

**Erfahrene Benutzer:** Sie können auch *genkernel --config* eingeben, damit Sie *genkernel* anpassen können, bevor die Kompilierung beginnt.

**Befehlsauflistung 50:** *genkernel* ausführen

Wenn Sie *genkernel* 1.2 (auf dem 1.4-20030803 x86/i686 GRP Set enthalten) verwenden, geben Sie folgendes ein:

```
# genkernel gentoo-sources
```

Verwenden Sie *genkernel* 1.4 oder neuer, brauchen Sie keinen speziellen Kernel angeben:

```
# genkernel
```

```
Gentoo Linux genkernel, version 1.4
```

```
Copyright 2003 Gentoo Technologies, Inc., Bob Johnson, Daniel Robbins
```

```
Distributed under the GNU General Public License version 2
```

Settings:

```
compile optimization: 1 processor(s)
source tree: /usr/src/linux-2.4.20-gaming-r3
config: gentoo (customized)
config loc: /etc/kernels/config-2.4.20-gaming-r3
initrd config: (default) /etc/kernels/settings
```

```
* Running "make oldconfig"...
* Logging to /var/log/genkernel.log...
* Starting 2.4.20-gaming-r3 build...
* Running "make dep"...
* Running "make bzImage"...
* Running "make modules"...
* Running "make modules_install"...
* Moving bzImage to /boot/kernel-2.4.20-gaming-r3...
```

```
* Building busybox...
* Creating initrd...

* Build completed successfully!

* Please specify /boot/kernel-2.4.20-gaming-r3 and /boot/initrd-2.4.20-gaming-r3
* when customizing your boot loader configuration files.
```

Sobald *genkernel* beendet ist, wird ein neuer Kernel, ein kompletter Satz an Modulen und die **initial root disk** (initrd) erzeugt. Notieren Sie sich den Namen des Kernels und des initrd, da Sie sie später für die Bootloader Konfigurationsdatei benötigen werden. Wir werden den Kernel und die initrd später in diesem Dokument zum Konfigurieren des Bootloaders verwenden. Die initrd wird direkt nach dem Booten gestartet, um eine Hardware Autoerkennung durchzuführen (wie auf unserer Live CD) bevor Ihr "wirkliches" System startet.

Lassen Sie uns nun einen weiteren Schritt durchführen, damit Ihr System wie die Live CD startet -- emergieren Sie *hotplug*. Während die initrd die Hardware automatisch erkennt, wird *hotplug* alles andere erkennen. Um *hotplug* zu emergieren und aktivieren, geben Sie folgendes ein:

**Befehlsauflistung 51:** emergieren und aktivieren von hotplug

```
# emerge -k hotplug
# rc-update add hotplug default
```

Da Sie nun Ihr System mit *genkernel* konfiguriert haben, können Sie den folgenden Abschnitt "Manuelle Kernelkonfiguration" überspringen.

## 16.3 Manuelle Kernelkonfiguration

Haben Sie sich dazu entschieden, nicht *genkernel* zur Erstellung Ihres Kernels einzusetzen, wird Sie dieser Abschnitt durch die Konfiguration und Installation eines Kernels von Hand begleiten. Achten Sie darauf, dass */usr/src/linux* eine symbolische Verknüpfung zu Ihren installierten Kernelquellen ist und automatisch von Portage während des emergieren gesetzt wurde. Haben Sie mehrere Kernelquellen installiert, müssen Sie */usr/src/linux* auf die richtigen Kernelquellen ändern.

### Warnung

Konfigurieren Sie Ihren eigenen Kernel, seien Sie mit der **grsecurity** Option vorsichtig. Wenn Sie zu scharfe Sicherheitseinstellungen vornehmen, kann dies dazu führen, dass bestimmte Programme (wie z.B. X) nicht einwandfrei laufen. Zweifeln Sie, so lassen Sie es aus.

### Notiz

Wenn Sie dieselbe Konfiguration wie auf der Live CD wünschen oder Ihre Konfiguration ausgehend davon anpassen möchten, sollten Sie `cd /usr/src/linux && cat /proc/config > .config && make oldconfig` ausführen. Wenn Sie nicht die *xfs-sources* verwenden, werden Sie zu ein paar Unterschieden zwischen Ihrem gewählten Kernel und den *xfs-sources* gefragt.

**Befehlsauflistung 52:** Konfigurieren des Linux Kernel

```
# cd /usr/src/linux
# make menuconfig
```

### Warnung

Damit Ihr Kernel später ordentlich läuft, gibt es einige Optionen, die Sie überprüfen sollten -- das heisst, sie sollten **eingeschaltet und nicht als Modul kompiliert** werden. Seien Sie sicher, dass "ReiserFS" eingeschaltet ist, wenn Sie ReiserFS Partitionen verwenden; das selbe gilt für "Ext3". Wenn Sie XFS verwenden, schalten Sie die "SGI XFS filesystem support" Option ein. Außerdem ist es immer eine gute Idee "Ext2" eingeschaltet zu haben, gleichgültig ob Sie es verwenden oder nicht.

Hier sind einige allgemeine Optionen, die Sie benötigen:

**Befehlsauflistung 53:** make menuconfig Optionen

```
Code maturity level options --->
  [*] Prompt for development and/or incomplete code/drivers"
(Sie benötigen dieses, um einige der Optionen weiter unten zu aktivieren.)
  ...

File systems --->
```

```

<*> Reiserfs support
(Nur benötigt, wenn Sie ReiserFS verwenden.)
...
<*> Ext3 journalling file system support
(Nur benötigt, wenn Sie ext3 verwenden.)
...
[*] Virtual memory file system support (former shm fs)
(Erforderlich für Gentoo Linux.)
...
<*> JFS filesystem support
(Nur benötigt, wenn Sie JFS verwenden.)
...
[*] /proc file system support
(Erforderlich für Gentoo Linux.)
[*] /dev file system support (EXPERIMENTAL)
[*] Automatically mount at boot
(Erforderlich für Gentoo Linux.)
[ ] /dev/pts file system for Unix98 PTYs
(Schaltet Sie es aus; es wird nicht gebraucht, solange Sie keinen 2.6er Kernel verwenden)
...
<*> Second extended fs support
(Nur benötigt, wenn Sie ext2 verwenden.)
...
<*> XFS filesystem support
(Nur benötigt, wenn Sie XFS verwenden.)

```

Verwenden Sie PPPoE um sich ins Internet einzuwählen (z.B. mit DSL), benötigen Sie die folgen Optionen (fest im Kernel kompiliert oder besser als Modul): "PPP (point-to-point protocol)", "PPP support for async serial ports", "PPP support for sync tty ports". Die beiden Komprimierungs-Optionen stören nicht, aber werden auch nicht unbedingt benötigt; gleiches gilt für die "PPP over Ethernet" Option, die nur von **rp-pppoe** verwendet wird, wenn es das PPPoE-Modul des Kernels verwenden soll.

Besitzen Sie einen IDE CD-Brenner, dann sollten Sie die SCSI Emulation aktivieren. Schalten Sie "ATA/IDE/MFM/RLI support" ---> "IDE, ATA, and ATAPI Block devices" ---> "SCSI emulation support" (üblicherweise als Modul) ein, dann unter "SCSI support" den "SCSI support", "SCSI CD-ROM support" und "SCSI generic support" (wieder als Modul) einschalten. Um diese Module automatisch beim Booten zu laden, führen Sie das Kommando `echo -e "ide-scsi\nsg\nsr_mod" >> /etc/modules.autoload.d/kernel-2.4` aus.

Vergessen Sie nicht die Unterstützung für Ihre Netzwerkkarte in den Kernel hinzuzufügen.

### Notiz

Wenn Sie es vorziehen, können Sie Gentoo Linux mit einem 2.2 Kernel installieren. Jedoch zahlen Sie dafür einen hohen Preis: Sie werden alle raffinierten Erweiterungen der 2.4er Kernel Serie verlieren (wie z.B. XFS und tmpfs Dateisysteme, iptables und mehr), obwohl der 2.2 Kernel mit ReiserFS und devfs Unterstützung erweitert werden kann. Die Gentoo Linux Boot Skripte benötigen entweder Unterstützung für tmpfs oder ramdisk im Kernel, sodass Benutzer von 2.2 Kernel die ramdisk Unterstützung im Kernel einkompiliert haben müssen (nicht als Modul). Es ist **wichtig**, dass die `gentoo=notmpfs` Option zur Kernel Startzeile in der `/boot/grub/grub.conf` oder der append Zeile in der `/etc/lilo.conf` hinzugefügt wird, sodass der 2.2 Kernel eine ramdisk für die Boot Skripte mounted anstelle von tmpfs. Wenn Sie sich dafür entscheiden, nicht devfs zu verwenden, dann sollte stattdessen die Option `gentoo=notmpfs,nodevfs` eingesetzt werden.

**Befehlsauflistung 54:** Kompilieren und Installieren des Kernels

```

# make dep && make clean bzImage modules modules_install
# cp /usr/src/linux/arch/i386/boot/bzImage /boot

```

## 16.4 Installation von zusätzlichen Hardware-spezifischen ebuilds

Abschließend installieren Sie noch kernelabhängig ebuilds für zusätzliche Hardware Ihres Systems. Hier ist eine Liste von ebuilds, die Sie emergen können:

| Ebuild        | Zweck   | Kommando                             |
|---------------|---|--------------------------------------|
| nvidia-kernel | Beschleunigte NVIDIA Grafik für XFree86                     | <code>emerge -k nvidia-kernel</code> |
| nforce-net    | On-board Ethernet Controller auf NVIDIA NForce(2) Mainboard | <code>emerge nforce-net</code>       |
| nforce-audio  | On-board Audio auf NVIDIA NForce(2) Mainboards              | <code>emerge nforce-audio</code>     |
| e100          | Intel e100 Fast Ethernet Adapters                           | <code>emerge e100</code>             |

|             |   |   |
|-------------|---|---|
| e1000       | Intel e1000 Gigabit Ethernet Adapters   | <i>emerge e1000</i>                             |
| emu10k1     | Creative Sound Blaster Live!/Audigy Unterstützung   | <i>emerge emu10k1</i>                           |
| ati-drivers | Beschleunigte ATI Radeon 8500+/FireGL Grafik für XFree86  | <i>emerge ati-drivers</i>                       |
| xfree-drm   | Beschleunigte Grafik für ATI Radeon bis 9200, Rage128, Matrox, Voodoo und andere Karten für XFree86 | <i>VIDEO_CARDS="ihrekarte" emerge xfree-drm</i> |

Die Pakete *nvidia-kernel*, *ati-drivers* und *xfree-drm* bedürfen einer weiteren Konfiguration zum Aktivieren. Alle anderen gelisteten ebuilds sollten während des Bootens durch das *hotplug* Paket automatisch erkannt werden. Wenn Sie *hotplug* nicht verwenden, fügen Sie die entsprechenden Module zu [/etc/modules.autoload.d/kernel-2.4](#) hinzu.

Mehr Informationen zu *xfree-drm* finden Sie in unserem [Direct Rendering Guide](#) (derzeit nur in Englisch).

## 16.5 Installation eines System Loggers

Ihr neuer angepasster Kernel (und seine Module) sind nun installiert. Wählen Sie jetzt einen System Logger, den Sie installieren möchten. Es gibt *sysklogd*, eine traditionelle Reihe an System Logger Diensten, als auch *msyslog*, *syslog-ng* und *metalog*. Im Zweifelsfall versuchen Sie *syslog-ng*, da es sehr flexibel und featurereich ist. Um den System Logger Ihrer Wahl zu emergieren, geben Sie **eine** der folgenden vier Möglichkeiten ein:

**Befehlsauflistung 55:** System Logger emergieren

```
# emerge -k app-admin/sysklogd
# rc-update add sysklogd default
oder
# emerge -k app-admin/syslog-ng
# rc-update add syslog-ng default
oder
# emerge -k app-admin/metalog
# rc-update add metalog default
oder
# emerge -k app-admin/msyslog
# rc-update add msyslog default
```

### Wichtig

Haben Sie *metalog* gewählt, lesen Sie [Kapitel 6, Abschnitt 3 der Gentoo Linux FAQ](#) zu metalogs Pufferung.

Sie können nun wahlweise zusätzlich einen Cron Dienst wählen, um regelmäßig auftretende Vorgänge automatisiert ausführen zu lassen. Zur Zeit gibt es *dcron*, *fcron* und *vcron*. Wenn Sie sich nicht entscheiden können, wählen Sie *vcron*.

**Befehlsauflistung 56:** Cron Dienst installieren

```
# emerge -k sys-apps/dcron
# rc-update add dcron default
# crontab /etc/crontab
oder
# emerge -k sys-apps/fcron
# rc-update add fcron default
# crontab /etc/crontab
oder
# emerge -k sys-apps/vcron
# rc-update add vcron default
```

Sie brauchen nicht *crontab /etc/crontab* bei Verwendung von *vcron* ausführen.

Weitere Informationen, um Programme und Dienste während des Systemsstarts auszuführen, finden Sie im [Gentoo Linux Init System Handbuch](#).

## 17. Installation von diversen benötigten Paketen

Falls Sie *rp-pppoe* benötigen, um sich ins Internet einzuwählen, so sollten Sie dieses nun installieren.

**Befehlsauflistung 57:** Installation von *rp-pppoe*

```
# USE="-X" emerge rp-pppoe
GRP Benutzer geben folgendes ein:
# emerge -K rp-pppoe
```

### Notiz

Das `USE="-X"` verhindert die Installation des optionalen X Interface für `pppoe`, was an dieser Stelle angegeben werden sollte, da sonst X und alle seine abhängigen Pakete installiert werden. Sie können `rp-pppoe` mit X Unterstützung später noch einmal neu bauen. Die GRP Version von `rp-pppoe` unterstützt das optionale X Interface. Verwenden Sie kein GRP, kompilieren Sie `rp-pppoe` wie im ersten Beispiel genannt.

### Notiz

Bedenken Sie, dass `rp-pppoe` nur installiert, aber nicht konfiguriert wird. Dazu rufen Sie nochmals `adsl-setup` auf, nachdem Sie das erste Mal von Ihrem Gentoo System gebootet haben.

Sie benötigen weitere Pakete aus dem Portage Tree, wenn Sie optionale Erweiterungen wie XFS, ReiserFS oder LVM verwenden.

### Befehlsauflistung 58: Hilfsprogramme für Dateisysteme emergent

Wenn Sie XFS verwenden, sollten Sie die XFS Tools installieren:

```
# emerge -k xfsprogs
```

Wenn Sie ReiserFS verwenden, sollten Sie die ReiserFS Tools installieren:

```
# emerge -k reiserfsprogs
```

Wenn Sie JFS verwenden, sollten Sie die JFS Tools installieren:

```
# emerge -k jfsutils
```

Wenn Sie LVM verwenden, sollten Sie das `lvm-user` Paket installieren:

```
# emerge -k lvm-user
```

Sind Sie ein Laptop-Benutzer und möchten auf die PCMCIA Slots während Ihres ersten richtigen Reboots zugreifen, installieren Sie das `pcmcia-cs` Paket.

### Befehlsauflistung 59: PCMCIA-cs emergent

```
# emerge -k sys-apps/pcmcia-cs
```

## 18. Benutzer-Management

### 18.1 Das Passwort für root vergeben

Bevor Sie es vergessen, sollten Sie root ein Passwort vergeben:

#### Befehlsauflistung 60: root Passwort vergeben

```
# passwd
```

### 18.2 Einen Benutzer anlegen

Als root auf einem Unix/Linux System zu arbeiten ist **gefährlich** und sollte so oft wie möglich vermieden werden. Von daher ist es **stark** empfehlenswert einen Benutzer für die allgemeinen Aufgaben anzulegen.

#### Befehlsauflistung 61: Benutzer hinzufügen

```
# useradd Neuer_Benutzer -m -G users,wheel,audio -s /bin/bash
# passwd Neuer_Benutzer
```

Ersetzen Sie natürlich `Neuer_Benutzer` durch Ihren Benutzernamen.

Wann immer Sie eine Aufgabe erledigen wollen, die nur root ausführen darf, benutzen Sie `su` - um die root-Privilegien zu erhalten oder werfen Sie einen Blick in das `sudo`-Paket.

## 19. Festlegen des Hostname

Ändern Sie die Datei `/etc/hostname` so, dass sie den Namen Ihres Rechners in einer einzelnen Zeile enthält, z.B. `meinrechner`.

**Befehlsauflistung 62:** Hostname festlegen

```
# echo meinrechner > /etc/hostname
```

Editieren Sie daraufhin die Datei `/etc/dnsdomainname`, sodass Sie Ihren DNS Domainnamen enthält, z.B. `meinedomain.lan`.

**Befehlsauflistung 63:** DNS Domainnamen konfigurieren

```
# echo meinedomain.lan > /etc/dnsdomainname
```

Befinden Sie sich in einer NIS Domain, schreiben Sie den Domainnamen in `/etc/nisdomainname`.

**Befehlsauflistung 64:** NIS Domainnamen konfigurieren

```
# echo nis.meinedomain.lan > /etc/nisdomainname
```

### Notiz

Eventuell erhalten Sie die Daten von Ihrem Netzwerkadministrator. Benutzen Sie den Rechner ausschließlich in einem lokalen Netzwerk, verwenden Sie keine echten Domains, mit denen Sie in Kollision geraten können. Aus diesem Grund gibt es die Top-Level-Domain `.lan`, um zu kennzeichnen, dass es sich ausschließlich um eine lokale Domain handelt.

## 20. Die `/etc/hosts` anpassen

Die Datei `/etc/hosts` enthält eine Liste von IP-Adressen und ihren zugehörigen Hostnamen. Sie wird vom System verwendet, um IP-Adressen von Hostnamen aufzulösen, die nicht in Ihrem Nameserver verzeichnet sind. Hier ist ein Beispiel für diese Datei:

**Befehlsauflistung 65:** Beispiel für `/etc/hosts`

```
127.0.0.1    localhost
# Die nächste Zeile enthält die IP-Adresse Ihres Rechners in Verbindung mit dem Hostnamen
192.168.1.1  meinrechner.meinedomain.lan
```

### Notiz

Wenn Sie Ihre IP-Adresse von einem DHCP-Server beziehen, ist es hilfreich, ihren Hostnamen hinter `localhost` einzutragen. Das hilft GNOME als auch anderen Programmen in der Namensauflösung.

## 21. Abschließende Netzwerkkonfiguration

### 21.1 Die Kernel Module laden

Fügen Sie die Namen von allen Modulen, die für ein ordentliches Funktionieren Ihres Systems erforderlich sind, in die Datei `/etc/modules.autoload.d/kernel-2.4` hinzu (Sie können zusätzliche Optionen für ein Modul auf derselben Zeile hinzufügen). Wenn Gentoo Linux bootet lädt es diese Module automatisch. Besonders wichtig sind Ihre Netzwerkkarten Module, wenn Sie diese als Modul kompiliert haben:

**Befehlsauflistung 66:** `/etc/modules.autoload.d/kernel-2.4`

```
Angenommen Sie verwenden eine 3COM Karte, schauen Sie im Verzeichnis
/lib/modules/<kernel version>/kernel/drivers/net nach dem Namen des
Moduls. In diesem Fall wäre es:
3c59x
```

### 21.2 Konfigurieren der Netzwerkkarte

Passen Sie die Datei `/etc/conf.d/net` für das erste Booten an.

**Befehlsauflistung 67:** Netzwerk Konfiguration zum Booten

```
# nano -w /etc/conf.d/net
```

Möchten Sie, dass eth0 automatisch seine IP-Adresse bezieht, setzen Sie *iface\_eth0* auf **dhcp**. Andererseits geben Sie Ihre IP-Adresse, die Broadcast-Adresse und die Netzmaske an. Haben Sie mehrere Netzwerkkarten, machen Sie das gleiche für *iface\_eth1*, *iface\_eth2* usw.

Fügen Sie nun das *net.eth0* Init-Skript zum default runlevel hinzu, **wenn** es sich nicht um eine PCMCIA Netzwerkkarte handelt:

**Befehlsauflistung 68:** Automatisch die Netzwerkkarte während des Bootens starten

```
# rc-update add net.eth0 default
```

Haben Sie mehr als eine Netzwerkkarte oder Tokenring-Karten, müssen Sie zusätzliche *net.ethx* oder *net.trx* Skripte erstellen, eins für jede Karte (x = 1, 2, ...):

**Befehlsauflistung 69:** Mehrere Netzwerkkarten

```
# cd /etc/init.d
# ln -s net.eht0 net.ethx
```

Fügen Sie nun jedes erstellte Init-Skript zum default runlevel hinzu (und wieder nur, wenn es sich nicht um eine PCMCIA Netzwerkkarte handelt):

**Befehlsauflistung 70:** net.ethx zum default runlevel hinzufügen

```
# rc-update add net.ethx default
```

## 21.3 Nur für PCMCIA Benutzer

Verwenden Sie eine PCMCIA Karte, schauen Sie in die Datei */etc/conf.d/pcmcia*, um festzustellen, dass alle Einstellungen korrekt sind und führen Sie das folgende Kommando aus:

**Befehlsauflistung 71:** PCMCIA Dienst automatisch starten

```
# rc-update add pcmcia boot
```

Dadurch werden die PCMCIA Treiber automatisch beim Ausführen des Netzwerkkarten Skripts mitgeladen. Die entsprechende Datei */etc/init.d/net.eth\** wird durch den PCMCIA Dienst automatisch gestartet.

## 22. Abschließende Konfiguration: Basiseinstellungen vornehmen (inklusive Tastaturbelegung einstellen)

**Befehlsauflistung 72:** Basiseingellungen

```
# nano -w /etc/rc.conf
```

Folgen Sie den Anleitungen in dieser Datei, um die Basiseinstellungen vorzunehmen. Die *CLOCK* Einstellung sollte Ihrem System entsprechend eingestellt werden. Für eine deutsche Tastaturbelegung ändern Sie *KEYMAP* in *KEYMAP="de-latin1"* (für weitere Tastaturbelegungen schauen Sie in das Verzeichnis */usr/share/keymaps*).

## 23. Konfigurieren des Bootloaders

### 23.1 Anmerkungen

Ganz im Sinne von Gentoo haben Benutzer mehr als einen Bootloader zur Wahl. Grundsätzlich können Sie derzeit zwischen den beiden Bootloadern *GRUB* und *LILO* wählen.

Es ist nicht nötig, beide Bootloader zu installieren, es wird sogar ein Nachteil sein! Von daher wählen Sie bitte nur einen.

Außerdem werden Sie Ihren Bootloader anders anpassen müssen, wenn Sie *genkernel* (mit dem Kernel und der *initrd*) verwenden oder einen Kernel von Hand kompiliert haben. Achten Sie auf die wichtigen Unterschiede.

## 23.2 GRUB konfigurieren

Der kritische Teil bei GRUB ist das Verständnis, wie GRUB Festplatten und Partitionen anspricht. Ihre Linux Partition */dev/hda1* ist unter GRUB (*hd0,0*). Beachten Sie die Klammern um *hd0,0* -- Sie sind von Nöten. Festplatten werden von Null an gezählt anstatt mit "a" beginnend. Partitionen starten mit Null anstatt mit Eins. Außerdem werden mit *hd* nur Festplatten angesprochen und keine ATAPI-IDE Geräte, wie CDROMs oder Brenner, jedoch kann es auch für SCSI-Festplatten verwendet werden (normalerweise erhalten sie höhere Nummern als IDE Geräte, außer das BIOS wurde so konfiguriert, dass es von SCSI booten soll). Angenommen, Sie haben eine Festplatte an */dev/hda*, ein CDROM an */dev/hdb* und einen Brenner an */dev/hdc*, eine weitere Festplatte an */dev/hdd* und keine SCSI Festplatten, dann ist */dev/hdd7* unter GRUB (*hd1,6*). Es mag ein wenig trickreich klingen - und es ist wirklich trickreich - aber Sie werden sehen, dass GRUB eine Tab-Completion anbietet, die sehr hilfreich ist, wenn Sie mehrere Festplatten und Partitionen besitzen und wenn Sie sich ein wenig in der GRUB-Nummerierung verlieren. Wenn Sie ein wenig Gefühl dafür haben, dann sollten Sie GRUB installieren.

Der einfachste Weg GRUB zu installieren, ist *grub* am Prompt einzugeben:

**Befehlsauflistung 73:** GRUB installieren

```
# emerge -k grub
# grub
```

Sie sehen nun das GRUB Prompt *grub>*. Geben Sie hier die richtigen Kommandos ein, um den GRUB boot record auf Ihre Festplatte zu installieren. In diesem Beispiel wird GRUB in den MBR (Master Boot Record) installiert, sodass Sie als erstes das GRUB Prompt sehen, wenn Sie Ihren Rechner einschalten. In diesem Fall sind die Kommandos folgende:

**Befehlsauflistung 74:** GRUB in den MBR installieren

```
grub> root (hd0,0) Ihre Boot Partition
grub> setup (hd0) Wo der boot record installiert werden soll (hier ist es der MBR)
```

**Befehlsauflistung 75:** GRUB nicht in den MBR installieren

Alternativ können Sie GRUB auch woanders als den MBR installieren

```
grub> root (hd0,0) Ihre Boot Partition
grub> setup (hd0,4) Wo der boot record installiert werden soll (hier ist es /dev/hda5)
grub> quit
```

Die Kommandos bewirken folgendes: Zuerst teilen Sie mit *root ( )* GRUB Ihre Boot Partition mit (im obigen Beispiel */dev/hda1* oder (*hd0,0*) in der GRUB Schreibweise). Das zweite Kommando *setup ( )* berichtet GRUB wo es den boot record hin installieren soll, sodass es nach seinen speziellen Dateien an der mit *root ( )* angegebenen Stelle schauen soll. In dem obigen Fall wird der boot record in den MBR der Festplatte geschrieben, sodass nur */dev/hda* (oder auch als (*hd0*) bekannt) angegeben werden muss. Wenn Sie einen anderen primären Bootloader und GRUB als den sekundären Bootloader verwenden wollen, können Sie den boot record in einer bestimmten Partition installieren. In diesem Fall geben Sie die bestimmte Partition anstatt der ganzen Festplatte an. Sobald der boot record erfolgreich installiert wurde, geben Sie *quit* ein, um GRUB zu beenden.

### Notiz

Die Tab Completion von GRUB kann so benutzt werden, dass Sie beispielsweise *root (* eingeben und danach die TAB-Taste drücken, wodurch Sie eine Liste der verfügbaren Geräte (nicht nur Festplatten) erhalten. Drücken Sie die TAB-Taste nachdem Sie *root (hd* eingegeben haben, wird GRUB Ihnen die verfügbaren Festplatten anzeigen. Und drücken Sie die TAB-Taste nachdem Sie *root (hd0,* eingegeben haben, zeigt GRUB Ihnen eine Liste der Partitionen auf dieser Festplatte. Mittels der Tab Completion können Sie recht effektiv den Syntax überprüfen und die richtige Wahl treffen.

Gentoo Linux ist nun installiert, aber Sie müssen noch die Datei `/boot/grub/grub.conf` erzeugen, um ein nettes GRUB Boot Menu zu erhalten.

### Wichtig

Um eine rückwärtige Kompatibilität zu GRUB zu bewahren, erzeugen Sie eine Verknüpfung von `grub.conf` zu `menu.lst`. Sie können dieses mit `ln -s /boot/grub/grub.conf /boot/grub/menu.lst` erledigen.

Erstellen Sie jetzt die Datei `grub.conf` (`nano -w /boot/grub/grub.conf`) mit dem folgendem Inhalt:

### Befehlsauflistung 76: grub.conf für GRUB

```
default 0
timeout 30
splashimage=(hd0,0)/boot/grub/splash.xpm.gz

# Wenn Sie Ihren eigenen Kernel erstellt haben, konfigurieren Sie ähnlich
zu dem folgenden:
title=Mein Gentoo Linux
root (hd0,0)
kernel (hd0,0)/boot/bzImage root=/dev/hda3

# Haben Sie genkernel verwendet, benutzen Sie folgendes:
title=Mein Gentoo Linux (genkernel)
root (hd0,0)
kernel (hd0,0)/boot/kernel-KV root=/dev/hda3
initrd (hd0,0)/boot/initrd-KV

# Beispiel für ein weiteres Betriebssystem
title=Windows XP
root (hd0,5)
chainloader (hd0,5)+1
```

### Warnung

Ersetzen Sie *KV* durch den installierten Kernel.

### Notiz

(hd0,0) muss ohne Leerzeichen in den Klammern geschrieben werden.

### Wichtig

Wenn Sie vorab die SCSI Emulation für einen IDE CD Brenner eingerichtet haben, dann benötigen Sie den Zusatz `hdx=ide-scsi` in der *kernel* Zeile der `grub.conf` ("hdx" entspricht dem Gerät Ihres Brenners).

Nachdem Sie die Datei gespeichert haben, ist Ihre Gentoo Linux Installation beendet. Wenn Sie den ersten Eintrag im Boot Menu wählen, wird GRUB ohne weiteres Gentoo Linux starten. Der zweite Teil in der Datei ist optional und zeigt Ihnen, wie Sie GRUB nutzen können, um Windows zu booten.

### Notiz

(hd0,0) sollte zu Ihrer boot Partition (`/dev/hda1` in dem obigen Beispiel) und `/dev/hda3` zu Ihrer root Partition zeigen. (hd0,5) beinhaltet den Windows Bootloader.

### Notiz

Der Pfad zum Kernel Image ist relativ zur Boot Partition. Wenn Sie beispielsweise eine separate Boot Partition unter (hd0,0) und Root Partition in (hd0,1) haben, werden alle Pfade in der obigen `grub.conf` zu `/bzImage`.

Wenn Sie weitere Optionen dem Kernel mitteilen müssen, fügen Sie diese einfach am Ende der *kernel* Zeile ein. Im obigen Beispiel wurde bereits eine Option (`root=/dev/hda3`) übergeben, aber Sie können weitere hinzufügen. Beispielsweise können Sie `devfs` ausschalten (nicht empfehlenswert, ausser Sie wissen, was Sie tun), indem Sie `gentoo=noddevfs` zum *kernel* Kommando hinzufügen.

### Notiz

Anders als in früheren Versionen von Gentoo Linux brauchen Sie nicht mehr `devfs=mount` hinzufügen, um `devfs` einzuschalten, da es nun standardmäßig aktiviert ist.

## 23.3 LILO konfigurieren

Während GRUB eine neue Alternative für viel Benutzer ist, ist es nicht immer die beste Wahl. LILO, der LINUXLoader ist der erprobtere Linux Bootloader. Hier sehen Sie, wie Sie LILO installieren, wenn Sie es anstelle von GRUB verwenden möchten:

Zuerst müssen Sie LILO emergen:

### Befehlsauflistung 77: LILO emergen

```
# emerge -k lilo
```

Nun muss LILO mit der Datei `/etc/lilo.conf` konfiguriert werden. Folgend eine Beispielkonfiguration:

**Befehlsauflistung 78:** Beispiel zur `lilo.conf`

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
lba32
default=linux

# Benutzen Sie die folgenden vier Zeilen als Beispiel, wenn Sie Ihren
Kernel selbst kompiliert haben
image=/boot/bzImage
    label=linux
    read-only
    root=/dev/hda3

# Haben Sie genkernel verwendet, tragen Sie folgendes ein
image=/boot/kernel-KV
    label=gk_linux
    root=/dev/hda3
    initrd=/boot/initrd-KV
    append="root=/dev/hda3 init=/linuxrc"

# Beispiel für ein weiteres Betriebssystem
other=/dev/hda1
    label=dos
```

### Warnung

Ersetzen Sie *KV* durch die Kernel-Version, die Sie installiert haben. Beachten Sie außerdem, dass *default=* auf das entsprechende Label zeigt (*gk\_linux* bei Verwendung von *genkernel*).

- *boot=/dev/hda* teilt LILO mit, dass es sich auf die erste Festplatte am ersten IDE Controller installieren soll.
- *map=/boot/map* gibt die *map* Datei an. Üblicherweise braucht dieser Eintrag nicht geändert werden.
- *install=/boot/boot.b* veranlasst LILO die angegebene Datei als den neuen Boot Sektor zu installieren. Üblicherweise braucht dieser Eintrag nicht geändert werden. Wenn dieser Eintrag fehlt, verwendet LILO eine eigene Datei anstatt von `/boot/boot.b`.
- Wenn *prompt* angegeben wird, zeigt LILO den klassischen *lilo:* Prompt beim Booten an. Obwohl es nicht empfehlenswert ist, dass Sie den Prompt abschalten, können Sie ihn entfernen und weiterhin einen Prompt erhalten, wenn Sie die [Shift]-Taste während des Bootens halten.
- *timeout=50* setzt die Zeitspanne, wie lange LILO auf eine Eingabe des Benutzers warten soll, bis es von selbst mit dem Eintrag *default* fortfährt. Der Wert ist in Zehntelsekunden anzugeben, und 50 der vorgegebene Wert.
- *lba32* beschreibt die Geometrie der Festplatte. Ein anderer hier üblicher Eintrag ist *linear*. Sie sollten diesen Eintrag nicht ändern, außer Sie wissen wirklich, was Sie tun. Möglicherweise können Sie Ihr System dazu bringen, dass es nicht bootet.
- *default=linux* zeigt auf das automatisch von LILO zu bootende Betriebssystem, das in einer Liste weiter unten in der Datei angegeben wird. Der Wert muss zu einem von *label=* angegebenen Wert gleich sein.
- *image=/boot/bzImage* gibt den mit diesem Abschnitt zu bootenden Linux Kernel an.
- *label=linux* ist ein Name für das Betriebssystem im aktuellen Abschnitt. In diesem Fall zeigt der *default* Eintrag auch auf diesen Namen.
- *read-only* veranlasst, dass die Root Partition (siehe nächste Zeile) zunächst schreibgeschützt gemountet wird und somit nicht während des Bootens verändert werden kann.
- *root=/dev/hda3* gibt LILO an, welche Partition als Root Partition verwendet werden soll.

Nachdem Sie Ihre `lilo.conf` editiert haben, müssen Sie LILO dazu bringen, die Informationen in den MBR (Master Boot Record) zu laden:

**Befehlsauflistung 79:** LILO ausführen

```
# /sbin/lilo
```

LILO ist nun konfiguriert und Ihr Rechner ist bereit, um Gentoo Linux zu starten!

## 23.4 Framebuffer verwenden

Möchten Sie Framebuffer verwenden sollten Sie `vga=xxx` als Parameter Ihrem Bootloader übergeben. `xxx` ist einer der Werte aus der folgenden Tabelle:

### **Bittiefe640x480800x6001024x7681280x1024**

|        |     |     |     |     |
|--------|-----|-----|-----|-----|
| 8 bpp  | 769 | 771 | 773 | 775 |
| 16 bpp | 785 | 788 | 791 | 794 |
| 32 bpp | 786 | 789 | 892 | 795 |

LILO Benutzer setzen `vga=xxx` am Kopf der Konfigurationsdatei ein.

GRUB Benutzer hängen `vga=xxx` an die Zeile `kernel (hd0,0)...` an.

## 24. Bootdiskette erstellen

### 24.1 GRUB Bootdiskette

#### **Wichtig**

Vergessen Sie nicht eine Diskette in Ihr Laufwerk einzulegen bevor Sie fortfahren.

Es ist immer eine gute Idee eine Bootdiskette bei der ersten Installation einer Linux Distribution zu erstellen. Dieses ist eine Sicherheitsmaßnahme und generell empfehlenswert. Lässt sich der Bootloader nicht in der chroot Umgebung installieren, **müssen** Sie eine Bootdiskette mit GRUB erstellen. Gehören Sie also zu dieser Gruppe von Benutzern, erstellen Sie eine GRUB Bootdiskette und wenn Sie das erste Mal Rebooten, installieren Sie GRUB in den MBR. Erstellen Sie Ihre Bootdiskette folgendermaßen:

**Befehlsauflistung 80:** GRUB Bootdiskette erstellen

```
# cd /usr/share/grub/i386-pc/  
# cat stage1 stage2 > /dev/fd0
```

Sie können nun von dieser Diskette rebooten. Beim `grub>` Prompt der Diskette können Sie die oben beschriebenen benötigten `root` und `setup` Kommandos ausführen.

### 24.2 LILO Bootdiskette

#### **Wichtig**

Vergessen Sie nicht eine Diskette in Ihr Laufwerk einzulegen bevor Sie fortfahren.

Wenn Sie LILO verwenden ist es ebenso eine gute Idee eine Bootdiskette zu erstellen:

**Befehlsauflistung 81:** LILO Bootdiskette erstellen

```
# dd if=/boot/ihr_kernel of=/dev/fd0  
(Das funktioniert nur, wenn Ihr Kernel kleiner als 1.4MB ist.)
```

## 25. GRP verwenden

GRP Benutzer können an dieser Stelle Binärpakete installieren:

**Befehlsauflistung 82:** Installieren von GRP

```
# emerge -k xfree
```

CD 1 enthält genügend Pakete, um ein funktionierendes System mit XFree86 zu installieren. Zusätzlich enthält CD 2 des 2-CD GRP Set weitere Applikationen, wie KDE, GNOME, Mozilla und andere. Zur Installation dieser Pakete müssen Sie erst Ihr neues Gentoo System rebooten (weiter unten im Abschnitt "Installation beendet!" beschrieben). Wenn Ihr Gentoo System von der Festplatte aus läuft, können Sie die zweite CD mounten und Dateien kopieren:

**Befehlsauflistung 83:** Binärpakete von CD 2 laden

```
# mount /dev/cdrom /mnt/cdrom
# cp -a /mnt/cdrom/packages/* /usr/portage/packages/
```

Nun können weitere Pakete in ähnlicher Weise installiert werden. Zum Beispiel:

**Befehlsauflistung 84:** KDE mittels GRP installieren

```
# emerge -k kde
```

## 26. Installation beendet!

Nun, Ihr Gentoo Linux ist installiert. Der einzige verbleibende Schritt ist, einige Konfigurationsdateien anzupassen, die chroot Umgebung zu beenden, Ihre Partitionen sicher zu unmounten und das System zu rebooten:

### Warnung

*etc-update* zeigt Ihnen eine Liste der zu überarbeitenden Konfigurationsdateien von denen es eine neuere Version gibt. Beachten Sie, dass keine der Konfigurationsdateien mit essentiellen Systemveränderungen darunter sind (wie z.B. */etc/fstab*, */etc/make.conf*, */etc/rc.conf*, ...). Spielen Sie die weniger wichtigen Dateien ein, löschen Sie die Updates der anderen oder schauen Sie sich die Unterschiede an und updaten diese von Hand.

**Befehlsauflistung 85:** Rebooten des Systems

```
# etc-update
# exit
(Dieses beendet die chroot Umgebung; Sie können genauso gut ^D (ctrl-d) drücken)
# cd /
# umount /mnt/gentoo/boot
# umount /mnt/gentoo/proc
# umount /mnt/gentoo
# reboot
```

### Notiz

Nach dem Rebooten sollten Sie *modules-update* ausführen, um die Datei */etc/modules.conf* zu erstellen. Anstatt diese Datei direkt zu verändern, sollten Sie Änderungen in den Dateien in */etc/modules.d/* vornehmen.

Wenn Sie Fragen haben oder sich bei der Entwicklung von Gentoo Linux beteiligen wollen, abonnieren Sie die [gentoo-user](#) und [gentoo-dev](#) Mailingliste (weitere Informationen auf der [Mailinglisten Webseite](#)). Außerdem ist die [Anleitung zur Gentoo Linux Desktop Konfiguration](#) recht praktisch für die weitere Konfiguration Ihres Gentoo Linux Systems. Der nützliche [Portage Benutzer Leitfaden](#) hilft Ihnen die Grundlagen von Portage zu verstehen. Darüberhinaus gibt es etliche weitere [Gentoo Linux Dokumente](#). Haben Sie Fragen, die die Installation betreffen, schauen Sie in die [FAQ \(häufig gestellte Fragen\)](#). Herzlich Willkommen und haben Sie Spaß mit Gentoo Linux!

## 27. Gentoo auf nicht weit verbreiteter Hardware

### 27.1 Hardware ATA RAID

Benutzer, die Gentoo auf einem Hardware ATA RAID installieren möchten, müssen bei den folgenden Schritten aufpassen, um ordnungsgemäß Gentoo Linux zu installieren:

- Starten Sie die Live CD mit der *doataraid* Kerneloption.
- Falls Sie vergessen haben *doataraid* während des Bootens auszuwählen oder eines der Module auf mysteriöse Weise nicht lädt, laden Sie dieses von Hand:

**Befehlsauflistung 86:** Laden des RAID Modul

```
# modprobe ataraid
Für Promise RAID Controller:
# modprobe pdcraid
Für Highpoint RAID Controller:
# modprobe hptraid
```

- Manche ATA RAID Controller verlangen einen Reboot nach der Partitionierung; andernfalls wird die Formatierung fehlschlagen.
- Mounten Sie den dev-Verzeichnisbaum in die neue Gentoo Umgebung, bevor Sie chroot ausführen:

**Befehlsauflistung 87:** /dev nach /mnt/gentoo/dev mounten

```
# mount -o bind /dev /mnt/gentoo/dev
```

- Während der Kernel Konfiguration wählen Sie die benötigten RAID Optionen aus:

**Befehlsauflistung 88:** RAID während der Linux Kernel Konfiguration

```
Für Highpoint RAID Controller:
ATA/IDE/HFM/RLL support --->
[*] HPT36X/37X chipset support
[*] Support for IDE Raid controllers
[*] Highpoint 370 software RAID
Für Promise RAID Controllers:
ATA/IDE/MFM/RLL support --->
[*] PROMISE PDC202{46|62|65|67} support
und/oder
[*] PROMISE PDC202{68|69|70|71|75|76|77} support
[*] Support for IDE Raid controllers
[*] Support Promise software RAID (Fasttrak(tm))
```

- Verwenden Sie GRUB, so fügen Sie `--stage2=/boot/grub/stage2` zum `setup` Kommando hinzu, während Sie `grub` ausführen:

**Befehlsauflistung 89:** Installation von GRUB auf Hardware RAID Systemen

```
grub> root (hd0,0)
grub> setup --stage2=/boot/grub/stage2 (hd0)
grub> quit
```

Stellen Sie ebenso in der GRUB Konfiguration sicher, dass `root` auf das entsprechende RAID Device zeigt:

**Befehlsauflistung 90:** grub.conf für RAID

```
title=Mein Gentoo Linux mit RAID
root (hd0,0)
kernel (hd0,0)/boot/bzImage root=/dev/ataraid/dXpY
```

- Benutzer von LILO müssen die `root` Option auf das entsprechende RAID Device zeigen lassen:

**Befehlsauflistung 91:** lilo.conf für RAID

```
image=/boot/bzImage
label=linux
read-only
root=/dev/ataraid/dXpY
```

Haben Sie weiterhin Probleme Gentoo Linux mit Ihrem Hardware RAID zum Laufen zu bekommen, schreiben Sie diese an <http://bugs.gentoo.org>.

Danke für die Wahl von Gentoo Linux und haben Sie viel Spaß mit Ihrer neuen Installation!



## >> Gentoo Linux/PPC 1.2 Installationsanleitung

[Bitte Kapitel auswählen]

### 1. Über die Installation

Die neue Boot-CD lässt sich auf jedem NewWorld Mac booten. OldWorld Macs benötigen eine funktionierende MacOS-Partition und den *Gentoo Installer.sit*, der sich im ISO befindet. Zusätzlich bieten wir Kernel-Treiber für einen großen Teil an Apple-Hardware und Programme zum Einrichten des Netzwerks, Aufbauen einer *ssh*-Verbindung oder zum Downloaden von Dateien.

Um von der "Build-CD" zu installieren wird mindestens ein PowerPC 603 Prozessor auf einem PowerPC PCI basierendem System mit 128 MB RAM benötigt (weniger ist möglich, aber die Installation kann schrecklich langsam werden oder sogar komplett versagen). Nubus-Maschinen werden noch nicht unterstützt. Um mit der Installation zu beginnen, ist eines unserer CD ISO Images von <http://www.ibiblio.org/gentoo/releases/build/> erforderlich. Für die Installation von Gentoo Linux/PPC 1.4 gibt es nicht nur gewöhnliche PPC-Images, sondern auch für G4-Prozessoren optimierte.

Nun lassen Sie uns einen kurzen Überblick des Installationsprozesses skizzieren: Sie erstellen Partitionen, dann die Dateisysteme und entpacken darin entweder ein stage1, stage2 oder stage3 tar-Archiv. Falls Sie ein stage1 oder stage2 tar-Archiv verwenden, sind entsprechende Schritte um zu stage3 zu gelangen erforderlich. Sobald das System in einem stage3-Status ist, können Sie es konfigurieren (die Konfigurationsdateien anpassen, den Bootloader installieren, usw.) und ein voll funktionsfähiges Gentoo Linux System booten. Abhängig von Ihrer gewählten Installationsstufe werden die folgende Schritte benötigt:

| stage tar-Archiv | Installationsanforderungen   |
|------------------|--|
| 1                | Partitionen und Dateisystem erstellen, emerge rsync, bootstrap, emerge system, emerge ppc-sources, abschließende Konfiguration |
| 2                | Partitionen und Dateisystem erstellen, emerge rsync, emerge system, emerge ppc-sources, abschließende Konfiguration            |
| 3                | Partitionen und Dateisystem erstellen, emerge rsync (optional), abschließende Konfiguration                                    |

### 2. Booten

Haben Sie sich für eine CD entschieden, booten Sie diese indem Sie entweder die 'C'-Taste während des Starten gedrückt halten oder bei OldWorld-Maschinen das Installationsprogramm aufrufen und dann BootX starten. Sie werden mit jeder Menge Text und einem *boot:*-Prompt am Ende des Bildschirms begrüßt. Drücken Sie dort Enter und Linux wird von der CD starten.

Darauf wird Ihnen eine kleine Liste der verfügbaren Befehle auf der Boot-CD, inklusive *nano* (einem Klon des Editors *pico*) und Anweisungen zur Netzwerkkonfiguration, angezeigt. Dann werden Sie zur Wahl Ihrer Tastaturbelegung aufgefordert, gefolgt von der automatischen Erkennung der PCI-Geräte. Die Geräteerkennung wird automatisch die passenden Kernelmodule für die gängigen PCI SCSI-Controller und Netzwerkkarten laden. Ist dieses erfolgt, können Sie Enter drücken und gelangen in eine minimale Boot-CD Linux-Umgebung. Sie sollten in der aktuellen Konsole ein root-Prompt ("#") sehen. Weitere Konsolen können mittels *ctrl+alt-F2*, *ctrl+alt-F3* (bei PowerBooks/iBooks möglicherweise *fn+ctrl+alt-F2*, *fn+ctrl+alt-F3*) usw. gefolgt von einem Enter aktiviert werden.

### 3. Laden von Kernel-Modulen

Nun können Sie nahezu mit der Installation von Gentoo Linux beginnen, denn der Gentoo PPC ISO Kernel beinhaltet jegliche Treiber für die meistens Standard-Macs. Für einige zusätzliche Geräte haben wir weitere Module bereitgestellt. Sie können sich im Verzeichnis */lib/modules/<kernel-version>/kernel/drivers/* einen Überblick über diese verschaffen. Um ein spezielles Modul zu laden, geben Sie ein:

#### Befehlsauflistung 1

```
# modprobe <Modulname>
      (ersetzen Sie <Modulname> durch Ihr gewünschtes Modul)
```

### 4. Konfigurieren und Installieren des Netzwerks

Die neue Boot-CD erlaubt Ihnen das Netzwerk zu konfigurieren und Programme wie *ssh*, *scp* oder *wget* selbst vor dem Installationsbeginn zu benutzen. Wenn Sie jetzt keine dieser Programme benötigen, sollten Sie dennoch schon das Netzwerk konfigurieren. Sobald es einmal eingestellt ist kann Portage während des Installationsprozesses in der *chroot*-Umgebung darauf zugreifen (wird für die Installation von Gentoo Linux benötigt).

## 4.2 DHCP

Netzwerkconfiguration mittels DHCP ist recht einfach. Sollte Ihr Internet-Anbieter kein DHCP bereitstellen, fahren Sie mit der [statischen Netzwerkconfiguration](#) fort.

**Befehlsauflistung 2:** Netzwerkconfiguration mit DHCP

```
# dhcpcd eth0
```

### Notiz

Einige Provider erwarten von Ihnen einen Hostnamen. Dafür hängen Sie die Option *-h myhostname* an das oben stehende *dhcpcd*-Kommando.

Falls Sie **dhcpConfig**-Warnungen erhalten, keine Panik -- die meisten Meldungen sind harmlos. Fahren Sie mit dem [Testen der Netzwerkconfiguration](#) fort.

## 4.3 Statische Netzwerkconfiguration

Sie müssen Ihre Netzwerkanbindung so weit konfigurieren, dass Sie die benötigten Quellen für den Installationsprozess herunterladen können. Außerdem wird das lokale Interface (localhost) benötigt. Bitte geben Sie dazu folgende Befehle in und ersetzen Sie *\$IFACE* durch Ihr Netzwerkinterface (üblicherweise *eth0*), *\$IPNUM* durch Ihre IP-Adresse, *\$BCAST* durch Ihre Broadcast-Adresse und *\$NMASK* durch Ihre Netzwerkmaske. Beim *route*-Befehl ersetzen Sie *\$GTWAY* durch Ihren Default-Gateway.

**Befehlsauflistung 3**

```
# /sbin/ifconfig $IFACE $IPNUM broadcast $BCAST netmask $NMASK
# /sbin/route add -net default gw $GTWAY netmask 0.0.0.0 metric 1
```

Nun sollten Sie die Datei [/etc/resolv.conf](#) erstellen, damit die Namensauflösung (Web/FTP-Seiten mittels Namen anstatt IP-Adressen finden) funktioniert.

### Notiz

Derzeit ist der einzige standardmäßig installierte Editor *nano*, ein kleiner und leicht zu benutzender E

### Notiz

Im Moment ist nur *nano*, ein kleiner, leicht zu bedienender Editor installiert. Verwenden Sie *nano* mit der Option *-w*, um den automatischen Zeilenumbruch abzuschalten.

Hier ist ein Beispiel der Datei [/etc/resolv.conf](#):

**Befehlsauflistung 4:** [/etc/resolv.conf](#) Vorlage

```
domain mydomain.com
nameserver 10.0.0.1
nameserver 10.0.0.2
```

Ersetzen Sie *10.0.0.1* und *10.0.0.2* durch die IP-Adressen Ihres primären und sekundären Domain-Name-Servers (DNS).

## 4.4 Testen der Netzwerkconfiguration

Da Ihre Netzwerkconfiguration jetzt durchgeführt ist, sollte der Befehl */sbin/ifconfig -a* ausgeben, dass Ihre Netzwerkkarte (hoffentlich) funktioniert (achten Sie auf **UP** und **RUNNING** in der Ausgabe!).

**Befehlsauflistung 5:** */sbin/ifconfig* bei einer funktionierenden Netzwerkconfiguration

```
eth0      Link encap:Ethernet  HWaddr 00:50:BA:8F:61:7A
          inet addr:192.168.0.2  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::50:ba8f:617a/10  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1498792 errors:0 dropped:0 overruns:0 frame:0
```

```
TX packets:1284980 errors:0 dropped:0 overruns:0 carrier:0
collisions:1984 txqueuelen:100
RX bytes:485691215 (463.1 Mb) TX bytes:123951388 (118.2 Mb)
Interrupt:11
```

## 4.5 Das Netzwerk ist jetzt online!

Der Netzwerkanschluss ist nun konfiguriert und nutzbar. Sie sind jetzt in der Lage, sich via *ssh*, *scp* und *wget* mit anderen Rechnern in Ihrem LAN oder im Internet zu verbinden.

## 5. Erstellen der Partitionen

Da nun das Netzwerk und der Festplatten-Controller angesprochen werden können, ist es an der Zeit die Festplatte für Gentoo Linux zu partitionieren.

Hier ist ein kurzer Überblick der Standard Gentoo Linux Partitionen: Wir erzeugen mindestens drei Partitionen: eine Swap Partition (wird als erweiterter Arbeitsspeicher benötigt), eine Root Partition (dort wird sich der Großteil von Gentoo Linux befinden) und eine spezielle Bootstrap Partition. Die Bootstrap Partition wird die Informationen des YABOOT Boot-Managers speichern. Benutzer von OldWorld-Maschinen benötigen die Bootstrap Partition nicht, da ihr System zuerst MacOS booten wird, um dann mit BootX den Linux-Kernel und die Startskripte zu initiieren.

Nun zu den diversen Dateisystemen. Momentan haben Sie vier Dateisysteme zur Wahl: XFS, ext2, ext3 (Journaling) und ReiserFS. ext2 ist ein erprobtes und das "wahre" Linux Dateisystem, jedoch unterstützt es kein Journaling. ext3 ist die neue Version von ext2, sowohl mit Metadaten Journaling als auch geordnete Schreibzugriffe und einem effektiven Daten-Journaling. Eines der beiden sollten Sie für Gentoo Linux/PPC verwenden. ReiserFS und XFS sind neue Dateisysteme mit guter Leistung, jedoch sind sie auf PPC-Systemen noch in der Erprobungsphase und somit nicht empfohlen.

### Wichtig

Falls Sie bisher **nur** MacOS/MacOS X auf Ihrem System laufen haben, müssen Sie Ihr komplettes System neu installieren. Es gibt keine Möglichkeit die MacOS-Partition(en) zu verkleinern. Weitere deutschsprachige Informationen zur Partitionierung gibt es z.B. unter <http://jules.killall5.de/writings/pismo/>.

### Warnung

Machen Sie vor dem Partitionieren ein Backup Ihrer auf der Festplatte gespeicherten Daten! Durch das Partitionieren können diese unwiderruflich gelöscht werden.

### Notiz

Erstellen Sie nun Ihre Partitionen mit dem Programm *mac-fdisk*, indem Sie *mac-fdisk /dev/hdx* eingeben (wobei *x* der Buchstabe Ihrer Festplatte ist, also *a*, *b*, *c* usw., bzw. verwenden Sie */dev/sdx* für SCSI-Festplatten). Sie benötigen mindestens eine Bootstrap Partition (*b* in *mac-fdisk* -- nur für NewWorld-Benutzer), eine Swap Partition (in etwa der doppelten Größe des RAM, jedoch sollte sie mindestens 512 MB betragen, falls Sie größere Pakete auf Ihrem System kompilieren möchten) und eine Root Partition. ? am *mac-fdisk*-Prompt gibt Ihnen eine Liste der verfügbaren Kommandos aus.

Da Sie nun ihre Partitionen mit *mac-fdisk* erstellt haben, ist es an der Zeit die Dateisysteme für die Daten zu erzeugen. Die Swap Partition erstellen Sie mit

### Befehlsauflistung 6

```
# mkswap /dev/<Swap Partition, z.B. hda3>
```

Sie können *mke2fs* zum Erzeugen von ext2-Dateisystemen verwenden:

### Befehlsauflistung 7

```
# mke2fs /dev/<Root Partition, z.B. hda4>
```

Falls Sie ext3 verwenden möchten, verwenden Sie *mke2fs -j*:

### Befehlsauflistung 8

```
# mke2fs -j /dev/<Root Partition, z.B. hda4>
```

### Warnung

Noch einmal: Wir empfehlen **nicht** ReiserFS und XFS zu verwenden! Obwohl ReiserFS und XFS auf vielen Systemen einwandfrei funktionieren, ist der häufigste Grund von Datenverlust auf PPC-Systemen eine Kombination aus ReiserFS oder XFS mit einem 2.4er-Kernel.

### Notiz

Mehr Informationen über ext3 unter Linux 2.4 finden Sie unter <http://www.zip.com.au/~akpm/linux/ext3/ext3-usage.html>.

## 6. Partitionen mounten

Aktivieren Sie nun Ihre Swap Partition, denn es könnte sein, dass Sie den virtuellen Speicher später noch benötigen:

### Befehlsauflistung 9

```
# swapon /dev/<Swap Partition, z.B. hda3>
```

Als nächstes erstellen Sie das Verzeichnis `/mnt/gentoo` und mounten Sie dorthin Ihre Root Partition:

### Befehlsauflistung 10

```
# mkdir /mnt/gentoo
# mount /dev/<Root Partition, z.B. hda4> /mnt/gentoo
```

Falls Sie für Gentoo Linux weitere Partitionen wie `/usr`, `/home` oder `/var` vorgesehen haben, mounten Sie diese nach `/mnt/gentoo/usr`, `/mnt/gentoo/home` bzw. `/mnt/gentoo/var`.

## 7. Mounten des CD-ROM

Obwohl Sie von der CD-ROM gebootet haben, ist diese noch nicht in unserer kleinen Linux-Umgebung gemountet. Um auf das komprimierte Installations-Archiv zuzugreifen (natürlich können Sie dieses auch aus dem [Internet herunterladen](#), wenn Ihr [Netzwerk](#) funktioniert), müssen Sie die CD-ROM mounten. Dazu tippen Sie folgendes ein:

### Befehlsauflistung 11

```
# mount /dev/cdroms/cdrom0 /mnt/cdrom -o ro -t iso9660
```

Auf der gemounteten CD sollten Sie die Datei `stage???.tbz2` mittels `ls /mnt/cdrom/build` sehen.

### Notiz

Falls Sie Ihr CD-ROM nicht mounten können, laden Sie sich das Installations-Archiv mit dem Programm `wget` von der [ibiblio-Webseite](#) herunter, z.B. durch Eingabe von `wget http://www.ibiblio.org/pub/Linux/distributions/gentoo/releases/build/1.2/stage1-ppc-1.2.tbz2`.

## 8. Entpacken des gewünschten Installations-Archivs

Nun ist es an der Zeit das gewünschte, gepackte Installations-Archiv nach `/mnt/gentoo` zu entpacken. Dann wechseln Sie mit `chroot` in die neue Gentoo Linux Installation.

### Wichtig

Achten Sie darauf, dass Sie das Archiv mit der Option `p` des Programms `tar` entpacken. Andernfalls werden die Rechte einiger Dateien nicht stimmen.

### Wichtig

Wenn Sie die "bau alles von Grund auf neu" (stage1) Installation gewählt haben, verwenden Sie das `stage1-ppc-1.2.tbz2`-Archiv. Natürlich können Sie auch mit stage2 oder stage3 beginnen, um Zeit auf Kosten der Flexibilität zu sparen (wir haben die Compiler Optimierung und `USE`-Variablen für Sie ausgewählt). Das stage3-Archiv beinhaltet zusätzlich eine Kopie des "Portage tree", das ein späteres `emerge rsync` überflüssig macht.

### Befehlsauflistung 12

```
# cd /mnt/gentoo
# tar -xvjpgf /mnt/cdrom/build/stage?-*.tbz2
# mount -o bind /proc /mnt/gentoo/proc
# cp /etc/resolv.conf /mnt/gentoo/etc/resolv.conf
```

### Befehlsauflistung 13

```
# chroot /mnt/gentoo /bin/bash
# env-update
>>> Regenerating /etc/ld.so.cache...
# source /etc/profile
#
```

Nachdem Sie diese Kommandos ausgeführt haben, befinden Sie sich "innerhalb" Ihrer neuen Gentoo Linux Umgebung.

## 9. Rsync

Als nächstes wird ein Durchlauf von *emerge rsync* benötigt. Dabei wird der Server [rsync.gentoo.org](http://rsync.gentoo.org) kontaktiert und die neueste Version des Portage-Verzeichnisses heruntergeladen:

### Befehlsauflistung 14

```
# emerge rsync
```

Das ungefähr 10 MB große Portage-Verzeichnis wird nun heruntergeladen.

### Wichtig

Die Adresse des rsync-Servers ist momentan [rsync://rsync.gentoo.org/gentoo-portage](http://rsync.gentoo.org/gentoo-portage). Falls Sie Probleme haben auf Gentoo rsync zuzugreifen, überprüfen Sie, dass die SYNC Variable in der Datei */etc/make.conf* auf diese Adresse gesetzt ist.

### Notiz

Sie können für den rsync-Befehl einen HTTP-Proxy einstellen, falls Sie sich hinter einer Firewall befinden. Dazu setzen Sie die Variable *RSYNC\_PROXY="hostname:port"* in der Datei */etc/make.conf* oder verwenden Sie diese als Umgebungsvariable.

## 10. Der Schritt von "stage1" zu "stage2"

### Wichtig

Falls Sie eine stage2- oder stage3-Archiv verwenden, dann haben wir den bootstrap-Prozess bereits für Sie durchgeführt. Es gibt keinen Grund, den bootstrap-Prozess ein weiteres Mal durchzuführen, außer, Sie entscheiden sich dafür nach dem *emerge rsync* die allerneuesten Gentoo Linux Komponenten zu installieren. Die meisten Benutzer, die stage2- oder stage3-Archive verwenden, werden **nicht** noch einmal den bootstrap-Prozess durchführen, da dieser selbst auf schnellen Rechnern bis zu einer Stunde in Anspruch nehmen kann. Dennoch empfehlen wir Ihnen die Lektüre dieses Kapitels und die Einstellungen in der Datei */etc/make.conf* vorzunehmen.

Da Sie nun eine funktionierende Kopie des Portage-Verzeichnisses haben, müssen Sie nun den Gentoo Linux bootstrap-Prozess durchlaufen. Dazu editieren zuerst die Datei */etc/make.conf*. In dieser Datei sollten Sie die *USE* Flags (mehr dazu in der [Gentoo Anleitung zu den USE Flags](#)) an Ihr System und Ihre Wünsche anpassen. Über die *USE* Flags können Sie steuern, welche optionale Funktionalität Sie in die Pakete eingebaut haben möchten. Normalerweise ist die vorgegebene Einstellung, **leere** oder ungesetzte *USE* Flags, ausreichend. Wichtig sind die für Ihr System passenden Einstellungen für *CHOST*, *CFLAGS* und *CXXFLAGS*. Auskommentierte Beispiele befinden sich in der Datei. Falls Sie sich hinter einer Firewall befinden, sollten Sie einen Proxy eintragen.

### Befehlsauflistung 15

```
# nano -w /etc/make.conf (Passen Sie die Einträge an)
```

### Notiz

Anwender, die ihr System wesentlich optimieren wollen oder müssen, sollten einen Blick in die Datei */etc/make.globals* werfen. Diese Datei beinhaltet alle Gentoo Linux Standardeinstellungen und sollte niemals verändert werden. Falls die Standardeinstellungen nicht ausreichend sind, können Sie neue Variablen in */etc/make.conf* setzen, da Einträge in der *make.conf* die vorgegebenen Einträge in der *make.globals* **überschreiben**. Falls Sie sich für das Optimieren mittels der *USE* Flags interessieren, schauen Sie in die Datei */etc/make.profile/make.defaults*. Wenn Sie vorhandene *USE* Flags abschalten wollen, fügen Sie in der Datei */etc/make.conf* ein entsprechendes *USE="-foo"* ein (dieses würde das Flag *foo* abschalten).

Jetzt können Sie den bootstrap-Prozess starten. Dieser benötigt etwa ein bis vier Stunden, ganz davon abhängig, wie fix Ihr Rechner ist. Dabei wird das entpackte Installations-Archiv dazu vorbereitet, den Rest des Systems zu kompilieren. Die GNU Compiler Suite sowie die GNU C Bibliothek werden gebaut. Dieses sind zeitaufwändige Vorgänge, die den Hauptteil des bootstrap-Prozesses ausmachen:

### Befehlsauflistung 16

```
# cd /usr/portage  
# scripts/bootstrap.sh
```

Der bootstrap-Prozess startet nun.

### Notiz

Portage verwendet normalerweise `/var/tmp` während des Kompilieren der Pakete, das mehrere hundert Megabyte an temporärem Speicher benötigt. Wenn Sie das Verzeichnis, in das Portage diese temporären Dateien ablegt, ändern möchten, können Sie dieses **vor** dem Starten des bootstrap-Prozesses durch Setzen der Umgebungsvariable `PORTAGE_TMPDIR` tun:

```
# export PORTAGE_TMPDIR="/otherdir/tmp"
```

Das Skript `bootstrap.sh` baut nun die Pakete `binutils`, `gcc`, `gettext` und `glibc`. Danach werden `binutils`, `gcc` sowie `gettext` mit der neuen `glibc` neu gebaut. Es muss nicht erwähnt werden, dass dieser Prozess seine Zeit braucht... Mache Sie ein kleines Nickerchen und wenn der Prozess dann durchgelaufen ist, haben Sie den Stand des stage2-Archivs erreicht.

## 11. Der Schritt von stage2 zu stage3

### Notiz

Falls Sie es noch nicht getan haben, editieren Sie die Datei `/etc/make.conf` nach Ihren Vorlieben (siehe auch Kapitel [Der Schritt von stage1 zu stage2](#)).

Sobald der bootstrap-Prozess abgeschlossen ist und Sie sich in stage2 befinden (nochmals, wenn Sie ein stage3-Archiv verwenden, dann werden diese Schritte nicht benötigt), kompilieren oder installieren Sie den Rest des Basissystems wie folgt:

### Befehlsauflistung 18

```
# export CONFIG_PROTECT=""
# emerge --pretend system
    [listet alle zu installierenden Pakete auf]
# emerge system
```

### Notiz

Die Umgebungsvariable `export CONFIG_PROTECT=""` stellt sicher, dass alle neuen Dateien, die in das Verzeichnis `/etc` installiert werden, die alten dort überschreiben (aus dem Paket `sys-apps/baselayout`). Dazu wird Portages neuer Mechanismus zum Schutz von Konfigurationsdateien umgangen. Genauer erfahren Sie, wenn Sie `emerge --help config` aufrufen.

Es dauert seine Zeit das ganze Basissystem zu installieren. Ihr Lohn ist es, dass es gänzlich auf Ihren Rechner optimiert ist. Leider müssen Sie einen Weg finden, sich bis zu diesem Zeitpunkt zu beschäftigen. Der Autor empfiehlt "Star Wars - Super Bombad Racing" für die PS2. Wenn `emerge system` abgeschlossen ist, besitzen Sie ein stage3 Gentoo Linux System.

## 12. Letzte Schritte: Zeitzone

### Notiz

Falls Sie es noch nicht getan haben, editieren Sie die Datei `/etc/make.conf` nach Ihren Vorlieben (siehe auch Kapitel [Der Schritt von stage1 zu stage2](#)).

Sie sollten ein stage3 System haben, das zur abschließenden Konfiguration bereit ist. Sie beginnen mit dem Einstellen der Zeitzone. Durch setzen der Zeitzone bevor der Kernel gebaut wird, stellen Sie sicher, dass Sie eine sinnvolle Ausgabe von `uname -a` erhalten.

Sie finden Ihre Zeitzonendatei in `/usr/share/zoneinfo/`. Erstellen Sie nun einen symbolischen Link:

### Befehlsauflistung 19

```
# ln -sf /usr/share/zoneinfo/pfad/zur/zeitzonendatei /etc/localtime
z.B. für Berlin:
# ln -sf /usr/share/zoneinfo/Europe/Berlin /etc/localtime
```

## 13. Letzte Schritte: Kernel, System Logger und Cron

Sie werden nun die ppc-Kernel-Quellen mit `emerge` installieren. Dazu geben Sie `emerge sys-kernel/ppc-sources` ein.

Sobald diese installiert sind, wird es an der Zeit, den eigenen Kernel zu bauen:

### Befehlsauflistung 20

```
# cd /usr/src/linux
# make menuconfig
# make dep && make clean vmlinux modules modules_install
# cp vmlinux System.map /boot
```

### Notiz

Sie sollten **nur** die *sys-kernel/ppc-sources* Gentoo Kernel für PowerPC-basierte Maschinen benutzen.

### Warnung

Für einen fehlerfrei funktionierenden Kernel, stellen Sie sicher, dass manche Optionen **nicht als Modul** kompiliert werden. Dazu aktivieren Sie "**Code maturity level options --> Prompt for development and/or incomplete code/drivers**", damit alle nötigen Optionen verfügbar sind. In der "**File systems**"-Sektion aktivieren Sie "**Device File System**" (beachten Sie, dass Sie die "**/dev/pts file system support**" Option **nicht** aktivieren müssen). Außerdem müssen Sie das "**Virtual Memory File system**" einschalten. Dann noch unter "**Macintosh Device Drivers**" den "**Support for ADB raw keycodes**" abschalten. Ebenso deaktivieren Sie die "**Enhanced RTC**" Option (in "**Character devices**"), da es keine PPC-freundliche ATM ist. Aber schalten Sie den "**Support for /dev/rtc**" unter "**General setup**" ein. Außerdem ist es immer eine gute Idee, ext2 weiterhin aktiviert zu lassen, egal, ob Sie es nutzen oder nicht. Benutzer mit OldWorld-Systemen sollten HFS einschalten, um den kompilierten Kernel auf die MacOS-Partition zu kopieren.

### Wichtig

Wenn Sie einen ATAPI CD-Brenner benutzen, sollten Sie die SCSI-Emulation im Kernel aktivieren. Die neuen cdrtools können zwar direkt auf ATAPI-Geräte ohne die SCSI-Simulation zugreifen, jedoch ist dieses noch als "unstable" eingestuft. Also, zur Aktivierung der SCSI-Emulation schalten Sie "**ATA/IDE/MFM/RLL support**" --> "**IDE, ATA and ATAPI Block devices**" --> "**SCSI emulation support**" (**ruhig als Modul**) ein; dann unter "**SCSI support**" --> "**SCSI support**", "**SCSI CD-ROM support**" und "**SCSI generic support**" (**ebenso als Modul**) einschalten. Falls Sie Module verwenden wollen, müssen Sie noch nach der Kompilation des Kernels `echo -e "ide-scsi\nsg\nsr_mod" >> /etc/modules.autoload` eingeben, um die Module automatisch beim Booten zu laden. Sie sollten außerdem `append="hdc=ide-scsi"` in Ihre `/etc/yaboot.conf` hinzufügen.

### Wichtig

Wenn Sie ein OldWorld-System besitzen, mounten Sie ihre HFS (MacOS) Partition, dann kopieren Sie den Kernel darauf, sodass BootX ihn beim Starten laden kann. BootX sucht automatisch nach Kernel im Verzeichnis `Linux Kernels` im System-Verzeichnis.

Ihr neuer, angepasster Kernel (und die Module) sind nun installiert. Sie müssen nun einen "System Logger" zur Installation wählen. Wir bieten mit **sysklogd** das traditionelle Paket an "system logging daemons". Ebenso gibt es **metallog**, zu dem Power-User von sysklogd aufgrund seiner schlechten Leistung umsteigen. Wenn Sie sich nicht sicher sind, versuchen Sie metalog, da es inzwischen ziemlich populär zu sein scheint. Um einen Logger zu installieren, tippen Sie **eine** der folgenden Möglichkeiten ein:

### Befehlsauflistung 21

```
# emerge app-admin/sysklogd
# rc-update add sysklogd default
oder
# emerge app-admin/metalog
# rc-update add metalog default
```

### Warnung

Falls Sie ein Version 1.2 stage3-Archiv verwenden, brauchen Sie keinen System Logger zu installieren. stage3-ppc-1.2.tbz2 beinhaltet sysklogd. Wenn Sie stattdessen metalog verwenden möchten, müssen Sie sysklogd sauber deinstallieren: Teilen Sie zuerst dem System mit, dass Sie sysklogd nicht weiter beim Booten starten möchten: `rc-update del sysklogd`. Dann deinstallieren Sie sysklogd mit `emerge unmerge sysklogd`.

Möglicherweise müssen Sie noch die sysklogd-Konfigurationsdatei in `/etc` löschen (halten Sie nach `cfgpro` in der Dateiliste beim deinstallieren Ausschau). Nun können Sie metalog mit `emerge metalog` installieren und durch `rc-update add metalog default` automatisch beim Booten starten.

### Wichtig

metalog schreibt seine Daten blockweise auf die Festplatte, weshalb Nachrichten nicht direkt in den Logdateien auftauchen. Falls Sie einen Daemon debuggen möchten, ist dieses Verhalten weniger sinnvoll. Während Ihr Gentoo Linux System läuft, können Sie metalog ein `USR1`-Signal senden, um vorübergehend die Pufferung von Nachrichten abzuschalten (sprich, `tail -f /var/log/everything/current` arbeitet nun wie erwartet in Realzeit); ein `USR2`-Signal schaltet die Pufferung wieder ein.

Sie können jetzt ein beliebiges Cron Paket zur Installation auswählen. Momentan bieten wir `dcron`, `fcron` und `vcron`. Wenn Sie sich nicht entscheiden können, wählen Sie `vcron`. Eines der Pakete installieren Sie wie folgt:

### Befehlsauflistung 22

```
# emerge sys-apps/dcron
# crontab /etc/crontab
oder
# emerge sys-apps/fcron
# crontab /etc/crontab
oder
# emerge sys-apps/vcron
# crontab /etc/crontab
```

Für weitere Informationen, wie Cron unter Gentoo Linux funktioniert, lesen Sie [diese Nachricht](#).

## 14. Letzte Schritte: Zusätzliche Pakete installieren

Wenn Sie Benutzer eines Laptops sind und Ihre PCMCIA-Slots nach dem ersten richtigen Booten benutzen wollen, sollten Sie das Paket *pcmcia-cs* installieren.

### Befehlsauflistung 23

```
# emerge sys-apps/pcmcia-cs
```

## 15. Letzte Schritte: Konfiguration des Systems

### 15.1 /etc/fstab

Ihr Gentoo Linux System ist nun annähernd betriebsbereit. Es müssen noch einige wichtige Systemdateien konfiguriert und der yaboot Bootmanager installiert werden.

Zuerst muss die Datei */etc/fstab* angepasst werden. Tragen Sie die Dateisysteme *ext2* oder *ext3* entsprechend Ihrer Konfiguration ein. Verwenden Sie das unten gelistete Beispiel und ersetzen Sie *ROOT* und *SWAP* durch die entsprechenden Geräte (z.B. *hda2* usw.). Sind Sie Benutzer eines NewWorld-Systems, tragen Sie **nicht** die bootstrap Partition ein!

### Befehlsauflistung 24: /etc/fstab

```
# /etc/fstab: static file system information.
#
# "noatime" schaltet "atimes" (man mount) ab und erhöht so die Performance. ("atimes"
# wird normalerweise nicht benötigt und "notail" erhöht die Leistung eines ReiserFS
# Dateisystems, allerdings auf Kosten der Speicherausnutzung.) Wenn Sie frei zwischen
# "tail" oder "notail" wählen wollen, lassen Sie die "noatime" Option sicherheitshalber
# weg.
# <fs>          <mountpoint>  <type>    <opts>          <dump/pass>

/dev/ROOT      /              ext3      noatime         0 1
/dev/SWAP      none          swap      sw              0 0
/dev/cdroms/cdrom0 /mnt/cdrom    iso9660   noauto,ro      0 0
proc           /proc        proc      defaults        0 0
```

### 15.2 Das Passwort für den Superuser root setzen

Bevor Sie es vergessen, setzen Sie das Passwort für den Superuser root mittels:

### Befehlsauflistung 25

```
# passwd
```

### 15.3 /etc/hostname

Tragen Sie in dieser Datei den kompletten Namen (Rechnername und Domainname) für Ihren Rechner in einer einzigen Zeile ein, z.B. *rechnername.domainname.lan*.

### 15.4 /etc/hosts

Diese Datei enthält eine Auflistung von IP-Adressen und Ihrem zugehörigen Rechnernamen. Sie wird vom System gebraucht, um IP-Adressen von Rechnernamen aufzulösen, die nicht Ihrem verwendeten Nameserver stehen. Hier ist ein Beispiel der Datei:

### **Befehlsauflistung 26:** /etc/hosts

```
127.0.0.1    localhost

# Die folgende Zeile enthält eine IP-Adresse ihres lokalen
# Netzwerks und dem Ihrem Rechner zugewiesenen Rechnernamen

192.168.1.1  rechnername.domainname.lan    rechnername
```

## **15.5 Abschließende Netzwerkkonfiguration**

Fügen Sie die Namen aller Module, die für den Betrieb Ihres System benötigt werden, in die Datei /**etc/modules.autoload** ein (Optionen können Sie in derselben Zeile an den Modulnamen anhängen). Wenn Gentoo Linux bootet, werden diese Module automatisch geladen. Vor allem sollten Sie dort das Modul der Netzwerkkarte eintragen, falls Sie es als Modul kompiliert haben:

### **Befehlsauflistung 27:** /etc/modules.autoload

```
sungem
```

Passen Sie vor dem ersten Booten die Datei /**etc/conf.d/net** Ihrer Netzwerkumgebung an:

### **Befehlsauflistung 28**

```
# nano -w /etc/conf.d/net
# rc-update add net.eth0 default
```

Besitzen Sie mehrere Netzwerkkarten, müssen Sie weitere **net.ethx** Skripte für jede Karte (x = 1, 2, ...) erstellen:

### **Befehlsauflistung 29:** Mehrere Netzwerkkarten

```
# cd /etc/init.d
# cp net.eth0 net.ethx
# rc-update add net.ethx default
```

Haben Sie eine PCMCIA-Karte installiert, prüfen Sie, ob die Einstellungen in der Datei /**etc/init.d/pcmcia** für Ihr System korrekt sind und fügen dann

### **Befehlsauflistung 30**

```
depend() {
    need pcmcia
}
```

am Kopf der Datei /**etc/init.d/net.ethx** ein, sodass PCMCIA-Treiber automatisch geladen werden, wenn die Netzwerkeinstellungen geladen werden.

Wechseln Sie öfter zwischen diversen Netzen (wenn Sie z.B. Ihren Laptop daheim und auf der Arbeit verwenden), wird Ihnen sicherlich das Programm *quickswitch* gefallen. Dort können Sie diverse Umgebungen anlegen und schnell zwischen diesen wechseln, ohne jedes mal die Konfigurationsdateien anzupassen.

## **15.6 Grundlegende Einstellungen (einschließlich des Tastaturlayouts)**

### **Befehlsauflistung 31:** Grundlegende Einstellungen

```
# nano -w /etc/rc.conf
```

Folgen Sie den Anweisungen in der Datei, um die Einstellungen anzupassen. Überprüfen Sie, dass *CLOCK* Ihrer Vorliebe entspricht. Das Tastaturlayout kann über die Variable *KEYMAP* eingestellt werden (in /**usr/share/keymaps/** finden Sie verschiedene Möglichkeiten; auch Mac-Benutzer sollten nun die Konfiguration aus dem Verzeichnis **i386** verwenden). Für eine deutsche Tastaturbelegung können Sie folgendes einstellen:

### **Befehlsauflistung 32:** Deutsche Tastaturbelegung

```
KEYMAP="de-latin1"
```

## 16. Letzte Schritte: Konfiguration des Bootmanagers

Der größte Unterschied von Linux auf x86-Maschinen ist der Bootloader. NewWorld-Maschinen benutzen *yaboot*. OldWorld-Maschinen verwenden BootX.

### 16.2 yaboot für NewWorld-Maschinen (automatische Konfiguration mit yabootconfig)

Sie können das neue und verbesserte Programm *yabootconfig* verwenden, das bei *yaboot-1.3.8-r1* und neueren Versionen beiliegt, um automatisch *yaboot* zu konfigurieren. *yabootconfig* wird die diversen Partitionen Ihres Rechners finden und eine Möglichkeit des Dual- oder Tripel-Boot mit Linux, Mac OS und Mac OS X bereitstellen.

Um *yabootconfig* zu verwenden, muss Ihre Festplatte eine bootstrap Partition besitzen und die Datei `/etc/fstab` entsprechend Ihrer Linux Partitionen konfiguriert sein. Das sollten Sie aber schon in den vorangegangenen Schritten erledigt haben. Bevor Sie starten, installieren Sie die letzte verfügbare Version von *yaboot*, indem Sie *emerge -u yaboot* eingeben. Das ist nötig, da die neueste Version immer in Portage verfügbar ist, aber nicht auf den Installations-Archiven.

An der Eingabeaufforderung geben Sie *yabootconfig* ein. Das Programm startet nun und es verlangt von Ihnen die Bestätigung, ob die gefundene bootstrap Partition die richtige ist. Geben Sie *Y* ein, falls es korrekt ist. Falls nicht, überprüfen Sie Ihre `/etc/fstab`. *yabootconfig* wird dann überprüfen, ob andere Betriebssysteme auf Ihrem Rechner installiert sind, die Datei `/etc/yaboot.conf` erstellen und *mkofboot* ausführen (*mkofboot* wird zur Formatierung der bootstrap Partition und Installation der *yaboot* Konfiguration dorthin verwendet).

Sie sollten die Datei `/etc/yaboot.conf` auf Korrektheit überprüfen. Wenn Sie Änderungen an `/etc/yaboot.conf` vorgenommen haben (wie z.B. Ändern des automatisch zu bootenden Betriebssystems), führen Sie darauf *ybin -v* aus, um Ihre bootstrap Partition zu erneuern.

### 16.3 yaboot für NewWorld-Maschinen (manuelle Konfiguration von /etc/yaboot.conf)

Wenn Sie aus irgendeinem Grund nicht *yabootconfig* die Konfigurationsdatei `/etc/yaboot.conf` erstellen lassen wollen, können Sie die bereits vorhandene Beispieldatei anpassen. Hier eine mögliche `yaboot.conf`:

**Befehlsauflistung 33:** `/etc/yaboot.conf`

```
## /etc/yaboot.conf
##
## Rufen Sie "man yaboot.conf" für Details auf. Machen Sie keine
## Änderungen, außer, Sie benötigen Sie wirklich!!
## Sehen Sie sich auch in /usr/share/doc/yaboot/example mögliche
## Konfigurationen an.
##
## Für ein Dual-Bootmenü, fügen Sie eine oder mehrere Zeilen wie
## bsd=/dev/hdaX, macos=/dev/hdaY, macosx=/dev/hdaZ hinzu.

## Unsere bootstrap Partition:
boot=/dev/hda9

## hd: ist für die OpenFirmware, um hda anzusprechen.
## führen Sie ofpath /dev/hda aus, um die richtigen Einstellung zu finden

device=hd:
## Partition, auf der die Kernel-Images zu finden sind
partition=11

root=/dev/hda11
delay=5
defaultos=macosx
timeout=30
install=/usr/lib/yaboot/yaboot
magicboot=/usr/lib/yaboot/ofboot
```

```

## Dieser Abschnitt kann dupliziert werden, um zwischen mehreren Kernel
## oder unterschiedlichen Boot-Optionen wählen zu können
#####

        image=/boot/vmlinux
        label=Linux
        sysmap=/boot/System.map
        read-only
#####

macos=/dev/hda13
macosx=/dev/hda12
enablecdboot
enableofboot

```

Sobald Sie die Datei `yaboot.conf` Ihren Wünschen entsprechend konfiguriert haben, führen Sie `mkofboot -v` aus, um die Einstellungen in der bootstrap Partition zu sichern. Vergessen Sie es nicht! Wenn alles gut verläuft und Sie dieselben Optionen wie in dem Beispiel oben verwendet haben, sehen Sie nach dem nächsten Reboot ein einfaches Bootmenü mit fünf Einträgen. Sollten Sie später die Datei `/etc/yaboot.conf` ändern, müssen Sie `ybin -v` ausführen, um die bootstrap Partition zu aktualisieren (mkofboot wird nur beim ersten Mal benötigt).

### Notiz

`yaboot` besitzt die Möglichkeit, einige Einstellungen während des Bootvorgangs vorzunehmen. Schauen Sie für weitere Informationen auf <http://penguinppc.org/projects/yaboot/>

## 16.4 BootX für OldWorld-Maschinen

### Wichtig

Führen Sie diese Schritte nach dem Reboot Ihres Rechners im nächsten Schritt aus.

Booten Sie Ihren Rechner in MacOS und öffnen Sie die BootX Erweiterung. Wählen Sie *Options...* und schalten Sie *Use specified RAM Disk* aus. Wenn Sie zu den BootX Haupteinstellungen zurückkehren, werden Sie nun eine Option vorfinden, in der Sie Ihre Festplatte und Partition der Root Partition angeben. Geben Sie dort die entsprechenden Werte ein.

BootX kann so eingestellt werden, dass Linux beim Booten gestartet wird. Dabei wird erst Ihr Rechner in MacOS gestartet, worauf BootX Linux lädt und startet. Schauen Sie auf die [BootX Homepage](#) für weitere Informationen.

## 17. Installation beendet!

Gentoo Linux ist installiert. Der letzte verbleibende Schritt ist die chroot-Umgebung zu verlassen, die Partitionen sicher zu unmounten und den Rechner zu rebooten:

### Befehlsauflistung 34

```

# exit Dieses beendet die chroot Umgebung; Sie können auch ^D eingeben
# cd /
# umount /mnt/gentoo/proc
# umount /mnt/gentoo
# reboot

```

### Notiz

Nach dem Reboot sollten Sie `update-modules` ausführen, um die Datei `/etc/modules.conf` zu erzeugen. Anstatt die Datei direkt editieren, sollten Sie Änderungen in `/etc/modules.d` vornehmen.

## 18. Extras

### 18.1 Emulieren von mehreren Maustasten

Sie können die folgenden Zeilen der Datei `/etc/sysctl.conf` hinzufügen, um mehrere Maustasten bei jedem Booten zu emulieren. Die Werte für die Tasten können Sie mit dem Programm `showkey` ermitteln. Hier ist ein Beispiel, um F11 und F12 zu verwenden:

### Befehlsauflistung 35

```
# Aktiviert die Maustasten-Emulation
dev.mac_hid.mouse_button_emulation = 1
# Setze zweite Maustaste auf 87 - F11
dev.mac_hid.mouse_button2_keycode = 87
# Setze dritte Maustaste auf 88 - F12
dev.mac_hid.mouse_button3_keycode = 88
```

## 18.2 Gentoo-Stats

Das "Gentoo Linux usage statistics project" wurde gestartet, um den Entwicklern einen Überblick der Gentoo-Benutzer zu verschaffen. Es sammelt Informationen über die Verbreitung von Gentoo Linux und hilft, Prioritäten bei den Entwicklern zu setzen. Die Installation ist freiwillig, jedoch wäre es willkommen, wenn Sie sich daran beteiligen würden. Generierte Statistiken können Sie sich unter <http://stats.gentoo.org/> ansehen.

Der gentoo-stats Server wird Ihrem System eine eindeutige ID vergeben. Diese ID wird dafür benötigt, dass jedes System nur einmal gezählt wird. Diese ID wird weder dafür verwendet, um ihr System individuell zu identifizieren, noch wird sie mit der IP-Adresse oder anderen persönlichen Informationen abgeglichen. Alle Vorkehrungen wurden getroffen, um Ihre Privatsphäre in der Entwicklung dieses Systems zu wahren. Nachfolgend sind die Informationen gelistet, die das *gentoo-stats* Programm überwachen wird:

- installierte Pakete und ihre Versionsnummer
- CPU Informationen: Geschwindigkeit (MHz), Hersteller, Modell, CPU Flags (wie "altivec")
- Speicher Informationen (installierter physischer RAM, Größe der angelegten Swap Partition(en))
- PCI Karten und Netzwerkkarten Chipsätze
- das Gentoo Linux Profil, das Ihr Rechner verwendet (dorthin, wo der Link [/etc/make.profile](#) zeigt)

Wir sind uns bewusst, dass eine Bekanntmachung von sensiblen Informationen eine Gefährdung für viele Gentoo Linux Benutzer darstellt (und ebenso für die Entwickler).

- Solange Sie nicht das gentoo-stats Programm ändern, wird niemals eine sensible Information, wie Ihr Passwort, Konfigurationsdateien, der Lieblingsfilm, ... übertragen
- Die Übermittlung Ihrer E-Mail Adresse ist optional und defaultmäßig ausgeschaltet.
- Die IP-Adresse von der Ihre Datenübertragung stammt wird niemals in der Art gespeichert, dass wir Sie identifizieren können. Es gibt keine Zusammenhänge zwischen der IP-Adresse und der System-ID.

Die Installation ist einfach -- führen Sie die folgenden Befehle aus:

**Befehlsauflistung 36:** gentoo-stats installieren

```
# emerge gentoo-stats    Installiert gentoo-stats
# gentoo-stats --new     Bezieht eine neue System-ID
```

Der zweite Befehl wird eine neue System-ID anfragen und sie in [/etc/gentoo-stats/gentoo-stats.conf](#) ablegen. Sie können diese Datei anschauen, um zusätzliche Konfigurationsparameter zu sehen.

Sie sollten das Programm regelmäßig laufen lassen (gentoo-stats benötigt keine Superuser Rechte). Fügen Sie folgende Zeile in Ihre [crontab](#) ein:

**Befehlsauflistung 37:** gentoo-stats mit cron updaten

```
0 0 * * 0,4 /usr/sbin/gentoo-stats --update > /dev/null
```

Das *gentoo-stats* Programm ist ein einfaches Perl-Script, das Sie sich mit Ihrem bevorzugten Textbetrachter oder Editor anschauen können: [/usr/sbin/gentoo-stats](#).

## 18.3 Weiterführende Informationen

Wenn Sie Fragen haben oder bei der Entwicklung von Gentoo Linux mithelfen möchten, tragen Sie sich auf der gentoo-ppc-user und gentoo-ppc-dev Mailing-Liste ein (es gibt Links dazu auf unserer [Mailinglisten-Seite](#) (bisher nur in Englisch)).

Außerdem gibt es eine [Anleitung zur Gentoo Linux Desktop Konfiguration](#), die Ihnen bei folgenden Konfiguration Ihres neuen Gentoo Linux Systems weiterhelfen wird. Weiterhin ist der [Portage Benutzer Leitfaden](#) für eine Einführung in Portage empfehlenswert.

Die [Gentoo Linux/PowerPC FAQ](#) (derzeit nur in Englisch) behandelt oft gestellte Fragen zum Betrieb von Gentoo Linux/PPC. Die allgemeine [Gentoo Linux FAQ](#) wird Ihnen vielleicht auch weiterhelfen, obwohl sie sich hauptsächlich an Benutzer von x86-Architekturen richtet. [Gentoo on PPC](#) (auf [forums.gentoo.org](http://forums.gentoo.org)) ist ein offenes Brett für Diskussionen rund um alle PowerPC Angelegenheiten. Für gewöhnlich gibt es direkte Diskussionen im Chat-Kanal *#gentoo-ppc* auf dem IRC-Server **irc.freenode.net**.

Herzlich Willkommen und viel Spaß mit Gentoo Linux!



## >> Kurzübersicht Installation (x86)

### 1. Kurzübersicht Installation (x86)

Die Installations ISOs finden sie auf einem der [Gentoo Spiegel](#). Detaillierte Beschreibungen der verschiedenen CDs sind im [Gentoo Store](#) verfügbar. CD1 beinhaltet alles, was Sie benötigen um Gentoo Linux schnell ohne Verbindung ins Internet zu installieren. CD2 ist optional und beinhaltet vor-kompilierte Pakete wie KDE, GNOME, OpenOffice, Mozilla, Evolution und andere.

Booten Sie von der ersten Gentoo CD. Drücken Sie <F1> und/oder <F2> um die verschiedene Boot Optionen zu sehen. Drücken Sie am Bootscreen <ENTER> um mit dem Default Kernel fortzufahren. Eventuell werden Sie eine Anzeige erhalten.

#### Befehlsauflistung 1: Erste Einstellungen

```
# date (Stellen Sie sicher, dass ihre Zeit korrekt ist. Falls nicht setzen Sie sie mit date MMD)
# modprobe module_name (Optional - lädt benötigte Module)
# net-setup eth0 (Konfiguration des Netzwerks)
# fdisk /dev/hda (Partitionierung der Festplatte)
```

Die empfohlene Partitionierung beinhaltet eine 64 MB große /boot Partition mit ext3, eine Swap Partition die doppelt so groß ist wie der Arbeitsspeicher, und den Rest als /-Partition mit ReiserFS.

Formatieren Sie Ihre Partitionen mit *mke2fs* (Ext2), *mke2fs -j* (Ext3), *mkreiserfs* (ReiserFS), *mkfs.xfs* (XFS), *mkfs.jfs* (JFS) und *mkswap* (swap partition). Zum Beispiel: *mke2fs -j /dev/hda3*.

Fahren Sie mit dem Mounten der Partitionen und dem Entpacken des gewünschten Stage-Tarballs fort.

#### Befehlsauflistung 2: Vorbereiten der Installation

```
(Aktivieren der Swap Partition) # swapon /dev/hdax
(Mounten der Root-Partition) # mount /dev/hdax /mnt/gentoo
(Erstellen des boot-Mountpunkt) # mkdir /mnt/gentoo/boot
(Mounten der Boot Partition) # mount /dev/hdax /mnt/gentoo/boot
(Wechseln zum Mountpunkt) # cd /mnt/gentoo
(Entpacken eines Stage-Tarball...) # tar -xvjpf /mnt/cdrom/stages/stage?-*tar.bz2
(oder Download des aktuellsten Tarball...) # links http://gentoo.oregonstate.edu/releases/x86
( ... und entpacken) # tar -xvjpf stage*
(Nur bei GRP: Entpacken des Portage Baum) # tar -xvjf /mnt/cdrom/snapshots/portage-*tar.bz2
(Nur bei GRP: Kopieren der Distfiles) # cp -R /mnt/cdrom/distfiles /mnt/gentoo/usr/portage
(Nur bei GRP: Kopieren der Packages) # cp -a /mnt/cdrom/packages /mnt/gentoo/usr/portage
(Auswählen eines Mirror) # mirrorselect -i -o >> /mnt/gentoo/etc/make.conf
(Kopieren der Nameserver Information) # cp /etc/resolv.conf /mnt/gentoo/etc/resolv.conf
(Mounten des Proc Dateisystem) # mount -t proc proc /mnt/gentoo/proc
(Chroot in die neue Umgebung) # chroot /mnt/gentoo /bin/bash
(Laden der notwendigen Variablen) # env-update; source /etc/profile
(Nur bei Netzwerk, GRP: aktualisieren Portage) # emerge sync
```

Jetzt können Sie Gentoo installieren:

#### Befehlsauflistung 3: Gentoo installieren

```
(Passen Sie USE, CFLAGS und CXXFLAGS an; auch CHOST bei Stage1) # nano -w /etc/make.conf
(Nur Stage1: Bootstrap des System) # cd /usr/portage; scripts/bootstrap.sh
(Nur Stage1 und Stage2: Installation des Basissystems) # emerge system
```

Als nächstes müssen Sie wichtige Informationen setzen:

#### Befehlsauflistung 4: Anpassen der Konfigurationsdateien

```
(Setzen der Zeitzone) # ln -sf /usr/share/zoneinfo/path/to/timezonefile /etc/localtime
(Editieren der fstab Datei) # nano -w /etc/fstab
```

Benutzen Sie folgendes als eine **Vorlage** (kopieren sie es nicht wörtlich) für [/etc/fstab](#):

#### Befehlsauflistung 5: /etc/fstab

| # <fs>             | <mountpoint> | <type>   | <opts>                | <dump/pass> |
|--------------------|--------------|----------|-----------------------|-------------|
| /dev/hdax          | /boot        | ext3     | noauto,noatime        | 1 2         |
| /dev/hdax          | /            | reiserfs | noatime               | 0 1         |
| /dev/hdax          | none         | swap     | sw                    | 0 0         |
| /dev/cdroms/cdrom0 | /mnt/cdrom   | iso9660  | noauto,ro,user        | 0 0         |
| none               | /proc        | proc     | defaults              | 0 0         |
| none               | /dev/shm     | tmpfs    | defaults              | 0 0         |
| /dev/hdax          | /mnt/dosc    | ntfs     | ro,noatime,umask=0222 | 0 0         |

## Fortfahren mit der Installation des Linux Kernel:

### Befehlsauflistung 6: Installation des Kernel

```
(Installation der Kernel Sourcen) # emerge -k sys-kernel/gentoo-sources
(Konfiguration des Kernel mit genkernel...) # emerge -k genkernel; genkernel
(oder (1) Manuelles bauen des Kernel) # cd /usr/src/linux; make menuconfig;
( (2) Inkludieren Sie VM fs, /proc fs, /dev fs, /dev fs auto mount at boot)
( (3) Kompilieren des Kernel) # make dep && make clean bzImage modules module
( (4) Den Kernel kopieren) # cp arch/i386/boot/bzImage /boot; cp System.ma
```

## Nun folgt die Installation der von Ihnen benötigten Tools:

### Befehlsauflistung 7: Installation von wichtigen System Tools

```
(System Logger; Auswahl: sysklogd, metalog, msyslog, syslog-ng) # emerge -k app-admin/syslog-ng
(Automatisches starten des Systemlogger beim Boot) # rc-update add syslog-ng default
(Installation eines Cron Daemon; Auswahl: vcron, dcron, fcron) # emerge -k sys-apps/vcron
(Automatisches starten des Cron Daemon automatically beim Boot) # rc-update add vcron default
(Nur genkernel Benutzer: Installation von hotplug) # emerge -k hotplug
(Nur genkernel Benutzer: Automatisches starten von hotplug beim Boot) # rc-update add hotplug default
(Nur nicht-ext2,ext3 Benutzer; Auswahl: reiserfsprogs, xfsprogs, jfsutils) # emerge -k reiserfsprogs
```

## Wenn Sie besondere Kernel Ebuilds benötigen, ist nun eine gute Zeit diese zu installieren:

### Befehlsauflistung 8: Installation besonderer Kernel Ebuilds

```
# emerge -k pcmcia-cs (oder emul0k1, nforce-net, nforce-audio, e100, e1000, ati-drivers, rp-pppc)
# VIDEO_CARDS="yourcard" emerge xfree-drm (für ATI Radeon bis 9200, Rage128, Matrox, Voodoo und ...)
```

## Beenden der Systemeinstellungen Ihres Gentoo System:

### Befehlsauflistung 9: Beenden der Systemeinstellungen Ihres Gentoo System

```
(Setzen des root Passwort) # passwd
(Erstellen eines Benutzers) # useradd benutzername -m -G users,wheel,aud
(Setzen des Passwort für diesen Benutzer) # passwd benutzername
(Setzen des System Hostnamen) # echo meinemaschine > /etc/hostname
(Setzen des System Domainname) # echo meinedomain.com > /etc/dnsdomainname
(Erstellen der Hosts-Datei, setzen Sie wenigstens "127.0.0.1 meinemaschine") # nano -w /etc/hosts
```

### Befehlsauflistung 10: Konfigurieren des Netzwerk

```
(Konfiguration des Netzwerk; dhcp-Benutzer sollten iface_eth0="dhcp" setzen) # nano -w /etc/conf.d/net
(Auflisten der Module die beim Start geladen werden sollen) # nano -w /etc/modules.autoload.d/kernel-2.6
(Konfiguration von grundlegenden Systemeinstellungen (folgen Sie den Kommentaren) # nano -w /etc/conf.d/net
(Nur nicht-PCMCIA: Starten des Netzwerk beim Boot) # rc-update add net.eth0 default
(Nur wenn Sie mehrere Netzwerkkarten haben:)
(1) Erstellen Sie Initscripte für jede Karte) # ln -s /etc/init.d/net.eth0 /etc/init.d/net.ethx
(2) Automatisches Starten von PCMCIA beim Boot) # rc-update add net.ethx default
(Nur PCMCIA: Kontrollieren Sie /etc/conf.d/pcmcia und laden von PCMCIA beim Boot) # rc-update add pcmcia default
```

## Installation eines Bootloader. Links grub, rechts lilo:

### Befehlsauflistung 11: Installation und Konfiguration eines Bootloader

```
# emerge -k grub | # emerge -k lilo
# grub | # nano -w /etc/lilo.conf
grub> root (hd0,0) (Boot Partition) | boot=/dev/hda
grub> setup (hd0) (Installieren in den MBR) | prompt
```

```

grub> quit
# nano -w /boot/grub/grub.conf
default 0
timeout 15
splashimage=(hd0,0)/grub/splash.xpm.gz

title=Gentoo Linux
root (hd0,0)
# Nur für Genkernel Benutzer:
kernel (hd0,0)/kernel-<Kernel Version> root=/dev/hda3
initrd (hd0,0)/initrd-<Kernel Version>
# Nicht-genkernel Benutzer:
kernel (hd0,0)/bzImage

# Für Dual-Boot:
title=Windows XP
root (hd0,5)
chainloader (hd0,5)+1

timeout=50
default=gentoo

# Nur für Genkernel Benutzer:
image=/boot/kernel-<Kernel Version>
root=/dev/hda3
label=gentoo

# Nicht-genkernel Benutzer:
image=/boot/bzImage
root=/dev/hda3
label=gentoo

# Für Dual-Boot:
other=/dev/hdaX
label=windows

# /sbin/lilo

```

Unmounten Sie nun alle Partitionen und rebooten Sie in Ihr neues System:

#### **Befehlsauflistung 12:** Beenden der Installation und Installation von X

```

(Aktualisieren von Konfigurationsdateien)      # etc-update
(Verlassen der Chroot-Umgebung)              # exit; cd /
(Unmounten der Partitionen)                  # umount /mnt/gentoo/boot /mnt/gentoo/proc /mnt/!
(Reboot; nehmen Sie die CD aus dem Laufwerk)  # reboot
(Nach dem Neustart:)
(Nur ADSL-Benutzer)                          # adsl-setup
(Nur GRP-benutzer, Benutzung der optionalen CD2 )
( (1) Mount CD2)                             # mount /dev/cdrom /mnt/cdrom
( (2) Kopieren der Pakete)                   # cp -a /mnt/cdrom/packages/* /usr/portage/packa
( (3) Installation zusätzlicher Software)    # emerge -k xfree gnome kde mozilla openoffice-b
( (4) Konfiguration des Xserver)            # /usr/X11R6/bin/xf86config

```

Sie können weitere Informationen in der [Gentoo Dokumentation](#) finden.



## >> Das Gentoo Linux alternative Installations Methoden HOWTO

[Bitte Kapitel auswählen]

### 1. Über dieses Dokument

Wenn die Standard Boot-von-CD Installation für Sie nicht funktioniert (oder Sie sie einfach nicht mögen), ist nun Hilfe verfügbar. Dieses Dokument soll eine Sammlung verschiedener alternativer Installations Methoden bieten. Sollten Sie eine alternative Installations Methode kennen, von der Sie meinen, dass sie auch für andere hilfreich sein könnte, zögern Sie bitte nicht [mir zu schreiben](#) (Englisch), bzw. sich an das [Team von Gentoo.de](#) zu wenden.

### 2. Booten der LiveCD mit Smart BootManager

Downloaden Sie den Smart BootManager <http://btmgr.sourceforge.net/index.php3?body=download.html>. Linux Sourcen oder Binärpakete sowie Windows .exe Versionen finden Sie dort genauso, wie viele Sprachpakete. Wie auch immer, in diesem Fall sollten Sie das Binärpaket herunterladen, da sich die Sourcen nicht mit neueren Versionen von NASM kompilieren lassen.

Entweder kompilieren Sie das Paket aus den Sourcen oder laden Sie nur das Binärpaket herunter. Es gibt verschiedene Optionen, die Sie während dem erstellen der Boot Diskette benutzen können, wie Sie hier sehen können.

#### Befehlsauflistung 1: Smart BootManager Optionen

```
sbminst [-t theme] [-d drv] [-b backup_file] [-u backup_file]

-t theme      select the theme to be used, in which the theme could be:
              us = English theme          de = German theme
              hu = Hungarian theme       zh = Chinese theme
              ru = Russian theme         cz = Czech theme
              es = Spanish theme         fr = French theme
              pt = Portuguese theme

-d drv        set the drive that you want to install Smart BootManager on;
              for Linux:
                /dev/fd0 is the first floppy driver,
                /dev/hda is the first IDE harddisk driver.
                /dev/sda is the first SCSI harddisk driver.
              for DOS:
                0   is the first floppy drive
                128 is the first hard drive;

-c            disable CD-ROM booting feature;

-b backup_file backup the data that will be overwritten for
              future uninstallation;

-u backup_file uninstall Smart BootManager, should be used alone;

-y            do not ask any question or warning.
```

#### Befehlsauflistung 2: Benutzen von sbminst um die Boot Diskette zu erstellen

```
# sbminst -t us -d /dev/fd0
```

#### Notiz

Ersetzen Sie fd0 mit dem Namen ihres Diskettenlaufwerks, sollte es anders heissen.

Schieben sie die Diskette nun in das Diskettenlaufwerk des Computer auf dem Sie die LiveCD starten möchten. Vergessen Sie nicht die LiveCD einzulegen und starten Sie die Maschine.

Sie werden vom Smart BootManager mit einem Dialog begrüßt. Wählen Sie Ihr CD-ROM Laufwerk aus und drücken Sie ENTER zum Starten der LiveCD. Nachdem Sie gebootet haben, können Sie mit der Standardinstallationsanleitung fortfahren.

Weitere Informationen zum Smart BootManager können Sie hier finden: <http://btmgr.sourceforge.net/>.

### 3. Knoppix Installation

Booten von der [Knoppix](#) LiveCD ist ein Weg ein voll funktionales System während des Kompilierens von Gentoo zur Verfügung zu haben. Tux Racer wird Ihnen helfen, die Zeit die der Bootstrapper benötigt totzuschlagen :)

Booten Sie die Knoppix CD. Generell funktioniert die Hardware Erkennung prächtig. Dennoch können Sie einige Boot Optionen anpassen.

Standardmäßig startet Knoppix einen KDE 3.1 Desktop. Was Sie sicherlich als erstes öffnen möchten, ist eine Konsole. Geben Sie hier `sudo passwd root` ein. Damit setzen Sie als erstes ein Root Passwort.

Als nächstes werden Sie mit `su` zu `root` und führen `usermod -d /root -m root` aus. Damit setzen Sie das Home-Verzeichnis auf `/root` (wie bei Gentoo) anstelle von `/home/root` (wie bei Knoppix). Wenn Sie dies nicht tun werden Sie beim Emergenz Fehler wie zum Beispiel `"/home/root: not found"` erhalten.

Führen Sie `exit` aus und werden direkt anschließend wieder mittels `su` `root`. Damit können wir sicherstellen, dass die Änderungen des `usermod` Kommandos eingelesen werden. An dieser Stelle können Sie mit Kapitel 6 der Standard Installationsanleitung fortfahren. Wenn Sie am Mounten des `proc`-Dateisystems angekommen sind, führen Sie anstelle des Kommandos in der Anleitung folgende Befehle aus:

**Befehlsauflistung 3:** Mounten benötigter Dateisysteme

```
# mount -o bind /proc /mnt/gentoo/proc
# mount -o bind /dev /mnt/gentoo/dev
```

### 4. Installation von Stage 1 ohne Netzwerkzugriff

Brennen eines LiveCD Images.

Holen Sie sich den aktuellsten Portage Snapshot von <http://distro.ibiblio.org/pub/linux/distributions/gentoo/snapshots/> (oder Ihrem beliebtesten [Gentoo Spiegel](#)). Legen Sie den Tarball entweder auf eine existierende Partition auf Ihrem Computer, oder brennen Sie ihn auf eine CD.

Sie benötigen ausserdem eine Paket Liste für die Stage Pakete. Speichern Sie das folgende Script auf dem gleichen Medium wie den Portage Snapshot, Sie werden es später benötigen.

**Befehlsauflistung 4:** `dl-list.sh`

```
#!/bin/bash

# set your defaults here:
user_defs() {

    # portage directory (without a trailing "/"):
    portage_dir="/usr/portage"

    # default download mirror (without a trailing "/"):
    gentoo_mirror="http://www.ibiblio.org/pub/Linux/distributions/gentoo"

    # default sourceforge mirror (unc, telia, belnet):
    sourceforge_mirror="unc"

    # fix gnu mirror entries
    gnu_url="ftp://ftp.gnu.org/pub/gnu"

}

#-----
```

```

# function to remove temporary files
cleanup() {

    rm -f $temp_file_1 $temp_file_2
    exit $1

}

# set user defaults
user_defs

# set the complete url for the sourceforge mirror
# (the \'s are needed because this goes in a sed command)
sourceforge_mirror_complete="http:\\\\$sourceforge_mirror.dl.sourceforge.net\\sourceforge"

# initialize counters
num_files=0
num_alt_urls=0
total_size=0

# initialize lists (arrays)
declare -a def_urls_arr
declare -a alt_urls_arr

# create 2 temporary files
temp_file_1=`mktemp -t dl-list.XXXXXX` || cleanup 1
temp_file_2=`mktemp -t dl-list.XXXXXX` || cleanup 1

# run "emerge -p <args>" (too easy to forget the "-p" in the command line...)
emerge -p $@ > $temp_file_1 || cleanup 1

# remove the lines that do not contain the word "ebuild"
sed -n -e '/ebuild/p' $temp_file_1 > $temp_file_2

# count how many lines were left
num_ebuilds=`wc -l $temp_file_2 | sed -e 's/\\(.*) \\(.*)/\\1/'`

# extract the useful information from those lines: category, package and version
#sed -e 's:\\(.*) \\(.*)/\\(.*)-\\([0-9].*) \\(.*) \\(.*):\\2 \\3 \\4:' $temp_file_2 > $temp_file_1
sed -e 's:\\(.*) \\(.*)/\\(.*)-\\([0-9].*):\\2 \\3 \\4:' $temp_file_2 > $temp_file_1

# display starting message :)
echo -n "Generating list " >&2

# process each package in turn
while read category package version rest
do

    # form the name of the digest file
digest_file="$portage_dir/$category/$package/files/digest-$package-$version"

    # process the contents of the digest file
    while read md5_flag md5_sum file_name file_size
    do

        # form the default url to download the file
        def_urls_arr[$num_files]="$gentoo_mirror/distfiles/$file_name"

        # increment the file counter
        num_files=$(( $num_files + 1 ))

        # update the size accumulator (in kilobytes)
        total_size=$(( $total_size + $file_size / 1024 ))

done < $digest_file

# form the "ebuild depend" command line
ebuild_depend_cmd="ebuild $portage_dir/$category/$package/$package-${version}.ebuild depend"

# execute the "ebuild depend" command
$ebuild_depend_cmd || cleanup 1

```

```

# form the name of the dependency file
dependency_file="/var/cache/edb/dep/$category/$package-$version"

# read in the 4th line from the dependency file,
# which contains the official download urls
alt_urls=`head -n 4 $dependency_file | tail -n 1`

# ignore empty url list
if [ -n "$alt_urls" ]
then

    # split the urls list into $1..$N
    set $alt_urls

    # process each url in turn
    for i in $@
    do

        # remove the (use)? strings from the url list
        alt_url_tmp=`echo "$i" | sed -e '/\?$/d`

        # remove the "mirror://gnome" urls
        alt_url_tmp=`echo "$alt_url_tmp" | sed -e '/^mirror:\\/\\/gnome/d`

        # remove the "mirror://kde" urls
        alt_url_tmp=`echo "$alt_url_tmp" | sed -e '/^mirror:\\/\\/kde/d`

        # remove the "mirror://gentoo" urls (already included)
        alt_url_tmp=`echo "$alt_url_tmp" | sed -e '/^mirror:\\/\\/gentoo/d`

        # translate the "mirror://sourceforge" urls into valid urls
        alt_url_tmp=`echo "$alt_url_tmp" | sed -e "s/mirror:\\/\\/sourceforge/$sourceforge_mirror_co

        # translate the "mirror://gnu" urls into valid urls
        alt_url_tmp=`echo "$alt_url_tmp" | sed -e "s/mirror:\\/\\/gnu/$gnu_url/"`
        # ignore empty urls
        if [ -n "$alt_url_tmp" ]
        then

            # add the url to the list
            alt_urls_arr[$num_alt_urls]=$alt_url_tmp

            # increment the alternate url counter
            num_alt_urls=$((num_alt_urls + 1))

        fi

    done

fi

# a progress bar :)
echo -n "." >&2

done < $temp_file_1

# display ending message :)
echo " done." >&2

# display default urls list
for i in ${def_urls_arr[@]}; do echo $i; done | sort

# display alternate urls list
for i in ${alt_urls_arr[@]}; do echo $i; done | sort

# display totals
echo "Totals:" $num_ebuilds "ebuilds," $num_files "files," $num_files "default urls," \n
$num_alt_urls "alternate urls," "${total_size}Kb." <&2

# remove temporary files and exit
cleanup 0

```

Ältere Versionen der LiveCD erwarten, dass Sie das root Passwort mit *passwd* ändern, bevor Sie sich manuell einloggen können.

Gehen Sie nun auf die erste Konsole zurück (Alt-F1, ohne chroot) und mounten die zweite CD nach `/mnt/gentoo/mnt/cdrom2`. Kopieren Sie den Stage Tarball von der CD und entpacken ihn nach `/mnt/gentoo/usr/portage`. Ausserdem kopieren Sie das `dl-list.sh` Script nach `/usr/sbin` und machen es ausführbar.

**Befehlsauflistung 5:** Mounten der Snapshot CD

```
# umount /mnt/cdrom
# mkdir /mnt/gentoo/mnt/cdrom2
# mount /dev/cdroms/cdrom0 /mnt/gentoo/mnt/cdrom2
# cp /mnt/gentoo/mnt/cdrom2/portage-$(date).tar.bz2 /mnt/gentoo/usr/portage
# cp /mnt/gentoo/mnt/cdrom2/dl-list.sh /mnt/gentoo/usr/sbin
# cd /mnt/gentoo/usr/portage
# tar xvjpf portage-$(date).tar.bz2
# chmod +x /mnt/gentoo/usr/sbin/dl-list.sh
```

Wechseln Sie nun wieder auf die F2 Konsole. Wenn Sie nun versuchen `bootstrap.sh` auszuführen wird es abbrechen, da es keine Sourcen herunterladen kann. Wir werden diese Dateien woanders laden und nach `/usr/portage/distfiles` (auf der F2 Konsole) schieben.

Sie benötigen eine Liste der Stage 1 Pakete: `glibc`, `baselayout`, `texinfo`, `gettext`, `zlib`, `binutils`, `gcc`, `ncurses` und deren Abhängigkeiten.

### Notiz

Beachten sie, dass Sie die Version jedes Pakets brauchen, die in Ihrem Portage Snapshot aktuell ist.

Benutzen Sie nun das `dl-list.sh` Script um die Paketliste zu generieren. Kopieren Sie die Liste nun auf eine Diskette.

**Befehlsauflistung 6:** `dl-list.sh` benutzen

```
# dl-list.sh glibc baselayout texinfo gettext zlib binutils gcc ncurses > stage1.list
# mount -t vfat /dev/fd0 /mnt/floppy
# cp /mnt/gentoo/stage1.list /mnt/floppy
# umount /mnt/floppy
```

Nehmen Sie die Diskette mit zu einem Rechner, der über einen Internetzugang verfügt und laden Sie die Liste mit `wget` herunter:

**Befehlsauflistung 7:** Laden der Sourcen mit `wget`

```
# wget -N -i stage1.list
```

Nachdem Sie alle benötigten Sourcen heruntergeladen haben legen Sie sie auf dem Zielsystem unter `/mnt/gentoo/usr/portage/distfiles` ab. Nun können Sie `bootstrap.sh` erfolgreich ausführen. Nutzen Sie das `dl-list.sh` Script auch, um die während Stage 2 / Stage 3 benötigten Sourcen zu laden.

## 5. Installation ohne Disks mit PXE Boot

### 5.1 Anforderungen

Sie benötigen am Client eine Netzwerkkarte die das PXE Protokoll zum booten benutzt, wie zum Beispiel viele 3COM Karten. Ausserdem benötigen sie ein BIOS, dass das booten von PXE unterstützt.

### 5.2 Basis Setup des Server

Erstellen Sie die Verzeichnisse, in dem Sie das Plattenlose System ablegen wollen. Erstellen Sie ein Verzeichnis namens `/diskless`, das ein Verzeichnis pro Client beinhaltet. Für den Rest dieses HowTo arbeiten wir mit dem Client "eta".

**Befehlsauflistung 8:** Verzeichnisse anlegen

```
# mkdir /diskless
# mkdir /diskless/eta
# mkdir /diskless/eta/boot
```

DHCP und TFTP Setup: Der Client bekommt die Boot Informationen per DHCP und lädt alle benötigten Dateien mit TFTP. Emergen Sie einfach DHCP und passen Sie es an Ihre Begebenheiten an. Fügen Sie nun folgendes in die `/etc/dhcp/dhcpd.conf` ein.

### Notiz

Dies gibt dem Client eine statische IP Adresse und den Pfad eines PXE Bootimage, hier pxegrub. Sie müssen die MAC Adresse der Netzwerk Karte des Client und das Verzeichnis, in dem die Client Dateien liegen, anpassen.

**Befehlsauflistung 9:** dhcp.conf

```
option option-150 code 150 = text ;
host eta {
hardware ethernet 00:00:00:00:00:00;
fixed-address ip.add.re.ss;
option option-150 "/eta/boot/grub.lst";
filename "/eta/boot/pxegrub";
}
```

Für TFTP, emerge `app-admin/tftp-hpa`. In `/etc/conf.d/in.tftpd`, tragen Sie folgendes ein:

**Befehlsauflistung 10:** in.tftpd

```
INTFTPD_PATH="/diskless"
INTFTPD_USER="nobody"
INTFTPD_OPTS="-u ${INTFTPD_USER} -l -vvvvvv -p -c -s ${INTFTPD_PATH}"
```

Setup GRUB: Zum PXE Booten benutze ich GRUB. Sie müssen ihn selber kompilieren, um die Unterstützung für PXE Image zu aktivieren. Dies ist aber ziemlich einfach. Zuerst laden Sie die aktuellste Version der GRUB Sourcen herunter. (`emerge -f grub` plaziert den Tarball in `/usr/portage/distfiles`). Kopieren Sie den Tarball nach `/diskless` und kompilieren Sie sich ein PXE fähiges Binary. Nachdem das Binary gebaut wurde, kopieren Sie es in das Boot Verzeichnis des Client. Passen Sie nun die entsprechende `grub.lst` Konfiguration an.

**Befehlsauflistung 11:** grub setup

```
# tar zxvf grub-0.92.tar.gz
# cd grub-0.92
# ./configure --help
// In den Optionen sehen Sie eine Liste Unterstützter Netzwerk Karten.
// Wählen Sie den zu Ihrer Karte passenden Treiber aus, er ist hier als $nic aufgeführt
# ./configure --enable-diskless --enable-$nic
# make
# cd stage2
# cp pxegrub /diskless/eta/boot/pxegrub
# nano -w /diskless/eta/boot/grub.lst
```

**Befehlsauflistung 12:** grub.lst

```
default 0
timeout 30

title=Diskless Gentoo
root (nd)
kernel /eta/bzImage ip=dhcp root=/dev/nfs nfsroot=ip.add.re.ss:/diskless/eta

// Für die nfsroot Option geben Sie die IP Adresse des Servers an
// das Verzeichnis wo die Daten des Plattenlosen Client liegen (auf dem Server).
```

Setup NFS: NFS ist ziemlich einfach zu konfigurieren. Sie müssen nur die Konfiguration in `/etc/exports` anpassen:

**Befehlsauflistung 13:** /etc/exports

```
# nano -w /etc/exports
NFS file systems being exported. See exports(5).
/diskless/eta eta(rw, sync, no_root_squash)
```

Aktualisieren Sie Ihre Host Datei: Ein wichtiger Schritt ist nun, ihre `/etc/hosts` Datei an Ihre Bedürfnisse anzupassen.

**Befehlsauflistung 14:** `/etc/hosts`

```
127.0.0.1 localhost

192.168.1.10 eta.example.com eta
192.168.1.20 sigma.example.com sigma
```

## 5.3 Erstellen des Systems auf dem Server

Rebooten Sie den Server mit einer Gentoo LiveCD. Folgen Sie der Standardinstallation ABER mit den folgenden unterschieden: Wenn Sie das Dateisystem mounten, führen Sie folgendes aus (wobei `hdaX` die Partition ist, auf der Sie das `/diskless` Verzeichnis erstellen). Sie brauchen keine weiteren Dateisysteme zu mounten, da alle Dateien in das `/diskless/eta` Verzeichnis kommen.

**Befehlsauflistung 15:** Mounten der Dateisysteme

```
# mount /dev/hda3 /mnt/gentoo
```

Stage Tarball und chroot: Dieses Beispiel benutzt einen Stage 3 Tarball. Mounten Sie `/proc` in Ihr `diskless` Verzeichniss und chrooten Sie in dieses, um mit der Installation fortzufahren. Folgen Sie den Anweisungen in der Installations Anleitung bis zur Kernel Konfiguration.

### Warnung

Seien Sie mit dem entpacken des Stage Tarball besonders vorsichtig. Sie wollen Ihn nicht über Ihre existierende Installation kopieren.

**Befehlsauflistung 16:** Entpacken des Stage Tarball

```
# cd /mnt/gentoo/diskless/eta/
# tar -xvjpf /mnt/cdrom/gentoo/stage3-*.tar.bz2
# mount -t proc /proc /mnt/gentoo/diskless/eta/proc
# cp /etc/resolv.conf /mnt/gentoo/diskless/eta/etc/resolv.conf
# chroot /mnt/gentoo/diskless/eta/ /bin/bash
# env-update
# source /etc/profile
```

Kernel Konfiguration: Wenn Sie `make menuconfig` zur Kernel Konfiguration ausführen, vergessen Sie nicht die folgenden Optionen zusätzlich zu den anderen empfohlenen zu aktivieren:

**Befehlsauflistung 17:** `menuconfig` Optionen

```
- Your network card device support

- Under "Networking options" :

[*] TCP/IP networking
[*] IP: kernel level autoconfiguration
[*] IP: DHCP support
[*] IP: BOOTP support

- Under "File systems --> Network File Systems" :

<*> NFS file system support
[*] Provide NFSv3 client support
[*] Root file system on NFS
```

Als nächstes konfigurieren Sie die `/etc/fstab` des Client.

**Befehlsauflistung 18:** `/etc/fstab`

```
# nano -w /etc/fstab
/dev/cdroms/cdrom0 /mnt/cdrom iso9660 noauto,ro 0 0
proc /proc proc defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
```

Bootloader: Installieren Sie keinen weiteren Bootloader, wir haben bereits einen: pxegrub. Beenden sie die Installation und starten Sie den Server neu. Starten Sie die Dienste, die Sie zum booten des neuen Client benötigen: DHCP, TFTP, und NFS.

**Befehlsauflistung 19:** Starten benötigter Dienste

```
# /etc/init.d/dhcp start
# /etc/init.d/tftpd start
# /etc/init.d/nfs start
```

## 5.4 Booten des neuen Client

Um den neuen Client problemlos zu booten, müssen Sie das BIOS und die Netzwerk Karte so konfigurieren, dass PXE als erste Boot Methode genutzt wird und nicht das booten von CD-ROM oder einer Diskette. Sollten Sie dazu Hilfe benötigen, schauen Sie in die Bedienungsanleitung oder auf die Webseite des Herstellers. Die Netzwerk Karte sollte mittels DHCP eine IP Adresse bekommen und das GRUB PXE Image per TFTP laden. Dann sollten Sie einen nettes schwarz/weisses GRUB Menü sehen, wo Sie den zu bootenden Kernel auswählen und ENTER drücken. Wenn alles in Ordnung ist sollte der Kernel booten, das root-Dateisystem per NFS gemountet und ein LogIn-Prompt bereitstellt werden.

## 6. Gentoo Installation aus einer existierenden Linux Distribution heraus

### 6.1 Anforderungen

Um Gentoo aus Ihrer existierenden Linux Distribution zu installieren benötigen Sie das chroot Kommando sowie eine Kopie eines beliebigen Stage Tarballs oder ein CD Image, dass Sie installieren möchten. Eine Verbindung ins Internet ist empfehlenswert, wenn Sie mehr installieren möchten, als in Ihrem Tarball enthalten ist. (BTW: Ein Tarball ist eine Datei, die auf .tbz, .tar.gz, .tar.bz2 oder .tbz2 endet). Der Autor benutzt RedHat Linux 7.3 als das "Host" System, aber das ist eher nicht wichtig. Also fangen wir an!

### 6.2 Überblick

Zunächst werden wir ein bisschen Platz für Gentoo schaffen, dann die neu gewonne Partition mounten, den Tarball entpacken, in das "Pseudo-System" chrooten und das Bauen von Gentoo starten. Nach dem Bootstrap Prozess werden wir einige Konfigurationen machem, um sicher zustellen, dass es bootet. Dann sind wir fertig zum Reboot und können Gentoo nutzen.

### 6.3 Wie kann ich Platz für Gentoo schaffen?

Die Root Partition ist das Dateisystem, dass als / gemounted wird. Ein kurzer Blick auf *mount* zeigt, wovon ich rede. Wir werden ausserdem *df* (disk free) benutzen, um zu sehen wieviel Platz uns bleibt. Beachten Sie, dass es nicht zwingend notwendig ist, die Größe der root-Partition zu ändern. Sie können auch die Größe anderer Partitionen ändern, darüber reden wir später.

**Befehlsauflistung 20:** Informationen zu bestehenden Dateisystemen

```
# mount
/dev/hdb2 on / type ext3 (rw)
none on /proc type proc (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
none on /dev/shm type tmpfs (rw)
# df -h
Filesystem      Size Used Avail Use% Mounted on
/dev/hdb2       4.0G 1.9G  2.4G  82% /
none            38M   0   38M   0% /dev/shm
```

Wie wir sehen hat die Partition */dev/hdb2*, die als / gemounted ist, noch 2.4 GB frei. In meinem Fall, werde ich die Partition verkleinern, so dass nachher noch 400 MB freier Platz und 2 GB für Gentoo

verfügbar sind. Nicht schlecht, da kann man schon ein bisschen was installieren. Für die meisten Benutzer sollte 1 GB ausreichen.

## 6.4 Installation von parted zum verändern der Partition(en)

Parted ist ein Werkzeug, dass von der GNU Stiftung bereitgestellt wird, ein altes und respektables großes Projekt dessen Software Sie in diesem Moment nutzen. Es gibt ein Tool, das uns in diesem Moment sehr hilfreich sein wird. Es heisst parted, 'Partition Editor'. Sie können es von <http://www.gnu.org/software/parted/> herunterladen.

### Notiz

Es gibt andere Tools, die genau dasselbe tun können, der Autor ist aber unsicher/uninteressiert ob auch PartitionMagic(tm) oder andere Software dieser Art für diesen Zweck geeignet ist. Es ist im Zweifelsfall Ihre Aufgabe dies heraus zufinden.

Schauen Sie auf der Webseite von parted, ob das Dateisystem, dessen Größe Sie ändern möchten, von parted unterstützt wird oder nicht. Wenn nicht haben Sie Pech und müssen Ihre Partitionen platt machen und ihr System neu installieren. Wenn Ihr Dateisystem von parted unterstützt wird, fahren Sie mit dem Download von parted fort. Nun haben wir ein Problem, wir wollen die Größe unserer root-Partition ändern, also müssen wir ein Mini-Linux von Diskette oder CD-ROM starten und parted auf eine Diskette kopieren.

### Notiz

Stellen Sie sicher, dass das was mit Ihrer Partition anstellen wollen auch von parted unterstützt wird.

Laden sie die tomsrftbt boot/root Diskette von <http://freshmeat.net/tomsrftbt> herunter, erstellen Sie nach der Anleitung eine Diskette und legen Sie eine neue Diskette für den nächsten Schritt ein.

### Notiz

Beachten Sie, dass Linux ein Synonym für "Es gibt mehrere Wege dies zu tun" ist. Sie könnten parted auch auf einer nicht gemounteten Partition ausführen oder eine anderes Mini-Linux auf die Diskette werfen. Vielleicht haben Sie ja auch noch Platz auf ihrer Festplatte frei, den Sie für Gentoo verwenden können.

**Befehlsauflistung 21:** Erstellen der Werkzeug Diskette

```
# mkfs.minix /dev/fd0
480 inodes
1440 blocks
Firstdatazone=19 (19)
Zonesize=1024
Maxsize=268966912
```

Wir werden nun parted bauen. Wenn Sie die Sourcen noch heruntergeladen und entpackt haben ist nun ein guter Zeitpunkt dafür. Wechseln Sie in das parted-Sourcen Verzeichnis und führen Sie das folgende Kommando aus, um parted zu kompilieren und auf die Diskette zu kopieren.

**Befehlsauflistung 22:** Bauen der Werkzeug Diskette

```
# mkdir /floppy; mount -t minix /dev/fd0 /floppy &&
export CFLAGS="-O3 -pipe -fomit-frame-pointer -static" && ./configure
&& make && cp parted/parted /floppy && umount /floppy
```

Gratulation, Sie sind nun soweit und können neustarten und die Größe der Partition ändern, nachdem Sie einen kurzen Blick auf die parted Dokumentation geworfen haben. Das resizen sollte auch auf größeren Festplatten nicht länger als 30 Minuten dauern, seien Sie vorsichtig. Mounten Sie `/dev/fd0` nach `/floppy`. Führen Sie nun parted aus, daran anschliessend kommen wir zu den Sachen die wirklich Spaß machen, die Installation von Gentoo. Rebooten Sie nun in Ihr altes Linux System zurück.

**Befehlsauflistung 23:** Kommandos die Sie ausführen müsseo nachdem Sie das Mini-Linux gestartet haben

```
# mount /dev/fd0 /floppy
# cd /floppy; ./parted [drive you wish to operate on]
(parted) print
Disk geometry for /dev/hdb: 0.000-9787.148 megabytes
Disk label type: msdos
Minor  Start      End        Type        Filesystem  Flags
1      0.031      2953.125  primary    ntfs
3      2953.125   3133.265  primary    linux-swap
```

```
2      3133.266  5633.085  primary  ext3
4      5633.086  9787.148  extended
5      5633.117  6633.210  logical
6      6633.242  9787.148  logical  ext3
```

(parted) **help resize**

```
resize MINOR START END          resize filesystem on partition MINOR
```

MINOR is the partition number used by Linux. On msdos disk labels, the primary partitions number from 1-4, and logical partitions are 5 onwards.

START and END are in megabytes

(parted) **resize 2 3133.266 4000.000**

### **Wichtig**

Seien Sie vorsichtig! Ihr Computer arbeitet. Werfen Sie einen Blick auf die Hardware LEDs, sie werden sehen dass ihr Computer wirklich arbeitet. Das resizen sollte zwischen 2 und 30 Minuten dauern.

Nachdem sie Platz für Gentoo geschaffen haben rebooten Sie in Ihr altes Linux System. Führen Sie nun die Schritte 6 bis 17 der Installationsanleitung durch.

Viel Spaß!



## >> Gentoo 1.4 Upgrade Anleitung

[Bitte Kapitel auswählen] 

### 1. Bevor sie anfangen

#### 1.1 Ein Hinweis für Ungeduldige

**WENN SIE NICHT SCHITT-FÜR-SCHRITT VORGEHEN WOLLEN** sollten sie nur das "Vorgehen in Kürze" in Abschnitt 2 lesen.

#### 1.2 Warnung an die Zartbesaiteden

##### Warnung

Die geschilderte Prozedur wurde zwar ausgiebig getestet, trotzdem kann es ihr System unbenutzbar machen!

Mit dieser Warnung im Kopf sollten sie wissen, dass die ganze Prozedur gefährlich und zeitaufwändig ist. Wenn ihr Computer bereits mit der Installation von **stage1** zu kämpfen hatte, sollten sie lieber kein Upgrade durchführen. Bitte lesen sie das gesamte Dokument, ehe sie anfangen und überlegen sie es sich zweimal, bevor sie das Upgrade starten. Wenn sie nicht viel Linux-Erfahrung haben, sollten sie von dem Ganze lieber die Finger lassen.

Einige der Texte in dieser Datei wurden aus der Datei README aus den Upgrade-Skripten von [Nicholas Jones](#) entnommen.

#### 1.3 Benötigte Dateien und allgemeine Informationen

Sie benötigen die vier Skripte von Nicholas Jones. Diese erhalten sie [hier](#). Nehmen sie die Dateien mit den Namen **update-step1.sh**, **update-step2.sh**, **update-step3.sh** and **update-step4.sh**. Vergessen sie nicht die Zugriffsrechte der Dateien zu ändern und sie ausführbar zu machen.

##### Warnung

**Starten sie die Skripte NIE alle mit einem Kommando!** z.B.: **update-step1.sh && update-step2.sh && ...**

**Skript 1** führt einige Tests durch, um zu sehen, ob alles in Ordnung ist. Außerdem wird es Sicherheitskopien von GCC, glibc, binutils, ncurses, gettext und texinfo erstellen. Sie werden diese benötigen, wenn etwas wirklich schief geht. Dies wäre sehr unangenehm (wir haben es bisher noch nicht erlebt, trotzdem kann es passieren), deshalb ist es besser darauf vorbereitet zu sein.

**Skript 2** startet den ersten Teil des Upgrades, es verändert die Profil-Systemlinks, spielt Portage ein und wird ein build+bootstap des **GCC** und der **binutils** durchführen.

##### Notiz

Um dieses Skript auch auf nicht-x86-Rechnern zum Laufen zu bringen, müssen sie die Variable ARCH am Skriptanfang anpassen. Sie können **x86**, **ppc**, **sparc** und **sparc64** verwenden.

**Skript 3** setzt das Upgrade fort und wird ein komplettes stage3-Bootstrap abhängig von ihrer `/etc/make.conf` durchführen, dies betrifft GCC, glibc, binutils .....

**Skript 4** ist optional und würde alle beschädigten Pakete die eventuell nach Skript 3 existieren unmerge und wieder re-emerge (löschen und neu einspielen). Sie haben ebenfalls die Möglichkeit, anstatt Skript 4 ein `emerge -e world` durchzuführen, um ihr komplettes System mit dem gcc 3.2 und den neuen CFLAGS neu zu bauen - allerdings wird dies ungleich länger dauern.

#### 1.4 Ratschläge der Autoren

**Übertakten sie ihr System während des Upgrades nicht!** Gustavo Felisberto (einer der Autoren) hatte eine sehr schmerzliche Erfahrung im Verlaufe seines Upgrades. Sein Dual-P3-866mhz-System war auf 1007MHz übertaktet - offenbar war das dem Hauptspeicher nicht recht und er erzeugte wiederholt Speicherzugriffsfehler.

Melden sie sich nachdem sie das zweite Skript gestartet haben NICHT ab, bis Skript 3 beendet wurde! Das Upgrade verändert **init** und **login** und wird die Konsolen auf Grund fehlender Bibliotheken sperren. Sie würden eine Auflistung von massenhaft Fehlermeldungen erhalten. Konsolenunabhängige Dienste wie SSH werden auch weiterhin funktionieren, vorausgesetzt es sind keine C++ Programme.

Verändern sie NIE ihre CFLAGS, bevor Skript 2 nicht beendet wurde. Durch das Hinzufügen von gcc3-Flags zur make.conf vor Skript 2 würde einige Teile des Skripts fehlschlagen.

## 2. Das Upgrade

### 2.1 Vorgehen in Kürze

#### Befehlsauflistung 1

```
# emerge rsync // Ein existierender Baum ist wichtig
# emerge -u world // Spart ihnen eine Menge Zeit und Nerven

# // Setzen sie ihre CFLAGS für bootstrap
# nano -w /etc/make.conf // Verwenden sie KEINE gcc 3.2-Spezifischen Flags.

# ./update-step1.sh // Kurzes Skript, das Sicherheitskopien von gcc,glibc,usw. anlegt
```

#### Warnung

**STARTEN SIE DAS SYSTEM NICHT NEU**, nachdem sie das Skript 2 begonnen haben!  
Ihr System wird wahrscheinlich nicht starten, bevor Skript 3 nicht vollständig durchgeführt wurde.

#### Notiz

An dieser Stelle sollten sie sich an sovielen Konsolen anmelden, die sie brauchen bis Skript 3 abgelesen ist.

```
# nano -w update-step2.sh // Überprüft die Variable ARCH. (x86,ppc,sparc,sparc64)
# ./update-step2.sh // Ändert das Profil, build-bootstrap gcc und glibc.
```

#### Warnung

Fehlermeldungen und Fehlschläge vor diesem Punkt sind meist auf die fehlerhafte CFLAGS zurück zu führen.

```
# nano -w /etc/make.conf // Verändern sie hier ihre CFLAGS.
# ./update-step3.sh // bootstraps gcc und erneuert Login-kritische Pakete.
# etc-update // Überprüfung ALLER Änderungen.
```

```
# // An dieser Stelle müssen sie entscheiden, ob sie alles neu übersetzen wollen oder aber
// relativ schnell ein laufendes System haben möchten.
```

```
# ./update-step4.sh // Optional. Erneuert nur beschädigte Pakete.
```

#### Warnung

Dies wird ALLE beschädigten Pakete entfernen, um sie anschließend wieder einzuspielen.

```
# emerge -e world // Optional. Übersetzt das gesamte System mit den von ihnen angegebenen gcc-Flags.
```

#### Notiz

Beide Möglichkeiten in der gegebenen Reihenfolge auszuführen sollte problemlos funktionieren.

### 2.2 Skript 1

Bevor sie damit anfangen irgendeines der Skripte auszuführen, sollten sie ihren Portage-Tree aktualisieren (*emerge rsync*) und gegebenenfalls ihr System aktualisieren (*emerge -u world*).

#### Befehlsauflistung 2

```
# emerge rsync
# emerge -u world // Dies ist zwar optional, aber eine gute Idee bevor sie fortfahren
# ./update-step1.sh
```

Skript 1 wird einige Tests auf ihrem System durchführen und gegebenenfalls das **gentoolkit** installieren, falls sie es noch nicht getan haben. Der wichtigste Teil ist aber, dass Sicherheitskopien wichtiger Pakete in dem Pfad `/usr/portage/packages/All` angelegt werden. Dies wird etwa 10 Minuten dauern.

### 2.3 Skript 2

Dieses Skript wird den System-Link von `/etc/make.profile` auf den richtige Pfad für das Upgrade setzen.

#### Warnung

Wenn sie nicht auf einer x86-Architektur arbeiten MÜSSEN sie die Variable ARCH am Anfang dieser Datei auf den richtigen Wert für **ihre Architektur** setzen (nochmal: gute Werte sind x86, ppc, sparc und sparc64).

### Befehlsauflistung 3

```
// Bezieht sich nur auf die vorangegangene Warnung:  
# nano -w update-step2.sh
```

Dieses Skript wird viel länger laufen, als das vorherige - in etwa 30 Minuten auf einem Dual-Pentium3-866MHz mit 256MB RAM.

### Befehlsauflistung 4

```
# ./update-step2.sh
```

## 2.4 Skript 3

Okay, jetzt haben sie das System bootfähig (eng. bootstrapped) gemacht und die Grundpakete mit allgemeinen Flags übersetzt. Jetzt ist es an der Zeit, erneut das Boot-System zu übersetzen - diesmal mit GCC 3.2 und den von ihnen angepassten Flags. Bearbeiten sie `make.conf` um sicherzustellen, dass alles in Ordnung ist.

### Befehlsauflistung 5

```
# nano -w /etc/make.conf  
# ./update-step3.sh
```

### Notiz

Wenn sie von GCC 2.x aktualisieren, beachten sie bitte, dass mit Version 3.x und darüber neue Flags eingeführt wurden. Schauen sie unter <http://www.freehackers.org/gentoo/gccflags/> nach, um Informationen darüber und andere GCC-Flags zu bekommen.

Jetzt einfach das Skript ausführen. Ich hätte gerne die Zeit gestoppt, um ihnen eine Vorstellung zu vermitteln, wie lang das Ganze dauert - aber dies ist wirklich systemabhängig. Fahren sie deshalb einfach fort und führen sie das Skript aus. Dies kann einige Zeit dauern, auf einem Dual-AthlonMP mit 1.5GHz z.B. 156 Minuten. Der Autor empfiehlt den Film Star Wars Episode IV, damit die Zeit schneller vergeht.

Nach der Abarbeitung werden sie einige weiterführende Anweisungen bekommen. Es ist sehr wichtig, dass sie ihre Konfigurationsdateien aktualisieren, da viele wichtige Dateien geändert wurden. Der einfachste Weg dies zu machen, ist es `etc-update` auszuführen.

### Befehlsauflistung 6

```
# etc-update
```

Dieses Upgrade wird zumeist Probleme mit **init** und **login** verursachen. Es gibt zwei Wege diese zu beseitigen, zum einen können sie `sync ; init u` ausführen oder einfach das System neu starten. SSH sollte auch weiterhin funktionieren, somit können sie sich jederzeit am Rechner anmelden, falls er blockiert ist. Starten sie den Rechner neu, falls sich die Konsole aufgehängt oder sie anfängt massenweise Fehlermeldungen auszugeben.

### Befehlsauflistung 7

```
# sync  
# init u
```

```
// Wenn sich die Konsole aufgehängt oder sie massenweise Fehlermeldungen ausgibt:  
# reboot
```

## 2.5 Skript 4

Wenn alles glatt geht haben sie jetzt ein fertiges GCC-3.2-System, allerdings teilweise mit beschädigten Paketen. Skript 4 wird diese für sie suchen und reparieren. Erneut sei darauf hingewiesen, dass dies ein optionaler Schritt ist - ihr Basissystem ist bereits nach dem 1.4-Profil aktualisiert. Der Grund dafür, dass einige Pakete kaputt sind ist, dass einige Bibliotheken neu übersetzt oder aktualisiert wurden und Programme die von ihnen abhängen nicht funktionieren werden, bis sie ebenfalls neu übersetzt wurden.

### Befehlsauflistung 8

```
# ./update-step4.sh
```

Dieser Vorgang wird sehr viel mehr Zeit in Anspruch nehmen, als alle anderen Schritte zusammen. Wenn sie jetzt schon Star Wars Episode IV gesehen haben, würde ich ihnen empfehlen sich die anderen vier Episoden anzuschauen. Sollten sie einen wirklich langsamen Rechner besitzen, ist die Sammlung aller Seinfeld Episoden (ja, alle 180) eine noch bessere Idee. Nur um ihnen eine Vorstellung zu geben: auf meinem 1.5GHz-Dual-AthlonMP hat der gesamte Vorgang 400 Minuten gedauert und er wurde bei einigen Paketen, wie z.B. Evolution und Mozilla, unterbrochen, so dass ich diese erneut übersetzen musste. Und wir alle wissen wie lange allein Mozilla dafür braucht.

Nach der Abarbeitung, wird Skript 4 die Pakete, die es nicht übersetzen konnte auflisten. Einige Paketübersetzungen werden fehlschlagen, da diese noch auf GCC-3.2 angepasst sind. Somit müssen sie entweder warten oder selbst einige Veränderung in den Ebuilds vornehmen, wenn sie wissen was sie tun!

#### **Notiz**

Skript 4 wird nicht alle ihre installierten Pakete neu übersetzen. Wenn sie wirklich alle Pakete auf ihrem System neu übersetzen wollen, müssen sie *emerge -e world* ausführen. Sie können dies anstatt oder nach Skript 4 ausführen, aber seien sie darauf vorbereitet, dass nicht alle ihre Pakete übersetzt werden können. Wenn sie das ganze nach Skript 4 starten, erhöhen sie die Chance, dass C++ Programme korrekt neu übersetzt werden.

### **3. Abschließende Überlegungen**

Wie sie feststellen werden, ist das ganze eine sehr heikle Angelegenheit. Einer der Autoren und verschiedene andere Leute sind sehr zufrieden mit dem Ergebnis. Der andere Autor hatte jedoch ein paar Probleme, darunter ein schwerwiegendes (KDE verweigerte die Arbeit, obwohl es eingespielt wurde).

Eine mögliche Alternative zum Upgrade ist eine Neuinstallation von Gentoo 1.4. Sichern sie einfach nur ihr */home* und ihr */etc* - sie werden ihre Benutzer glücklich machen, wenn sie deren Dateien sichern, und mit den gesicherten Konfigurationsdateien wird die Installation des neuen Systems einfacher. Es ist wirklich wichtig, dass sie jedes Paket auf ihrem System aktualisieren und sicherstellen, dass deren Konfigurationsdateien vorhanden sind bevor sie dies versuchen, da das neue System die neuste Version aller Pakete installiert. Durch gegensätzliche Konfigurationen könnten Programme fehlerhaft starten oder sich abnormal verhalten.



## >> Portage Benutzer Leitfaden

[Bitte Kapitel auswählen] 

### 1. Zu Beginn

#### 1.1 Portage auf den aktuellen Stand bringen

Nachdem Sie Gentoo Linux installiert und ein wenig mit ihm gearbeitet haben, ist es möglich, dass Sie Fehler in einigen Paketen entdecken oder Sie wollen einfach nur die neusten Software Pakete installieren oder Ihre momentanen Pakete aktualisieren. Um dies durchzuführen, müssen Sie sich unseren neusten Portage Baum herunterladen, bzw. Ihre lokale Kopie auf den neusten Stand bringen. Wir haben verschiedene Anonyme rsync-Server, die Sie benutzen können um dies zu tun. Hier erfahren Sie nun, wie es geht.

Benutzen Sie folgendes Kommando um Ihren Portage Baum zu synchronisieren.

##### Befehlsauflistung 1

```
# emerge rsync
```

##### Warnung

Bitte beachten Sie, dass *emerge sync* automatisch die **--clean** Option aktiviert, das heisst alle Ihre Änderungen in */usr/portage* werden wieder gelöscht. Um Ihre eigenen Ebuilds aus dem Portage Baum auszulagern, benutzen Sie die **PORTAGE\_OVERLAY** Funktion.

Wenn Sie anfangen viele Ebuilds zu schreiben, Bugs zu fixen und mehr, möchten Sie vielleicht Gentoo Linux Developer werden. Wenden Sie sich für weitere Informationen an [Daniel Robbins](#) oder [Seemant Kulleen](#).

#### 1.2 Portage updaten

Bevor Sie unseren Portage Baum verwenden können, ist es wichtig, dass Sie Ihre Portage auf den aktuellen Stand bringen. Dies tun Sie mit folgendem Kommando:

##### Befehlsauflistung 2: Portage aktualisieren

Dies Zeigt an, welche Pakete aktualisiert werden sollen.

```
# emerge -up system
```

Dies aktualisiert die entsprechenden Pakete

```
# emerge -u system
```

Nun benutzen Sie die neueste Version von Portage. Ab jetzt können Sie unser ebuild System benutzen, um Ihre installierte Software auf den neusten Stand zu bringen.

## 2. Einführung von emerge

### 2.1 emerge --pretend

Bevor Sie ein Paket installieren, ist es grundsätzlich eine gute Idee zu schauen, welche Abhängigkeiten benötigt und welche Pakete aktualisiert werden müssen. *emerge --pretend* oder *emerge -p* kann dies für sie tun.

##### Befehlsauflistung 3: Emerge benutzen

```
# emerge -p xchat
```

These are the packages that I would merge, in order.

```
Calculating dependencies..... done!  
[ebuild  U] sys-libs/zlib-1.1.3-r2 to /  
[ebuild  U] dev-libs/glib-1.2.10 to /  
[ebuild N  ] media-libs/jpeg-6b-r2 to /  
[ebuild N  ] x11-base/xfree-4.0.3-r3 to /
```

```
[ebuild N ] x11-libs/gtk+-1.2.10-r1 to /
[ebuild N ] media-libs/giflib-4.1.0-r3 to /
[ebuild N ] media-libs/tiff-3.5.6_beta to /
[ebuild N ] media-libs/imlib-1.9.10 to /
[ebuild N ] net-irc/xchat-1.4.3 to /
```

In diesem bestimmten Fall werden wir **xchat** auf einer Maschine ohne X installieren. So hat *emerge --pretend* richtig festgestellt, dass zunächst etliche Abhängigkeiten erfüllt werden müssen. Besonders **sys-libs/zlib** und **dev-libs/glib** müssen aktualisiert werden und ausserdem müssen noch einige Ebuilds (natürlich auch **x11-base/xfree**) eingespielt werden.

## 2.2 USE und emerge

Oben habe ich *emerge --pretend* auf einem System ausgeführt, welches *gnome* nicht in der *USE* Variable in */etc/make.conf* definiert hat. Das bedeutet, dass eine optionale GNOME Unterstützung, falls vorhanden, nicht aktiviert wird. Wie auch immer, **xchat** **hat** optionale GNOME Unterstützung. Nun lassen Sie uns einen Blick auf die Ausgabe von *emerge --pretend* werfen, nachdem ich *gnome* zur *USE* Umgebungsvariable in */etc/make.conf* ergänzt habe:

### Befehlsauflistung 4: Emerge in Kombination mit USE Flags

```
# emerge -p xchat
```

These are the packages that I would merge, in order.

```
Calculating dependencies..... done!
```

```
[ebuild N ] media-libs/jpeg-6b-r2 to /
[ebuild N ] gnome-base/libghttp-1.0.9 to /
[ebuild N ] media-libs/audiofile-0.2.1 to /
[ebuild N ] media-sound/esound-0.2.22-r2 to /
[ebuild N ] gnome-base/gnome-env-1.0 to /
[ebuild N ] gnome-base/libxml-1.8.11 to /
[ebuild N ] gnome-base/ORBit-0.5.8 to /
[ebuild N ] gnome-base/oaf-0.6.5 to /
[ebuild U] dev-libs/glib-1.2.10 to /
[ebuild N ] net-libs/libwww-5.3.2-r1 to /
[ebuild N ] media-libs/giflib-4.1.0-r3 to /
[ebuild N ] dev-util/guile-1.4-r3 to /
[ebuild U] sys-libs/zlib-1.1.3-r2 to /
[ebuild N ] x11-base/xfree-4.0.3-r3 to /
[ebuild N ] x11-libs/gtk+-1.2.10-r1 to /
[ebuild N ] media-libs/tiff-3.5.6_beta to /
[ebuild N ] media-libs/imlib-1.9.10 to /
[ebuild N ] gnome-base/gnome-libs-1.2.13 to /
[ebuild N ] gnome-base/glibwww-0.2-r1 to /
[ebuild N ] gnome-base/gdk-pixbuf-0.11.0 to /
[ebuild N ] gnome-base/gconf-1.0.0 to /
[ebuild N ] gnome-base/gnome-vfs-1.0.1 to /
[ebuild N ] gnome-base/control-center-1.4.0.1 to /
[ebuild N ] gnome-base/scrollkeeper-0.2 to /
[ebuild N ] dev-util/xml-18n-tools-0.8.1 to /
[ebuild N ] gnome-base/libglade-0.16-r1 to /
[ebuild N ] gnome-base/gnome-core-1.4.0.4 to /
[ebuild N ] net-irc/xchat-1.4.3 to /
```

Wie Sie sehen erkennt *emerge* Dank der gesetzten *USE* Variable, dass *xchat* optionale GNOME Unterstützung beinhalten sollte. Damit diese optionale GNOME Unterstützung korrekt kompiliert werden kann, muss zunächst GNOME installiert werden. *emerge* erkennt dies und ergänzt verschiedene Pakete, die zur GNOME Installation benötigt werden zur Liste der zu installierenden Pakete. In einigen Fällen kann es vorkommen, dass Ihre *USE* Variable nicht richtig gesetzt ist. Dadurch kann es passieren, dass *emerge* unerwartete optionale Erweiterungen aktiviert oder ausschliesst. Deshalb empfehle ich, dass Sie vor dem eigentlichen *emerge* immer ein *emerge --pretend* ausführen. Dies gilt besonders für neue oder ungewöhnliche Ebuilds. Dadurch wissen Sie, was Sie erwartet. Sobald alles OK aussieht können Sie mit dem eigentlichen *emerge* durch Weglassen der *--pretend* Option fortfahren:

### Befehlsauflistung 5: xchat emergent

```
# emerge xchat
```

Nachdem alle Abhängigkeiten installiert sind (soweit es welche gibt; nicht alle Pakete haben welche) wird der xchat Source nach [/usr/portage/distfiles](#) runtergeladen, verifiziert (md5), entpackt, compiliert und in ein temporäres Verzeichnis Namens "Sandbox" installiert. Danach werden sie ins lokale Filesystem integriert und eine Paketdatenbank in [/var/db/pkg/net-irc/xchat-1.4.3/CONTENTS](#) erstellt. Diese Datenbank enthält die installierten Dateien und ihre md5-Checksummen.

Um zu sehen, welche USE Flags in Zusammenhang mit einem bestimmten Paket benutzt werden können fügen Sie das `-v` oder `--verbose` Argument an `emerge -p` an:

**Befehlsauflistung 6:** Benutzen von emerge mit `--verbose`

```
# emerge -pv gentoo-sources
```

These are the packages that I would merge, in order:

Calculating dependencies ...done!

```
[ebuild U ] sys-kernel/gentoo-sources-2.4.20-r5 -build +crypt -evms2
-aavm -usagi
```

## 2.3 Herausfinden, was sich geändert hat

Wenn Sie herausfinden möchten, was sich zwischen der bereits installierten und der neueren Version eines Pakets geändert hat fügen Sie `--changelog` oder `-l` als Argument an das `emerge` Kommando an:

**Befehlsauflistung 7:** Anzeigen des ChangeLog

```
# emerge -pl mozilla
```

These are the packages that I would merge, in order:

Calculating dependencies ...done!

```
[ebuild U ] net-www/mozilla-1.3-r1 [1.2.1-r5]
```

```
*mozilla-1.3-r1
```

```
22 Mar 2003; Martin Schlemmer <azarah@gentoo.org> mozilla-1.3-r1.ebuild :
Add Gtk2 patch. Add default/prefs/xft.js when Xft is enabled. Some other
long overdue cleanups.
```

```
*mozilla-1.3
```

```
21 Mar 2003; Jay Kwak <jayskwak@gentoo.org> mozilla-1.3.ebuild :
Add XIM input patch for GTK
```

```
18 Mar 2003; Martin Schlemmer <azarah@gentoo.org> mozilla-1.3.ebuild :
New version.
```

```
13 Mar 2003; Olivier Reisch <doctomoe@gentoo.org> mozilla-1.2.1-r5.ebuild :
Marked ppc stable
```

```
*mozilla-1.3_beta
```

```
23 Feb 2003; Martin Schlemmer <azarah@gentoo.org> mozilla-1.3_beta.ebuild :
New version.
```

## 3. Pakete upgraden

Der Standardweg um Pakete unter Gentoo Linux zu aktualisieren ist `emerge --update` oder `emerge -u`.

**Befehlsauflistung 8:** Benutzen von emerge `-u`

```
# emerge -u xchat
```

Portage benutzt ein sogenanntes "sicheres" deinstallieren; es "löscht" nur Original Dateien. Wenn eine Datei überschrieben oder modifiziert wurde, bleibt diese im Dateisystem erhalten (vorausgesetzt, dass Sie eine neuere Version installiert haben). Wenn Sie also eine alte Version von xchat nach der Installation einer neuen Version löschen (unmergen), wird die ausführbare Datei von xchat nicht gelöscht, solange sie einen neueren Zeitstempel bzw. eine andere md5-Checksumme hat. Dieses sichere Löschen ist grossartig, da es sicherstellt, dass jeder Zeit eine Version der Anwendung verfügbar ist. Würden Sie zuerst löschen, würde xchat für einige Minuten nicht verfügbar sein, bis die neue Version heruntergeladen, kompiliert und installiert wäre.

### **Wichtig**

Portage hat nun eine spezielle Funktion, die "config file protection", eine Schutzfunktion für Konfigurationsdateien. Die Aufgabe dieser Funktion ist es zu verhindern, dass neue Pakete bereits existierende Konfiguration verändern. Schutz der Konfigurationsdateien ist automatisch für /etc und die KDE Konfigurationsverzeichnisse aktiviert. Um weitere Informationen dazu zu erhalten geben sie `emerge --help config` ein.

## **4. Auflösen von blockenden Paketen**

Aktuell installierte Pakete können in manchen Fällen die Installation anderer Pakete blockieren. Dies kann passieren, wenn die Funktionalität eines Pakets in ein anderes verschoben wurde oder zwei Pakete zueinander inkompatibel sind. Ein blockendes Paket muss zunächst deinstalliert werden, bevor das geblockte Paket installiert werden kann.

**Befehlsauflistung 9:** Emergen eines Pakets, das geblockt ist

```
# emerge -pv libbonobo
```

These are the packages that I would merge, in order:

```
Calculating dependencies ...done!  
[blocks B      ] gnome-base/bonobo-activation (from pkg gnome-base/libbonobo-2.4.0)  
[ebuild      U ] gnome-base/ORBit2-2.8.1 [2.6.3] -doc +ssl  
[ebuild      U ] gnome-base/libbonobo-2.4.0 [2.2.3] -doc
```

In dem oben dargestellten Beispiel blockt bonobo-activation die Installation von libbonobo-2.4.0.

**Befehlsauflistung 10:** Entfernen eines blockierenden Pakets aus dem System

```
# emerge -C bonobo-activation  
# emerge libbonobo
```

Das Entfernen von bonobo-activation aus dem System erlaubt Ihnen libbonobo-2.4.0 erfolgreich zu installieren.

### **Wichtig**

Das Ausführen von `unmerge (emerge -C)` entfernt ein Paket auch dann, wenn es als Abhängigkeit für andere Pakete benötigt wird. Dies kann Ihr System in einen Status bringen, in dem einige Programme oder gar das ganze System nicht mehr funktionieren.

## **5. Das Verhalten von Portage kontrollieren**

Wenn Sie Anpassungen am Verhalten von Portage vornehmen möchten, sollten Sie `/etc/make.conf` editieren. Diese Datei enthält Variablen (oder Beispiele für Variablen), mit denen Sie das Verhalten von Portage ändern können. Wenn Sie zum Beispiel ändern möchten, wie Portage Sourcecode herunterlädt sollten Sie `FETCHCOMMAND` an Ihre Bedürfnisse anpassen.

`/etc/make.conf` enthält eine Vielzahl von Beispielen für Variableneinstellungen aus denen Sie die für Sie optimalen ableiten können. Sie sollten auch einen Blick in die `make.conf` Manpage (`man make.conf`) werfen und/oder das [Portage Handbuch](#) lesen.

Wenn Sie eine Variable nur für einen einzigen Lauf benötigen, können Sie sie anstelle des Editierens der `/etc/make.conf` als Umgebungsvariable setzen. `AUTOCLEAN="no" emerge kde` zum Beispiel deaktiviert das autocleaning während `emerge kde`.

## **6. Was sind maskierte Pakete?**

Viele Leute sind neugierig, warum ein neu veröffentlichtes Paket nicht in einem *emerge -u world* enthalten ist. Ein gutes Beispiel ist xfree-4.3.0 (aktuell als diese Anleitung geschrieben wurde). Wenn Sie *emerge sync* gefolgt von einem *emerge -u world* ausführen werden Sie feststellen, das xfree nicht aktualisiert werden soll. Warum das?

Der Grund ist, dass bestimmte Pakete als "masked" markiert sind. Das heisst, dieses Paket wird nicht automatisch aktualisiert, solange Sie nicht bestimmte Aktionen ausführen um dies dennoch zu erreichen. Für eine Erklärung, wie die Installation von maskierten Pakete funktioniert, schauen Sie sich den [Maskierte Pakete FAQ](#) Thread in unseren [Gentoo Foren](#) an.



## >> Portage Handbuch

[Bitte Kapitel auswählen]

# 1. Portage: Ein Überblick

## 1.1 Überblick

Portage ist ein sehr mächtiges und fortgeschrittenes Paket-Management-System. Seine Flexibilität und Fähigkeit als einfaches Werkzeug zum Kompilieren von Software oder als Herzstück einer brandaktuellen Linux Distribution zu dienen, ist nahezu einzigartig. Die Gentoo Linux Distribution wurde um Portage herum entwickelt.

Gentoo Linux wird oftmals als "Meta-Distribution" bezeichnet. Gentoo besteht aus Portage und über 4000 Anleitungen zum Kompilieren von Paketen, sogenannten **ebuilds**. Diese **ebuilds** geben Portage die Anweisungen wie ein bestimmtes Softwarepaket kompiliert und installiert werden soll. Durch die Benutzung von **Profilen** und dem Kommandozeilen Programm **emerge** können Benutzer und Entwickler Portage dazu nutzen um die Pakete zu installieren und zu pflegen, die die Grundlage des Betriebssystems und der darauf laufenden Anwendungen darstellen.

Ein Gentoo Linux wird "on-the-fly" kompiliert, d.h. direkt auf den entsprechenden Rechner angepasst und erstellt. Der Installationsprozess umfasst das Erstellen eines funktionierenden Compilers sowie einer minimalen Umgebung, in der Portage Quellcode aus dem Internet laden kann, um den Rest des "Systemkerns" und etwaige Anwendungen zu installieren. Auch wenn Portage die Benutzung von vorkompilierten Anwendungen unterstützt, stellt diese nur einen Kompromiss dar und wird nur zur Installation auf langsamen Maschinen bzw. von Entwicklern die schnell ein bestimmtes Paket wiederherstellen wollen verwendet. Desweiteren gibt es einem die Möglichkeit, Pakete auf einer schnellen Maschine zu kompilieren, um sie dann auf einem langsamen Rechner zu installieren.

Durch das "on-the-fly" Kompilieren und die starke Konfigurierbarkeit von Portage, sind nur sehr wenige Gentoo Installationen gleich. Im Grunde wird bei der Installation von Gentoo Linux eine angepasste Linux Distribution erstellt, die sich an den Optionen, wie sie in der Portage Konfiguration und den ebuilds selber definiert sind, orientiert.

Auf den ersten Blick mag die Idee hinter Portage ähnlich dem BSD Ports System sein. Beide kompilieren Pakete aus den Sourcen und erlauben dem Benutzer Software sicher zu installieren bzw. zu deinstallieren. Und beide lösen Abhängigkeiten automatisch auf. Viele Ideen von Portage wurden beim BSD Ports System ausgeliehen, jedoch handelt es sich bei Portage definitiv nicht um eine weitere "Ports-Kopie".

Das Portage System ist eine Verbindung aus einem auf Python basierenden Kern und auf Bash Scripten basierten **ebuilds**. Anstatt mit Makefiles und dem *make* Kommando zu hantieren, verbindet Portage die Möglichkeiten von Python und das einfach zu handhabende Shellscripting mit einigen objektbasierten Eigenschaften, um ein einzigartig mächtiges System zu erstellen. Dieses setzt Portage an die Spitze aller aktuellen Ports Systeme.

Einige der von Portage angebotenen erweiterten Funktionen sind die Möglichkeit verschiedene Versionen und Überarbeitungen des gleichen Pakets zu halten, Auflösung der entsprechenden Abhängigkeiten, feinstrukturiertes Paketmanagement, sichere Installation via Sandbox, Schutz von bestehenden Konfigurationsdateien, Profile und vieles mehr. Viele dieser Funktionen werden noch im Laufe dieses Handbuchs genauer erläutert.

## 1.2 Umgebungsabhängige Auflösung von Abhängigkeiten und Unterstützung von Features

Das Portage System ist im Hinblick auf die dem Benutzer gebotene Flexibilität einzigartig. Traditionelle BSD Ports Systeme tendieren dazu nur jeweils eine Version eines Paketes zu installieren. Portage hat diese Begrenzungen nicht. Es können mehrere Versionen eines Paketes installiert werden. Paketabhängigkeiten (Pakete, die zum Kompilieren andere Pakete nötig sind) können entweder mit ihrem Namen oder mittels zusätzlich angehangener Versionsnummer definiert werden. Das ermöglicht mehrere Versionen eines Paketes zeitgleich zu pflegen.

Das Abhängigkeits-System unterstützt ebenfalls umgebungsbedingte Abhängigkeiten. Portage hat dafür ein leistungsstarkes Konzept, die sogenannten **USE Settings**. Durch ändern einer Konfigurations-Variablen in einer Portage Konfiguration, ist es möglich bestimmte Unterstützung für Funktionen oder

Bibliotheken während des Kompilierens zu aktivieren bzw. abzuschalten. Dies ist ein sehr flexibles und leistungsstarkes System, welchem wir uns im nächsten Kapitel widmen werden.

Zusätzlich unterstützt Portage Konzept der **SLOTS**. Während der Entwicklung von Gentoo Linux wurde klar, dass wir öfters mehrere Versionen bestimmter Pakete (wie z.B. Bibliotheken) benötigen, um die Ansprüche anderer Pakete zu erfüllen. Der traditionelle Weg um dieses Problem zu lösen war, verschiedene Versionen eines Paketes als unterschiedliche Pakete mit leicht abweichenden Namen zu behandeln.

Anstatt verschiedene Versionen als eigene Pakete zu behandeln, brachten die Entwickler Portage bei, wie mehrere Versionen eines Paketes gleichzeitig durch Benutzung von **SLOTS** zu handhaben sind. Als Beispiel sei hier die gängige Freetype Bibliothek genannt. Die 1.x Reihe von Freetype ist mit der 2.x Reihe inkompatibel, jedoch sind oftmals beide Versionen nötig, um die Abhängigkeiten verschiedener Pakete zu erfüllen. Die meisten Distributionen und Ports Systeme tendieren dazu, ein "freetype" Paket für Freetype 1.x und ein "freetype2" Paket für Freetype 2.x anzubieten. Dies betrachten wir als beinahe komplett beschädigtes Paketmanagement System. Wir ergänzen einfach die *SLOT* Nummer 1 zur ersten und Nummer 2 zur zweiten Version. Mit dieser Information ist Portage in der Lage beide Zweige zu pflegen und, falls nötig, beide auf höhere Versionen zu updaten.

## 1.3 Profile

Portage unterstützt ein weiteres Konzept, sogenannte **Profile**. Ein Profil enthält eine Liste von Paketnamen und Versionen mit Anweisungen und einigen Standardoptionen, welche von Portage benutzt werden sollen. Ein Profil enthält Informationen welche Pakete und welche Versionen jeweils erlaubt bzw. verboten oder als zwingend notwendig behandelt werden sollen. Der Benutzer kann zwischen verschiedenen Profilen wechseln in dem er einfach einen Symlink ändert (</etc/make.profile>). Dieses erscheint einfach, aber es erlaubt Portage den Kern einer Distribution anzupassen und als hochwertiges Build System zu dienen.

Der gesamte Aufwand, der bei der Entwicklung von Gentoo Linux betrieben wurde, resultiert in einer Sammlung von **ebuilds** und einem Profil. Dieses Profil beschreibt welche Pakete als Kernpakete für den Betrieb des Systems wichtig sind. Das Profil erlaubt darüber hinaus den Entwicklern bestimmte Pakete zu blockieren bzw. freizuschalten. Dies ist auch nützlich, um defekte Pakete temporär zu deaktivieren. Die **ebuild** Dateien definieren einfach nur wie bestimmte Pakete von Portage kompiliert und installiert werden sollen, die durch das Profil verlangt bzw. erlaubt werden.

## 2. Portage konfigurieren

### 2.1 Überblick

Dieses Kapitel soll verschiedene Aspekte der Konfiguration von Portage abdecken, die für Benutzer wie Entwickler wichtig sind. Portage ist ein sehr flexibles System und Sie müssen verstehen, wie man Portage richtig konfiguriert, um Ihr System genau auf Ihre Bedürfnisse abstimmen zu können.

Bitte beachten Sie, dass in diesem Dokument der Begriff "Benutzer" Personen mit administrativen Rechten definiert. Um die Portage Konfiguration zu ändern und neue Pakete zu installieren bzw. zu entfernen benötigen Sie Root-Rechte.

### 2.2 Portage Konfigurations Dateien

Fast alle Konfigurations-Optionen, die weiter unten erklärt werden, befinden sich in den Dateien </etc/make.conf>, </etc/make.profile/make.defaults> und </etc/make.globals>. </etc/make.conf> enthält einige Optionen, die von Portage benutzt werden. Portage wird erst aktuell gesetzte Umgebungsvariablen für jegliche Optionen überprüfen. Werden keine Umgebungsvariablen gefunden, wird die Datei </etc/make.conf> überprüft. Können dort auch keine Optionen gefunden werden, wird </etc/make.profile/make.defaults> überprüft. Ist dort ebenso keine Option vorhanden, werden die Standardwerte aus der </etc/make.globals> verwendet. Beachten Sie, dass alle Optionen, die in </etc/make.conf> definiert sind, von den Optionen, die in </etc/make.globals> gemacht wurden, fast immer überschrieben werden. Alle Optionen in </etc/make.conf> und </etc/make.globals> können systemweit als global betrachtet werden, solange es Portage betrifft.

Wenn Sie überprüfen wollen, ob eine Option bereits definiert wurde, ist es zu empfehlen, dass Sie zuerst </etc/make.conf> überprüfen und erst dannach </etc/make.globals>. Sofern nicht anders angegeben, werden Optionen, die in </etc/make.globals> definiert wurden, von den Optionen in </etc/make.conf> überschrieben.

### 2.3 USE Einstellungen

Das USE-System ist eine flexible Möglichkeit, um sämtliche Features, direkt beim Kompilierprozess von Gentoo Linux oder später für einzelne Pakete ein- bzw. auszuschalten. Dies erlaubt eine Administrierbarkeit, wie die Pakete in Verbindung mit den angegebenen Features in der USE-Variable, kompiliert werden. Zum Beispiel, wenn ein Paket Unterstützung für GNOME bietet, können Sie diese abschalten, indem Sie in der USE-Variable `-gnome` angeben. Wenn Sie jedoch die GNOME-Unterstützung für das Paket aktivieren wollen, dann setzen Sie in der USE-Variable `gnome` ein.

Der Effekt der USE Flags auf Pakete ist abhängig davon, ob die Software und das ebuild Paket selbst, das jeweils gesetzte USE Flag unterstützen. Wenn dem nicht so ist, so bewirkt auch das jeweilige USE Flag nichts an der Software. Auch werden einige Paketabhängigkeiten bei mancher Software nicht als optional angesehen, dadurch haben gesetzte USE Flags keine Auswirkung auf diese vorgeschriebenen Abhängigkeiten. Eine Liste der USE Flags, die von einem Paket benutzt werden, kann durch `emerge --verbose --pretend ebuild` herausgefunden werden.

Eine Liste der USE Flags, die von Gentoo Linux benutzt werden, können Sie aus der Datei `/usr/portage/profiles/use.desc` beziehen. Jedes USE Flag ist jeweils in einer Zeile mit einer zusätzlichen Beschreibung erklärt, welches Feature es aktiviert.

Portage bestimmt, welche USE Flags ein- bzw. ausgeschaltet sind aufgrund einer Überprüfung von bis zu vier verschiedenen Stellen, wo USE definiert sein kann. Diese Stellen bewirken, dass die USE Flags, die dort definiert sind, "aufgestapelt" werden. Es durchforstet jede einzelne Stelle und merkt sich jedes USE Flag der früheren Stellen, ob sie ein- bzw. ausgeschaltet sind und fügt neu gefundene Features hinzu, so dass es eine lange USE Variable ergibt. Wird zum Beispiel ein Feature an einer früheren Stelle abgeschaltet und an der aktuellen Stelle, die Portage durchforstet, eingeschaltet, so wird das Feature trotz Abschaltens an der früheren Stelle wieder eingeschaltet.

Portage überprüft die Stellen in denen USE Flags definiert werden in der Reihenfolge, wie sie in der Variable **USE\_ORDER** in `/etc/make.globals` angegeben ist. Möchten Sie eine Stelle ausschalten, dann entfernen Sie diese einfach in **USE\_ORDER**.

Die folgenden Absätze beschreiben jeden Ort, der in **USE\_ORDER** mit der Portage Standard Konfiguration definiert wurde.

## Defaults

Portage Profile können einen Satz von Standard USE Flags beinhalten. Diese Standard Features sind bei jedem Profil dabei und werden in der Datei `make.defaults` angegeben. Da `/etc/make.profile` als eine Verknüpfung zum aktuell benutzten Portage Profil fungiert, ist der Satz an standard USE Flags des aktuellen Profils die Datei `/etc/make.profile/make.defaults`. Es ist nicht zu empfehlen Änderungen in dieser Datei vorzunehmen, da zukünftige Profiländerungen sie wieder überschreiben würden.

## Auto

Diese Pakete sind in der Datei `use.defaults` im Portage Profil (`/etc/make.profile/use.defaults`) definiert. Jeder Eintrag beinhaltet eine USE Flag Bezeichnung und das dazugehörige Paket. Wenn das für ein USE Flag angegebene Paket installiert ist, wird das Feature als eingeschaltet betrachtet. Wenn Sie zum Beispiel das Paket `x11-base/xfree` installieren und das USE Flag X nicht abschalten, so wird diese solange global eingeschaltet, wie das Paket installiert ist. Es ist nicht zu empfehlen, diese Datei zu bearbeiten da zukünftige Profiländerungen, diese überschreiben würden.

## Conf

Wenn eine *USE* Variable in `/etc/make.conf` definiert ist, werden die *USE* Flags, die in jener Variable angegeben wurden, verwendet. Wenn es keine USE Variable in `make.conf` gibt, dann wird `/etc/make.globals` überprüft. Solch ein Eintrag könnte wie folgt aussehen:

### Befehlsauflistung 1: Beispielhafte USE Variable

```
USE="slang readline gpm berkdb gdbm tcpd pam libwww ssl gb tk
lm_sensors lvm ldap tex bonobo sdl gtk xfs evo pda ldap
mmx mitshm perl python guile ruby postgres dvd 3dnow tcl
lcms gif sdl vorbis ogg oss libg++ directfb decss snmp
gnome X opengl mozilla pdflib gpg -nls gd xface jpilot
-kde -qt -esd -motif -alsa oggvorbis"
```

USE Flags werden eingeschaltet, indem Sie einfach in der USE Variable angegeben werden. USE Flags können jedoch auch mittels Negation durch ein - (Minus) vor dem Feature-Namen deaktiviert werden. Zum Beispiel schaltet `gnome` GNOME Support ein, während `-motif` den MOTIF Support abschaltet.

Die *USE* Variable in `/etc/make.conf` ist der empfohlene Ort, wo Sie USE Flags ein- bzw. ausschalten. Diese Datei wird nicht automatisch von Portage überschrieben. Es wird empfohlen, dass Sie hier die USE Flags eintragen, die nicht von den vorangehenden Stellen ein bzw. ausgeschaltet werden sollen.

## Env

Die USE Flags kann man von Hand aus mit seiner eigenen Shell Umgebungsvariable überschreiben.

**Befehlsauflistung 2:** USE Flags temporär setzen

```
# export USE="--gnome"
# emerge net-im/gaim
```

oder

```
# USE="--gnome" emerge net-im/gaim
```

Dies erlaubt die Benutzung von speziellen USE Flags für ein oder mehrere Pakete. Diese von Ihnen per Hand definierten USE Flags bleiben bestehen, solange Sie in jener Shell Umgebung damit arbeiten. Verlassen Sie Ihre Umgebung ("exit"), ist auch das von Hand gesetzte USE Flag weg.

## Notiz

Auch wenn Portage die aktuellen USE Flags zum Zeitpunkt der Installation des Pakets speichert, so sind diese Einstellungen nicht dauerhaft gespeichert. Sollte dieses Paket in der Zukunft erneut installiert bzw. aktualisiert werden, werden nur die aktuell im System definierten USE Flags verwendet und keinesfalls die zum Zeitpunkt der Erstinstallation geltenden.

## 2.4 Compiler Optionen

Die Compiler Optionen, welche von Portage zum kompilieren von Paketen genutzt werden, können in `/etc/make.conf` gesetzt werden indem *CHOST*, *CFLAGS* und *CXXFLAGS* editiert werden. Die *CHOST* Einstellung gibt an, auf welcher Plattform Sie kompilieren. *CFLAGS* und *CXXFLAGS* geben Compiler Einstellungen für C bzw. C++ an.

Es werden Ihnen einige kommentierte Standard Einstellungen für unterschiedliche Plattformen angeboten. Diese Einstellungen können als getestet und stabil angesehen werden. Diese zu ändern, könnte zu Fehlern in der Software und dem Compiler führen. Bitte seien Sie beim Ändern der Standard Einstellung vorsichtig, da sie zu einem unbenutzbaren bzw. beschädigten System führen können.

Benutzer eines Multiprozessor Systems können einige Vorteile erhalten, indem sie in `/etc/make.globals` die Option *MAKEOPTS* editieren. Diese Option wird während des Kompilierungs Vorganges an *make* übergeben um mehrere *gcc* Instanzen zu aktivieren.

## 2.5 Lage der Verzeichnisse

Portage erlaubt es Ihnen zu bestimmen in welchen Verzeichnissen Pakete kompiliert und verschiedene Dateien gespeichert werden sollen. Die meisten Benutzer werden diese Optionen nicht benötigen. Folgende Optionen können genutzt werden:

- *PORTDIR* - Position des Portage Trees
- *DISTDIR* - Lokaler Cache der heruntergeladenen Pakete
- *PKGDIR* - Ort der lokal erzeugten tbz2 Pakete
- *RPMDIR* - Ort der lokal erzeugten RPM Pakete
- *PORTAGE\_TMPDIR* - Temporärer Platz zum Kompilieren der Pakete
- *BUILD\_PREFIX* - Gehört zu *PORTAGE\_TMPDIR*
- *PKG\_TMPDIR* - Gehört zu *PKGDIR*

## 2.6 Schutz der Konfigurationsdateien

Portage erlaubt den Schutz aller Konfigurationsdateien in bestimmten Verzeichnissen. Portage wird in geschützten Verzeichnissen keine Dateien überschreiben. Wenn ein Paket versucht, bereits existierende Dateien zu überschreiben, wird die neue Datei nach dem Muster `._cfg0000_name` umbenannt. Dies geschieht damit der Benutzer später die neue Datei einsehen kann, um die Unterschiede manuell einzufügen.

Geschützte Verzeichnisse können über *CONFIG\_PROTECT* entweder in `/etc/make.conf` oder `/etc/make.globals` definiert werden. Bestimmte Dateien und Unterverzeichnisse unterhalb geschützter Verzeichnisse können durch *CONFIG\_PROTECT\_MASK* vom Schutz ausgenommen werden.

Das Folgende ist nur ein Beispiel und sollte nicht blind kopiert werden:

**Befehlsauflistung 3:** Beispielhafte Angabe zum Schutz von Konfigurationsdateien

```
CONFIG_PROTECT="/etc /usr/share/config /usr/kde/2/share/config"  
CONFIG_PROTECT_MASK="/etc/gconf /etc/init.d /etc/pam.d"
```

Benutzer können mehr Informationen über den Schutz von Konfigurationsdateien erhalten, indem sie folgendes in ihrer Shell eingeben:

**Befehlsauflistung 4:** Hilfe zum Schutz von Konfigurationsdateien aufrufen

```
# emerge --help config
```

## 2.7 FEATURES

Portage bietet eine ganze Menge Optionen, die auf Entwickler abzielen, welche regeln wollen wie sich Portage verhält und nach der Installation verschiedener Paketstufen aufräumt. Diese Optionen sollten nur für Entwickler interessant sein. Bei Benutzern könnten sie Probleme verursachen.

Eine Liste einzuschaltender Funktionen kann unter **FEATURES** entweder in `/etc/make.conf` oder `/etc/make.globals` gesetzt. Hier eine Liste der verfügbaren Funktionen:

- *digest* : Erstellt automatisch einen Digest für neue Ebuilds.
- *cvs*  : Ergänzt neue Ebuild Digests automatisch ins CVS.
- *sandbox*  : Aktiviert das Sandboxverfahren.
- *noclean*  : Portage räumt nach dem kompilieren nicht auf.
- *noauto*  : Führt keine wichtigen Ebuild Schritte automatisch aus.
- *distcc*  : Benutzt das verteilte Kompilieren mit distcc.
- *ccache*  : Behält kompilierte Objektdateien, sodass eine erneute Kompilierung weniger Zeit benötigt.
- *buildpkg*  : Erstellt Binärpakete für jedes Paket, das Sie emergent.
- *userpriv*  : Kompiliert nicht mit root-Privilegien.
- *usersandbox*  : Benutzt eine Sandbox, wenn **userpriv** aktiviert ist.
- *keeptemp*  : Entfernt nicht die temporären Dateien (`$_T`) nach dem mergen.

Einige dieser Funktionen können durch Setzen der folgenden Variablen angepasst werden:

- *CCACHE\_SIZE*  : Damit wird festgelegt, wie viel Platz *ccache* benutzen kann. Standard sind 2GB.
- *DISTCC\_HOSTS*  : Legt fest, welche Rechner Teil des verteilten Kompilierens mit *distcc* sind. Alle diese Rechner müssen den *distccd* Daemon laufen haben.

## 2.8 Portage SYNC Einstellungen

Portage bietet die Möglichkeit den Portage Tree mittels *rsync* aufzufrischen. Finden Sie einen anderen Weg, können Sie angeben, welche Methode Portage verwenden soll, indem Sie ihn in der *SYNC* Option in der `/etc/make.conf` angeben.

### RSYNC

*rsync* ist der gewöhnliche Weg Ihren Portage Tree aktuell zu halten. Die *rsync* Methode wird in der `/etc/make.conf` gesetzt:

**Befehlsauflistung 5:** Den Portage Tree mittels *rsync* aktuell halten

```
SYNC="rsync://rsync.gentoo.org/gentoo-portage"
```

### Notiz

Die *rsync*-Methode überschreibt blind alle Veränderungen, die an Ihrem lokalen Portage Tree vorgenommen wurden. Falls Sie lokale Änderungen behalten möchten, müssen Sie `PORTDIR_OVERLAY="/ein/verzeichnis/fuer/Ihre/Änderungen"` setzen.

### Entwickler CVS

Entwickler mit vollem Zugang zum CVS können auch per **emerge** den lokalen Tree mit dem CVS abgleichen lassen, welches sie mittels SSH erreichen können.

Laden Sie einfach den CVS Tree mittels Ihres Accounts herunter ("checkout"), verschieben Sie diesen nach `/usr/portage` und benutzen Sie folgende **SYNC** Option:

**Befehlsauflistung 6:** Den Portage Tree mittels Entwickler-CVS aktuell halten

```
SYNC="cvs://Ihr_account@cvs.gentoo.org:/home/cvsroot"
```

## 2.9 Spiegelungen (Mirrors)

Das Gentoo Projekt bietet eine lokale Spiegelung von allen Quellpaketen, die in ebuilds im Portage Tree angegeben sind. Meistens werden Source Tar-Archive usw. auf langsamen Servern gespeichert, die ziemlich oft ausfallen können. Auch Entwickler entfernen regelmässig alte Versionen Ihrer Software von FTP-Servern, wenn neue Versionen herausgegeben werden. Um das Leben von Benutzer, die die Gentoo Distribution nutzen, zu erleichtern, spiegeln wir diese Dateien. Das erlaubt ein schnelleres und sicheres Herunterladen der Archivdateien von Spiegel-Servern, die näher an Ihrem Standort liegen.

Wann auch immer Sie ein Paket mergen und dazu das Quellpaket benötigen, versucht Portage zu allererst unseren Spiegel-Server für die angeforderten Dateien zu erreichen. Wenn sich auf diesem Server die benötigten Dateien nicht befinden, versucht Portage den jeweiligen HTTP- oder FTP-Server, der im ebuild angegeben ist.

Die Angabe des Spiegel-Server, den Portage benutzt, erfolgt in der `GENTOO_MIRRORS` Option, die sich in der Datei `/etc/make.conf` befindet. Das Folgende ist die standard Einstellung:

**Befehlsauflistung 7:** Download-Mirrors angeben

```
GENTOO_MIRRORS="http://www.ibiblio.org/gentoo"
```

Für einen Gentoo-Spiegel-Server in Ihrer Nähe, besuchen Sie die Gentoo Webseite oder fragen Sie in der lokalen Mailingliste nach einem für Sie günstigeren Server. Das Programm `mirrorselect` überprüft anhand von pings, welcher der für Sie am schnellsten erreichbare Server ist und fügt optional eine Liste von Servern in die `GENTOO_MIRRORS` Option in der `/etc/make.conf` ein. Schauen Sie in die [Installationsanleitung](#) für die Benutzung von `mirrorselect`.

## 2.10 Download-Programm

Das Programm, das Portage zum Download der benötigten Dateien verwendet, kann durch die Optionen `FETCHCOMMAND` und `RESUMECOMMAND` angegeben werden. Einige Beispiele werden in `/etc/make.conf` und `/etc/make.globals` gezeigt. Als Standardprogramm verwendet Portage `wget`, welches den meisten Anforderungen genügen sollte.

### Notiz

Portage gibt Informationen über HTTP und FTP Proxies an das jeweilige Download-Programm weiter, die in den `HTTP_PROXY` und `FTP_PROXY` Optionen definiert sind.

## 2.11 Proxies

Portage kann angewiesen werden, zum Download HTTP oder FTP Proxies zu nutzen. Die Proxies können bei den Optionen `HTTP_PROXY` und `FTP_PROXY` entweder in `/etc/make.conf` oder in `/etc/make.globals` angegeben werden. Auch hier sollten die Einstellungen vorzugsweise in `/etc/make.conf` erfolgen. Wenn HTTP und FTP über den gleichen Proxy erfolgen soll, können Sie auch einfach die Option `PROXY` nutzen.

Im Folgenden ein Beispiel:

**Befehlsauflistung 8:** HTTP- und FTP-Proxies setzen

```
HTTP_PROXY="http://192.168.1.1:8080"  
FTP_PROXY="http://192.168.1.1:8080"
```

oder

```
PROXY="http://192.168.1.1:8080"
```

Portage kann zusätzlich angewiesen werden einen HTTP Proxy zur Verwendung von `rsync` zu nutzen. Dies kann durch die Option `RSYNC_PROXY` in `/etc/make.conf` oder als Umgebungsvariable aktiviert werden.

Im Folgenden ein Beispiel:

**Befehlsauflistung 9:** RSYNC-Proxy setzen

RSYNC\_PROXY="192.168.1.1:8080"

## Notiz

Sollten Sie hinter einer Firewall sitzen und rsync scheint Ihren HTTP Proxy nicht nutzen zu können, können Sie Ihren Portage Tree durch einen Snapshot Tar-Archiv updaten. Schauen Sie in unsere [FAQ](#) für weitere Informationen.

## 2.12 Verschiedene Optionen

Die folgenden Optionen können für einige Nutzer sinnvoll sein:

- *NOCOLOR* : Erlaubt es dem Benutzer, das Verwenden von Farben bei Ausgaben von emerge zu deaktivieren.
- *CLEAN\_DELAY* : Dadurch verzögert Portage das Entfernen von Paketen, um Nutzern die Möglichkeit zum Abbruch zu geben. Diese Zeit kann hier angegeben werden. Zum Abschalten der Verzögerung setzen Sie den Wert auf "0".

## 3. Paketmanagement

### 3.1 Updaten des Portage Trees

Der Portage Tree, der in `/usr/portage` liegt, enthält die Bibliothek der "Bauanleitungen" für verschiedene Pakete (sogenannte *ebuilds*). Darüber hinaus enthält der Tree auch noch Informationen zu dem Profil und der *package.mask*, die wichtig sind, um das System aktuell zu halten. Um immer die aktuellsten Versionen und neuesten Bugfixes zu haben, ist es wichtig den Tree regelmäßig mit dem offiziellen Tree abzugleichen. Sie können den Portage Tree mittels folgendem Kommando updaten:

**Befehlsauflistung 10:** Den Portage-Tree updaten

```
# emerge sync
```

Die von Portage genutzte Methode, kann geändert werden. Schauen Sie in die [Portage SYNC Settings](#) im [Portage Konfigurieren](#) Kapitel für weitere Informationen.

### 3.2 Pakete installieren (mergen)

Der Vorgang des Kompilierens und Installierens eines Paketes durch Portage wird als *mergen* bezeichnet. Portage kompiliert Pakete und installiert diese temporär in ein "Abbild-Verzeichnis", in dem es die zu installierenden Dateien aufzeichnet. Diese Dateien werden dann aus dem "Abbild-Verzeichnis" ins Root (/) Dateisystem integriert (merged).

Das *emerge* Kommando dient als Front-End des Portage Systems. Das Installieren und Entfernen von Paketen wird durch dieses Kommando und seine diversen Argumente kontrolliert.

Um die neueste, unmaskierte Version eines Paketes zu installieren, geben Sie einfach den Paketnamen, wie folgt ein (Beispiel):

**Befehlsauflistung 11:** Ein Paket installieren

```
# emerge galeon
```

Dieses Kommando wird zunächst alle benötigten Abhängigkeiten (unter Berücksichtigung der USE Flags) und dann die neueste und unmaskierte Version von Galeon kompilieren und installieren. Galeon könnte auch mit vollem Namen inklusive Kategorie angegeben werden: *net-www/galeon*

Das *emerge* Kommando akzeptiert auch die direkte Angabe von ebuilds. Dies erlaubt dem Benutzer auch ältere Versionen eines bestimmten Paketes oder ebuilds von Drittanbietern zu installieren. Beachten Sie, dass dadurch alle Maskierungen für das Paket übergangen werden und Ihre *ACCEPT\_KEYWORDS* Einstellung ignoriert wird. Das Folgende ist ein Beispiel:

**Befehlsauflistung 12:** Ein Paket unter Angabe der Version installieren

```
# emerge /usr/portage/net-www/galeon/galeon-1.2.0-r3.ebuild
```

Zusätzlich zur Angabe des Paketnamen oder ebuild, unterstützt *emerge* verschiedene weitere Argumente. Eines dieser Argumente ist *--pretend*, vielleicht eines der nützlichsten. Durch dieses Argument wird das geplante Vorgehen nicht durchgeführt. Stattdessen gibt Portage eine Liste aller zu

installierenden Pakete aus. Das Folgende zeigt eine Auflistung der Pakete, die während der Installation der neuesten Version des Kdevelop Paketes installiert würden:

**Befehlsauflistung 13:** Darstellung der zu installierenden/aufzufrischenden Pakete

```
# emerge --pretend kdevelop
```

These are the packages that I would merge, in order.

```
Calculating dependencies ...done!  
[ebuild N ] kde-base/kdelibs-2.2.2-r4 to /  
[ebuild N ] dev-util/kdbg-1.2.2 to /  
[ebuild U ] app-text/psutils-1.17 to /  
[ebuild U ] app-text/a2ps-4.13b-r3 to /  
[ebuild U ] app-text/jadetex-2.20 to /  
[ebuild N ] app-text/sgmltools-lite-3.0.3-r2 to /  
[ebuild N ] kde-base/kdoc-2.2.2-r1 to /  
[ebuild N ] net-www/htdig-3.1.5-r2 to /  
[ebuild N ] app-text/enscript-1.6.3-r1 to /  
[ebuild N ] kde-base/kdebase-2.2.2-r2 to /  
[ebuild N ] app-doc/qt-docs-2.3.1 to /  
[ebuild N ] dev-util/kdevelop-2.0.2 to /
```

Mit *N* gekennzeichnete Pakete sind noch nicht auf ihrem Rechner installiert, würden aber durch die angegebene Aktion eingespielt werden. Pakete, die mit einem *U* gekennzeichnet sind, befinden sich bereits in einer älteren Version auf Ihrem System und werden durch diese Aktion aktualisiert.

Folgende Parameter sind ausserdem Verfügbar:

**--fetchonly** : Lädt alle benötigten Dateien herunter, die für das Kompilieren notwendig sind, sowie alle Abhängigkeiten, die dadurch entstehen.

**--emptytree** : Diese Option lässt Portage vortäuschen, dass keine der Abhängigkeiten oder Pakete, auf denen das zu installierende Paket beruht, installiert sind. Dies lässt sich sehr gut mit der Option **--pretend** verbinden, um eine komplette Liste der Abhängigkeiten für jedes einzelne Paket anzeigen zu lassen. Alle Abhängigkeiten mit Ausnahme von *glibc* werden dargestellt.

**--nodeps** : Mit dieser Option versucht Portage nur die angegebene Pakete zu "mergen" und ignoriert sämtliche Abhängigkeiten. Bitte beachten Sie, dass diese Option zu Problemen führen kann, wenn Sie die Pakete, von denen das jeweilige Paket abhängig ist, nicht bereits installiert haben.

**--onlydeps** : Mit dieser Option ist es möglich, nur die Abhängigkeiten des jeweiligen Paketes zu "mergen", jedoch **nicht** das ausgewählte Paket selbst.

**--noreplace** : Wenn Sie Pakete zum "mergen" angeben, die bereits installiert sind, Sie jene aber nicht durch neue ersetzen wollen, hilft Ihnen diese Option weiter.

**--usepkg** : Anstatt das angegebene Paket zu kompilieren, versucht Portage mit dieser Option ein vorkompiliertes *tbz2* Paket von einer angegebenen Stelle zu installieren . Jene Stelle ist in der *PKGDIR* Shellumgebungsvariable anzugeben.

**--debug** : Um eine noch detailliertere Ausgabe zu bekommen, was während der Aktion mit Portage passiert, benutzen Sie diese Option. Normalerweise werden Ausgaben "menschlich lesbarer" dargestellt. So haben Sie zum Beispiel als Entwickler die Möglichkeit, syntaktische Fehler in den Bash Script basierten *ebuild* Dateien zu finden.

**--autoclean** : Zwingt **emerge** zum totalen Bereinigen von paketspezifischen temporären Verzeichnissen für Kompilervorgänge, bevor das Paket kompiliert wird. Portage erledigt dies bei der standard Konfiguration von selbst, dadurch ist diese Option nur für Entwickler interessant, die dieses Verhalten abgeschaltet haben.

**--verbose** : Sagt *emerge*, dass es im ausführlichen Modus laufen soll. Zusammen mit **--pretend** werden die möglichen USE Flags ausgegeben.

### 3.3 "Unmergen" (Deinstallieren) von Paketen

Der Vorgang des "unmergens" ist, dass die Dateien, die mit einem installierten Paket verbunden sind, gelöscht werden. Damit ist die Software vom System entfernt und kann nicht mehr benutzt werden, bis Sie jenes Paket wieder "mergen".

Pakete werden mittels des *emerge* Befehls und dem Parameter *unmerge*, gefolgt vom Namen des Paketes entfernt. Das folgende Beispiel beseitigt alle installierten Versionen vom *ltrace* Paket.

**Befehlsauflistung 14:** Ein Paket deinstallieren

```
# emerge unmerge ltrace
```

oder

```
# emerge unmerge dev-util/ltrace
```

Portage erlaubt ausserdem das "unmergen" von spezifischen Versionen eines Paketes. Bereiche werden durch = (exakte version), < (kleiner als), > (größer als), <=(kleiner als oder gleich), und >= (größer als oder gleich) dargestellt. Das folgende Beispiel würde alle Versionen, die gleich und älter des Paketes *ltrace* in der Version 0.3.15 sind, "unmergen".

**Befehlsauflistung 15:** Bestimmte Versionen eines Pakets deinstallieren

```
# emerge unmerge \<=dev-utils/ltrace-0.3.15
```

Wenn Sie Bereiche für Pakete benutzen, so stellen Sie sicher, das Sie jeweils für die Zeichen > und < ein Backslash davorsetzen, sodass Ihre Shell in diesem Fall dies nicht falsch interpretiert. Ausserdem ist es von Nöten, die Kategorie des Paketes, wie im Beispiel gezeigt, anzugeben. Für andere Beispielausdrücke zu Bereichen, führen Sie den Befehl *emerge --help* aus.

### Warnung

Das "unmergen" von Paketen kann gefährlich sein. Wenn Sie ein Paket des Grundsystems entfernen, verliert Ihr System an Funktionalität und bei entfernten Bibliotheken droht funktionsuntüchtige Software. **Portage warnt Sie nicht, wenn Sie Pakete des Grundsystems oder gar Abhängigkeiten anderer Pakete entfernen.**

Wenn das zu entfernende Paket tatsächlich installiert ist, wird das Programm *emerge* exakt anzeigen, welche Pakete entfernt werden und wartet eine gewisse Anzahl an Sekunden ab, um den Benutzer die Möglichkeit zu geben, den Vorgang mittels der Tastenkombination Control-C abzubrechen.

Beginnt erstmal der Vorgang des "unmergens", sehen Sie eine lange Liste von Dateinamen, die mit dem Paket verbunden sind. Manche dieser Dateinamen haben ein Merkmal (flag), das an der linken Seite der Datei angezeigt wird. Die Merkmale *!mtime*, *!empty*, und *cfgpro* verdeutlichen, weshalb einige Dateien nicht entfernt worden sind, als das Paket "unmerged" wurde. Dateien ohne jegliche Merkmale wurden erfolgreich vom System entfernt.

Das Merkmal *!mtime* sagt aus, dass die Datei nach der Installation des Pakets geändert wurde. Das bedeutet, dass jemand nach dem "mergen" des Paketes diese Datei bearbeitet hat oder zu einem späteren Zeitpunkt andere Pakete sie überschrieben haben. Dies erlaubt es, dass Pakete aktualisiert werden können, ohne die Gefahr, dass wichtige Dateien entfernt werden.

Das Merkmal *!empty* weist auf Verzeichnisse hin, welche Portage verbietet zu entfernen, da sie nicht leer sind (mehrere Pakete teilen sich oft das selbe Verzeichnis, welches das Paket, was "unmerged" wird, entweder selbst gehört oder ebenfalls benutzt). Der Konfigurationsdatei Schutz-Mechanismus tritt dann ein, wenn Sie das *cfgpro* Merkmal sehen. Das bedeutet, ein neueres Paket, was installiert wird, übernimmt den Besitz jener Konfigurationsdateien und Portage verweigert die Entfernung dieser Dateien.

### Warnung

Dateien werden immer dem letzten installierten Paket zugeordnet. Dies ist abhängig von der Reihenfolge der Installation und unabhängig von der aktuellen Versions- oder Revisionsnummer der Pakete, die installiert sind. Wenn ein Paket eine Datei besitzt, wird diese immer mit deinstalliert, auch wenn eine ältere Version eines Paketes diese Datei installiert hat, solange der Nutzer diese Datei nicht manuell geändert hat.

## 3.4 System Update

Portage unterstützt die Möglichkeit installierte Pakete mit einem einzigen Befehl zu aktualisieren. Das System-Update-Feature ermöglicht es Ihnen, die Kernpakete ihres Systems zu Versionen zu aktualisieren, die von den Gentoo-Entwicklern empfohlen werden und zum einwandfreien Betrieb von Gentoo Linux notwendig sind. Ein System-Update aktualisiert nur Pakete, die als essentiell angesehen werden. Also nur die Pakete, die im Portage Profil angegeben sind, werden als absolut wichtig für den Betrieb und die Aktualität des Systems erachtet.

Um ein System-Update zu starten, geben sie den folgenden Befehl ein:

**Befehlsauflistung 16:** Das Kernsystem aktualisieren

```
# emerge --update system
```

Portage wird nun, abhängig von den Versionen und Paketen, welche sie derzeit installiert haben, die Updates kompilieren und installieren, die vom aktuellen Portage-Profil empfohlen werden. Sie haben die Möglichkeit sich mit der Option **--pretend** eine Liste mit den Paketen die installiert bzw. aktualisiert werden, ausgeben zu lassen, wenn das oben gezeigte Beispiel ausgeführt würde.

### Notiz

Wie sie aus der Gentoo-Installationsanleitung erfahren können, ist einer der ersten Schritte der Befehl *emerge system*, um das Grundsystem zu installieren. Mit *emerge --update system* werden diese Basispakete auf den aktuellsten Stand gebracht.

## 3.5 World Update

Portage unterstützt außerdem die Möglichkeit, alle nicht-essentiellen Pakete mit einem einzigen Befehl zu aktualisieren. Das Portage-System besitzt dafür einen gewissen Grad an "Intelligenz", die es ermöglicht, ein System mit verschiedene Versionen von Paketen, die in Konflikt zueinander stehen, sicher zu aktualisieren .

Das world-update-Feature von Portage überprüft das Systemprofil, die Liste der blockierten Pakete (*package.mask*), das world-Profil und die Abhängigkeiten (inkl. Versionkontrolle) von Paketen, die im world-Profil stehen. Dadurch findet es heraus, welche Pakete aktualisiert werden müssen. Ein Paket wird nur aktualisiert, wenn es eine neue Version gibt und das Paket im world-Profil aufgeführt wird oder ein anderes Paket, das im world-Profil steht von ihm abhängt. Selbstverständlich darf das Paket oder eine spezielle Version dessen nicht durch das Systemprofil oder die *package.mask* blockiert sein.

Portage versucht nun alle Pakete, die im world-Profil aufgeführt sind auf die neuste verfügbare und unblockierte Version zu aktualisieren. Desweiteren überprüft Portage auch die Abhängigkeiten von jedem Paket im world-Profil und wird versuchen diese auf die neuste Version zu aktualisieren. Dabei wird eine Versionkontrolle durchgeführt, sodass die Versionhierarchie bestehen bleibt. Außerdem dürfen diese Pakete weder durch das Systemprofil noch durch *package.mask* blockiert sein. Schließlich werden die *SLOTS* überprüft, die in einem vorangegangenen Kapitel besprochen wurden.

Benutzer, die andere Distributionen und ihre Paket-Management-Systeme neben Portage kennen, sind vielleicht etwas darüber irritiert, dass Portage nicht nur ein blindes Aktualisieren der Pakete vornimmt, einfach nur anhand der Versionsnummern (Wie es bis Gentoo 1.0 gehandhabt wurde).

Viele der Pakete, die in Gentoos Portage-Tree sind, stehen in verschiedenen Versionen zur Verfügung. Eine ältere oder neuere Version eines Paketes kann mit der Software, die auf sie aufbaut inkompatibel sein. Blindes Aktualisieren von Bibliotheken und Programmen, ohne Rücksicht darauf, dass sie von anderen Paketen gebraucht werden, kann schnell zu vielen schwerwiegenden Problemen führen. Um dies zu verhindern lässt Portage beim Aktualisieren Vorsicht walten und bezieht die Abhängigkeiten aller Pakete, basierend auf den Angaben in den einzelnen ebuilds, mit ein.

Das Herz von Portages-World-Update ist das World-Profil. Anders als das System-Profil, welches nur von den Entwicklern gewartet wird und nie vom Benutzer verändert werden sollte, wird das World-Update-Profil indirekt mit der Zeit durch Aktionen des Benutzers erstellt. Das world-Profil funktioniert in etwa wie eine "Favoriten-Liste". Pakete die vom Benutzer manuell mit Hilfe von *emerge* installiert werden, werden in der Datei *world* aufgezeichnet. Diese Datei findet sich unter */var/cache/edb/world*. Portage macht dies, da sie ihm mitgeteilt haben es zu installieren (per *emerge*) und es annimmt, dass sie ein Interesse daran haben, das Paket immer auf dem aktuellsten Stand zu halten.

Die *world* Datei besteht aus einem Paketnamen mit Kategorie pro Zeile und sollte in etwa wie folgt aussehen:

**Befehlsauflistung 17:** Inhalt einer world Datei

```
net-im/gaim
net-www/skipstone
net-www/galeon
app-editors/vim
app-text/ispell
net-mail/evolution
dev-util/ltrace
sys-fs/xfsprogs
-net-www/mozilla-0.9.8-r3
sys-apps/attr
```

```
sys-apps/dmapi
sys-kernel/linux-sources
sys-apps/acl
app-office/gnucash
app-cdr/xcdroast
```

Nahezu alle Einträge in diesem Beispiel wurden von Portage automatisch hinzugefügt, als der Benutzer eines der Pakete manuell "ein-merge-te". Diese Pakete werden aktualisiert, wenn eine neuere Version verfügbar ist.

### Notiz

Um Zeit zu sparen und sicher zu stellen, dass alle Ihre bevorzugten Pakete aktuell gehalten werden, können Sie die **world** Datei selbst bearbeiten und so Einträge für diese Pakete hinzufügen. Wenn Sie eine älteren Version von Portage aktualisieren, müssen sie das world-Profil erstellen und dem System bekannt machen. Bei aktuellen Installationen von Gentoo und Portage sollte das world-Profil während der Installation erzeugt werden.

Ein interessanter Eintrag ist der für das Mozilla-Paket (`=net-www/mozilla-0.9.8-r3`). Dieser Eintrag wurde von Hand hinzugefügt, um eine exakte Version festzulegen. Paketeinschränkungen, wie sie im Abschnitt "Unmergen (Deinstallieren) von Paketen" in diesem Handbuch besprochen wurden, können dazu verwendet werden, Portage zu zwingen nur spezielle Versionen beim Aktualisieren von Paketen zu verwenden. Der obige Eintrag hat z.B. den Effekt, dass Portage auf das Paket mozilla-0.9.8-r3 als einzig verfügbare Version festgelegt ist. Somit wird dieses Paket im Verlaufe eines World-Update nie aktualisiert werden.

World-Updates werden durch den folgenden Befehl gestartet:

**Befehlsauflistung 18:** World Update ausführen

```
# emerge --update world
```

Portage wird nun versuchen alle Pakete die in der **world** Datei stehen und (wenn nötig) deren Abhängigkeiten aktualisieren. Abhängigkeiten werden auf die neueste verfügbare Version, die vom zu aktualisierenden Paket gebraucht wird, gebracht. Pakete die nicht in der **world** Datei aufgeführt sind und keine Abhängigkeiten von den vorher genannten Paketen sind, werden nicht aktualisiert.

Wenn *emerge* gegen Ihren Willen einige Pakete downgraden will, benutzen Sie *--upgradeonly*:

**Befehlsauflistung 19:** Erlaube nur das Upgraden

```
# emerge --upgradeonly world
```

### Warnung

Portage wird keine Dateien in Verzeichnissen überschreiben, die durch die "Configuration File Protection" (Schutz von Konfigurationsdateien) geschützt sind. Es ist notwendig, dass Sie selbst die Unterschiede zwischen Ihren bestehenden und den neuen Dateien, die von Portage generiert wurden, ausgleichen. Wenn Sie Ihre Konfigurationsdateien nicht aktualisieren, werden verschiedene Programme nicht mehr funktionieren. Bitte schauen sie für weitere Informationen unter "Schutz der Konfigurationsdateien" im Kapitel "Portage konfigurieren" nach oder benutzen sie den Befehl *emerge --help config*.

Um eine Liste mit den Paketen zu sehen, die von einem World-Update betroffen sind, können Sie das Argument *--pretend* verwenden, so wie es bereits in einem vorangegangenen Abschnitt in diesem Kapitel besprochen wurde.

### Notiz

Durch ein World-Update wird gleichzeitig auch ein System-Update durchgeführt. Außerdem können Kernpakete nicht auf bestimmte Versionen in der **world** Datei festgelegt werden, da sie vom aktuellen Portage-Profil immer überschrieben werden!

Ein praktischer Nebeneffekt der Art wie World-Update arbeitet, ist für Benutzer interessant, die ein komplettes neu-Kompilieren aller installierten Pakete auf einem System wünschen. Da World-Update alle Pakete und deren Abhängigkeiten, die in der Datei **world** stehen aktualisiert, gibt einem die Option *--emptytree* die Möglichkeit eine neu-Kompilierung sämtlicher Pakete und aller Abhängigkeiten - mit Ausnahme der glibc - zu erzwingen. Das ist z.B. für Leute nützlich, die ihre Compiler-Optionen oder ihre USE Variable geändert haben und wollen, dass diese Veränderungen von der gesamten Software die sie benutzen verwendet wird - ohne dass sie nun jedes Paket selbst erneut "mergen" müssen. Dazu müssen sie einfach die Datei **world** mit allen Paketen, die sie verwenden auffüllen und den folgenden Befehl verwenden:

**Befehlsauflistung 20:** Das System komplett neu kompilieren

```
# emerge --update world --emptytree
```

Sie können die Option `--pretend` mit diesem Befehl verwenden, um eine Liste mit den Paketen, welche neu-Kompiliert werden, zu bekommen.

### 3.6 System aufräumen

Portage hat die Fähigkeit verschiedene Versionen eines Paketes parallel zu installieren. Es gibt einige Pakete in Gentoo's Portage-Tree die diese Funktion nutzen (z.B. zur Kompatibilitätssicherung, wenn ältere Programme mit neueren Versionen inkompatibel sind).

Denken Sie daran, dass wenn eine neuere Version eines Paketes installiert wird, in den meisten Fällen ein Großteil des älteren Paketes überschrieben wird und alles was zurückbleibt sind einige Dokumentationsdateien und andere für das System unwichtige. Mit der Zeit können diese "Dateileichen" sehr viel Festplattenplatz verschwenden.

Um dies zu verhindern bietet Portage einen einfachen Weg an, Rückstände veralteter Dateiversionen vom Benutzersystem zu entfernen. Diese Funktionalität ergibt sich aus der `emerge`-Option `clean` und kann folgendermaßen benutzt werden:

**Befehlsauflistung 21:** Das System aufräumen

```
# emerge clean
```

`emerge` wird nun eine Liste mit Paketversionen und -revisionen ausgeben die entfernt werden und die Versionen die erhalten bleiben. Außerdem gibt es dem Benutzer Zeit, die Aktion mit Control+C abzubrechen. Auf einem normalen System werden nun eine Vielzahl von Aktionen durchgeführt, die eine lange Liste mit Dateien, die entweder gelöscht oder erhalten wurden, ausgibt.

Naheliegenderweise wird Portage die Aufräumaktion auf die `world` Datei (alle installierten Pakete) anwenden. Sie können den Umfang der Säuberung durch Optionen wie `world`, `system`, eine Liste von Paketnamen oder eine Einschränkung auf Paketversionen, wie es im Abschnitt "Unmerge" in diesem Kapitel besprochen wurde, beeinflussen.

Beim Herausfinden, welche Paketversionen entfernt werden sollen, überprüft Portage die verschiedenen Profile, die Beziehungen zu anderen Paketen und den SLOT eines Paketes. Vorausgesetzt dass alle Paketabhängigkeiten für alle Pakete richtig definiert sind, wird `emerge clean` nur veraltet Pakete vom System entfernen und nicht solche, deren Entfernung die Funktionalität des Systems beeinträchtigen würde.

### 3.7 Pakete säubern

Portage bietet außerdem die Funktion ein Paket zu säubern (engl. *prune*). `prune` ist eine unsichere Variante von `clean`. Es entfernt alle Versionen aller Pakete, ausgenommen der zuletzt installierten Version. Es führt nur wenige Überprüfungen aus, die `clean` durchführt und kann grundlegende Abhängigkeiten von Ihrem System entfernen! Wenn Sie diese Option nutzen, können sie sehr schnell ihr System unbrauchbar machen. Somit wird diese Variante nicht empfohlen und sollte nur in wenigen Ausnahmefällen verwendet werden.

Die Aktion `prune` akzeptiert die selben Optionen wie die Aktion `clean` und kann wie folgt angewendet werden:

**Befehlsauflistung 22:** Pakete säubern

```
# emerge prune
```

### 3.8 Den Portage-Tree durchsuchen

Portage-Trees, wie jener der das Herzstück von Gentoo Linux bildet, können sehr groß sein. Der Befehl `emerge` bietet eine Suchfunktion an, die Suchanfragen in Form eines regulären Ausdrucks, dieser muss von Anführungszeichen eingeschlossen sein, akzeptiert. Reguläre Ausdrücke sind sehr komplizierte Biester und Anwendern die sich dafür interessieren, sei ein gutes Buch zum Thema empfohlen.

Die meisten einfachen Suchen könne ohne Wissen, wie ein regulärer Ausdruck zu bilden ist durchgeführt werden. Das folgende ist ein Beispiel für eine einfache Suche nach einem Paket, das "gcc" heisst oder "gcc" im Namen hat:

**Befehlsauflistung 23:** Nach ebuilds suchen

```
# emerge search gcc
```

Für jeden Treffer gibt der Befehl den Paketnamen, die neuste Version, die neuste installierte Version, seine Homepage und eine Beschreibung über die Software im Paket aus.

### 3.9 Hilfe erhalten

Mehr Informationen über die zahlreichen Optionen und Aktionen die *emerge* unterstützt, erhalten Sie durch die Eingabe von:

**Befehlsauflistung 24:** Hilfe zu emerge aufrufen

```
# emerge --help
```

### 3.10 Nützliche Werkzeuge

Verschiedenste Werkzeuge wurden von Gentoo-Benutzern erstellt, um das Arbeiten mit *emerge* zu erleichtern. Diese Werkzeuge sind im Paket *app-admin/gentoolkit* im Gentoo-Portage-Tree zu finden.

- *etc-update* : Shell-Skripte für vim, die dabei helfen die Dateien in */etc* abzugleichen (eine falsche Benutzung kann gefährlich sein!)
- *qpkg* : Werkzeug für Datenbankabfragen
- *epm* : ein weiteres Abfragewerkzeug für die Datenbank mit RPM-ähnlicher Syntax
- *etcat* : zeigt und erläutert die USE Flags für ein Paket

## 4. Was sind maskierte Pakete (masked packages)?

Viele Leute sind verwundert, warum ein neu erschienenes Paket nicht bei einem *emerge -u world* enthalten ist. Ein gutes Beispiel ist *xfree-4.3.0* (die Version zur Zeit des Schreibens). Wenn Sie *emerge sync* gefolgt von einem *emerge -u world* ausführen, werden Sie kein Update zu *xfree* in der Liste sehen. Warum?

Der Grund ist, dass bestimmte Pakete als "masked" markiert sind. Das Paket wird nicht automatisch auf eine neuere Version gebracht oder installiert, solange Sie nicht selbst Hand anlegen. Für eine Beschreibung, wie Sie die Installation von maskierten Paketen durchführen, möchten wir Sie auf die [Masked Packages FAQ](#) in unserem [Gentoo Forum](#) verweisen.



## >> Gentoo Anleitung zu den USE Flags

[Bitte Kapitel auswählen] ▣

### 1. Einführung

#### 1.1 Was ist überhaupt der Sinn von USE flags?

Das was mich am meisten als User zu Gentoo hingezogen hat war die Tatsache, dass es eine stark anpassbare Distribution ist. Da Gentoo eine sourcenbasierte Distribution darstellt, hat der Benutzer die absolute Kontrolle über alle Funktionen und Unterstützungen, die in die installierten Anwendungen einfließen. Darüber hinaus mag ich es nicht Dinge installieren zu müssen, die ich **wirklich** nicht brauche.

Und nun, wie genau setzt Gentoo dies um? Durch das Definieren von *USE* flags. Grundsätzlich sind dies Schlüsselwörter, die Optionen definieren welche systemweit zur Konfiguration der Anwendungen während des Compilations Prozesses genutzt werden.

#### 1.2 Standard USE Flags

Natürlich definiert Gentoo für Sie bereits standardmäßig einige sichere Flags. Diese finden Sie in Ihrer `/etc/make.profile/make.defaults` Datei.

Bei Gentoo-1.2 sind diese *USE* Flags wie folgt definiert:

##### Befehlsauflistung 1

```
USE="3dnow apm arts avi berkdb crypt cups encode gdbm gif gpm gtk imlib ipv6
java jpeg kde libg++ libwww mikmod mmx motif mpeg ncurses nls oggvorbis opengl pam
pdflib png python qt qtmt quicktime readline sdl slang spell ssl svga tcpd truetype
X xml2 xmms xv"
```

##### Warnung

Anstatt diese Datei zu bearbeiten, wie es leider einige machen, sollten Sie auf jeden Fall die Datei `/etc/make.conf` bearbeiten. In dieser Datei finden Sie:

##### Befehlsauflistung 2

```
#USE=" "
```

Entfernen Sie einfach das Kommentarzeichen `#` und ergänzen Sie Ihre eigenen USE Flags zwischen den Anführungszeichen.

##### Wichtig

Beachten Sie, dass *-flag* ein Flag deaktiviert und *flag* das entsprechende Flag aktiviert. Wenn ein Flag nicht in `/etc/make.profile/make.defaults` noch in `/etc/make.conf` zu finden ist, ist diese Funktion deaktiviert

Einfach das Kommentierungszeichne `#` entfernen. Dann die gewünschten Flags ergänzen. "Aber was ist wenn ich nun ein Flag aus der `make.defaults` Datei entfernen möchte?" Dies lässt sich ganz einfach durch ein Ergänzen von `"-flag"` in `/etc/make.conf` erreichen. Meine `make.conf` sieht z.B. so aus:

##### Befehlsauflistung 3

```
USE="-gnome -kde qt -arts -nls python perl oggvorbis opengl sdl
      -postgres jpeg png truetype xml xml2 dvd avi aalib mpeg encode fbcon
      mmx"
```

Da Sie nun wissen wie man extra Flags aktiviert und existierende Flags deaktiviert, schauen wir uns mal an was diese Flags überhaupt machen.

### 2. Vorhandene USE Flags

#### 2.1 HARDWARE FLAGS

Für Ihren Prozessor gibt es drei Flags: *mmx*, *3dnow*, und *sse*. *3dnow* ist anwendbar auf bestimmte AMD K-6, Athlon und höhere CPUs. *mmx* is anwendbar auf Pentium-MMX und höhere CPUs inklusive

Athlons. *sse* ist eine Funktion für höhere Pentiums. Wenn Sie einen Pentium4 mit SSE2 haben, können Sie ebenfalls dieses Flag nutzen.

Nun für Ihre Grafikkarte. Wenn Sie eine 3dfx Karte haben, aktivieren Sie das *3dfx* Flag. Sollten Sie eine Voodoo3 Karte besitzen, können Sie das *voodoo3* Flag aktivieren. Bei einer Matrox Karte schalten Sie einfach das *matrox* Flag ein.

Laptopnutzer werden wohl die *apm* und *pcmcia* Flags aktivieren. Wenn Sie keine Unterstützung für CardBus haben wollen, schalten sie *nocardbus* ein. die *pnp* Unterstützung stellt PNP in pcmcia-cs zur Verfügung.

Sollte Sie einen Palm Pilot oder anderen PDA haben, den Sie mit Ihrer Gentoo Maschine verbinden möchten, sollten sie *pda* nutzen.

Wenn Sie vorhaben eine Digitalkamera zu nutzen, sollten Sie *gphoto2* aktivieren. Dies wird eine entsprechende Unterstützung in die relevanten Anwendungen compilieren.

Wenn Sie einen Drucker nutzen möchten, können Sie mit *cups* die CUPS Drucker Unterstützung aktivieren. Mit *slp* schalten Sie "Service Locator Protocol" Unterstützung für CUPS ein.

## 2.2 BILDSCHIRM FLAGS

Wenn Sie das XFree86 System nutzen möchten, sollten Sie *X* aktivieren. Mit *X* können Sie zusätzlich auch *dga* einschalten, um direkten Grafik Zugang (Direct Graphic Acces - DGA) zu erhalten. Das *opengl* Flag aktiviert OpenGL Unterstützung in vielen Anwendungen. Um die XVideo Erweiterung von XFree86 nutzen zu können, müssen Sie *xv* aktivieren. Nach einem Besucher im IRC Channel #gentoo bei openprojects.net (|Nagash|) "macht xv Mplayer gut"

Falls Sie auch Unterstützung für die Framebuffer Konsole (wie sie vom Kernel zur Verfügung gestellt wird - Der Pinguin beim Booten ist ein Beispiel) wünschen, schalten Sie das *fbcon* Flag ein.

Um bestimmte Anwendungen mit DirectFB zu nutzen, müssen Sie *directfb* einschalten.

## 2.3 DESKTOP FLAGS

Sind Sie KDE Nutzer? Dann sind folgende Flags für Sie:

*kde* -- KDE Unterstützung aktivieren

*qt* -- Erlaubt das linken auf die Qt-Bibliotheken

*qtmt* -- Erlaubt das linken auf die "multi-threaded" Qt-Bibliotheken

*arts* -- Das KDE Sound System (artsd)

### **Wichtig**

Bitte beachten Sie, dass falls Sie keine KDE Unterstützung wünschen mindestens *kde* und *arts* deaktiviert sind (-*kde* -*arts*).

Sie bevorzugen GNOME? Dann benutzen Sie:

*gnome* -- um Anwendungen mit GNOME Unterstützung zu erstellen

*gtk* -- um Anwendungen mit GTK+ Unterstützung zu erstellen

## 2.4 GNOME FLAGS

Es gibt viel mehr Flags für GNOME Nutzer.

*bonobo* -- Unterstützung für GNOMEs CORBA durch Bonobo

*evo* -- Unterstützung für Evolution (eine zu Outlook ähnliche Anwendung)

*gb* -- Unterstützung für Gnome Basic

*gtkhtml* -- Unterstützung für die GTK-HTML Bibliothek

### **Warnung**

Um GNOME Unterstützung zu deaktivieren, ergänzen Sie einfach *-gnome* zu Ihrer USE Zeile in [/etc/make.conf](#)

## 2.5 GRAFIK BIBLIOTHEKEN

Es gibt viele Bibliotheken, die Multimedia Unterstützung liefern. Fangen wir also mit den Grafik Bibliotheken an. Im vorigen Abschnitt haben wir bereits die GTK und Qt Toolkit Bibliotheken gesehen. Andere mögliche Optionen sind:

*motif* -- für OpenMotif Unterstützung

*tcltk* -- für Tcl und/oder Tk Unterstützung

*aalib* -- Unterstützung für ASCII Grafiken

*imlib* -- Unterstützung für Imlib (image loading and rendering library)

*ncurses* -- Unterstützung für Ncurses Konsolen Darstellung - dies bieten eine menge Anwendungen

*readline* -- eine andere auf der Konsole basierende Bibliothek

*sdl* -- SDL (Simple Direct Layer media library) wird von vielen Spielen gebraucht

*svga* -- Einige Anwendungen bieten Unterstützung von SVGAlib. NICHT für PPC Benutzer!

*lcms* -- Bietet Farbmanagement Unterstützung für die Imagemagick Bibliothek

Mit den folgenden Flags können Sie Unterstützung für folgende Grafikformate aktivieren:

*gif* -- Unterstützung für GIF Grafiken (libungif oder giflib)

*jpeg* -- Unterstützung für JPEG Grafiken (libjpeg)

*png* -- Unterstützung für PNG Grafiken

*tiff* -- Unterstützung für TIFF Grafiken

## 2.6 VIDEO BIBLIOTHEKEN

Sie schauen gerne Videos? Dann möchten Sie vielleicht einige der folgenden Flags aktivieren:

*avi* -- Unterstützung für die avifile Bibliothek und optional der Win32 Codec Bibliothek

*mpeg* -- Unterstützung für die libmpeg3 Bibliothek

*quicktime* -- Unterstützung für Quicktime mittels OpenQuickTime und Quicktime4Linux

## 2.7 SOUND BIBLIOTHEKEN

Es gibt eine ganze menge Sound Bibliotheken.

*alsa* -- ALSA Soundtreiber (Advanced Linux Sound Architecture)

*esd* -- ESD Sounddaemon (Enlightened Sound Daemon)

*mikmod* -- Eine Bibliothek für MikMod Sound Module

*oss* -- Das Open Sound System

*nas* -- Network Audio Sound

## 2.8 ANDERE MULTIMEDIA FLAGS

*ggi* -- Unterstützung für Nicht-X Video Treiber

*dvd* -- Unterstützung für das Abspielen und Entschlüsseln von DVD via libcss Bibliothek

*xmms* - Das X MultiMedia System

*oggvorbis* -- Unterstützung für das Ogg Vorbis Format

*flash* -- Unterstützung für das Erstellen von Flash Dateien mittels Ming

*encode* -- Zum encoden verschiedener Multimedia Formate

## **2.9 NETZWERK FLAGS**

Natürlich möchten Sie Ihre Gentoo Box an ein Netzwerk anschliessen ;)

*ipv6* -- Unterstützung für IPv6

*snmp* -- Unterstützung für das Simple Network Management Protocol zum Betreuen von entfernten Maschinen

## **2.10 SICHERHEITS FLAGS**

Wenn Sicherheit für Sie ein Thema ist (was es sein sollte!), empfehlen wir das aktivieren von PAM:

*pam* -- Pluggable Authentication Modules

*ssl* -- Secure Socket Layer -- wird von SSH benötigt. Apache und verschiedene Mailclients unterstützen dies ebenfalls um sichere Verbindungen zu etablieren.

*crypt* -- Unterstützung von mdecrypt, mhash und GPG Verschlüsselung

*socks5* -- Unterstützung für Socks5

## **2.11 MAIL FLAGS**

*imap* -- Unterstützung für IMAP

*ldap* -- LDAP-Unterstützung (Lightweight Directory Access Protocol)

*tcpd* -- Unterstützung für TCP Wrapper

*xface* -- Unterstützung für Xface, Sylpheed und Sylpheed-Claws

## **2.12 MOZILLA FLAGS**

*mozilla* -- Unterstützung für Mozilla einer Anwendungen

Beim Einspielen von Mozilla via emerge sollten Sie sich auch folgendes anschauen:

*mozaccess-builtin* -- Ergänzt Zusatzpakete wie Text-zu-Sprache oder Braille Unterstützung

*mozctl* -- Ergänzt Pango ähnliche Module

*mozirc* -- Ergänzt einen IRC (Internet Relay Chat) Client zu Mozilla

*mozxmlterm* -- Ergänzt ein xterm-ähnliches Terminal mit XML Unterstützung zu Mozilla

## **2.13 BÜRO-SPEZIFISCHE FLAGS**

*spell* -- Unterstützung für Rechtschreibprüfung via ispell/aspell/pspell

*truetype* -- Unterstützung für FreeType und FreeType2 Schriftarten

*xml* -- Unterstützung für die XML Bibliothek (Version 1)

*xml2* -- Unterstützung für Gnomes XML Bibliothek (Version 2)

*pdflib* -- PDF (Adobes Portable Document Format) Unterstützung

*plotutils* -- Unterstützung fürs Plotten in gnuplot

*tetex* -- Unterstützung für TeTeX eines der besten Publishingtools

## 2.14 SPRACHEN

*guile* -- Guile ist ein Interpreter für Scheme

*java*

*perl*

*python*

*ruby*

*slang* -- S-Lang ist eine leistungsstarke Sprache

*libwww* --

*libg++* -- Ergänzt C++ Module

*atlas* --

*pic*--

## 2.15 DATENBANK FLAGS

Sie haben einige Datenbanktypen zur Auswahl für die Unterstützung möglich ist:

*berkdb* -- BerkleyDB für MySQL

*mysql* -- MySQL

*postgres* -- PostgreSQL

*odbc* -- Unterstützung für UnixODBC (Open Database Connectivity)

*innodb* -- Unterstützung für MySQL transaction

*gd* -- GNU database Bibliotheken

*gdbm* -- dito

## 2.16 FILESYSTEM FLAGS

*acl* -- Unterstützung für die XFS Utilities von SGI

*afs* -- OpenAFS

*samba* -- Unterstützung für SAMBA

## 2.17 WEITERE NÜTZLICHE FLAGS

*nls* -- Unterstützung für NLS (Native Language Support), verwendet die gettext Bibliotheken, wenn Sie nur eine Sprache verwenden ist dies nicht nötig

*doc* -- Hierdurch wird weitere Dokumentation erstellt

*gpm* -- Ergänzt Mausunterstützung für die Konsole

## 2.18 FLAGS, WELCHE NICHT BERÜHRT WERDEN DÜRFEN!

Diese Flags dürfen **nicht** berührt werden! Sie werden hier nur genannt um diese Dokumentation zu vervollständigen. Diese Flags werden automatisch mit Ihrem Profil gesetzt. Solange Sie nicht **genau** wissen was Sie tun, aktivieren oder deaktivieren Sie diese Flags nicht in Ihrer Konfiguration.

*x86* -- Identifiziert Ihren Rechner als x86 kompatibel.

*ppc* -- Identifiziert Ihren Rechner als PowerPC.

*sparc* -- Identifiziert Ihren Rechner als Sparc.

*sparc64* -- Identifiziert Ihren Rechner als 64-bit Sparc.



## >> Gentoo Linux Init System

[Bitte Kapitel auswählen] 

### 1. Einleitung

Gentoo Linux benutzt ein Init-System das größtenteils über Abhängigkeiten kontrolliert wird. Es sollte einfach zu warten und doch stark und flexibel genug sein für jede Art von Konfiguration. Diesen Text sollte man nicht als eine Einleitung in die inneren Mechanismen verstehen, sondern vielmehr als eine einfache Anleitung um mit Gentoo's Init-System arbeiten zu können. Leute die ernsthaft daran interessiert sind, die inneren Mechanismen zu verstehen .... lesen Sie den Quelltext ;-)

### 2. Runlevel

Im Gegensatz zu anderen Init-Systemen, bestehen Gentoos Runlevel nicht aus festen Namen oder Nummern, sondern vielmehr aus eigenen Namen, die auf die standard Runlevel von init abgebildet werden.

#### Notiz

Standardmäßig gibt es drei Runlevel, namentlich: "**boot**", "**default**" und "**nonetwork**".

Das "boot" Runlevel sollte der standard Typ für die meisten Setups sein, und ist, wie der Name sagt, das erste Runlevel das nach dem booten ausgeführt wird. Als nächstes kommt "**default**", welches, wie der Name schon andeutet, das standard Runlevel ist und nach dem booten gestartet wird. Zuletzt folgt "**nonetwork**", welches ausschließlich als Beispiel dient.

Die Runlevels liegen in `/etc/runlevels`, in einem Unterverzeichnis das nach dem Namen des Runlevels benannt wurde; dieses Unterverzeichnis enthält symbolische Links zu den Diensten, die in diesem Runlevel geladen werden sollen.

#### Notiz

Der bevorzugte Weg, um einen Service hinzuzufügen oder zu löschen wird später in dem Abschnitt "Über rc-update" behandelt.

Wie bereits erwähnt, kann der Name frei gewählt werden, solange die Datei `/etc/inittab` entsprechend dem neuen Namen des Runlevels angepasst wird.

#### Wichtig

Eine Ausnahme dieser Regel, die jedoch erwähnt werden sollte, stellt das Runlevel "**boot**" dar.

#### Warnung

Bitte ändern Sie den Namen des "**boot**" Runlevels NIEMALS, da es einige Dinge kaputt machen würde.

Die ganze Arbeit wird vom `/sbin/rc` Script erledigt, mit diesem können Sie auch im laufenden Betrieb zwischen den virtuellen Runleveln wechseln.

### 2.2 Virtuelle Runlevel

Da Runlevel nicht statisch auf die von Init gebunden sind, kann man wesentlich mehr haben als init unterstützt. Das ermöglicht es dem Benutzer nach seinen Bedürfnissen Profile und Virtuelle Runlevel zu erstellen

Zum Beispiel könnte ein Laptop Benutzer zwei standard Runlevel haben, "online" und "offline". Das würde erlauben ein aktives Runlevel zu benutzen, wenn die PCMCIA Netzwerkkarte eingesteckt ist, und ein weiteres Runlevel, wenn sie es nicht ist. Die PCMCIA Scripts könnten dann so konfiguriert werden, dass sie `"/sbin/rc online"` oder `"/sbin/rc offline"` aufrufen, um die richtigen Dienste zu starten oder zu stoppen, jeweils abhängig vom Status der PCMCIA Netzwerkkarte.

### 2.3 Runlevel und XFree86

Nach Gentoos Weg, Dinge zu erledigen haben wir kein Runlevel für X, sondern stattdessen ein startup-Script. Es hat den Namen "xdm" und kann zu jedem Runlevel hinzugefügt werden, wenn der Benutzer dieses wünscht.

#### Notiz

Dies sollte das hauptsächlich genutzte Runlevel des Nutzers sein.

#### Warnung

Es zum Boot Runlevel zu ergänzen kann in unerwünschten Nebeneffekten enden.

Wenn xdm, gdm oder kdm vor Ihren Gettys gestartet wird, startet X standardmäßig auf der nächsten freien Konsole. Auf langsameren Rechner ist das kein Problem, solange der Desktop Manager am Ende

des Runlevel Init Vorganges startet. Die Gettys werden vor X starten, welches dann auf Konsole 7 startet. Auf schnelleren Rechnern ist dies nicht der Fall. X wird hier vorher auf Konsole zwei starten. Wenn nun die Gettys starten, übernehmen diese die Kontrolle über die Tastatur, dadurch verliert der Desktop Manager die Kontrolle der Tastatur.

Dies wird dadurch verhindert, dass das Desktop Manager Startscript über ein "Extra-Runlevel", hier das Runlevel "a", geführt wird. Da das Runlevel "a" kein richtiges Runlevel ist, ruft unser "xdm" Script einfach "*telinit*" auf. Dadurch werden alle Dienste des Runlevels "a" erst nach dem aktuellen Runlevel gestartet, also nachdem die Gettys gestartet wurden.

#### **Notiz**

Sie können mehr Informationen über Runlevel "a" erhalten, indem Sie die man pages von init lesen.

## **3. RC-Scripts**

Rc-Scripts sind Scripts, die die grundsätzlichen Funktionen, sowie die Abhängigkeiten von Diensten beim Start definieren. Sie liegen in [/etc/init.d/](#).

### **3.2 Grundlayout eines RC-Skriptes**

**Befehlsauflistung 1:** rc-script Layout

```
#!/sbin/runscript

depend() {
    need bar
}

start() {
    ebegin "Starting foo"
    /sbin/foo
    eend $? "Failed to start foo"
}

stop() {
    ebegin "Stopping foo"
    kill $(cat /var/run/foo.pid)
    eend $? "Failed to stop foo"
}
```

#### **Notiz**

Der Interpreter ist `"/sbin/runscript"`.

#### **Notiz**

Die "depend" Funktion ist optional.

#### **Notiz**

Jedes rc-Script braucht mindestens die "start" Funktion.

### **3.3 Kontrollieren des Start-up**

Die generelle Startreihenfolge der Dienste innerhalb eines Runlevels ist alphabetisch. Dies liegt an der Sortierung der Ausgabe von [/bin/lis](#).

Die erste Möglichkeit von der Startreihenfolge abzuweichen sind Abhängigkeiten. Alternativ können auch die sogenannten "Order Types" genutzt werden.

## **4. Abhängigkeitstypen**

Die meisten Dienste sind mit anderen Diensten verbunden und hängen sogar von ihnen ab.

Zum Beispiel benötigt Postfix ein laufendes Netzwerk, sowie einen Systemlogger.

Samba benötigt ebenfalls ein laufendes Netzwerk. Wenn CUPS zum Drucken benutzt werden soll, muss cupsd auf jeden Fall vor Samba gestartet werden. Beachten Sie, das CUPS nicht unbedingt zum Starten von Samba nötig ist.

Wir haben zwei verschiedene Möglichkeiten, um Abhängigkeiten zwischen verschiedenen Diensten zu definieren. Diese Abhängigkeiten sind immer gültig, egal ob ein Runlevel als ganzes gewechselt wird oder ein Service einfach nur manuell nach dem Booten gestartet wird.

## 4.2 Der NEED Abhängigkeitstyp

Dieser Typ wird benutzt, wenn ein Dienst für das Starten des aktuellen Dienstes dringend nötig ist.

**Befehlsauflistung 2:** Logger und net werden als NEED Abhängigkeit definiert

```
depend() {
    need net logger
}
```

### Notiz

Die Dienste, welche nach **NEED** genannt werden, sind dringend erforderlich, damit der aktuelle Dienst starten kann. Der aktuelle Dienst wird also nicht starten, wenn eine der Abhängigkeiten nicht starten sollte.

### Wichtig

Jeder Dienst, der in der **NEED** Zeile steht, wird gestartet, auch wenn dieser NICHT zum aktuellen oder "**boot**" Runlevel ergänzt wurde.

**NEED** ist deshalb eine "starke" Abhängigkeit.

## 4.3 Der USE Abhängigkeitstyp

Dieser Dienst ist nicht unbedingt zum Starten des aktuellen Dienstes nötig, sollte jedoch vor dem aktuellen gestartet werden.

**Befehlsauflistung 3:** Portmap wird als USE Abhängigkeit zu netmount ergänzt

```
depend() {
    use portmap
}
```

Netmount kann standardmäßig mit NFS Mounts umgehen, aber wird nur von Portmap abhängen, wenn dieses zum aktuellen oder zum Boot Runlevel ergänzt wurde. Jeder Benutzer, der NFS Mounts nutzt, sollte portmap zum Default Runlevel hinzufügen. Dadurch wird Portmap als USE Abhängigkeit gesehen und vor netmount gestartet werden.

### Wichtig

Jeder Dienst in der **USE** Zeile **\*muss\*** im aktuellen oder im Boot Runlevel vorhanden sein, damit er als gültige **USE** Abhängigkeit anerkannt werden kann.

**USE** ist dadurch eine "schwache" Abhängigkeit.

### Notiz

Sollte ein Dienst in der **USE** Zeile nicht starten, so wird der aktuelle Dienst trotzdem gestartet werden, da der Dienst der **USE** Zeile nicht zwingend für den Start des aktuellen Dienstes nötig ist.

## 5. Kontrollieren der Reihenfolge ohne Abhängigkeiten

Sollte keine abhängige Verbindung zwischen zwei Diensten bestehen, es aber trotzdem nötig oder gewünscht sein, einen Dienst nach einem bestimmten anderen zu starten, können die **AFTER** und **BEFORE** Optionen genutzt werden.

### Notiz

Diese beiden Typen sind nur während dem Wechseln eines Runlevels gültig.

Optional können diese beiden Typen ein "\*" Wildcard für das integrieren aller anderen Dienste enthalten:

**Befehlsauflistung 4:** Ein Beispiel für AFTER

```
depend() {
    after *
}
```

Dies wird den Dienst **\*nach\*** allen anderen Diensten starten.

### 5.2 Die BEFORE Option

der aktuelle Dienst wird **\*vor\*** den in der **BEFORE** Zeile aufgelisteten Diensten gestartet.

**Befehlsauflistung 5:** Lässt foo vor dem Dienst bar starten (Auszug von foo)

```
depend() {
    before bar
}
```

## 5.3 Die AFTER Option

Der aktuelle Dienst wird **\*nach\*** den in der **AFTER** Zeile gelisteten Diensten gestartet.

**Befehlsauflistung 6:** Lässt bar nach foo starten (Auszug von bar)

```
depend() {
    after foo
}
```

## 6. Virtuelle Dienste

Dienste, wie die meisten Dinge in der heutigen Unix Welt, kommen in verschiedenen Farben und Geschmäckern daher. Normalerweise hat der Nutzer / Administrator die Wahl zu bestimmen welche genutzt werden.

Ein Beispiel dafür sind Systemlogger. Zum Zeitpunkt des Schreibens dieses Dokumentes, hat der Nutzer von Gentoo Linux die Auswahl aus vier verschiedenen. Alle Dienste, die einen Logger benötigen, können nicht alle vier mittels **NEED** Option anfordern. Die **USE** option ist zu schwach.

Dies ist der Punkt, an dem Virtuelle Dienste und die **PROVIDE** Option ins Spiel kommen.

### 6.2 Die PROVIDE Option

Die **PROVIDE** Option definiert einen virtuellen Dienst, welche andere Dienste mittels **NEED** und **USE** aufrufen können.

**Befehlsauflistung 7:** Sysklogd liefert logger

```
depend() {
    provide logger
}
```

### 6.3 Der Virtuelle Dienst LOGGER

**LOGGER** ist ein vordefinierter virtueller Dienst, welcher von allen Systemloggern geliefert wird. Er kann entweder mit **NEED** oder **USE** genutzt werden.

### 6.4 Der virtuelle Dienst NET

Der NET Dienst ist ein anderer virtueller Dienst, jedoch im Gegensatz zu **LOGGER** liefert er nicht einen eindeutigen Dienst.

#### Wichtig

Um den virtuellen Dienst **NET** zu unterstützen, muss ein Dienst:

- zum aktuellen oder Boot Runlevel angehören.
- ein "net" vorangestellt haben.
- der Teil nach "net" muss den Namen des eigentlichen Netzwerk Interfaces tragen (z.B. net.eth0 oder net.ppp1).

Für jeden gültigen net.\* Dienst, wird \$IFACE auf den Namen des Netzwerk Interfaces gesetzt(z.B. "eth0" für net.eth0).

## 7. Standard Kommandozeilen Optionen

Jeder Dienst kann mit einer der standard Optionen aufgerufen werden. All diese genannten sind bereits definiert, mit Ausnahme von **START** und **STOP**, welche dem Benutzer als Funktionen in seinem Rc-Script definieren sollte.

#### Wichtig

Die **start()** Funktion **muss** definiert werden.

#### Notiz

Die **stop()** Funktion ist nicht ganz so wichtig und kann weggelassen werden.

### Notiz

In der Regel wird der Benutzer nur **start()**, **stop()** und **restart()** definieren. Der Rest wird intern ablaufen und sollte in Ruhe gelassen werden.

**Befehlsauflistung 8:** Start des HTTPD Dienstes

```
# /etc/init.d/httpd start
```

### Notiz

Kommandozeilen Optionen können gestapelt, bzw. aufgereiht werden.

**Befehlsauflistung 9:** Pausieren / Starten von net.eth0

```
# /etc/init.d/net.eth0 pause start
```

## 7.2 Die START/STOP Option

**START**, startet den Dienst inklusive aller seiner Abhängigkeiten.

**STOP**, stoppt den Dienst inklusive aller Dienste, welche von ihm abhängig sind.

## 7.3 Die RESTART Option

Damit **RESTART** funktioniert, muss der Dienst vorher gestartet sein. Dadurch wird der Dienst und alle Dienste, die von ihm abhängen neu gestartet.

### Wichtig

Sollte eine eigene **restart()** Funktion definiert sein, sollte der Nutzer zum Starten und Stoppen "**svc\_start()**" und "**svc\_stop()**" verwenden.

### Notiz

Dies ist nötig, damit alle abhängigen Dienste richtig gehandhabt werden.

## 7.4 Die PAUSE Option

Dies wird den Dienst stoppen, nur im Gegensatz zu **STOP** wird kein abhängender Dienst angehalten.

## 7.5 Die ZAP Option

Setzt den Status eines Dienstes auf gestoppt.

### Notiz

Beachten Sie, dass hier kein Kommando aus der **stop()** Funktion ausgeführt wird. Deshalb sollte der Benutzer alle nötigen "Aufräumarbeiten" manuell durchführen.

## 7.6 Die INEED/NEEDSME Optionen

**INEED** listet alle Dienste auf, die vom aktuellen Dienst gebraucht werden.

**NEEDSME** listet alle Dienste auf, die den aktuellen Dienst brauchen.

## 7.7 Die IUSE/USESME Optionen

**IUSE** listet alle Dienste auf, die der aktuelle Dienst benutzt (siehe USE Option / Typ weiter oben).

**USESME** listet alle Dienste auf, die den aktuellen Dienst nutzen (siehe USE Option / Typ weiter oben).

## 7.8 Die BROKEN Option

Diese listet alle fehlenden Dienste (falls es welche gibt) auf, die der aktuelle Dienst braucht.

## 8. Eigene Kommandozeilen Optionen ergänzen

Eigene Kommandozeilen Optionen zu ergänzen ist relativ einfach. Eine Funktion mit dem Namen der Option muss im Rc-Script definiert werden und zur **\$opts** Variable ergänzt werden, siehe unten:

**Befehlsauflistung 10:** foo als eigene Option

```
opts="{opts} foo"
```

```
foo() {  
    .....  
}
```

## 9. Konfiguration

Konfiguration sollte generell durch Umgebungsvariablen erfolgen. Diese sollten *\*nicht\** im Rc-Script definiert werden, sondern in einer der drei möglichen Konfigurations Dateien.

Eine, die speziell für das Rc-Script da ist und zwei globale Konfigurations Dateien:

**Befehlsauflistung 11:** Konfigurations Dateien für Rc-Scripte

```
/etc/conf.d/<name of rc-script>  
/etc/conf.d/basic  
/etc/rc.conf
```

### Notiz

Diese drei werden in der aufgelisteten Reihenfolge eingelesen.

### Wichtig

Alle **NET** Dienste lesen also `/etc/conf.d/net` aus.

## 10. Utilities und hilfreiche Scripts

### 10.1 Das rc-update Utility

Rc-update ist das Hauptwerkzeug, um Dienste zu einem bestimmten Runlevel zu ergänzen oder von einem solchen zu entfernen. Es wird darüber hinaus auch "depscan" aufrufen, um den Abhängigkeits Cache zu aktualisieren.

**Befehlsauflistung 12:** Metalog zum standard Runlevel ergänzen

```
# rc-update add metalog default
```

**Befehlsauflistung 13:** Metalog vom standard Runlevel entfernen

```
# rc-update del metalog default
```

### Notiz

Um mehr Hilfe zu erhalten, rufen Sie einfach rc-update ohne weitere Argumente auf

### 10.2 Das depscan.sh Helfer Script

Um eine vollständige Dokumentation zu bieten, wird depscan.sh hier erwähnt. Es wird genutzt, um einen Abhängigkeits Cache zu erstellen, dieser ist eigentlich eine Aufzeichnung der Abhängigkeiten zwischen den Diensten.

Es sollte immer dann gestartet werden, wenn ein neues Rc-Script nach `/etc/init.d/` ergänzt wird. Da jedoch rc-update dieses Script automatisch aufruft, wird der normale Benutzer normalerweise nicht mit depscan in Berührung kommen.



## >> Gentoo Linux Kernel Guide

[Bitte Kapitel auswählen] 

### 1. Einleitung

So wie bei Allem in Gentoo-Linux ist es auch das Ziel des Gentoo-Kernel-Teams Ihnen, dem Anwender, so viele Wahlmöglichkeiten wie möglich zu geben. Wenn man sich die Ausgabe von *emerge -s sources* ansieht, wird deutlich, wie viele Kernel bei Gentoo zur Auswahl stehen. In diesem Dokument möchten wir einen kurzen Überblick über die Ziele der einzelnen Patchesätze geben die wir entwerfen und außerdem die anderen Kernelquellen erklären, die wir anbieten.

### 2. Die Auswahlmöglichkeiten, Teil I

#### 2.1 gentoo-sources

Für die meisten Anwender sind die empfehlenswertesten Kernelquellen die *gentoo-sources*. Das *gentoo-sources* Paket enthält speziell auf Leistung getrimmte Kernelpatches die bestimmte Aufgaben wie z.B. Kompilieren, gleichzeitiges Musikhören und im Web surfen optimieren sollen. Viele von Ihnen haben möglicherweise noch nie ein System benutzt, bei dem man gleichzeitig kompilieren und die tägliche Arbeit erledigen kann.

Sie werden vielleicht feststellen, dass die Benutzung der *vanilla-sources* (den offiziellen Kernelquellen von <http://www.kernel.org>) dazu führt, dass normale Aufgaben wie Musikhören, Mausbewegungen und ähnliches "Sprünge" aufweisen können.

Die *gentoo-sources* enthalten ein erweitertes ACPI-Subsystem und basieren auf Con Kolivas High-Performance Kernelpatches (den sog. *ck-sources*). Wir unterstützen außerdem grSecurity (einen Satz von sicherheitsbezogenen Patches mit Unterstützung für ACLs), EVMS(2) (ein hochflexibles Speicherverwaltungs-System mit einfacher Partitionsgrößenveränderung), JFS (ein Hochleistungs-Dateisystem von IBM), die neuesten NTFS-Treiber und vieles, vieles mehr.

Da die *gentoo-sources* auf Höchstleistung ausgelegt sind, stellen sie außerdem eine gute Grundlage für Spiele dar.

Die folgenden USE-Flags können zur Auswahl von optionalen Patches gesetzt werden:

#### Flag Beschreibung

aavm Andrea Arcangeli's Speicherverwaltung nutzen

evms2EVMS 2.0.1 anstelle von EVMS 1.2.1 nutzen

crypt Kryptographie-Patches benutzen

usagi USAGI beibehalten, superfreeswan, patch-int und loop-jari nicht beibehalten

#### 2.2 vanilla-sources

Die nächsten Kernelquellen mit denen Sie als Linux-Anwender vielleicht vertraut sind, sind die *vanilla-sources*. Wie ich vorher schon kurz erwähnt habe, sind dies die offiziellen Kernelquellen von [http://www.kernel.org/](http://www.kernel.org). Diese Quellen werden (entgegen dem weit verbreitetem Glauben) nicht von Linus Torvalds selbst, sondern von Marcelo Tosatti verwaltet. Linus ist der "Chef" der aktiven Kernelentwicklung und da er eben nur eine Person ist, übergibt er die Verwaltung des stabilen Kernel-Zweigs an Personen, denen er dies zutraut. So ist zum Beispiel Alan Cox Verwalter der Kernel-Serie 2.2 und Marcelo Tosatti wurde eben Verwalter der Kernel-Serie 2.4. Dies ist exakt das, worauf alle Patchesätze der Serie 2.4 basieren. Marcelo hat bei der Verwaltung hervorragende Arbeit geleistet und man kann auf diesen Kernel in Sachen Stabilität, Aktualität (betrifft nicht noch in der Entwicklung befindliche Teile) und Hardwareunterstützung wirklich zählen.

Die *vanilla-sources* sind möglicherweise die stabilsten aller verfügbaren Kernelquellen, da sie auch die am meisten getesteten sind und alle anderen Kernelquellen auf ihnen basieren. Wenn Sie keines der von den anderen Kernels mitgebrachten Extras brauchen, sollten Sie sich für die *vanilla-sources* entscheiden.

#### 2.3 gs-sources

Für Anwender, die Arbeitsplatzleistung gegenüber Verlässlichkeit und Hardwareunterstützung bevorzugen, haben wir die *gs-sources* im Angebot. GS steht für Gentoo Stable (wie kreativ, gell?).

Dieser Patch-Satz ist im Hinblick auf Unterstützung von neuester Hardware erweitert und getestet und stellt sicher, dass kritische Server dann verfügbar sind, wenn sie benötigt werden. Dieser Kernel lässt einige der überaggressiven in den *gentoo-sources* verfügbaren Leistungs-Tuning-Patches außen vor, aber die hervorragende Leistung die man von den Vanilla Kernels kennt und liebt sind immer noch enthalten. Dort wo es ohne Stabilitätseinbußen möglich ist, fügen wir serverbezogene Patches dem Kernel hinzu.

Dieser Kernel bietet Unterstützung für das neueste ACPI Subsystem, EVMS, ECC (wird für Hochverfügbarkeitskernel benötigt), verschlüsselte Loopbackgeräte, NTFS, Win4Lin und XFS. Er enthält außerdem Aktualisierungen für IDE, ext3, diverse Netzwerkkarten und noch einige andere Patches.

Mit anderen Worten sind diese Quellen perfekt für Server und Hochverfügbarkeitssysteme.

Die folgenden USE-Flags können zur Auswahl von optionalen Patches gesetzt werden:

#### **FlagsBeschreibung**

crypt Kryptographie-Patches benutzen

## **2.4 gentoo-test-sources**

*gentoo-test-sources* werden nach viel Testarbeit und Qualitätssicherung der Nachfolger der *gentoo-sources* werden. Patches für die *gentoo-sources* werden zu Testzwecken zuerst an den *gentoo-test-sources* angewendet. Wenn Sie also die Leistung der *gentoo-sources* zusammen mit den neuesten Patches haben wollen, dann benutzen sie die *gentoo-test-sources*.

Die folgenden USE-Flags können zur Auswahl von optionalen Patches gesetzt werden:

#### **Flag Beschreibung**

aavm Andrea Arcangeli's Speicherverwaltung nutzen

evms2EVMS 2.0.1 anstelle von EVMS 1.2.1 nutzen

crypt Kryptographie-Patches benutzen

usagi USAGI beibehalten, superfreeswan, patch-int und loop-jari nicht beibehalten

## **2.5 hardened-sources**

Die *hardened-sources* enthalten neben Patches für die verschiedenen Unterprojekte von Gentoo Hardened auch einige andere Stabilitäts- und Sicherheitserweiterungen. Besuchen Sie <http://www.gentoo.org/proj/en/hardened/> für weitere Informationen.

Die folgenden USE-Flags können zur Auswahl von optionalen Patches gesetzt werden:

#### **Flags Beschreibung**

selinuxErsetze grSecurity mit Unterstützung für SELinux

## **2.6 xfs-sources**

Die *xfs-sources* enthalten Unterstützung für EVMS, ACPI, grSecurity und - was Sie sich vielleicht schon gedacht haben - die neuesten Patches für die Unterstützung von XFS. Falls Sie es wissen möchten: die Gentoo LiveCD benutzt die *xfs-sources* :-).

Mehr Informationen über XFS gibt es bei <http://oss.sgi.com/projects/xfs/>.

Die folgenden USE-Flags können zur Auswahl von optionalen Patches gesetzt werden:

#### **FlagsBeschreibung**

crypt Kryptographie-Patches benutzen

## **2.7 Architektur-spezifische Kernel**

*alpha-sources*, *arm-sources*, *hppa-sources*, *mips-sources*, *ppc-sources* und *sparc-sources* sind - wie der Name vermuten lässt - für die bestmögliche Lauffähigkeit auf den entsprechenden Plattformen angepasst. Sie enthalten ausserdem einige der oben und unten genannten Hardware- und Feature-Patchsätze.

## **2.8 ppc-sources-benh**

Die *ppc-sources-benh* Ebuilds bieten zusätzliche Hardwareunterstützung für die *ppc-sources*-Kernel. Sie sind ein wenig experimenteller als die *ppc-sources*.

## 2.9 ppc-sources-crypto

Die *ppc-sources-crypto* Ebuilds bieten Crypto-API Unterstützung für den Gentoo Linux PPC-Kernel. Mehr Informationen über CryptoAPI erhalten Sie unter <http://www.kernel.org/about/>.

## 2.10 ppc-sources-dev

Die *ppc-sources-dev*-Pakete bieten die Entwicklerquellen für die *ppc-sources*. Jeder Patch der in die *ppc-sources* einfließen soll muß seine Funktionsfähigkeit erst in den *ppc-sources-dev* beweisen.

## 2.11 compaq-sources

Die *compaq-sources* bieten die Kernel Quellen von RedHat für Alpha, die von Compaq verwaltet werden.

## 3. Die Auswahlmöglichkeiten, Teil II

Nun werde ich versuchen, kurz einige der anderen in *sys-kernel/\*-sources* befindlichen Kernelquellen, die Sie bei *emerge -s sources* vorbeihuschen gesehen haben, zu erklären. Lassen Sie sie uns in alphabetischer Reihenfolge durchgehen.

### 3.2 aa-sources

Zu allererst haben wir die *aa-sources*. Hierbei handelt es sich um Andrea Arcangelis Patch-Satz. Sein Kernelpatch hat einige der am aggressivsten getunten VM (Virtual Memory) Patches die bekannt sind. Als ich das letzte Mal nachgesehen habe, enthielten Sie auch SGIs XFS-Dateisystem und den O(1)-Scheduler von Ingo Molar (welcher der Default-Scheduler in Linux 2.6 werden wird).

Sie bieten außerdem Unterstützung für User-Mode-Linux (siehe auch unseren [UML Guide \(englisch\)](#) für weitere Informationen) und den neuesten TUX-Webserver (ein im Kernel befindlicher Webserver).

Wenn Sie bei anderen Kernels Probleme mit der Speicherverwaltung haben, dann könnten die *aa-sources* die Lösung des Problems darstellen. Wenn Sie die Linux Speicherverwaltung für Ihr System optimieren wollen, dann sollten die *aa-sources* **auf jeden Fall** das sein, was Sie brauchen.

Besuchen Sie <http://www.kernel.org/pub/linux/kernel/people/andrea/kernels/v2.4> um mehr Informationen über alle Patches in diesen Kernelquellen zu erhalten.

### 3.3 ac-sources

Als Nächstes haben wie die *ac-sources*. Es handelt sich hierbei um Alan Cox Patchsatz für die Kernel Serie 2.4. In diesem Patchsatz finden Sie den O(1) Scheduler, die neuesten Aktualisierungen für das IDE-System von 2.4 und viele andere Patches die auf eine mögliche Integration in die Kernel Serie 2.4 warten.

Dieser Kernel ist dafür bekannt, anständige Unterstützung für einige zusätzliche Hardware-Komponenten zu haben und kann ein Kandidat für Sie sein, wenn Sie einen stabilen aber weniger konservativen Kernel als die *vanilla-sources* möchten.

Besuchen Sie <http://www.kernel.org/pub/linux/kernel/people/alan/linux-2.4/> um einen Überblick über Alan Cox Arbeit zu bekommen.

### 3.4 ck-sources

Die *ck-sources* sind Con Kolivas Kernelpatch-Satz. Dieser Kernel ist **höchst** optimiert für Desktopleistung mit dem Nachteil von Durchsatz und ein paar Fähigkeiten des Schedulers bei der Priorisierung von Anwendungen. Con Kolivas "benchmarkt" Kernel, um die bestmögliche Kombination der einzelnen Features für die Benutzung auf Desktops zu finden. Besuchen Sie <http://kernel.kolivas.org>, um mehr Informationen über Con und seine Patches zu bekommen.

### 3.5 development-sources

Die *development-sources* sind die Kernel des aktuellen Entwicklerzweiges. Dies ist der Zweig des Linux-Kernels, den Linus Torvalds selbst verwaltet. Man kann sich diesen Kernel als einen Spielplatz vorstellen, wo alle Features für den nächsten stabilen Kernelzweig angepasst, weiterentwickelt und getestet werden bevor sie an die übrigen Linux-Anwender weitergegeben werden.

Wenn Sie die neuesten Treiber, Erweiterungen und das neueste experimentelle Kernsystem möchten, dann ist dieser Kernel das was Sie haben wollen. Bedenken Sie aber bitte, das diese Kernelquellen **äußerst** experimentell sind und es **nicht empfehlenswert** ist, sie auf einem unbedingt benötigten System einzusetzen.

### Warnung

Wir weisen Sie hiermit darauf hin, dass Gentoo-Linux keine Fähigkeiten der *development-sources* oder der darauf basierenden sources unterstützt, da sich diese zu oft ändern und evtl. Schäden anrichten können.

## 3.6 gaming-sources

Die *gaming-sources* basieren auf den *ck-sources* und sind folglich auf Höchstleistung getrimmt. Sie enthalten außerdem die neuesten Patches für Multimedia-Hardware (Grafikkarten, Soundkarten, usw.).

Wenn Sie ein Hardcore-Gamer sind, dann ist dies der Kernel Ihrer Wahl.

## 3.7 mm-sources

Die *mm-sources* basieren auf den *development-sources* und enthalten Andrew Mortons Patchsatz. Dieser enthält einige andere Patches, wie z.B. erweiterte Attribute für ext2/3 und ACLs (Access Control Lists), Page Table Sharing, den Orlov Allocator, nicht-lineares Mapping-Verhalten, usw.

Wenn Sie sich wirklich auf Messers Schneide bewegen wollen und der Meinung sind die *development-sources* wären nur etwas für Memmen, dann versuchen Sie die *mm-sources*.

## 3.8 mosix-sources

Die *mosix-sources* sind speziell für MOSIX-Operationen bei Clustern angepasst. Ein Cluster ist ein Satz von Knoten (=PCs) mit spezieller Software, die es Ihnen erlaubt Aufgaben verteilt zu erledigen. Mit Clustern entfällt die Notwendigkeit für Supercomputer, um grössere Aufgaben zu erledigen. Wenn Sie weitere Informationen möchten, dann besuchen Sie bitte <http://www.mosix.org>.

## 3.9 openmosix-sources

Die *openmosix-sources* sind für die Unterstützung von openMosix Systemen optimiert (dasselbe wie MOSIX, aber Open Source). Für weitere Informationen besuchen Sie bitte <http://www.openmosix.org>.

## 3.10 redhat-sources

Die *redhat-sources* sind - wie der Name schon vermuten lässt - die Quellen des RedHat Kernels. Dank der Vorteile von Open Source kann jeder an den Früchten der Arbeit, die die RedHat-Ingenieure in ihren Kernel investieren, teilhaben. Wir von Gentoo bieten ein entsprechendes Ebuild an, damit Sie auf möglichst einfachem Wege in den Genuß kommen.

## 3.11 rsbac-sources

Die *rsbac-sources* enthalten die Patches von <http://www.rsbac.org>. RSBAC steht für **Rule Set Based Access Control**. Diese Kernelpatches erlauben Ihnen, Benutzer anhand von Regeln anstatt Kombinationen von UID-/GID-Berechtigungen zu authentifizieren.

## 3.12 selinux-sources

Die *selinux-sources* von <http://www.nsa.gov/selinux> sind Patches zur Unterstützung der LSM (Linux Security Modules) und Flask Security Architecture.

## 3.13 usermode-sources

Die *usermode-sources* sind die Quellen, die Sie für einen Usermode-Kernel brauchen. Dies ist ein spezieller Kernel, der dafür entwickelt wurde, um Linux innerhalb von Linux innerhalb von Linux... auszuführen. User-Mode Linux ist zu Testzwecke und zur Unterstützung von

virtuellen Servern vorgesehen. Für weitere Informationen über diese phantastische Hommage an die Skalierbarkeit von Linux, besuchen Sie bitte <http://user-mode-linux.sourceforge.net>.

Für weitere Informationen über UML und Gentoo, lesen Sie bitte [den Gentoo UML Guide](#).

### 3.14 win4lin-sources

Die *win4lin-sources* sind zur Unterstützung der Win4Lin Werkzeuge speziell angepasst. Diese erlauben Linux-Anwendern viele Anwendungen, die eigentlich für Microsoft Windows entwickelt wurden mit fast normaler Geschwindigkeit auszuführen. Weitere Informationen gibt es unter <http://www.netraverse.com/>.

### 3.15 wolk-sources

Die *wolk-sources* enthalten den **Working Overloaded Linux Kernel** von <http://sourceforge.net/projects/wolk>. Dieser Kernel enthält viele verschiedene Patches, die alle mit größter Vorsicht in den Kernel eingepasst wurden. Dies erlaubt Ihnen als Anwender fast jeden davon entweder bei der Kompilierung des Kernels zu berücksichtigen oder dies eben zu lassen - der Kernel selber wird bei fast jeder Kombination von einkompilierten Patches funktionieren.

Wenn Sie eine bestimmte Kombination von Patches brauchen, die Sie nicht in anderen Kernelquellen finden können, dann ist WOLK sicherlich einen Versuch wert.



## >> Gentoo Linux Sicherheitsleitfaden

[Bitte Kapitel auswählen]

### 1. Einleitung

Dieser Guide ist für Leute gedacht, die Gentoo Linux in einer Server-Basierten Umgebung einsetzen, oder das Gefühl haben mehr Sicherheit zu brauchen.

#### Notiz

Wenn Sie nach dem Lesen dieses Guides an noch mehr Sicherheit interessiert sind, sollten Sie einen Blick auf das [Hardened Gentoo Projekt](#) werfen

### 2. Was vor der Installation beachtet werden sollte

#### 2.1 Physische Sicherheit

Egal wieviele Sicherheitsmassnahmen Sie integrieren, sie können leicht umgangen werden, wenn der Hacker direkten Zugriff auf Ihre Maschine hat. Stellen Sie sicher, dass Ihre Hardware nicht einfach so zugänglich ist. Zum Beispiel könnten Sie Ihre Maschine in einem speziellen Serverraum einschliessen. Die Gehäuse zu versiegeln ist auch eine gute Idee. Um das höchste Niveau an Sicherheit zu erreichen, können Sie Ihr BIOS so einstellen, dass es nur von der Festplatte bootet. Deaktivieren Sie auch das Booten von Diskette und von CD-ROM. Für den Übervorsichtigen kann es auch sinnvoll sein, das BIOS-Passwort zu aktivieren. BIOS-Passwörter sind auch eine gute Idee für Notebook-Benutzer.

#### 2.2 Dämon/Dienst Planung

Dokumentieren Sie, welche Dienste die Maschine anbieten soll oder anbieten darf. Dies wird Ihnen helfen ein besseres Partitionsschema für das System zu erstellen. Es kann auch das Aufspüren von Eindringlingen erheblich erleichtern. Natürlich brauchen Sie das nicht zu dokumentieren, wenn Sie nur einen oder ein paar Computer benutzen und Sie der einzige Nutzer sind. Zum Beispiel: Wenn die Maschine als Firewall agieren soll sollten auf der Maschine **keine** Dienste ausser vielleicht sshd laufen.

Dokumentieren Sie dies und die aktuelle Version von SSH - es wird Ihnen helfen, das zu aktualisierende System wiederzufinden - für den Fall, dass jemand ein Sicherheitsloch in sshd findet. Dies wird Ihnen auch dabei helfen festzulegen, wer Zugriff auf das System haben sollte.

#### 2.3 Partitions-Schemata

Goldene Regeln:

- Jedes Verzeichnis auf das ein Benutzer Schreibrechte haben muss (`/home` und `/tmp`, `/var`) sollte auf einer separaten Partition liegen und Disk-Quotas benutzen. Portage benutzt `/var/tmp` zum kompilieren, folglich muss diese Partition gross sein. Dies reduziert das Risiko, daß ein Benutzer "/" komplett füllen kann.
- Jedes Verzeichnis, in das nicht in der Distribution enthaltene Pakete installiert werden sollen, sollten auf einer separaten Partition liegen. Nach dem [Filesystem Hierarchy Standard](#) ist dies `/opt` oder `/usr/local`. Wenn diese separate Partitionen sind, bleiben Sie bei einer eventuellen Neuinstallation des Systems bestehen.
- Versuchen sie, statische Daten in eine eigene Partition verschieben und diese Partition nur lesbar einhängen. Wenn sie wirklich übervorsichtig sind, dann könnten Sie statische Daten auch auf einem nur lesbaren Medium speichern - zum Beispiel einer CD-ROM.

#### 2.4 Der Benutzer root

Der Benutzer 'root' ist der mächtigste Benutzer im System und sollte nie für irgendetwas Anderes als administrative Aufgaben eingesetzt werden. Wenn ein Angreifer root-Zugang erreicht, dann können Sie Ihrem System nicht mehr länger trauen - Sie haben dann keine andere Wahl, als neu zu installieren.

Goldene Regeln bezüglich 'root'

- Erstellen Sie immer einen Benutzer für die tägliche Arbeit. Wenn dieser Benutzer root-Zugang benötigt, dann fügen Sie diesen Benutzer zur Gruppe wheel hinzu. Dies erlaubt einem normalen benutzer "nach root zu su'en".
- Lassen Sie X oder irgendeine andere Benutzeranwendung niemals als root laufen.

- Benutzen Sie immer absolute Pfadangaben, wenn Sie als root angemeldet sind. Sonst ist es möglich, root eine andere Anwendung ausführen zu lassen, als er denkt (wenn zum Beispiel jemand an PATH manipuliert hat und root ohne `su - su'ed`). Dann wird root den Pfad des Nutzers benutzen.
- Wenn ein Benutzer nur ein paar Kommandos, anstatt von allen, die root benutzen kann, dann überlegen Sie, vielleicht auf sudo zurückzugreifen - aber seien Sie vorsichtig damit.
- Verlassen Sie nie den Terminal, wenn root angemeldet ist!

Gentoo hat einen allgemeinen Schutz gegen normale Benutzer, die versuchen su einzusetzen. Die Standardeinstellung von PAM besagt, dass ein Benutzer in der Gruppe wheel sein muss, um su benutzen zu dürfen.

## 2.5 Sicherheitsrichtlinien

Es gibt verschiedene Gründe, weshalb Sicherheitsrichtlinien benötigt werden.

- Sie können nicht behaupten ein sicheres Netzwerk zu haben, ohne jemals definiert zu haben, was sicher ist.
- Es ist fast unmöglich potentielle Angreifer zu erwischen, Netzwerkprobleme zu lösen, oder Prüfungen zu dirigieren ohne den Netzwerkverkehr abzuhören oder in private Home-Verzeichnisse zu sehen. Aber dieses Reinhören ohne Erlaubnis des Nutzers ist in den meisten Ländern illegal. Und da 60% aller Angriffe von innerhalb eines Unternehmens kommen ist es wichtig, dass Sie die Augen offen halten.
- Sie können von Ihren Anwendern nicht erwarten, dass sie sich Gedanken über Sicherheit machen, wenn Sie niemals erklärt haben, warum es wichtig ist oder wie sie sich selbst und ihre Kollegen schützen sollten.
- Gute Richtlinien und Netzwerkdokumentation zahlen sich immer aus - egal wie.
- Die Polizei oder die Staatliche Kriminalbehörde kann Ihnen nicht helfen den Angreifer dingfest zu machen, wenn diese nicht wissen wie Ihre Netzwerkkonfiguration aussieht oder welche Dienste Sie anbieten.
- Was werden Sie tun, wenn es einen Angriff gab? Sie müssen definieren, was Sie tun würden und wen Sie informieren würden. Würden Sie bei jeder Gelegenheit die Polizei oder ein CERT-Team anrufen? Die würden Sie nicht ernst nehmen.

Dies sollte eigentlich darlegen warum es wichtig ist Richtlinien auf Systemen mit mehr als einem Benutzer festzulegen und warum es wichtig ist die Anwender zu erziehen.

Eine Richtlinie ist ein Dokument (oder mehrere Dokumente) mit Antworten auf die Fragen "wer", "wo", "warum" und "was". Jeder Anwender in Ihrem System/Netzwerk sollte es lesen, verstehen und unterschreiben. Es ist wichtig, daß Sie sich die Zeit nehmen den Anwendern beim Verstehen der Richtlinie zu helfen: dem Grund, weshalb diese Richtlinie unterschrieben werden muß und was passiert, wenn sie direkt gegen die Richtlinie verstossen (dies sollte in der Richtlinie aufgeführt sein). Dies sollte mindestens einmal im Jahr wiederholt werden, da sich die Richtlinie ändern kann, aber auch um die Anwender immer wieder aufs Neue daran zu erinnern.

### Notiz

Erstellen Sie Richtlinien, die einfach zu lesen und in jedem Zusammenhang sehr präzise sind.

Eine Sicherheitsrichtlinie sollte mindestens die folgenden Punkte beinhalten:

#### Akzeptable Anwendung

- Bildschirmschoner
- Behandlung von Kennwörtern
- Herunterladen von Programmen
- Wissen darüber, ob diese überwacht werden
- Benutzung von Antiviren-Software
- etc.

#### Behandlung von sensitiven Daten (jegliche Schriftliche Form, Papier oder Digital)

- Sauberer Schreibtisch und verschlossene, vertrauliche Informationen
- PC herunterfahren vorm Verlassen
- Benutzung von Verschlüsselung
- Behandlung von Schlüsseln für vertraute Mitarbeiter
- Behandlung von vertraulichem Material auf Reisen

#### Behandlung der Computerausstattung auf Reisen

- Behandlung des Laptops auf Reisen und bei Hotelaufenthalten

Die Richtlinie für die IT-Abteilung kann sich von der für die normalen Angestellten leicht unterscheiden.

Die Sicherheitsrichtlinie kann riesig werden und wichtige Informationen können leicht vergessen gehen. Die Richtlinie für die IT-Abteilung kann Informationen enthalten, die gegenüber den normalen Benutzern als vertraulich gelten. Somit ist es sinnvoll, sich in kleinen Richtlinien fortzubewegen: Richtlinie für akzeptable Bedienung, Richtlinie für Passwörter, für E-Mail und für Fernzugriff.

Beispiele für Richtlinien können beim [The SANS Security Policy Project](#) gefunden werden. Wenn Sie ein kleines Netzwerk haben und diese Richtlinie für zu gross halten, dann sollten sie einen Blick auf das [RFC2196](#) werfen, das ein Sicherheitshandbuch darstellt

## 3. Die Sicherheit nach/während der Installation anziehen

### 3.1 USE Flags

Die `/etc/make.conf`-Datei enthält die benutzerdefinierten und die `/etc/make.profile/make.defaults`-Datei die Standard USE Flags. Die wichtigen Flags für diesen Guide sind *PAM* (Pluggable Authentication Module), *tcpd* (TCP Wrapper) und *SSL* (Secure Socket Layer). Diese sind in den Standard USE Flags enthalten.

### 3.2 GRUB Passwort

Grub unterstützt 2 verschiedene Wege für Passwortkontrolle in seiner Konfigurationsdatei(`/boot/grub/grub.conf`): Zum einen normalen Text und zum anderen md5+salt Verschlüsselung.

Fügen Sie ein Passwort in `/boot/grub/grub.conf` ein.

**Befehlsauflistung 1:** `/boot/grub/grub.conf`

```
timeout 5
password changeme
```

Dies wird das Passwort **changeme** hinzufügen; wenn kein Passwort eingegeben ist, wird die Standard-Boot-Einstellung genommen.

Sollte ein md5-Passwort genommen werden, dann müssen Sie das Passwort ins Crypt-Format konvertieren (*man crypt*), daß das selbe Format wie die Shadow-Datei hat. Zum Beispiel könnte das verschlüsselte Passwort **changeme** so aussehen: **\$1\$T7/dgdIJ\$dJM.n2wZ8RG.oEiIOwJUUs.**

Oder das Passwort direkt in der GRUB Shell konvertieren:

**Befehlsauflistung 2:** md5crypt in der GRUB Shell

```
#!/sbin/grub

GRUB version 0.92 (640K lower / 3072K upper memory)

[ Minimal BASH-like line editing is supported. For the first word, TAB
  lists possible command completions. Anywhere else TAB lists the possible
  completions of a device/filename. ]

grub> md5crypt

Password: *****
// Typed changeme
Encrypted: $1$T7/dgdIJ$dJM.n2wZ8RG.oEiIOwJUUs.

grub> quit
```

Dann kopieren und fügen Sie das Passwort in `/boot/grub/grub.conf` ein.

**Befehlsauflistung 3:** `/boot/grub/grub.conf`

```
timeout 5
password --md5 $1$T7/dgdIJ$dJM.n2wZ8RG.oEiIOwJUUs.
```

Der Zeitablauf von 5 Sekunden wird sinnvoll, wenn das System fernbedient wird und bei einem Neustart ohne Tastatureingaben auskommen muss. Mehr Informationen über Grub-Passwörter können Sie bekommen, wenn Sie `info grub` ausführen.

### 3.3 LILO Passwort

LILO unterstützt auch zwei Arten des Behandelns von Passwörtern : Global und Imagerelativ -- beide in Klartext.

Das globale wird am Anfang der Konfigurationsdatei gesetzt:

**Befehlsauflistung 4:** /etc/lilo.conf

```
password=changeme
restricted
delay=3
```

Im anderen Fall fügen Sie es einfach beim entsprechenden Image hinzu.

**Befehlsauflistung 5:** /etc/lilo.conf

```
image=/boot/bzImage
  read-only
  password=changeme
  restricted
```

Wenn die *restricted*-Option nicht angegeben wurde, dann wird jedes Mal nach einem Passwort gefragt.

Um die Änderungen an lilo.conf zu übernehmen, müssen Sie */sbin/lilo* ausführen.

### 3.4 Einschränkung der Konsolenbenutzung.

*/etc/securetty* enthält Terminaltypen die es Ihnen ermöglichen/erlauben festzulegen, von welchen *TTY* Geräten aus root sich einloggen darf.

Wir empfehlen, dass sie alle Zeilen bis auf *vc/1* auskommentieren. Dies stellt sicher, dass sich root nur einmal und nur an einem Terminal einloggen kann.

#### Notiz

Benutzer in der wheel Gruppe können weiterhin auf anderen Konsolen per *su* - root werden.

**Befehlsauflistung 6:** /etc/securetty

```
vc/1
```

## 4. Mehr Protokolle (Logs)

Zusätzliche Protokolle sollten hinzugefügt werden um Warnungen oder Fehler aufzuspüren, die vor einem momentanen oder bereits durchgeführten Angriff warnen könnten. Angreifer beobachten ein Netzwerk oder durchsuchen dies oft, bevor sie angreifen.

Es ist auch unersetzlich, dass die Protokolldateien einfach zu lesen und zu verwalten sind. Gentoo Linux gibt ihnen die Möglichkeit bei der Installation zwischen drei verschiedenen Protokollierungsprogrammen zu wählen.

### 4.2 Loggen: Syslogd

Syslogd ist das gängigste Protokollierungsprogramm für Linux und Unix. Es beinhaltet keine Protokollrotation. Diese Eigenschaft wird durch das Verwenden von */usr/sbin/logrotate* in einem Cron Job und korrekt konfigurierten Einstellungen in */etc/logrotate.conf* übernommen. Wie oft die Protokollrotation stattfinden sollte hängt von der Systembelastung ab.

Hierunter sehen Sie die Standard Konfiguration */syslog.conf* mit einigen zusätzlichen Features. Wir haben die *cron* und *tty* Zeilen unkommentiert und eine Remote Logging Server hinzugefügt. Um die Sicherheit weiter zu erhöhen, können Sie Logs an zwei Orten schreiben lassen.

**Befehlsauflistung 7:** /etc/syslog.conf

```
# /etc/syslog.conf      Configuration file for syslogd.
#
#                       For more information see syslog.conf(5)
#                       manpage.
#                       This is from Debian, we are using it for now
#                       Daniel Robbins, 5/15/99
#
```

```

# First some standard logfiles.  Log by facility.
#

auth,authpriv.*                /var/log/auth.log
*.*;auth,authpriv.none         -/var/log/syslog
cron.*                          /var/log/cron.log
daemon.*                        -/var/log/daemon.log
kern.*                          -/var/log/kern.log
lpr.*                           -/var/log/lpr.log
mail.*                          /var/log/mail.log
user.*                          -/var/log/user.log
uucp.*                          -/var/log/uucp.log
local6.debug                    /var/log/imapd.log

#
# Logging for the mail system. Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                       -/var/log/mail.info
mail.warn                       -/var/log/mail.warn
mail.err                        /var/log/mail.err

# Logging for INN news system
#
news.crit                       /var/log/news/news.crit
news.err                       /var/log/news/news.err
news.notice                    -/var/log/news/news.notice

#
# Some `catch-all' logfiles.
#
*.=debug;\
    auth,authpriv.none;\
    news.none;mail.none        -/var/log/debug
*.=info;*.=notice;*.=warn;\
    auth,authpriv.none;\
    cron,daemon.none;\
    mail,news.none            -/var/log/messages

#
# Emergencies and alerts are sent to everybody logged in.
#
*.emerg                         *
*.=alert                        *

#
# I like to have messages displayed on the console, but only on a virtual
# console I usually leave idle.
#
daemon,mail.*;\
    news.=crit;news.=err;news.=notice;\
    *.=debug;*.=info;\
    *.=notice;*.=warn         /dev/tty8

#Setup a remote logging server
*.*                             @logserver

# The named pipe /dev/xconsole is for the `xconsole' utility.  To use it,
# you must invoke `xconsole' with the `-file' option:
#
#   $ xconsole -file /dev/xconsole [...]
#
# NOTE: adjust the list below, or you'll go crazy if you have a reasonably
#       busy site..
#
#daemon.*,mail.*;\
#   news.crit;news.err;news.notice;\
#   *.=debug;*.=info;\
#   *.=notice;*.=warn         |/dev/xconsole

local2.*                        -/var/log/ppp.log

```

Der Angreifer wird höchstwahrscheinlich versuchen seine Spuren zu verwischen, indem er die Protokolldateien bearbeitet oder löscht. Sie können es für den Angreifer schwerer machen indem sie

kann nach Programmnamen protokollieren (wie syslogd) und beinhaltet reguläre Ausdrucksübereinstimmung und die Möglichkeit Kommandos auszuführen. Sehr gut um Handeln zu können, wenn nötig.

Die Standard Konfiguration ist zunächst ausreichend. Wenn Sie benachrichtigt werden wollen, wenn z.B. ein Anmeldevorgang fehlschlägt benutzen Sie eines der folgenden Skripte.

Für Postfix.

**Befehlsauflistung 8:** /usr/local/sbin/mail\_pwd\_failures.sh für Postfix

```
#!/bin/sh
echo "$3" | mail -s "Warning (program : $2)" root
```

Für Qmail:

**Befehlsauflistung 9:** /usr/local/sbin/mail\_pwd\_failures.sh für Qmail

```
#!/bin/sh
echo "To: root
Subject:Failure (Warning: $2)
$3
" | /var/qmail/bin/qmail-inject -f root
```

Denken Sie daran das Skript mit `/bin/chmod +x /usr/local/sbin/mail_pwd_failures.sh` ausführbar zu machen.

Unkommentieren Sie dann die Zeile unter "Password failures" in `/etc/metalog/metalog.conf` wie folgt:

**Befehlsauflistung 10:** /etc/metalog/metalog.conf

```
command = "/usr/local/sbin/mail_pwd_failures.sh"
```

## 4.4 Loggen: Syslog-ng

Syslog-ng enthält einige derselben Funktionen wie Syslog und Metalog mit einem kleinen Unterschied. Es ermöglicht die Filterung von Nachrichten basierend auf Level und Inhalt (wie Metalog), bietet entferntes Protokollieren (wie syslog) und kann Protokolle von syslogd verarbeiten. Sogar Streams von Solaris, schreiben an ein TTY, Ausführen von Programmen und die Nutzung als Protokollierungsserver sind möglich. Grundlegend ist dies das Beste aus beiden Protokollierern kombiniert mit einer erweiterten Konfiguration.

Eine klassische, leicht modifizierte Konfigurationsdatei.

**Befehlsauflistung 11:** /etc/syslog-ng/syslog-ng.conf

```
options { long_hostnames(off); sync(0); };

#Quelle von der das Log gelesen werden soll
source src { unix-stream("/dev/log"); internal(); };
source kernsrc { file("/proc/kmsg"); };

#Ziele festlegen
destination authlog { file("/var/log/auth.log"); };
destination syslog { file("/var/log/syslog"); };
destination cron { file("/var/log/cron.log"); };
destination daemon { file("/var/log/daemon.log"); };
destination kern { file("/var/log/kern.log"); };
destination lpr { file("/var/log/lpr.log"); };
destination user { file("/var/log/user.log"); };
destination mail { file("/var/log/mail.log"); };

destination mailinfo { file("/var/log/mail.info"); };
destination mailwarn { file("/var/log/mail.warn"); };
destination mailerr { file("/var/log/mail.err"); };

destination newscrit { file("/var/log/news/news.crit"); };
destination newserr { file("/var/log/news/news.err"); };
destination newsnotice { file("/var/log/news/news.notice"); };

destination debug { file("/var/log/debug"); };
destination messages { file("/var/log/messages"); };
```

```

filter f_daemon { facility(daemon); };
filter f_kern { facility(kern); };
filter f_lpr { facility(lpr); };
filter f_mail { facility(mail); };
filter f_user { facility(user); };
filter f_debug { not facility(auth, authpriv, news, mail); };
filter f_messages { level(info..warn)
    and not facility(auth, authpriv, mail, news); };
filter f_emergency { level(emerg); };

filter f_info { level(info); };
filter f_notice { level(notice); };
filter f_warn { level(warn); };
filter f_crit { level(crit); };
filter f_err { level(err); };
filter f_failed { match("failed"); };
filter f_denied { match("denied"); };

#Filter und Ziele verbinden
log { source(src); filter(f_authpriv); destination(authlog); };
log { source(src); filter(f_syslog); destination(syslog); };
log { source(src); filter(f_cron); destination(cron); };
log { source(src); filter(f_daemon); destination(daemon); };
log { source(kernsrc); filter(f_kern); destination(kern); };
log { source(src); filter(f_lpr); destination(lpr); };
log { source(src); filter(f_mail); destination(mail); };
log { source(src); filter(f_user); destination(user); };
log { source(src); filter(f_mail); filter(f_info); destination(mailinfo); };
log { source(src); filter(f_mail); filter(f_warn); destination(mailwarn); };
log { source(src); filter(f_mail); filter(f_err); destination(mailerr); };

log { source(src); filter(f_debug); destination(debug); };
log { source(src); filter(f_messages); destination(messages); };
log { source(src); filter(f_emergency); destination(console); };

#Standard-Log
log { source(src); destination(console_all); };

```

Sehr einfach zu konfigurieren, aber es ist auch sehr einfach etwas zu übersehen, da die Konfigurationsdatei riesig ist. Der Autor verspricht zudem noch einige zusätzliche Funktionen wie Verschlüsselung, Authentifizierung, Komprimierung und MAC (Mandatory Access Control) Kontrolle. Mit diesen Optionen wird es perfekt sein für Netzwerkprotokollierung, da der Angreifer die Protokolle nicht ausspionieren kann.

Syslog-ng hat auch noch andere Vorteile - es muss nicht als root laufen!

## 5. Partitionen mounten

Mountet man eine ext2, ext3 oder eine reiserfs Partition, so gibt es mehrere Optionen die man in `/etc/fstab` einfügen kann. Diese Optionen sind:

- *nosuid* - Ignoriert das SUID bit und behandelt es einfach wie eine normale Datei.
- *noexec* - Verhindert das Ausführen von Dateien von dieser Partition.
- *nodev* - Ignoriert Geräte.

Leider können diese Einstellungen leicht umgangen werden, indem man einen nicht-direkten Pfad ausführt. Jedoch wenn man `/tmp` auf *noexec* setzt, stoppt das etwa 99% aller Script-Kiddies, da deren Exploits so gestaltet sind dass sie direkt von `/tmp` ausgeführt werden.

### Befehlsauflistung 12: /etc/fstab

```

/dev/sda1 /boot ext2 noauto,noatime 1 1
/dev/sda2 none swap sw 0 0
/dev/sda3 / reiserfs notail,noatime 0 0
/dev/sda4 /tmp reiserfs notail,noatime,nodev,nosuid,noexec 0 0
/dev/sda5 /var reiserfs notail,noatime,nodev 0 0
/dev/sda6 /home reiserfs notail,noatime,nodev,nosuid 0 0
/dev/sda7 /usr reiserfs notail,noatime,nodev,ro 0 0
/dev/cdroms /cdrom0 /mnt/cdrom iso9660 noauto,ro 0 0
proc /proc proc defaults 0 0

```

## Warnung

Setzt man `/tmp` in `noexec` Modus, kann dies dazu führen, dass einige Scripts nicht richtig ausgeführt werden.

## Notiz

Plattenquotas werden im Kapitel [Quotas](#) behandelt

## Notiz

Beachten Sie dass ich `/var` weder in `noexec` noch in `nosuid` Modus setze, obwohl Dateien von diesem Mountpunkt normalerweise niemals ausgeführt werden. Der Grund dafür ist, dass `qmail` in `/var/qmail` installiert ist und berechtigt sein muss eine `suid`-Datei auszuführen und auf sie zuzugreifen. Ich setze `/usr` in `read-only` Modus, da ich hier nichts verändere solange ich Gentoo nicht aktualisiere. Dann mountete ich das Dateisystem erneut in `read-write` Modus, aktualisiere und mountete dann erneut in `read-only`.

## Notiz

Selbst wenn sie `qmail` nicht benutzen, braucht Gentoo trotzdem noch die Ausführberechtigung in `/var/tmp`, da dort `ebuilds` hergestellt werden. Jedoch kann hierfür ein alternativer Pfad eingerichtet werden, wenn Sie darauf bestehen `/var` im `noexec` Modus zu betreiben.

## 6. Einschränkungen für Benutzer/Gruppen

### 6.1 /etc/security/limits.conf

Die Kontrolle von Ressourcenbegrenzungen kann sehr effektiv sein, wenn es darum geht eine lokale DoS Attacke zu verhindern oder die maximal erlaubten Logins für eine Gruppe oder einen Benutzer zu handhaben.

**Befehlsauflistung 13:** `/etc/security/limits.conf`

```
*      soft core      0
*      hard core      0
*      hard nproc     15
*      hard rss       10000
*      -  maxlogins   2
@dev   hard core     100000
@dev   soft nproc    20
@dev   hard nproc    35
@dev   -  maxlogins  10
```

Wenn Sie dabei sind den Wert von `nproc` oder `maxlogins` gleich 0 zu setzen, sollten sie diesen Benutzer vielleicht lieber löschen. Das Beispiel oben setzt die Einstellungen für die Gruppe `dev` für Prozesse, Kerndateien und `maxlogins`. Der Rest erhält einen Standardwert.

## Notiz

`/etc/security/limits.conf` ist Teil des PAM Paketes und wird nur auf Pakete angewendet, die PAM benutzen.

### 6.2 /etc/limits

`/etc/limits` ist recht ähnlich zur Limit-Datei `/etc/security/limits.conf`. Der einzige Unterschied ist das Format und daß diese nur auf Benutzern oder Wild-Cards (aber keinen Gruppen) beruht. Werfen wir einen Blick auf die Konfiguration:

**Befehlsauflistung 14:** `/etc/limits`

```
*      L2 C0 U15 R10000
kn     L10 C100000 U35
```

Hier setzen wir die Standardeinstellungen und eine spezielle Einstellung für den Anwender `kn`. Limits sind ein Teil des Shadow-Paketes und betreffen nur das Shadow-Login-Programm. Es ist nicht notwendig irgendwelche Beschränkungen in dieser Datei zu setzen, wenn Sie die PAM-Einstellung in `/etc/make.conf` vorgenommen haben und PAM vollständig konfiguriert haben.

### 6.3 Quotas

## Warnung

Stellen Sie sicher, dass ihr Dateisystem Quotas unterstützt. ReiserFS zum Beispiel tut es nicht!

Die Anwendung von Quotas auf einem Dateisystem verhindert, daß Anwender den Datenträger mit Ihren Daten überfüllen oder überhaupt schreiben können. Die Kernel-Option wird bei der Kernelkonfiguration unter `File systems->Quota support` aktiviert. Nehmen Sie die Einstellung vor, kompilieren Sie den Kernel neu und starten Sie mit diesem Ihren Computer neu.

Starten Sie die Installation mit *emerge quota*. Passen Sie Ihre */etc/fstab* an, indem Sie *usrquota* und *grpquota* bei den Partitionen hinzufügen, für die Sie die Nutzungsbeschränkung festlegen wollen.

**Befehlsauflistung 15:** */etc/fstab*

```
/dev/sda1 /boot ext2 noauto,noatime 1 1
/dev/sda2 none swap sw 0 0
/dev/sda3 / reiserfs notail,noatime 0 0
/dev/sda4 /tmp ext3 notail,noatime,nodev,nosuid,noexec,usrquota,grpquota 0 0
/dev/sda5 /var ext3 notail,noatime,nodev,usrquota,grpquota 0 0
/dev/sda6 /home ext3 notail,noatime,nodev,nosuid,usrquota,grpquota 0 0
/dev/sda7 /usr reiserfs notail,noatime,nodev,ro 0 0
/dev/cdroms/cdrom0 /mnt/cdrom iso9660 noauto,ro 0 0
proc /proc proc defaults 0 0
```

Auf jeder Partition auf der Sie Quotas aktiviert haben, erstellen Sie nun die Quota-Dateien (*quota.user* und *quota.group*) und setzen diese in die Wurzel der Partition.

**Befehlsauflistung 16:** Erstellen der Quota-Dateien

```
# touch /tmp/quota.user
# touch /tmp/quota.group
# chmod 600 /tmp/quota.user
# chmod 600 /tmp/quota.group
```

Dieser Schritt muss auf jeder Partition durchgeführt werden, auf der Quotas aktiviert wurden. Nachdem Sie die Quota-Dateien erstellt und konfiguriert haben, müssen Sie das *quota* Initskript dem Default Runlevel hinzufügen.

**Befehlsauflistung 17:** Quota zum Default Runlevel hinzufügen

```
# rc-update add quota default
```

Wir werden das System nun so konfigurieren, dass die Quotas einmal wöchentlich gecheckt werden. Dazu fügen Sie folgende Zeile in die */etc/crontab* ein.

**Befehlsauflistung 18:** Quota Check in der crontab

```
0 3 * * 0 /sbin/quotacheck -avug
```

Nachdem Sie den Rechner neu gestartet haben, ist es an der Zeit, die Quotas für die Benutzer und Gruppen festzulegen. *edquota -u kn* wird den in *\$EDITOR* festgelegten Editor starten (Standard ist *nano*), damit Sie die Quotas des Benutzers *kn* bearbeiten können. *-g* wird genau dasselbe, allerdings für Gruppen machen.

**Befehlsauflistung 19:** Bearbeiten der Quotas für den Benutzer *kn*

```
Quotas for user kn:
/dev/sda4: blocks in use: 2594, limits (soft = 5000, hard = 6500)
          inodes in use: 356, limits (soft = 1000, hard = 1500)
```

Für weitere Informationen lesen Sie bitte **man edquota** oder [Das Quota Mini-Howto](#)

## 6.4 /etc/login.defs

Wenn die Richtlinie besagt, dass die Anwender jede Woche ihr Passwort ändern müssen, dann setzen Sie die Variable *PASS\_MAX\_DAYS* auf 14 und *PASS\_WARN\_AGE* auf 7. Es wird ausserdem empfohlen, dass Sie alternde Passwörter benutzen, da Brute-Force Angriffe jedes Passwort finden können - alles nur eine Frage der Zeit. Wir empfehlen ausserdem, dass Sie *LOG\_OK\_LOGINS* auf *yes* setzen.

## 6.5 /etc/login.access

Die *login.access* ist auch ein Teil des *Shadow*-Paketes, das eine Login Zugangs-Kontrolltabelle anbietet. Die Tabelle wird benutzt um zu kontrollieren, wer und wer nicht einloggen darf, basierend auf dem Benutzernamen, dem Gruppennamen oder dem Hostnamen von dem der Versuch gestartet wird. Normalerweise sind alle Anwender des Systems berechtigt sich anzumelden; aus diesem Grunde ist die Datei nur mit Kommentaren und Beispielen gefüllt. Je nachdem wie Sie Ihren Server oder Ihren Arbeitsplatzrechner schützen empfehlen wir die Datei so anzupassen, das niemand anderes als Sie selbst (also der Administrator) Zugang zur Konsole bekommt.

## Notiz

Diese Einstellungen sind nicht für root anwendbar.

**Befehlsauflistung 20:** /etc/login.access

```
-:ALL EXCEPT wheel sync:console
-:wheel:ALL EXCEPT LOCAL .gentoo.org
```

## Wichtig

Seien Sie vorsichtig bei der Bearbeitung der Datei. Wenn Sie nicht aufpassen, dann können Sie sich aussperren wenn Sie nicht über root-Rechte verfügen.

## Notiz

Diese Einstellungen wirken sich nicht auf SSH aus, da SSH /bin/login normalerweise nicht ausführt. Dies kann durch die Benutzung von "UseLogin yes" in /etc/ssh/sshd\_config ermöglicht werden. Das führt dazu, dass SSH login benutzt und die Einstellungen benutzt werden.

Dies erstellt Loginzugriff so, dass Mitglieder von wheel sich an der Konsole einloggen können oder wenn ihre Quelle die gentoo.org Domäne ist. Vielleicht ein wenig zu paranoid, aber sicher ist sicher.

## 7. Dateiberechtigungen.

### 7.1 Von allen lesbar

Normale Benutzer sollten zu Konfigurationsdateien oder Passwörtern keinen Zugang haben. Ein Angreifer kann Passwörter aus einer Datenbank oder von einer Webseite stehlen und verunstalten oder noch schlimmer: Daten löschen. Deswegen ist es notwendig, dass die Berechtigungen korrekt gesetzt sind. Wenn Sie sicher sind, dass eine Datei nur von root benutzt wird, geben Sie ihr die Berechtigung 0600 und ordnen Sie diese mit *chown* dem richtigen Benutzer zu.

### 7.2 Welt/Gruppen-Schreibbar.

**Befehlsauflistung 21:** Auffinden von Dateien und Verzeichnissen, die von allen schreibbar sind

```
# /usr/bin/find / -type f \( -perm -2 -o -perm -20 \) \
-exec ls -lg {} \; 2>/dev/null >writable.txt
# /usr/bin/find / -type d \( -perm -2 -o -perm -20 \) \
-exec ls -ldg {} \; 2>/dev/null >>writable.txt
```

Dies schafft eine riesige Datei mit Berechtigungen von allen Dateien, die entweder Schreibberechtigungen für alle oder eine Gruppe haben. Überprüfen Sie die Berechtigungen und eliminieren Sie die für alle schreibbaren Dateien durch das Ausführen von /bin/chmod o-w für die Dateien.

### 7.3 SUID/SGID Dateien

SUID/SGID Dateien (Dateien bei denen das superuser bit gesetzt wurde) ist ein Weg für normale Benutzer Dinge zu tun die normalerweise nur root darf. Diese Dateien können zu lokalen root-Brüchen führen (wenn sie Sicherheitslöcher enthalten), da so eine Datei mit root Berechtigungen ausgeführt wird. Diese Dateien sind gefährlich und sollten unter allen Umständen vermieden werden. Wenn Sie die Dateien nicht benutzen. Führen sie *chmod 0* aus oder entfernen sie das Paket (*unmerge*) aus dem sie stammen (überprüfen sie das Paket mit *qpkg -f*. Wenn sie es nicht längst installiert haben, tun Sie dies mit *emerge gentoolkit*). Ansonsten schalten Sie das *suid* bit einfach mit *chmod-s* ab.

**Befehlsauflistung 22:** Auffinden von setuid Dateien

```
# /usr/bin/find / -type f \( -perm -004000 -o -perm -002000 \) \
-exec ls -lg {} \; 2>/dev/null >suidfiles.txt
```

Dies erzeugt eine Datei mit einer Liste aller SUID/SGID Dateien.

**Befehlsauflistung 23:** Liste der setuid binären Dateien

```
/bin/su
/bin/ping
/bin/mount
/bin/umount
/var/qmail/bin/qmail-queue
/usr/bin/chfn
```

```
/usr/bin/chsh
/usr/bin/crontab
/usr/bin/chage
/usr/bin/expiry
/usr/bin/sperl5.6.1
/usr/bin/newgrp
/usr/bin/passwd
/usr/bin/gpasswd
/usr/bin/procmail
/usr/bin/suidperl
/usr/lib/misc/pt_chown
/usr/sbin/unix_chkpwd
/usr/sbin/traceroute
/usr/sbin/pwdb_chkpwd
```

Standardmäßig hat Gentoo Linux nicht viele SUID Dateien (es hängt davon, was Sie installiert haben), aber Sie könnten eine Liste wie die obige erhalten. Viele dieser Befehle sollten nicht von normalen Benutzern benutzt werden, sondern nur von root. Schalten sie das suid bit bei *ping*, *mount*, *umount*, *chfn*, *chsh*, *newgrp*, *suidperl*, *pt\_chown* und *traceroute* aus. Sie tun dies mit dem Befehl *chmod -s* bei jeder einzelnen Datei. Entfernen Sie das bit nicht von *su*, *qmail-queue* oder *unix\_chkpwd*. Dies würde dazu führen, dass Sie nicht mehr su benutzen und mail empfangen könnten. Durch entfernen des bits entfernen Sie die Möglichkeit, dass ein normaler User (oder Angreifer) root Zugriff durch eine dieser Dateien erlangen kann.

Die einzigen SUID Dateien die ich auf meinem System habe sind *su*, *passwd*, *gpasswd*, *qmail-queue*, *unix\_chkpwd* und *pwdb\_chkpwd*. Aber wenn sie X benutzen, könnten sie einige mehr haben, denn X benötigt diesen Zugriff.

## 8. PAM (Pluggable Authentication Modules)

PAM ist eine Sammlung von shared libraries, die eine Alternative für Authentifizierungen in Programmen darstellen. Die PAM Einstellungen von Gentoo Linux sind relativ vernünftig, aber es gibt immer Platz für Verbesserungen. Zunächst installieren wir cracklib.

**Befehlsauflistung 24:** Installieren von cracklib

```
# emerge cracklib
```

**Befehlsauflistung 25:** /etc/pam.d/passwd

```
auth    required pam_pwdb.so shadow nullok
account required pam_pwdb.so
password required pam_cracklib.so difok=3 retry=3 minlen=8 dcredit=2 ocredit=2
password required pam_pwdb.so md5 use_authtok
session required pam_pwdb.so
```

Dies fügt die cracklib hinzu, welche sicherstellt, dass der Benutzer eine minimale Passwortlänge von 8 Zeichen benutzt; Bestehend aus mindestens 2 Zahlen, 2 Unterschiedlichen und es müssen mindestens 3 Zeichen anders sein als beim letzten Passwort. Dies zwingt den Benutzer ein gutes Passwort zu wählen (Passwortrichtlinien). In der Dokumentation von [PAM](#) finden Sie weitere Optionen.

**Befehlsauflistung 26:** /etc/pam.d/sshd

```
auth    required pam_pwdb.so nullok
auth    required pam_shells.so
auth    required pam_nologin.so
auth    required pam_env.so
account required pam_pwdb.so
password required pam_cracklib.so difok=3 retry=3 minlen=8 dcredit=2 ocredit=2 use_authtok
password required pam_pwdb.so shadow md5
session required pam_pwdb.so
session required pam_limits.so
```

Jeder andere Dienst der nicht mit einer PAM Datei in */etc/pam.d* konfiguriert ist wird die "andere" Regel benutzen. Die Standardeinstellung sind auf *deny* gesetzt, so wie es sein sollte. Jedoch habe ich gerne viele Protokolle und deswegen habe ich *pam\_warn.so* hinzugefügt. Die letzte Konfiguration ist *pam\_limits* welche von */etc/security/limits.conf* kontrolliert wird. Siehe das [passende Kapitel](#) hierzu.

**Befehlsauflistung 27:** /etc/pam.d/other

```
auth      required pam_deny.so
auth      required pam_warn.so
account   required pam_deny.so
account   required pam_warn.so
password  required pam_deny.so
password  required pam_warn.so
session   required pam_deny.so
session   required pam_warn.so
```

## 9. TCP Wrappers

Ist ein Weg um Zugang zu kontrollieren für Dienste die normalerweise von inetd ausgeführt werden (welches Gentoo nicht hat) aber es kann auch von xinetd und anderen Diensten benutzt werden.

### Notiz

Der Dienst sollte tcpd in seinem Serverargument (in xinetd) ausgeführt werden. Schauen Sie für mehr Informationen in das xinetd Kapitel.

**Befehlsauflistung 28:** /etc/hosts.deny

```
ALL:PARANOID
```

**Befehlsauflistung 29:** /etc/hosts.allow

```
ALL: LOCAL @wheel
time: LOCAL, .gentoo.org
```

Wie Sie sehen können ist das Format sehr ähnlich dem in [/etc/login.access](#). Tcpsd unterstützt einen spezifischen Dienst und sie arbeiten nicht im selben Gebiet von Sicherheit. Diese Einstellungen gelten nur für Dienste die TCP Wrapper benutzen.

Es ist auch möglich Befehle auszuführen wenn auf einen Dienst zugegriffen wird (kann benutzt werden wenn Weiterleiten für Benutzer die sich einwählen aktiviert wird) aber es nicht empfohlen, da Menschen dazu neigen mehr Probleme zu schaffen als sie versuchen zu beheben. Ein Beispiel könnte sein, dass sie ein Script konfigurieren um email zu senden jedes mal wenn jemand die deny-Regel trifft, aber ein Angreifer könnte so eine DoS Attacke ausführen indem er darauf weiter zugreift. Dies schafft viel I/O und viele mails, deswegen tun Sie es nicht! Lesen Sie *man 5 hosts\_access* für weitere Informationen.

## 10. Kernelsicherheit

### 10.1 Funktionsentfernung

Eine grundlegende Regel ist die Entfernung von allem was sie nicht brauchen. Das schafft einen kleinen Kernel und entfernt auch die Verwundbarkeiten die in Treibern oder anderen Eigenschaften liegen können.

Ziehen Sie auch in betracht loadable module support(=ladbare-Modulunterstützung) auszuschalten. Auch wenn es möglich ist Module ohne diese Eigenschaft hinzuzufügen (root kits), wird es doch schwerer für den normalen Angreifer root kits über Kernelmodule zu installieren.

### 10.2 /proc (kernel flags)

Viele Kernel Parameter können durch das /proc Dateisystem verändert werden, oder durch die Benutzung von *sysctl*.

Um dynamisch Kernelparameter und -variablen sofort zu ändern benötigen Sie *CONFIG\_SYSCTL* in Ihrem Kernel. Die ist voreingestellt im Standard 2.4 Kernel.

**Befehlsauflistung 30:** Entfernen von ping-Paketen

```
# /bin/echo "1" > /proc/sys/net/ipv4/icmp_echo_ignore_all
```

Dies sperrt icmp Typ 0 (auch bekannt als Ping) Pakete. Der Grund hierfür ist, dass icmp Nutzlast mit anderen Informationen beinhalten kann als Sie denken. Administratoren benutzen Ping als Diagnoseprogramm und beschweren sich oft wenn sie Ping nicht benutzen können. Es gibt keinen Grund für einen Außenstehenden die Möglichkeit zu haben Ping zu benutzen, aber ab und zu kann es hilfreich für Eingeweihte sein, diese Möglichkeit zu haben. Das Problem kann dadurch gelöst werden, indem man icmp type 0 in der Firewall deaktiviert.

**Befehlsauflistung 31:** Ignorieren von broadcast-Pings

```
# /bin/echo "1" > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

Dies sperrt Antworten auf Anfragen, Sie wollen schliesslich nicht ein Schlumpferverstärker werden. Schlumpferverstärker oder X-mass trees ist eine Methode die es einem Angreifer erlaubt einen moderaten Teil von Traffic zu senden und geradezu eine Explosion von Traffic zu verursachen am beabsichtigten Ziel.

**Befehlsauflistung 32:** Sperren von source routed Paketen

```
# /bin/echo "0" > /proc/sys/net/ipv4/conf/all/accept_source_route
```

Akzeptieren sie keine source routed Pakete. Angreifer können source routing benutzen um Traffic zu erzeugen der vorgibt aus dem Netzwerk zu kommen, jedoch weitergeleitet wurde den Pfad von dem es ursprünglich kam. Sperren sie Source Routing denn es wird selten für legitime Zwecke genutzt.

**Befehlsauflistung 33:** Sperren von Umleitungsakzeptanz

```
# /bin/echo "0" > /proc/sys/net/ipv4/conf/all/accept_redirects
```

Sperren Sie ICMP Umleitungsakzeptanz. ICMP Umleitungen können benutzt werden um Ihre routing tables zu verändern, möglicherweise zu einem schlimmen Ende.

**Befehlsauflistung 34:** Schutz gegen bad error messages

```
# /bin/echo "1" > /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
```

Schalten sie den Schutz gegen bad error messages ein.

**Befehlsauflistung 35:** Ermöglichen von rückwärtiger Pfadfilterung

```
# for i in /proc/sys/net/ipv4/conf/*; do
    /bin/echo "1" > $i/rp_filter
done
```

**Notiz**

Wenn Sie IP forwarding aktivieren, werden Sie auch dieses Resultat erhalten.

Stellen Sie reverse path filtering an. Dies hilft durch automatisches Ablehnen von Quelladressen, die nicht mit dem Netzwerkinterface übereinstimmen, dabei sicherzustellen, dass Pakete legitime Quelladressen benutzen. Dies hat Sicherheitsvorteile, da es IP Spoofing verhindert.

**Warnung**

Die Nutzung von reverse path filtering kann auch ein Problem darstellen, wenn sie asymmetrisches Routing benutzen (Pakete von Ihnen zu einem Host nehmen einen anderen Weg als Pakete vom host zu Ihnen) oder wenn Sie einen Non-Routing Host betreiben, der verschiedene IP-Adressen an verschiedenen Interfaces hat.

**Befehlsauflistung 36:** Protokollieren aller spoofed, source routed und umgeleiteten Pakete

```
# /bin/echo "1" > /proc/sys/net/ipv4/conf/all/log_martians
```

Protokollieren von spoofed, source routed und umgeleiteten Pakete.

**Befehlsauflistung 37:** Aktivieren von IP forwarding

```
# /bin/echo "0" > /proc/sys/net/ipv4/ip_forward
```

Stellen Sie sicher, dass IP forwarding ausgeschaltet ist. Wir wollen es nur für einen multi-homed Host.

Alle diese Einstellungen werden zurückgesetzt, wenn die Maschine neu gestartet wird. Daher schlage ich vor, dass Sie folgendes Script zum *rc-update add proccparam default* Default Runlevel hinzufügen und ausführbar *chmod +x /etc/init.d/proccparam* machen.

**Befehlsauflistung 38:** /etc/init.d/proccparam

```
#!/sbin/runscript

depend() {
    use checkroot
}
```

```

start() {
  ebegin "Setting /proc options."
  /bin/echo "1" > /proc/sys/net/ipv4/icmp_echo_ignore_all
  /bin/echo "1" > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
  /bin/echo "0" > /proc/sys/net/ipv4/conf/all/accept_source_route
  /bin/echo "0" > /proc/sys/net/ipv4/conf/all/accept_redirects
  /bin/echo "1" > /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
  for i in /proc/sys/net/ipv4/conf/*; do
    /bin/echo "1" > ${i}/rp_filter
  done
  /bin/echo "1" > /proc/sys/net/ipv4/conf/all/log_martians
  /bin/echo "0" > /proc/sys/net/ipv4/ip_forward
  eend 0
}

```

## 10.3 Grsecurity

Der Patch von [Grsecurity](#) ist Standard im Gentoo Kernel aber per Voreinstellung deaktiviert. So aktivieren Sie ihn:

Konfigurieren Sie Ihren Kernel wie Sie es normalerweise auch würden und konfigurieren Sie dann die Grsecurity Option: (wählen sie customized) und stellen sie folgende Optionen ein:

- Buffer Overflow Protection
  - Openwall non-executable stack
  - GCC trampoline support
- Filesystem Protections
  - Proc restrictions
  - Linking restrictions
  - Secure file descriptors
  - Chroot jail restrictions (aktivieren sie alle Optionen unterhalb dieser)
- Kernel Auditing
  - Log execs within chroot
  - (Un)Mount logging
  - Signal logging
  - Fork failure logging
  - Log set\*ids to root
  - Time change logging
- Executable Protections
  - Dmesg restriction
  - Randomized PIDs
  - Altered default IPC permissions (kann verhindern, dass einige Programme korrekt ausgeführt werden)
  - Restricted ptrace
- Network Protections
  - Randomized IP IDs
  - Randomized TCP source ports
  - Altered Ping IDs
  - Randomized TTL
- Miscellaneous Features
  - BSD-style core dumps (erzeugt core dumps wie core.named)

Jetzt kompilieren und installieren sie ihren Kernel mit verbesserter Sicherheit.

## 10.4 Kerneli

[Kerneli](#) ist ein Patch der Verschlüsselung zum existierenden Kernel hinzufügt. Durch patchen des Kernel erhalten Sie neue Optionen wie: Kryptographische Chiffrierung, Zusammenfassungsverfahren und Kryptographische-Schleifenfilter.

### Warnung

Der Kerneli Patch ist momentan nicht in einer stabilen Version für den neuesten Kernel verfügbar, also Vorsicht beim Gebrauch.

## 10.5 Andere Kernel Patches

- [The OpenWall Project](#) (nicht für 2.4 Kernel)
- [Linux Intrusion Detection System](#)

- [Rule Set Based Access Control](#)
- [NSA's security enhanced kernel](#)
- [Wolk](#)

Und es gibt wahrscheinlich vieles mehr.

## 11. Sichern von Diensten

### 11.1 Benutzung von xinetd

xinetd ist ein Ersatz für inetd (welchen Gentoo nicht hat), den Internet-Dienst-Daemon. Er unterstützt Zugriffskontrolle basierend auf den Adressen der entfernten Hosts und der Zugriffszeit. Es beinhaltet auch ausführliche Protokollfähigkeiten, inklusive Serverstartzeit, Adresse des entfernten Hosts, entfernter Benutzername, Serverlaufzeit und geforderte Abläufe.

Wie bei allen anderen Diensten ist es wichtig eine gute Standardkonfiguration zu haben. Da aber *xinetd* von root benutzt wird und Protokolle unterstützt, von denen Sie möglicherweise die Funktionsweise nicht verstehen, raten wir Ihnen es nicht zu benutzen. Wenn Sie es aber doch benutzen wollen, fügen Sie so mehr Sicherheit hinzu:

**Befehlsauflistung 39:** Installieren von xinetd

```
# emerge xinetd tcp-wrappers
```

Ergänzen Sie die Konfigurationsdatei um:

**Befehlsauflistung 40:** /etc/xinetd.conf

```
defaults
{
  only_from      = localhost
  instances      = 10
  log_type       = SYSLOG authpriv info
  log_on_success = HOST PID
  log_on_failure = HOST
  cps            = 25 30
}

# Dies konfiguriert pserver (cvs) durch xinetd mit den folgenden Einstellungen:
# maximal 10 Instanzen (10 Verbindungen gleichzeitig)
# Begrenzung von pserver auf tcp
# benutzen des Benutzer-cvs um diesen Dienst laufen zu lassen
# Anbinden der Schnittstelle an nur 1 IP
# Zulassen von Zugriff von 10.0.0.*
# Begrenzung der Zeit in der Entwickler auf das cvs
# zugreifen können von 08Uhr bis 17Uhr
# Benutzung von tcpd wrappers (Zugriffskontrolle kontrolliert durch
# /etc/hosts.allow und /etc/hosts.deny)
# max_load ist an der Maschine auf 1.0 gesetzt
# die disable flag (sperrern) steht auf nein, aber ich bevorzuge sie zu
# haben, für den Fall das es gesperrt werden sollte
service cvspserver
{
  socket_type      = stream
  protocol         = tcp
  instances        = 10
  protocol         = tcp
  wait             = no
  user             = cvs
  bind             = 10.0.0.2
  only_from        = 10.0.0.0
  access_times     = 8:00-17:00
  server           = /usr/sbin/tcpd
  server_args      = /usr/bin/cvs --allow-root=/mnt/cvsdisk/cvsroot pserver
  max_load         = 1.0
  log_on_failure   += RECORD
  disable          = no
}
```

Für mehr Informationen lesen Sie *man 5 xinetd.conf*.

## 11.2 SSH

Die einzige Sicherheitsverstärkung die OpenSSH benötigt, ist eine stärkere Verschlüsselung basierend auf Public Key Verschlüsselung. Zu viele Seiten (wie <http://www.sourceforge.net>, <http://www.php.net> und <http://www.apache.org>) haben wegen Passwortlecks oder schlechten Paswörtern unter unauthorisiertem Eindringen in ihre Systeme gelitten.

**Befehlsauflistung 41:** /etc/ssh/sshd\_config

```
#Aktivieren Sie nur Version 2
Protocol 2

#Kein direkter root Zugriff
PermitRootLogin no

#Benutzung von RSA Schlüsselauthentifizierung
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile      .ssh/authorized_keys

# Sperren von .rhost Dateien und normaler Passwordauthentifizierung
RhostsAuthentication no
PasswordAuthentication no
PermitEmptyPasswords no

AllowHosts *.gentoo.org

#Nur Mitglieder der wheel oder admin erhalten Zugriff
AllowGroups wheel admin

#Von allen Leuten in diesen beide Gruppen erhalten nur kn und ns Zugriff
AllowUsers kn bs

#Hinzufügen des Protokollierungslevels
SyslogFacility AUTH
LogLevel INFO

#bind
ListenAddress 127.0.0.1
```

Jetzt ist das einzige was Ihre Benutzer machen müssen einen Schlüssel zu erstellen (auf der Maschine von der sie sich einloggen wollen) mit folgendem Befehl:

**Befehlsauflistung 42:** Erstellen eines RSA Schlüsselpaars

```
# /usr/bin/ssh-keygen -t rsa
```

Und tippen Sie einen Passsatz (auch Mantra genannt) ein:

**Befehlsauflistung 43:** Ausgabe von ssh-keygen

```
Generierung des öffentlichen/privaten rsa Schlüsselpaars.
Geben sie den Dateinamen ein unter dem der Schlüssel gespeichert wird (/home/kn/.ssh/id_rsa):[En
Verzeichnis erstellt '/home/kn/.ssh'.
Passsatz eingeben (leer für keinen Passsatz): [Passsatz eingeben]
Denselben Passsatz erneut eingeben: [Erneut Passsatz eingeben]
Ihre Identifikation wurde in /home/kn/.ssh/id_rsa gespeichert.
Ihr öffentlicher Schlüssel wurde in /home/kn/.ssh/id_rsa.pub gespeichert.
Der Fingerabdruck des Schlüssels ist:
07:24:a9:12:7f:83:7e:af:b8:1f:89:a3:48:29:e2:a4 kn@knielsen
```

Dies fügt zwei Dateien zu Ihrem `~/ssh/` Verzeichnis mit den Namen `id_rsa` und `id_rsa.pub` hinzu. Die Datei `id_rsa` ist Ihr privater Schlüssel und sollte von anderen Leuten außer Ihnen Ferngehalten werden. Die andere Datei `id_rsa.pub` soll an jeden Server verteilt werden zu dem Sie Zugriff haben. Fügen Sie den Schlüssel in das home Verzeichnis des Benutzers in `~/ssh/authorized_keys` ein, so sollte der Benutzer die Möglichkeit haben sich einzuloggen.

Ihre Benutzer sollten diesen privaten Schlüssel gut verwahren. Packen Sie es auf ein Medium, dass sie immer mit sich tragen oder lassen Sie es auf ihrer Workstation (fügen Sie dies in die [Passwortrichtlinien](#) ein).

Mehr über [OpenSSH](#) finden Sie auf der Webseite.

## 11.3 X

XFree ist von Haus aus als Xserver konfiguriert. Dies kann gefährlich sein, denn X benutzt unverschlüsselte TCP-Verbindungen und wartet auf xclients.

### Wichtig

Wenn Sie diesen Dienst nicht brauchen, dann deaktivieren Sie ihn!

Aber wenn Sie Ihren Arbeitsplatz als Xserver betreiben möchten, dann benutzen Sie das Kommando `/usr/X11R6/bin/xhost` nur mit äußerster Vorsicht. Dieses Kommando erlaubt Clients von anderen Rechnern sich mit Ihrer Anzeige zu verbinden und diese auch zu nutzen. Dies kann sinnvoll sein, wenn Sie eine X-Anwendung von einem anderen Rechner brauchen und die einzige Verbindung zwischen den Rechnern ein Netzwerk ist. Die Syntax lautet `/usr/X11R6/bin/xhost +hostname`.

### Warnung

Benutzen Sie nie das `xhost +` feature! Dies wird jedem Client erlauben sich mit Ihrem X zu Verbinden und dieses unter Kontrolle bringen. Wenn ein Angreifer Zugang zu Ihrem X erlangt, dann kann er Ihre Tastatureingaben überwachen und Kontrolle über Ihren Desktop erlangen.

Eine sichere Lösung ist, dieses Feature vollständig zu deaktivieren indem Sie X mit `startx -- -nolisten tcp` starten oder dies auf Dauer über eine entsprechende Einstellung in der Konfigurationsdatei verhindern.

**Befehlsauflistung 44:** `/usr/X11R6/bin/startx`

```
defaultserverargs="-nolisten tcp"
```

Um sicherzustellen, dass `startx` bei einem Emergen einer neuen XFree Version überschrieben werden kann, müssen Sie die Datei beschützen. Fügen Sie die folgende Zeile in die `/etc/make.conf` ein:

**Befehlsauflistung 45:** `/etc/make.conf`

```
CONFIG_PROTECT_MASK="/usr/X11/bin/startx"
```

Wenn sie einen grafischen Login Manager benutzen, benötigen Sie einen anderen Ansatz.

Für `gdm` (Gnome Display Manager)

**Befehlsauflistung 46:** `/etc/X11/gdm/gdm.conf`

```
[server-Standard]
command=/usr/X11R6/bin/X -nolisten tcp
```

Für `xdm` (X Display Manager) und `kdm` (KDE Display Manager)

**Befehlsauflistung 47:** `/etc/X11/xdm/Xservers`

```
:0 local /usr/bin/X11/X -nolisten tcp
```

## 11.4 FTP

Das Benutzen von FTP (File Transfer Protocol) ist im Allgemeinen eine schlechte Idee. Es benutzt unverschlüsselte Daten, lauscht auf zwei Ports (normalerweise 20 und 21), und anonyme Logins sind das, wonach Angreifer gerne suchen (um WareZ zu verteilen). Da das ftp-Protokoll einige Sicherheitslücken enthält, benutzen Sie bitte alternativ `sftpd` oder HTTP. Wenn dies nicht möglich sein sollte, dann sichern Sie Ihre Dienste so gut wie nur möglich ab und bereiten Sie sich vor.

## 11.5 Pure-ftpd

Pure-ftpd ist ein Abkömmling des originalen `trollftpd`. Für mehr Sicherheit und Funktionalität wurde es von Frank Dennis modifiziert.

Benutzen Sie virtuelle Benutzer (niemals Systemkonten) indem Sie die AUTH-Option aktivieren. Setzen Sie diese auf `-lpuredb:/etc/pureftpd.pdb` und erstellen Sie Ihre Benutzer mittels `/usr/bin/pure-pw`.

**Befehlsauflistung 48:** `/etc/conf.d/pure-ftpd`

```

## Anzahl der gleichzeitigen Verbindungen - insgesamt und je IP ##
MAX_CONN="-c 30"
MAX_CONN_IP="-C 10"

## Keine Uploads erlauben, wenn die Partition voller als dieser Wert hier ist##
DISK_FULL="-k 90%"

AUTH="-lpuredb:/etc/pureftpd.pdb"

## Diverse andere ##
MISC_OTHER="-A -E -X -U 177:077 -d -4 -L100:5 -I 15"

```

Desweiteren konfigurieren Sie die *MISC\_OTHER* Einstellung so, dass folgendes verboten ist: keine Anonyma (-E) und chroot auf jeden (-A). Benutzer können keine Dateien lesen oder schreiben, die mit einem . (Punkt) beginnen (-X), maximale Leerlaufzeit (-I), Rekursion begrenzen (-L) und einen sinnvollen *umask*.

### Warnung

Benutzen Sie **nicht** die Option -w oder -W ! Wenn Sie eine Warez Seite möchten, dann hören Sie nun bitte auf dieses Dokument zu lesen!

Mehr dazu gibts auf <http://www.pureftpd.org>

## 11.6 Proftpd

Proftpd hat einige Sicherheitsprobleme, aber es hat den Anschein als seien die meisten repariert worden. Weitere Verbesserungen wären:

### Befehlsauflistung 49: /etc/proftpd/proftpd.conf

```

ServerName "Mein ftp Daemon"
#Zeigen Sie nicht den Ident des Servers
ServerIdent on "Hau ab!"

#Vereinfacht es virtuelle Benutzer anzulegen
RequireValidShell off

#Benutzen Sie eine alternative Passwort- und Gruppendatei (passwd benutzt das Crypt-Format)
AuthUserFile "/etc/proftpd/passwd"
AuthGroupFile "/etc/proftpd/group"

# Berechtigungen
Umask 077

# Timeouts und Beschränkungen
MaxInstances 30
MaxClients 10 "Nur 10 Verbindungen erlaubt"
MaxClientsPerHost 1 "Sie sind schon eingeloggt"
MaxClientsPerUser 1 "Sie sind schon eingeloggt"
TimeoutStalled 10
TimeoutNoTransfer 20
TimeoutLogin 20

#jeden "chroot"-en
DefaultRoot ~

#nicht als root laufen lassen
User nobody
Group nogroup

#Jeden Transfer aufzeichnen
TransferLog /var/log/transferlog

#Probleme mit Zeichenersetzung
DenyFilter \*.*

```

Alles Andere hängt von Ihnen und Ihren Lesekenntnissen ab (<http://www.proftpd.org>).

## 11.7 Vsftpd

Vsftpd (dies steht für "Very Secure ftp") ist ein kleiner FTP-Dämon mit einer einfachen Standardkonfiguration. Er ist einfach und hat nicht so viele Funktionen (wie z.B. virtuelle Benutzer) wie sie PureFTP und ProFTP anbieten.

**Befehlsauflistung 50:** /etc/vsftpd

```
anonymous_enable=NO
local_enable=YES

#nur lesbar
write_enable=NO

#Aufzeichnen von Übertragungen aktivieren
xferlog_std_format=YES

idle_session_timeout=20
data_connection_timeout=20
nopriv_user=nobody

chroot_list_enable=YES
chroot_list_file=/etc/vsftpd/chrootlist

ls_recurse_enable=NO
```

Wie Sie sehen können, gibt es keine Möglichkeit für diesen Dienst individuelle Rechte und eine standardmäßige chroot-Aktion zu konfigurieren. Aber wenn es um die anonymen Einstellungen geht, dann entwickelt sich dies zum Vorteil. Manchmal kann es gut sein einen anonymen FTP-Server z.B. zum Verteilen von Open-Source zu haben und dieser Server passt hierfür perfekt.

## 11.8 Apache

Apache (1.3.26) kommt mit einer recht gut eingestellten Konfigurationsdatei, aber auch hier müssen wir einige Dinge verbessern wie z.B. verbinden mit einer Adresse und Verhindern des Datenverlustes bei der Übertragung. Folgende Optionen sollten sie in der Konfigurationsdatei anpassen:

Wenn Sie *ssl* in Ihrer */etc/make.conf* vor der Installation von Apache gesetzt haben, dann sollten Sie Zugang zu einem SSL-fähigen Server haben. Fügen Sie folgende Zeile ein um dieses Feature zu aktivieren.

**Befehlsauflistung 51:** /etc/conf.d/apache

```
HTTTPD_OPTS="-D SSL"
```

**Befehlsauflistung 52:** /etc/apache/conf/apache.conf

```
#Lassen Sie ihn auf die richtige IP hören
Listen 127.0.0.1
BindAddress 127.0.0.1
#Es ist keine gute Idee nobody oder nogroup für
#jeden Prozess der nicht als root läuft zu benutzen -
#(erstellen Sie den Benutzer apache mit der Gruppe apache)
User apache
Group apache
#Wir halten Apache davon ab, die Serverversion auszuplaudern
ServerSignature Off
ServerTokens min
```

Apache wird mit *--enable-shared=max* und *--enable-module=all* kompiliert. Dies wird von Vorneherein alle Module aktivieren, sodass Sie alle Module in der *LoadModule*-Sektion (also *LoadModule* und *AddModule*) auskommentieren müssen, die Sie nicht benötigen. Starten Sie den Dienst neu, indem Sie */etc/init.d/apache restart* ausführen.

Die Dokumentation gibt es auf <http://www.apache.org>

## 11.9 Qmail

Qmail wird als der sicherste Mail-Server angesehen. Er wurde mit Sicherheit (und Paranoia) im Hinterkopf geschrieben. Es erlaubt von Haus aus kein Relaying und hatte seit 1996 kein Sicherheitsloch. Starten Sie einfach ein *emerge qmail* und konfigurieren Sie es danach.

## 11.10 BIND

### Wichtig

Bind ist bekannt für seine lausige Sicherheits-Vergangenheit und sollte nicht leicht genommen werden. Wie jeder andere Dienst sollte auch Bind **niemals** als root laufen, ändern Sie die Standard-Konfiguration für diesen Dienst also bitte nicht.

Sie finden Dokumentation zu Bind beim [Internet Software Consortium](#). Das "BIND 9 Administrator Reference Manual" ist auch in [doc/arm](#) verfügbar.

## 11.11 Djbdns

Über dkbdns gibt es nicht viel zu sagen - ausser das der Autor bereit ist [Geld](#) darauf zu verwetten dass es sicher ist. Also versuchen Sie es : <http://www.djbdns.org/>. Es arbeitet anders als Bind v.9 aber Sie werden einen Einstieg finden.

## 11.12 Samba

Samba ist ein Protokoll für Dateiaustausch mit Microsoft/Novell-Netzwerken und es sollte **nicht** über das Internet benutzt werden. Dennoch benötigt es Absicherung.

**Befehlsauflistung 53:** /etc/samba/smb.conf

```
[global]
#An ein Interface binden
interfaces = eth0 10.0.0.1/32

#Sicherstellen, dass die Passwörter verschlüsselt werden
encrypt passwords = yes
directory security mask = 0700

#Kommunikation von 10.0.0.* erlauben
hosts allow = 10.0.0.

#Benutzerauthentifizierung aktivieren
#(Also nicht den "Share-Mode" benutzen)
security = user

#Privilegierte Accounts verbieten
invalid users = root @wheel

#Maximale Größe die SMB für einen Share anzeigt (kein Limit)
max disk size = 102400

#Richtlinie für Passwörter
min password length = 8
null passwords = no

#Wenn möglich PAM benutzen
obey pam restrictions = yes
pam password change = yes
```

Stellen Sie sicher, dass die Berechtigungen für jedes Share richtig eingestellt sind und denken Sie daran die [Dokumentation](#) zu lesen.

Starten Sie nun den Server neu und fügen Sie die Benutzer hinzu, die Zugriff auf diesen Server haben sollen. Dies wird über das Aufrufen von [/usr/bin/smbpasswd](#) mit dem Parameter -a ermöglicht.

## 11.13 Chroot oder Virtuelle Server.

Einen Dienst zu chrooten stellt eine Möglichkeit dar einen Dienst (oder Benutzer) auf für ihn vorgesehene Ressourcen zu beschränken und zu verhindern, daß er Zugang zu Bereichen (oder Informationen) erlangt, die zu einem unberechtigten Besitz von root-Rechten führen könnte. Indem man einen Dienst als ein anderer Benutzer als root laufen lässt (nobody, apache, named) kann ein Angreifer nur Zugriff auf die Ressourcen des entsprechenden Accounts bekommen. Dies bedeutet, dass ein Angreifer nie root-Rechte erlangen kann, selbst wenn der entsprechende Dienst eine Sicherheitslücke hätte.

Einige Dienste wie zum Beispiel pure-ftpd und bind haben eingebaute Fähigkeiten für "chrooting" und andere Dienste bieten dies nicht. Wenn der Dienst es anbietet, dann benutzen Sie es, andernfalls müssen Sie in die Materie einsteigen und einen eigenen Benutzer erstellen. Lassen Sie es uns nun versuchen und eine eigene chroot-Umgebung aufbauen. Um einen Einstieg zu finden und zu sehen wie chroot arbeitet versuchen wir es zuerst mit bash (als einfachen Einstieg ins Lernen).

Erstellen Sie das `/chroot` Verzeichnis mittels `mkdir /chroot`. Nun müssen wir herausfinden, mit welche dynamischen Bibliotheken `bash` benötigt (wenn sie mit `-static` kompiliert wurde, dann ist dieser Schritt nicht nötig).

Das folgende Kommando wird eine Liste der von bash benutzten Bibliotheken ausgeben.

**Befehlsauflistung 54:** Benutzte Bibliotheken auflisten

```
# ldd /bin/bash
libncurses.so.5 => /lib/libncurses.so.5 (0x4001b000)
libdl.so.2 => /lib/libdl.so.2 (0x40060000)
libc.so.6 => /lib/libc.so.6 (0x40063000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2 (0x40000000)
```

Nun erstellen wir die Umgebung für bash

**Befehlsauflistung 55:** chroot-Umgebung für bash erstellen

```
# mkdir /chroot/bash
# mkdir /chroot/bash/bin
# mkdir /chroot/bash/lib
```

Nun kopieren wir die von bash benutzten Bibliotheken (`/lib`) in das chroot und kopieren das Kommando `bash` in das Verzeichnis `/bin` im chroot. Dies wird die originale Umgebung herstellen - nur eben mit weniger Funktionalität. Nachdem das Kopieren abgeschlossen wurde versuchen Sie ein `chroot /chroot/bash`. Wenn Sie ein Prompt bekommen, dass als aktuelles Verzeichnis `/` angibt, dann hat alles funktioniert. Wenn nicht, dann werden Sie unter Umständen eine Fehlermeldung bekommen, die die fehlende Datei angibt. Manche dynamische Bibliotheken bauen aufeinander auf.

Sie werden feststellen, dass innerhalb der chroot-Umgebung nichts anderes als `echo` funktioniert. Dies ist deshalb so, weil wir keine anderen Kommandos in unserer chroot-Umgebung haben und `echo` ein in bash eingebauter Befehl ist.

Dies ist in etwa der Weg den Sie gehen würden, um einen "ge-chrooteten" Dienst zu erstellen. Der einzige Unterschied ist, dass Dienste manchmal auf Geräten und Konfigurationsdateien in `/etc` basieren. Kopieren Sie diese einfach in die chroot-Umgebung (Geräte können mit `cp -a` kopiert werden) und editieren Sie das Init-Script sodaß es die chroot-Umgebung vor der Ausführung verwendet. Es kann schwierig sein herauszufinden, welche Konfigurationsdateien und Geräte ein Dienst benutzt. Dies ist der Punkt, an dem `strace` nützlich wird. Starten Sie den Service mit `/usr/bin/strace bash` und suchen Sie nach `open`, `read`, `stat` und vielleicht noch `connect`. Dies wird Ihnen eine Idee darüber geben, welche Dateien Sie kopieren müssen. Aber in den meisten Fällen kopieren Sie einfach die `passwd`-datei (vorher editieren und die Benutzer entfernen, die mit dem Dienst nichts zu tun haben), `/dev/zero`, `/dev/log` und `/dev/random` in die neue Umgebung.

Ein weiterer Weg eine sichere Umgebung zu erstellen besteht darin, eine virtuelle Serverumgebung zu erstellen. Dies wird eine Kopie der existierenden Linuxinstallation erstellen und es virtuell booten. Dies bedeutet, dass wenn der Server angegriffen wird in Wirklichkeit nur der virtuelle Server angegriffen wird, nicht die echte Installation.

Beispiele von virtuellen Servern:

- [Usermode Linux](#) und ein HowTo über [Usermode Linux](#).

## 12. Firewalls

### 12.1 Eine Firewall

Oftmals wird eine Firewall als die ultimative Sicherheitsmassnahme bezeichnet - was aber nur bedingt stimmt. In den meisten Fällen kann eine falsch konfigurierte Firewall ein System sogar noch mehr verunsichern. Eine Firewall ist auch eine Software und sollte genau so wie jeder andere Dienst behandelt werden, denn auch hier können Bugs vorhanden sein (die hier Sicherheitslöcher sind).

Also denken Sie nach, bevor Sie eine Firewall in Betrieb nehmen! Brauchen Sie wirklich eine? Wenn Sie der Meinung sind, daß sie eine brauchen, dann verfassen Sie eine Richtlinie wie sie funktionieren sollte, welcher Art sie sein soll und wer sie betreiben sollte.

Firewalls werden für folgende beiden Zwecke verwendet:

- Um Benutzer (Würmer/Angreifer) draussen zu halten
- Um Benutzer (Angestellte/Kinder) drinnen zu halten

Es gibt im Allgemeinen drei Arten von Firewalls:

- Paket-Filter
- >>>> S.o please translate >>>> Circuit relay
- Applikationsgateway

Eine Firewall sollte auf einer dedizierten Maschine ohne weitere Dienste laufen (und wenn, dann höchstens noch ssh) und so abgesichert werden wie dieser Leitfaden es vorschlägt.

## 12.2 Paket-Filter

Jeglicher Netzwerkverkehr basiert auf Paketen. Viel Verkehr wird in kleinere Pakete transformiert (da diese einfacher zu handhaben sind) und bei der Ankunft am Ziel wieder in der richtigen Reihenfolge wieder zusammengesetzt. Jedes Paket enthält Informationen darüber wie es wohin transportiert werden soll. Und genau diese Informationen macht sich eine Firewall mit Paketfilter zu nutze. Filtern basiert auf:

- Erlauben oder verbieten von Paketen entsprechend der Quell-/Ziel-IP-Adresse
- Erlauben oder verbieten von Paketen entsprechend des Quell-/Ziel-Ports
- Erlauben oder verbieten von Paketen entsprechend dem verwendeten Protokoll
- Erlauben oder verbieten von Paketen entsprechend von bestimmten Einstellungen im Protokoll

Normalerweise wird nur anhand der Daten im Kopf eines Paketes und nicht im eigentlichen Inhalt vollzogen.

Schwächen:

- Adressinformationen in einem Paket könnten gefälscht (**gespoofed**) an den Sender übermittelt werden
- Daten oder Anfragen im erlaubten Paket könnten ungewollte Daten enthalten die ein Angreifer zu seinen Zwecken benutzen könnte um z.B. Schwächen in den Diensten oder hinter der Firewall zu benutzen.
- Normalerweise kann ein Fehler die Firewall unbrauchbar machen

Vorteile:

- Einfach und schnell zu implementieren
- Kann Warnungen vor Angriffen verursachen, bevor diese stattfinden (erkennen von Portscans)
- Geeignet um SYN-Attacken zu beenden

Beispiele für freie Paketfilter für Linux:

- [Iptables](#)
- [Ipchains](#)
- [SmoothWall](#)

## 12.3 Circuit Relay

Oder auch Circuit Level Gateways sind Firewalls, die Verbindungen validieren bevor die Erlaubnis für den Datenaustausch erteilt wird. Dies bedeutet, dass Pakete entsprechend dem Inhalt des Paketkopfes erlaubt oder verboten werden; dies aber hängt davon ab, ob die Verbindung an beiden Enden gültig entsprechend konfigurierbaren Regeln ist, bevor sie geöffnet oder Daten ausgetauscht werden. Filtern basiert auf:

- Quell-/Zieladresse
- Quell-/Zielport
- Zeitraum
- Protokoll
- Nutzer

- Passwort

Jeglicher Verkehr wird validiert, überwacht und Verkehr ohne diese Informationen wird verboten.

Schwächen:

- Operiert auf der Transportebene und kann u.U. grundlegende Veränderungen in der Programmierung die normalerweise die Transportfunktionen regelt erfordern.

## 12.4 Applikationsgateway

Der Gateway auf der Applikationsebene ist ein Proxy für eine Applikation, die Daten mit dem Remotesystem unter Verwendung seiner Clients austauscht. Er wird vor der Öffentlichkeit hinter einer DMZ oder einer Firewall ohne Verbindung zur Aussenwelt gesichert. Der Filter basiert auf:

- Erlauben oder verbieten basierend auf Herkunft/Ziel
- Entsprechend dem Paketinhalt
- Dateizugriff abhängig von Dateityp oder -Erweiterung beschränken

Vorteile:

- Dateien können zwischengespeichert werden - das erhöht die Netzwerkleitung
- Detailliertes aufzeichnen von Verbindungen
- Skaliert perfekt (manche Proxy-Server können die zwischengespeicherten Daten teilen)
- Kein direkter Zugriff von Aussen
- Kann Inhalte "on the fly" modifizieren

Schwächen:

- Die Konfiguration ist komplex

Applikationsgateways werden als die sicherste Lösung angesehen, da sie nicht als root laufen müssen und Richtung Internet nicht öffentlich sind.

Beispiel eines freien Applikationsgateways:

- [Squid](#)

## 12.5 Iptables

Um iptables ans Laufen zu kriegen, muss es im Kernel aktiviert werden. Ich habe sie als Module eingefügt (das Kommando iptables wird diese wenn benötigt laden) und den Kernel neu kompiliert. Für mehr Informationen zur Konfiguration von iptables lesen Sie [Iptables Tutorial Chapter 2: Preparations](#). Nachdem Sie den Kernel neu kompiliert haben (oder noch während der Kernel kompiliert wird) müssen Sie die iptables-Kommandos hinzufügen. Führen Sie einfach nur *emerge iptables* aus und alles sollte funktionieren.

Nun probieren Sie bitte ob alles funktioniert, indem Sie *iptables -L* ausführen. Wenn irgendetwas nicht funktioniert, dann sollten Sie die Konfiguration nochmals überprüfen.

Iptables ist der neue und extrem verbesserte Paketfilter in Linux 2.4.x. Es ist der Nachfolger von ipchains aus dem Linux 2.2.x Kernel. Eine der großen Verbesserungen ist, dass iptables nun in der Lage ist "stateful" Packet Filtering zu performen. Mit "stateful" Packet filtering is es möglich die Spur jeder errichteten TCP Verbindung zu verfolgen.

Eine TCP Verbindung besteht aus einer Serie von Paketen die Informationen über die Quelladresse, die Zieladresse und einem Zähler der das richtige Zusammensetzen der Daten ermöglicht. TCP ist im Gegensatz zu UDP ein verbindungsorientiertes Protokol, UDP ist verbindungslos.

Bei der Prüfung der Header der TCP Pakete kann ein "stateful" Paketfilter bestimmen, ob ein empfangenes TCP Paket zur einer bestehenden Verbindung gehört oder nicht und das Paket entweder akzeptieren oder wegwerfen.

Mit einem "stateless" Paketfilter is es möglich, dem Paket Filter Pakete durch das manipulieren des TCP Paket Header unterzuschieben die eigentlich gedroppt werden sollten. Dies kann durch das manipulieren des SYN Flag oder anderer Flags im TCP Header erreicht werden. Mit "stateful" Packet Filtering ist es möglich solche Pakete zu droppen, da Sie keiner bestehenden Verbindung zuzuordnen

sind. Damit wird auch die Möglichkeit von "stealth scans" verhindert, da solche Pakete ebenfalls keiner bestehenden Verbindung zuzuordnen sind.

Iptables bietet einige weitere Möglichkeiten wie zum Beispiel Wiederholungsbegrenzung (rate limiting). Diese Fähigkeit ist extrem nützlich, wenn man einen sicheren DoS (Denial of Service)-Angriff wie auch einen SYN-Angriff verhindern will.

Eine TCP-Verbindung wird durch einen sogenannten Drei-Wege-Handschlag aufgebaut. Wenn eine TCP Verbindung sendet der Client ein Paket mit einem SYN Flag an den Server. Wenn die Server-Seite das SYN Paket empfängt reagiert sie, in dem sie ein SYN+ACK Paket an den client schickt. Wenn der SYN+ACK vom Client empfangen wird, erkennt dieser wiederum mit einem dritten ACK Paket die Verbindung an.

Ein SYN-Angriff geschieht, wenn nur ein SYN-Paket gesendet wird, aber das Senden des SYN+ACK Paket fehlschlägt. Der Client kann ein Paket mit einer gefälschten IP-Adresse senden, da es keine Antwort benötigt. Der Server wird beim Empfang eines SYN Paket einen Eintrag in die Liste halb-geöffneter Verbindungen machen und dann auf den finalen ACK warten, bis der Eintrag gelöscht wird. Der Queue hat eine begrenzte Anzahl von Slots, wenn alle Slots belegt sind können keine neuen Verbindungen aufgebaut werden. Wenn der ACK nicht innerhalb eines begrenzten Zeitraums beim Server ankommt timed die Verbindung aus, der Eintrag wird aus der Queue gelöscht. Die Timeout Einstellungen variieren, liegen aber typischerweise im Bereich von 30 bis 60 Sekunden oder mehr. Die Clientseite initiiert die Attacke durch das Aussenden einer größtmöglichen Zahl von SYN Paketen mit verschiedenen Source IP Adressen. Dadurch wird die Liste halb-geöffneter Verbindungen schnell gefüllt, so dass andere Clients davon abgehalten werden eine Verbindung zu diesem Server aufzubauen.

Hier wird das Ratenlimit besonders hilfreich. Es ist möglich die Anzahl von SYN-Paketen von einer bestimmten Quelle zu begrenzen, aber durch Gebrauch von `-m limit --limit 1/s` begrenzt dies das Limit der SYN-Pakete für eins pro Quelle und daher begrenzt die SYN-Flut auf unsere Ressourcen.

Jetzt einiger praktischer Kram!

Wenn iptables in den Kernel geladen wird, hat es 5 Aufhänger an die Sie ihre Regeln hängen können. Sie heißen *INPUT*, *OUTPUT*, *FORWARD*, *PREROUTING* und *POSTROUTING*. Diese Listen nennt man Ketten, da sie per zugefügter Regel funktionieren und überprüfen die Regeln eine nach der anderen in der Reihenfolge wie sie hinzugefügt wurden. Wenn eine Regel auf ein Paket nicht zutrifft wird es an die nächste Regel weitergeleitet.

Sie können Regeln direkt in die 5 Hauptketten setzen oder Ketten erstellen und diese als Regel zu einer existierenden Kette hinzufügen. Iptables unterstützt die folgenden Optionen.

| <b>Option:</b> | <b>Beschreibung:</b>                                 |
|----------------|--|
| -A             | Anhängen   |
| -D             | Löschen  |
| -I             | Einfügen   |
| -R             | Ersetzen   |
| -L             | Auflisten  |
| -F             | Löscht alle Regeln in der Kette oder in allen Ketten |
| -Z             | Keine Counter in der Kette oder in allen Ketten      |
| -C             | Teste dieses Paket an der Kette                      |
| -N             | Erstellen einer neuen benutzerdefinierten Kette      |
| -X             | Löschen einer benutzerdefinierten Kette              |
| -P             | Richtlinie der Kette bezüglich des Ziels ändern      |
| -E             | Ändern des Kettennamens                              |
| -p             | Protokoll  |
| -s             | Quelladresse/maske                                   |
| -d             | Zieladresse/maske                                    |
| -i             | Eingabename (Ethernetname)                           |
| -o             | Ausgabename (Ethernetname)                           |
| -j             | Jump (Ziel für Regel)                                |
| -m             | Erweiterter Treffer (Kann erweiterung benutzen)      |
| -n             | Numerische Ausgabe von Adressen und Ports            |
| -t             | Zu ändernde Tabelle                                  |
| -v             | Ausführliche Ausgabe                                 |
| -x             | Zahlen Erweitern (exakte Werte anzeigen)             |
| -f             | Nur auf die zweiten oder weitere Pakete achten       |
| -V             | Paketversion   |
| --line-numbers | Zeilennummern mit ausgeben                           |

Zuerst werden wir versuchen alle ICMP-Pakete an unsere Maschine zu blocken - nur um uns mit iptables vertraut zu machen.

**Befehlsauflistung 56:** Alle ICMP-Pakete blocken

```
# iptables -A INPUT -p icmp -j DROP
```

Zuerst legen wir die Kette fest, an die es angehängt werden soll, dann das Protokoll und dann das Ziel. Das Ziel kann eine Benutzer spezifizierte Regel oder eines der speziellen Ziele *ACCEPT*, *DROP*, *REJECT*, *LOG*, *QUEUE*, *MASQUERADE* sein. In diesem Fall benutzen wir *DROP* daß das Paket ohne irgendeine Antwort an den Client fallen lässt.

Nun versuchen Sie ein *ping localhost*. Es wird nicht möglich sein, eine Antwort zu bekommen, da das komplette ICMP-Protokoll eingehend geblockt wird. Es wird auch nicht möglich sein, andere Maschinen azupingen, da die Pakete nicht mehr von den anderen Rechnern in unseren Rechner kommen können. Nun leeren Sie die Kette um ICMP wieder zum Laufen zu bekommen.

**Befehlsauflistung 57:** Alle Regeln leeren (Flush)

```
# iptables -F
```

Nun sehen wir uns die Zustandsmaschinerie in iptables an. Wenn wir eine Prüfung bezüglich des Verbindungszustandes an eth0 haben wollen, könnten wir dies folgendermassen aktivieren:

**Befehlsauflistung 58:** Pakete die zu einer bereits bestehenden Verbindung gehören akzeptieren

```
# iptables -A INPUT -i eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Dies wird in der INPUT-Kette alle Pakete die zu einer bereits bestehenden oder einer verwandten Verbindung gehören akzeptieren. Man könnte auch jedes Paket, dass nicht in der Zustandstabelle abgedeckt wurde fallen lassen, indem man *iptables -A INPUT -i eth0 -m state --state INVALID -j DROP* direkt davor aufruft. Das aktiviert die Zustandssteuerung in iptables da es die Erweiterung state lädt. Wenn Sie nun von Aussen mit Ihrer Maschine, dann könnten Sie *--state NEW* benutzen. Iptables enthält einige unterschiedliche Module für unterschiedliche Anwendungszwecke. Einige dieser Module sind:

| Modul/<br>Treffer | Beschreibung   | Erweiterte Optionen  |
|-------------------|--|--|
| mac               | Prüfung auf die Quell-MAC-Adressen der eingehenden Pakete. | --mac-source   |
| state             | Prüfung auf Zustand  | --state (passende Werte sind ESTABLISHED,RELATED, INVALID, NEW)                    |
| limit             | Trefferrate begrenzen                                      | --limit, --limit-burst   |
| owner             | Prüfung auf diverse Fähigkeiten des Paketgenerators        | --uid-owner userid --gid-owner groupid --pid-owner processid --sid-owner sessionid |
| unclean           | Diverse Gültigkeitsprüfungen auf den Paketen               |  |

Lassen Sie uns nun eine benutzerdefinierte Kette erstellen und in einer der existierenden Ketten einbetten:

**Befehlsauflistung 59:** Benutzerdefinierte Kette erstellen

```
// Neue Kette mit einer Regel erstellen
# iptables -X mychain
# iptables -N mychain
# iptables -A mychain -i eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT
// Die Standardrichtlinie sagt, dass jeglicher ausgehender Verkehr erlaubt ist, aber eingehender
# iptables -P OUTPUT ACCEPT
# iptables -P INPUT DROP
// And add it to the INPUT chain
# iptables -A INPUT -j mychain
```

Indem man die Regel in die INPUT-Kette einpasst bekommt man die Richtlinie: Alles darf raus, aber alles reinkommende wird verworfen ("gedroppt").

Wenn Sie mehr Dokumentation haben wollen, dann werfen Sie einen Blick auf die [Netfilter/iptables Dokumentation](#)

Schauen wir uns nun ein komplettes Beispiel an. In diesem Falle sagt meine Firewall/Gateway Richtlinie:

- Verbindungen zur Firewall wird nur über SSH erlaubt (Port 22)
- Das lokale Netz soll Zugriff auf HTTP, HTTPS und SSH haben (DNS sollte auch erlaubt sein)
- ICMP-Verkehr könnte kritische Daten enthalten und sollte deswegen nicht erlaubt sein. Natürlich gibt es einige Ausnahmen
- Portscans sollten erkannt und aufgezeichnet werden
- SYN-Angriffe sollten abgewehrt werden
- Jeglicher anderer Verkehr sollte blockiert und aufgezeichnet werden

**Befehlsauflistung 60:** /etc/init.d/firewall

```
#!/sbin/runscript
IPTABLES=/sbin/iptables
IPTABLESSAVE=/sbin/iptables-save
IPTABLESRESTORE=/sbin/iptables-restore
FIREWALL=/etc/firewall.rules
DNS1=212.242.40.3
DNS2=212.242.40.51
#Innen
IIP=10.0.0.2
IINTERFACE=eth0
LOCAL_NETWORK=10.0.0.0/24
#Aussen
OIP=217.157.156.144
OINTERFACE=eth1

opts="${opts} showstatus panic save restore showoptions rules"

depend() {
    need net proctparam
}

rules() {
    stop
    ebegin "Setze interne Regeln"

    einfo "Setze Standardregel auf Fallenlassen"
    $IPTABLES -P FORWARD DROP
    $IPTABLES -P INPUT DROP
    $IPTABLES -P OUTPUT DROP

    #Standardregel
    einfo "Erstelle Zustands-Kette"
    $IPTABLES -N allowed-connection
    $IPTABLES -F allowed-connection
    $IPTABLES -A allowed-connection -m state --state ESTABLISHED,RELATED -j ACCEPT
    $IPTABLES -A allowed-connection -i $IINTERFACE -m limit -j LOG --log-prefix \
        "Böses Paket von ${IINTERFACE}:"
    $IPTABLES -A allowed-connection -j DROP

    #ICMP Verkehr
    einfo "Erstelle ICMP-Kette"
    $IPTABLES -N icmp_allowed
    $IPTABLES -F icmp_allowed
    $IPTABLES -A icmp_allowed -m state --state NEW -p icmp --icmp-type \
        time-exceeded -j ACCEPT
    $IPTABLES -A icmp_allowed -m state --state NEW -p icmp --icmp-type \
        destination-unreachable -j ACCEPT
    $IPTABLES -A icmp_allowed -p icmp -j LOG --log-prefix "Bad ICMP traffic:"
    $IPTABLES -A icmp_allowed -p icmp -j DROP

    #Eingehender Verkehr
    einfo "Erstelle Kette für eingehenden SSH-Verkehr"
    $IPTABLES -N allow-ssh-traffic-in
    $IPTABLES -F allow-ssh-traffic-in
    #Flood-Schutz
    $IPTABLES -A allow-ssh-traffic-in -m limit --limit 1/second -p tcp --tcp-flags \
        ALL RST --dport ssh -j ACCEPT
    $IPTABLES -A allow-ssh-traffic-in -m limit --limit 1/second -p tcp --tcp-flags \
        ALL FIN --dport ssh -j ACCEPT
```

```

$IPTABLES -A allow-ssh-traffic-in -m limit --limit 1/second -p tcp --tcp-flags \
    ALL SYN --dport ssh -j ACCEPT
$IPTABLES -A allow-ssh-traffic-in -m state -state RELATED,ESTABLISHED -p tcp -dport ssh -j ACC

#Ausgehender Verkehr
einfo "Erstelle Kette für ausgehenden SSH-Verkehr"
$IPTABLES -N allow-ssh-traffic-out
$IPTABLES -F allow-ssh-traffic-out
$IPTABLES -A allow-ssh-traffic-out -p tcp --dport ssh -j ACCEPT

einfo "Erstelle Kette für ausgehenden DNS-Verkehr"
$IPTABLES -N allow-dns-traffic-out
$IPTABLES -F allow-dns-traffic-out
$IPTABLES -A allow-dns-traffic-out -p udp -d $DNS1 --dport domain \
    -j ACCEPT
$IPTABLES -A allow-dns-traffic-out -p udp -d $DNS2 --dport domain \
    -j ACCEPT

einfo "Creating outgoing http/https traffic chain"
$IPTABLES -N allow-www-traffic-out
$IPTABLES -F allow-www-traffic-out
$IPTABLES -A allow-www-traffic-out -p tcp --dport www -j ACCEPT
$IPTABLES -A allow-www-traffic-out -p tcp --dport https -j ACCEPT

#Portscanner fangen
einfo "Erstelle Portscan-Erkennungs-Kette"
$IPTABLES -N check-flags
$IPTABLES -F check-flags
$IPTABLES -A check-flags -p tcp --tcp-flags ALL FIN,URG,PSH -m limit \
    --limit 5/minute -j LOG --log-level alert --log-prefix "NMAP-XMAS:"
$IPTABLES -A check-flags -p tcp --tcp-flags ALL FIN,URG,PSH -j DROP
$IPTABLES -A check-flags -p tcp --tcp-flags ALL ALL -m limit --limit \
    5/minute -j LOG --log-level 1 --log-prefix "XMAS:"
$IPTABLES -A check-flags -p tcp --tcp-flags ALL ALL -j DROP
$IPTABLES -A check-flags -p tcp --tcp-flags ALL SYN,RST,ACK,FIN,URG \
    -m limit --limit 5/minute -j LOG --log-level 1 --log-prefix "XMAS-PSH:"
$IPTABLES -A check-flags -p tcp --tcp-flags ALL SYN,RST,ACK,FIN,URG -j DROP
$IPTABLES -A check-flags -p tcp --tcp-flags ALL NONE -m limit \
    --limit 5/minute -j LOG --log-level 1 --log-prefix "NULL_SCAN:"
$IPTABLES -A check-flags -p tcp --tcp-flags ALL NONE -j DROP
$IPTABLES -A check-flags -p tcp --tcp-flags SYN,RST SYN,RST -m limit \
    --limit 5/minute -j LOG --log-level 5 --log-prefix "SYN/RST:"
$IPTABLES -A check-flags -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
$IPTABLES -A check-flags -p tcp --tcp-flags SYN,FIN SYN,FIN -m limit \
    --limit 5/minute -j LOG --log-level 5 --log-prefix "SYN/FIN:"
$IPTABLES -A check-flags -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP

# Ungültige Zustände in den Ketten einpassen
einfo "Passe Ketten in INPUT an"
$IPTABLES -A INPUT -m state --state INVALID -j DROP
$IPTABLES -A INPUT -j icmp_allowed
$IPTABLES -A INPUT -j check-flags
$IPTABLES -A INPUT -i lo -j ACCEPT
$IPTABLES -A INPUT -j allow-ssh-traffic-in
$IPTABLES -A INPUT -j allowed-connection

einfo "Passe Ketten in FORWARD an"
$IPTABLES -A FORWARD -m state --state INVALID -j DROP
$IPTABLES -A FORWARD -j icmp_allowed
$IPTABLES -A FORWARD -j check-flags
$IPTABLES -A FORWARD -o lo -j ACCEPT
$IPTABLES -A FORWARD -j allow-ssh-traffic-in
$IPTABLES -A FORWARD -j allow-www-traffic-out
$IPTABLES -A FORWARD -j allowed-connection

einfo "Passe Ketten in OUTPUT an"
$IPTABLES -A OUTPUT -m state --state INVALID -j DROP
$IPTABLES -A OUTPUT -j icmp_allowed
$IPTABLES -A OUTPUT -j check-flags
$IPTABLES -A OUTPUT -o lo -j ACCEPT
$IPTABLES -A OUTPUT -j allow-ssh-traffic-out
$IPTABLES -A OUTPUT -j allow-dns-traffic-out

```

```

$IPTABLES -A OUTPUT -j allow-www-traffic-out
$IPTABLES -A OUTPUT -j allowed-connection

#erlaube den Clients über NAT (Network Address Translation) zu routen
$IPTABLES -t nat -A POSTROUTING -o $IINTERFACE -j MASQUERADE
eend $?
}

start() {
  ebegin "Starte Firewall"
  if [ -e "${FIREWALL}" ]; then
    restore
  else
    einfo "${FIREWALL} existiert nicht. Benutze Standardregeln."
    rules
  fi
  eend $?
}

stop() {
  ebegin "Halte Firewall an"
  $IPTABLES -F
  $IPTABLES -t nat -F
  $IPTABLES -X
  $IPTABLES -P FORWARD ACCEPT
  $IPTABLES -P INPUT ACCEPT
  $IPTABLES -P OUTPUT ACCEPT
  eend $?
}

showstatus() {
  ebegin "Status"
  $IPTABLES -L -n -v --line-numbers
  einfo "NAT status"
  $IPTABLES -L -n -v --line-numbers -t nat
  eend $?
}

panic() {
  ebegin "Setze Panikregeln"
  $IPTABLES -F
  $IPTABLES -X
  $IPTABLES -t nat -F
  $IPTABLES -P FORWARD DROP
  $IPTABLES -P INPUT DROP
  $IPTABLES -P OUTPUT DROP
  $IPTABLES -A INPUT -i lo -j ACCEPT
  $IPTABLES -A OUTPUT -o lo -j ACCEPT
  eend $?
}

save() {
  ebegin "Sichere Firewallregeln"
  $IPTABLESSAVE > $FIREWALL
  eend $?
}

restore() {
  ebegin "Stelle Firewallregeln wieder her"
  $IPTABLESRESTORE < $FIREWALL
  eend $?
}

restart() {
  svc_stop; svc_start
}

showoptions() {
  echo "Benutzung: $0 {start|save|restore|panic|stop|restart|showstatus}"
  echo "start)      Wird die Standardeinstellung wieder herstellen oder anderndfalls zu Regeln z
  echo "stop)      Alle Regeln löschen und alles akzeptieren"
  echo "rules)      zu Einstellungen der neuen regeln zwingen"
}

```

```

echo "save)          speichert die Regeln in ${FIREWALL}"
echo "restore)       stellt die Regeln von ${FIREWALL} wieder her"
echo "showstatus)    Status anzeigen"
}

```

Kostenloser Ratschlag für das Erstellen einer Firewall:

- Erstellen Sie vor der Implementierung eine Richtlinie für die Firewall
- Halten Sie sie einfach
- Erlangen Sie Wissen über die Protokolle (lesen Sie das [RFC \(Request For Comments\)](#))
- Denken Sie daran, dass eine Firewall ein weiteres Paket ist, das als root läuft
- Testen Sie die Firewall

Wenn Sie denken, dass iptables schwer zu verstehen sind oder es zu lange dauert eine sinnvolle Firewall zu erstellen, dann könnten Sie auch [Shorewall](#) benutzen. Es benutzt im Grunde genommen iptables um eine Firewall zu erstellen, aber es konzentriert sich auf Regeln und nicht auf spezielle Protokolle.

## 12.6 Squid

Squid ist ein sehr leistungsstarker Proxy Server, er hat eingebaute Filter, lehnt Traffic auf Grund folgender Merkmale ab: Zeit, regelmäßiger Ausdruckspfad/URI, Quell- und Zieladresse (IP), Domäne, Browser, der authentifizierte Benutzername, Mime-Typ und Port (Protokoll). Wahrscheinlich habe ich einige Funktionen vergessen, aber es ist schwer jede Funktion der gesamten Funktionsliste abzudecken.

Im folgenden Beispiel habe ich einen Banner Filter hinzugefügt anstatt eines Filters basierend auf pornographischen Seiten. Der Grund dafür ist, dass Gentoo.org **nicht** als eine pornographische Seite aufgelistet werden sollte. Ausserdem will ich meine Zeit nicht damit verbringen einige gute Seiten für Sie zu finden.

In diesen Fall diktiert meine Richtlinie:

- Surfen (HTTP/HTTPS) ist während der Arbeitszeiten erlaubt (MO-FR 8-16 und SA 8-13), wenn sie länger da sind, sollten sie arbeiten und nicht surfen.
- Downloaden ist nicht erlaubt (.exe, .com, .arj, .zip, .asf, .avi, .mpg, .mpeg etc.)
- Banner sind unerwünscht, daher werden sie herausgefiltert und mit einem transparenten GIF ersetzt (hier können Sie kreativ werden!)
- Jede andere kommende oder gehende Verbindung mit dem Internet ist nicht erlaubt.

Dies wird in 4 **einfachen** Schritten implementiert .

**Befehlsauflistung 61:** /etc/squid/squid.conf

```

# Anbinden an eine IP und einen Port
http_port 10.0.2.1:3128

# Standardkonfiguration
hierarchy_stoplist cgi-bin ?
acl QUERY urlpath_regex cgi-bin \?
no_cache deny QUERY

# Hinzufügen von grundlegenden Listen der Zugriffskontrolle
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255

# Hinzufügen wer auf diesen Proxy Server zugreifen kann
acl localnet src 10.0.0.0/255.255.0.0

# Und welche Ports
acl SSL_ports port 443
acl Safe_ports port 80
acl Safe_ports port 443
acl purge method PURGE

# Hinzufügen von Listen zur Zugriffskontrolle basierend
# auf regelmäßigen Ausdrücken innerhalb von URLs
acl archives urlpath_regex "/etc/squid/files.acl"
acl url_ads url_regex "/etc/squid/banner-ads.acl"

```

```

# Hinzufügen von Listen zur Zugriffskontrolle basierend
# auf Datum und Uhrzeit
acl restricted_weekdays time MTWHF 8:00-17:00
acl restricted_weekends time A 8:00-13:00

acl CONNECT method CONNECT

# Erlauben von Managmentzugriff von Localhost
http_access allow manager localhost
http_access deny manager

# Nur Purge Anfragen von Localhost erlauben
http_access allow purge localhost
http_access deny purge

# Verweigern von Anfragen an unbekannte Ports
http_access deny !Safe_ports

# Verweigern von CONNECT an alle ausser SSL Ports
http_access deny CONNECT !SSL_ports

# Meine eigenen Regeln

# Hinzufügen einer Seite zur Darstellung,
# wenn ein Banner entfernt wurde
deny_info NOTE_ADS_FILTERED url_ads

# Dann diese verweigern
http_access deny url_ads

# Verweigern aller Archive
http_access deny archives

# Begrenzung des Zugriffs auf Arbeitszeiten
http_access allow localnet restricted_weekdays
http_access allow localnet restricted_weekends

# Verweigern von allem anderen
http_access deny all

```

Als nächstes fügen Sie alle Dateitypen ein, von denen sie wollen, dass ihre Benutzer sie nicht herunterladen können. Ich habe zip, viv, exe, mp3, rar, ace, avi, mov, mpg, mpeg, au, ra, arj, tar, gz und z Dateien gewählt.

**Befehlsauflistung 62:** /etc/squid/files.acl

```

\[Zz][Ii][pP]$
\[Vv][Ii][Vv].*
\[Ee][Xx][Ee]$
\[Mm][Pp]3$
\[Rr][Aa][Rr]$
\[Aa][Cc][Ee]$
\[Aa][Ss][Ff]$
\[Aa][Vv][Ii]$
\[Mm][Oo][Vv]$
\[Mm][Pp][Gg]$
\[Mm][Pp][Ee][Gg]$
\[Aa][Uu]$
\[Rr][Aa]$
\[Aa][Rr][Jj]$
\[Tt][Aa][Rr]$
\[GgZz]$
\[Zz]$

```

**Notiz**

Beachten Sie bitte die [] mit Gross- und Kleinbuchstaben fuer jeden Buchstaben. Dies dient dazu, dass niemand es umgehen kann indem er eine Datei mit AvI abrufen anstatt avi.

Als nächstes fügen wir die regelmäßigen Ausdrücke ein um Banner zu identifizieren. Sie werden wahrscheinlich viel kreativer sein als ich:

### **Befehlsauflistung 63:** /etc/squid/banner-ads.acl

```
/adv/*.gif$
/[Aa]ds/*.gif$
/[Aa]d[Pp]ix/
/[Aa]d[Ss]erver
/[Aa][Dd]/*.[[GgJj]][IiPp][FfGg]$
/[Bb]annerads/
/adbanner.*.[[GgJj]][IiPp][FfGg]$
/images/ad/
/reklame/
/RealMedia/ads/*.
^http://www\.submit-it.*
^http://www\.eads.*
^http://ads\.
^http://ad\.
^http://ads02\.
^http://adaver.*\.
^http://adforce\.
adbot\.com
/ads/*.gif.*
_ad\.*cgi
/Banners/
/SmartBanner/
/Ads/Media/Images/
^http://static\.wired\.com/advertising/
^http://*\.dejane\.com/ads/
^http://adfu\.blockstackers\.com/
^http://ads2\.zdnet\.com/adverts
^http://www2\.burstnet\.com/gifs/
^http://www\.valueclick\.com/cgi-bin/cycle
^http://www\.altavista\.com/av/gifs/ie_horiz\.gif
```

Nun der letzte Teil: Wir wollen diese Datei anzeigen, wenn das Banner entfernt wird. Es ist grundlegend eine halbe HTML Datei mit einem 4x4 transparenten GIF Bild.

### **Befehlsauflistung 64:** /etc/squid/errors/NOTE\_ADS\_FILTERED

```
<HTML>
<HEAD>
<META HTTP-EQUIV="REFRESH" CONTENT="0; URL=http://www.insecurity.dk/images/4x4.gif">
<TITLE>FEHLER: Die angeforderte URL konnte nicht angezeigt werden</TITLE>
</HEAD>
<BODY>
<H1>Anzeige gefiltert!</H1>
```

### **Notiz**

Schliessen sie die <HTML> <BODY> Tags nicht. Dies wird von Squid erledigt.

Wie Sie sehen können hat Squid eine Vielzahl von Möglichkeiten und ist sehr effektiv zum Filtern und als Proxy. Es kann sogar alternative Squid Proxys benutzen um an sehr grosse Netzwerke angepasst zu werden. Die Konfiguration, die ich hier aufgelistet habe ist hauptsächlich fuer kleine Netzwerke mit 1-20 Benutzern geeignet.

Jedoch die Kombination von Paketfilterung (iptables) und dem Anwendungsgateway (squid) ist wahrscheinlich die beste Lösung, selbst wenn Squid selber an einem sicheren Ort stationiert ist und niemand von ausserhalb darauf zugreifen kann. Wir müssen uns weiterhin Gedanken machen um Angriffe von Innen.

Nun müssen Sie den Proxy Server in die Einstellungen des Browsers Ihrer Benutzers einbinden. Das Gateway verhindert, dass die Benutzer jeglichen Kontakt mit der Aussenwelt haben, solange sie nicht den Proxy benutzen.

### **Notiz**

In Mozilla geschieht dies in Bearbeiten->Einstellungen->Erweitert->Proxies (bzw. Edit->Preferences->Advanced->Proxies).

Es kann auch transparent geschehen, indem man iptables benutzt um den gesamten Traffic an einen Squid Proxy weiterzuleiten. Dies kann erreicht werden indem man eine Weiterleitungs/Prerouting Regel fürs Gateway hinzufügt:

### **Befehlsauflistung 65:** Ermöglichen von Portweiterleitung an unseren Proxy Server

```
# iptables -t nat -A PREROUTING -p tcp --dport 80 -j DNAT --to proxyhost:3128
# iptables -t nat -A PREROUTING -p tcp --dport 443 -j DNAT --to proxyhost:3128
```

## 12.7 Also was haben wir gelernt?

Wir lernten, dass:

- Eine Firewall selbst ein Risiko sein kann. Eine schlecht konfigurierte Firewall ist schlechter als überhaupt keine.
- Wie man ein grundlegendes Gateway und einen transparenten Proxy erstellt
- Der Schlüssel zu einer guten Firewall ist das Protokoll zu kennen, das Sie zulassen wollen
- Dass IP Traffic nicht immer legitime Daten beinhaltet, z.B. ein ICMP Paket mit zusätzlicher Nutzlast
- Wie man SYN Angriffe vereitelt
- Filtern von HTTP Traffic indem man anstössige Bilder und Downloads von Viren entfernt.
- Kombinieren von Paketfiltern und Anwendungsgateways geben eine bessere Kontrolle.

Nun, wenn sie **wirklich** müssen, schaffen Sie sich eine Firewall an, die ihre Bedürfnisse deckt.

## 13. Auffinden von Eindringlingen

### 13.1 Aide (Advanced Intrusion Detection Environment)

Aide ist ein Host basierendes Eindringlingserkennungssystem (eine kostenlose Alternative zu Tripwire). Und wenn Sie bereits mit Tripwire vertraut sind, sollten sie keine Schwierigkeiten haben die Konfigurationsdateien für Aide zu erlernen.

Die Konfigurationsdatei basiert auf regelmäßigen Ausdrücken, Makros und Regeln für Dateien und Verzeichnisse. Wir haben die folgenden Makros:

#### MakroBeschreibung

ifdef wenn definiert

ifndef wenn nicht definiert

define definiert eine Variable

undef undefiniert eine Variable

ifhost wenn "hostname"

ifnhostwenn "hostname" nicht

endif Endif muss benutzt werden nach jedem der obrigen Makros ausser define und undef

#### Syntax

@@ifdef "name"

@@ifndef "name"

@@define "name"

"value"

@@undef "name"

@@ifhost "hostname"

@@ifnhost "hostname"

@@endif

Diese Makros sind sehr praktisch, wenn sie mehr als ein Gentoo System haben und auf allen Aide benutzen wollen. Aber nicht alle Maschinen oder vielleicht sogar Benutzer benutzen denselben Dienst.

Als nächstes haben wir Gruppen von Flags um überprüfungen an Dateien und Ordnern durchzuführen. Diese sind eine Kombination aus Berechtigungen, Dateieigenschaften und kryptographischen Hashes/Checksummen.

#### Flag Beschreibung

p Berechtigungen

i Inode

n Anzahl der Links

u Benutzer

g Gruppe

s Grösse

b Blockzahl

m mtime

a atime

c ctime

S überprüfung ob die Grösse wächst

md5 MD5 Checksumme

sha1 SHA1 Checksumme

rmd160RMD160 Checksumme

tiger Tiger Checksumme

R p+i+n+u+g+s+m+c+md5

L p+i+n+u+g

E Leere Gruppe

> Wachsende Protokolldatei p+u+g+i+n+S

Und wenn Aide mit mhash Unterstützung kompiliert ist, hat es noch einige weitere Funktionen:

### Flag Beschreibung

havalHAVAL Checksumme  
gost GOST Checksumme  
crc32CRC32 Checksumme

Nun können sie ihre eigenen auf den oben genannten Flags basierenden Regeln definieren, indem Sie diese folgendermassen kombinieren:

### Befehlsauflistung 66: Erstellen eines Regelsatzes für AIDE

```
All=R+a+sha1+rmd160
Norm=s+n+b+md5+sha1+rmd160
```

Das letzte was wir tun müssen um unsere eigene Konfigurationsdatei zu erstellen ist zu schauen wie man diese Regeln einer Datei oder einem Verzeichnis hinzufügt. Grundlegend tippen sie einfach den Datei- oder Verzeichnisnamen und die Regel ein. Aide wird alle Dateien rekursiv hinzufügen, solange Sie nicht etwas anderes angeben.

### FlagBeschreibung

! Diese Datei oder dieses Veriechnis nicht hinzufügen.  
= Dieses Verzeichnis hinzufügen, aber nicht rekursiv.

Lassen sie uns also ein vollständiges Beispiel betrachten

### Befehlsauflistung 67: /etc/aide/aide.conf

```
@@ifndef TOP DIR
@@define TOPDIR /
@@endif

@@ifndef AIDEDIR
@@define AIDEDIR /etc/aide
@@endif

@@ifhost smbserve
@@define smbactive
@@endif

# Der Ort der Datenbank die gelesen werden soll.
database=file:@@{AIDEDIR}/aide.db

# Der Ort der Datenbank, die erstellt werden soll.
database_out=file:aide.db.new

verbose=20
report_url=stdout

# Regeldefinition
All=R+a+sha1+rmd160
Norm=s+n+b+md5+sha1+rmd160

@@{TOPDIR} Norm
!@@{TOPDIR}etc/aide
!@@{TOPDIR}dev
!@@{TOPDIR}proc
!@@{TOPDIR}root
!@@{TOPDIR}tmp
!@@{TOPDIR}var/log
!@@{TOPDIR}var/run
!@@{TOPDIR}usr/portage
@@ifdef smbactive
!@@{TOPDIR}etc/smb/private/secrets.tdb
@@endif
=@@{TOPDIR}home Norm
```

Im obrigen Beispiel definieren wir einige Makros, die angeben wo das topdir startet und wo das Aide Verzeichnis ist. Aide überprüft die `/etc/aide/aide.db` Datei wenn die Intigrität einer Datei überprüft wird. Jedoch wenn ein Update vorgenommen wird oder eine neue Datei erstellt wird, speichert es die Informationen in `/etc/aide/aide.db.new`. Dies geschieht, damit die ursprünglich Datenbankdatei nicht automatisch überschrieben wird. Die Option `report_URL` ist eine "noch kommende" Funktion die

wirklich noch keine Bedeutung hat. Die Absicht der Autoren war es, dass es möglich wäre eine Email zu senden oder vielleicht sogar ein Script auszuführen.

Nach der Konfiguration sollten Sie Ihre Datenbankdatei erstellen indem Sie `aide -i` ausführen und dann die Datei `/etc/aide/aide.db.new` nach `/etc/aide/aide.db` kopieren und den Check zu cron hinzufügen durch `crontab -e` als `root`.

### Notiz

Abhängig von ihren CPU, Festplattenzugriff und den benutzten Flags für Dateien, kann dies einige Zeit in Anspruch nehmen

**Befehlsauflistung 68:** Aide als cronjob einrichten

```
0 3 * * * /usr/bin/aide -u
```

### Notiz

Denken sie daran es so einzustellen, dass sie die Post für `root` bekommen. Ansonsten werden Sie niemals wissen was Aide berichtet.

In diesem Fall läuft es einmal um 03 Uhr. Dies geschieht dann, denn ich will die Benutzer beim Arbeiten nicht stören. Beachten Sie, dass ich die `-u` (Update) Option benutzen statt `-C` (überprüfen). Denn `-u` überprüft auch die Dateien und überschreibt nicht die Originaldatenbankdatei, was Zeit spart, denn so müssen sie nur eine Datei kopieren, wenn Veränderungen festgestellt werden. überprüfen sie einfach die Veränderungen um zu sehen, ob Sie es selbst waren oder ein Angreifer der die Veränderungen gemacht hat bevor sie diese kopieren!

Nun gibt es einige Probleme damit die Datenbankdateien lokal zu speichern, denn der Angreifer wird (Wenn er weiss, dass Aide installiert ist) höchstwahrscheinlich versuchen die Datenbankdatei zu verändern, ein Update bei der Datenbankdatei durchzuführen oder `/usr/bin/aide` zu verändern. Deswegen sollten sie eine CD erstellen oder anderes Medium auf das Sie eine Kopie der Datenbankdatei und der Aide Binärdateien ablegen.

Weitere Informationen gibt es auf der [AIDE](#) Projektseite.

## 13.2 Snort

Snort ist ein "Network Intrusion Detection System (NIDS)". Zur Installation und Konfiguration benutzen Sie die folgenden Beispiele.

**Befehlsauflistung 69:** Dem System einen Anwender hinzufügen

```
# user add snort -d /var/log/snort -s /dev/null
# chown -R snort /var/log/snort
```

**Befehlsauflistung 70:** /etc/conf.d/snort

```
PID FILE=/var/run/snort_eth0.pid
MODE="full"
NETWORK="10.0.0.0/24"
LOG DIR="/var/log/snort"
CONF=/etc/snort/snort.conf
SNORT_OPTS="-D -s -u snort -dev -l $LOGDIR -h $NETWORK -c $CONF"
```

**Befehlsauflistung 71:** /etc/snort/snort.conf

```
// Schritt 1
var HOME_NET 10.0.0.0/24
var EXTERNAL_NET any
var SMTP $HOME_NET
var HTTP_SERVERS $HOME_NET
var SQL_SERVERS $HOME_NET
var DNS_SERVERS [10.0.0.2/32,212.242.40.51/32]
var RULE_PATH ./

// Schritt 2
preprocessor frag2
preprocessor stream4: detect_scans detect_state_problems detect_scans disable_evasion_alerts
preprocessor stream4_reassemble: ports all
preprocessor http_decode: 80 8080 unicode iis_alt_unicode double_encode iis_flip_slash full_whit
preprocessor rpc_decode: 111 32771
preprocessor bo: -noprute
preprocessor telnet_decode
```

```

// Schritt 3
include classification.config

// Schritt 4
include $RULE_PATH/bad-traffic.rules
include $RULE_PATH/exploit.rules
include $RULE_PATH/scan.rules
include $RULE_PATH/finger.rules
include $RULE_PATH/ftp.rules
include $RULE_PATH/telnet.rules
include $RULE_PATH/smtp.rules
include $RULE_PATH/rpc.rules
include $RULE_PATH/rservices.rules
include $RULE_PATH/dos.rules
include $RULE_PATH/ddos.rules
include $RULE_PATH/dns.rules
include $RULE_PATH/tftp.rules
include $RULE_PATH/web-cgi.rules
include $RULE_PATH/web-coldfusion.rules
include $RULE_PATH/web-iis.rules
include $RULE_PATH/web-frontpage.rules
include $RULE_PATH/web-misc.rules
include $RULE_PATH/web-attacks.rules
include $RULE_PATH/sql.rules
include $RULE_PATH/x11.rules
include $RULE_PATH/icmp.rules
include $RULE_PATH/netbios.rules
include $RULE_PATH/misc.rules
include $RULE_PATH/attack-responses.rules
include $RULE_PATH/backdoor.rules
include $RULE_PATH/shellcode.rules
include $RULE_PATH/policy.rules
include $RULE_PATH/porn.rules
include $RULE_PATH/info.rules
include $RULE_PATH/icmp-info.rules
include $RULE_PATH/virus.rules
# include $RULE_PATH/experimental.rules
include $RULE_PATH/local.rules

```

#### **Befehlsauflistung 72: /etc/snort/classification.config**

```

config classification: not-suspicious,Not Suspicious Traffic,3
config classification: unknown,Unknown Traffic,3
config classification: bad-unknown,Potentially Bad Traffic, 2
config classification: attempted-recon,Attempted Information Leak,2
config classification: successful-recon-limited,Information Leak,2
config classification: successful-recon-largescale,Large Scale Information Leak,2
config classification: attempted-dos,Attempted Denial of Service,2
config classification: successful-dos,Denial of Service,2
config classification: attempted-user,Attempted User Privilege Gain,1
config classification: unsuccessful-user,Unsuccessful User Privilege Gain,1
config classification: successful-user,Successful User Privilege Gain,1
config classification: attempted-admin,Attempted Administrator Privilege Gain,1
config classification: successful-admin,Successful Administrator Privilege Gain,1

# NEW CLASSIFICATIONS
config classification: rpc-portmap-decode,Decode of an RPC Query,2
config classification: shellcode-detect,Executable code was detected,1
config classification: string-detect,A suspicious string was detected,3
config classification: suspicious-filename-detect,A suspicious filename was detected,2
config classification: suspicious-login,An attempted login using a suspicious username was detected,2
config classification: system-call-detect,A system call was detected,2
config classification: tcp-connection,A TCP connection was detected,4
config classification: trojan-activity,A Network Trojan was detected, 1
config classification: unusual-client-port-connection,A client was using an unusual port,2
config classification: network-scan,Detection of a Network Scan,3
config classification: denial-of-service,Detection of a Denial of Service Attack,2
config classification: non-standard-protocol,Detection of a non-standard protocol or event,2
config classification: protocol-command-decode,Generic Protocol Command Decode,3
config classification: web-application-activity,access to a potentially vulnerable web application,1
config classification: web-application-attack,Web Application Attack,1

```

```
config classification: misc-activity,Misc activity,3
config classification: misc-attack,Misc Attack,2
config classification: icmp-event,Generic ICMP event,3
config classification: kickass-porn,SCORE! Get the lotion!,1
```

Weitere Informationen gibt es auf der [Snorts](#) Webseite.



## >> Gentoo Distcc Dokumentation

[Bitte Kapitel auswählen]

### 1. Einleitung

#### 1.1 Was ist distcc?

Distcc ist ein Programm, welches sich um die Verteilung von Kompilierungsprozessen an teilnehmende Rechner kümmert. Es besteht aus einem Serverteil, *distccd* und einem Clientprogramm, *distcc*. Mit etwas Konfigurationsaufwand funktioniert distcc mit [ccache](#) und Portage zusammen.

### 2. Einrichtung

#### 2.1 Abhängigkeiten

**Befehlsauflistung 1:** Distcc Abhängigkeiten (Version 2.8 bis 2.10)

```
>=sys-apps/portage-2.0.46-r11
// Distcc-2.11 und höher erfordern >=sys-apps/portage-2.0.49-r6
>=sys-devel/gcc-config-1.3.1
sys-apps/shadow
// (Version 2.8 bis 2.11): Distcc hat folgende weitere Abhängigkeiten, wenn Sie gtk in Ihren USE
>=x11-libs/gtk+-2.2.1
```

**Befehlsauflistung 2:** Distcc Abhängigkeiten (Version 2.11 und höher)

```
>=sys-apps/portage-2.0.49-r6
>=sys-devel/gcc-config-1.3.1
sys-apps/shadow
// In dieser Revision können Sie zwischen einer Gnome und GTK GUI Überwachung auswählen
// Für GTK:
>=x11-libs/gtk+-2.0.0
>=gnome-base/libglade-2.0.0
x11-libs/pango
// Für Gnome:
>=x11-libs/gtk+-2.0.0
>=gnome-base/libglade-2.0.0
x11-libs/pango
>=gnome-base/libgnomeui-2.0.0.0
>=gnome-base/libgnome-2.0.0
```

#### 2.2 Installing Distcc

Distcc unter Portage einzurichten ist sehr einfach. Setzen Sie einfach Ihre gewünschten USE Flags und führen Sie **emerge distcc** aus. Allerdings gibt es einige Optionen über die Sie bescheid wissen sollten.

Distcc kommt mit einem grafischen Überwachungstool, um einzelne Aufgaben, die an andere PCs verteilt werden, aufzuzeichnen. Als Gnome Benutzer sollten sie 'gnome' in Ihren USE Flags setzen. Verwenden Sie kein Gnome, wollen jedoch die Funktionalität des grafischen Tools nicht missen, dann sollten Sie 'gtk' in den USE Flags gesetzt haben.

#### 2.3 Portage so einrichten, dass es Distcc benutzt

Distcc unter Portage einzurichten ist sehr einfach. Folgen Sie den nachfolgenden Schritten auf jedem Computer, auf dem Sie distcc verwenden möchten:

**Befehlsauflistung 3:** Einrichten von Distcc und Portage

```
# emerge distcc
# nano -w /etc/make.conf
```

**Wichtig**

Sollten Sie distcc-2.11.1 oder höher unter  $\geq$ portage-2.0.49-r6 verwenden, dann genügt es zur Einrichtung die Zielrechner (siehe nächste Codezeile) einzutragen, die MAKEOPTS in `/etc/make.conf` anzupassen, sowie 'distcc' zu FEATURES in `/etc/make.conf` einzutragen.

#### Notiz

Tragen Sie in Ihre FEATURES-Flags den Parameter "distcc" ein.

#### Notiz

unkommentieren Sie: `DISTCC_DIR=DISTCC_TMPDIR` in `/etc/make.conf`.

#### Notiz

Desweiteren müssen Sie die Zeile `DISTCC_DIR=${PORTAGE_TMPDIR}/portage/.distcc` in der letzten Zeile des Dokumentes eintragen.

Als nächsten Schritt müssen Sie angeben, welche Rechner benutzt werden sollen. Um dies zu erreichen, können Sie `distcc-config` benutzen. Dieses Tool ermöglicht Ihnen, eine Liste von teilnehmenden Rechnern anzugeben. Hier ein Beispiel für eine solche Liste:

#### Befehlsauflistung 4: Beispielhafte Liste der Rechnerdefinition

```
192.168.0.1          192.168.0.2          192.168.0.3
192.168.0.1/2       192.168.0.2          192.168.0.3/10
192.168.0.1:4000/2  192.168.0.2/1        192.168.0.3:3632/4
@192.168.0.1        @192.168.0.2:/usr/bin/distccd  192.168.0.3
```

Das mag nun alles etwas kompliziert aussehen, aber in den meisten Fällen wird eine Variation aus Zeile 1 oder 2 für Sie funktionieren. Nun eine Erklärung zu den einzelnen Zeilen: Zeile 1 ist eine mit Leerzeichen getrennte Auflistung von Rechnern, da alle mit Standardeinstellungen laufen. Zeile 2 zeigt eine Liste in der die maximale Anzahl der Prozesse, die zu einem Rechner an einem Zeitpunkt gesendet und verarbeitet werden sollen, definiert ist. Die Anzahl der zu einem Zeitpunkt zu sendenden Prozesse wird durch `/N` definiert. Da wohl die meisten Benutzer Zeile 3 und 4 nicht benutzen werden, verweise ich an die [Distcc-Dokumentation \(englisch\)](#).

Folgendes Beispiel zeigt die Einrichtung der Konfiguration, wie sie in Zeile 1 zu sehen ist:

#### Befehlsauflistung 5: Beispielaufruf, um teilnehmende Rechner bekannt zu geben (Beispiel: Zeile 1)

```
# /usr/bin/distcc-config --set-hosts "192.168.0.1 192.168.0.2 192.168.0.3"
```

Nun müssen Sie nur noch einen Eintrag in der Datei `/etc/make.conf` abändern, um distcc in Portage zu integrieren. Hierzu ändern Sie `MAKEOPTS` so ab, dass Sie `-jN` (wobei N eine Ganzzahl ist) auf einen Wert setzen, der üblicherweise der "Anzahl der Prozessoren + 1" der teilnehmenden Rechner entspricht.

#### Befehlsauflistung 6: Letzte Schritte der Konfiguration: make.conf

```
# nano -w /etc/make.conf
MAKEOPTS=-jN
```

#### Wichtig

Vergessen Sie bitte nicht, den Distcc-Dienst durch ein `/etc/init.d/distccd start` zu starten!

## 2.4 Distcc so einrichten, dass es mit Automake funktioniert

Dies ist in manchen Fällen einfacher als die Einrichtung es Portage-Teiles. Sie müssen zunächst Ihre `PATH` Umgebungsvariablen so anpassen, dass `/usr/lib/distcc/bin` vor dem Pfad von gcc (`/usr/bin`) gestellt wird. Wie auch immer, es gibt eine Vorsichtsmaßnahme zu beachten. Wenn Sie `ccache` benutzen, so müssen Sie den Distcc-Teil hinter dem `ccache`-Teil stellen. Dies würde in etwa wie folgt aussehen:

#### Befehlsauflistung 7: PATH setzen

```
# export PATH="/usr/lib/ccache/bin:/usr/lib/distcc/bin:${PATH}"
```

Dann, wenn Sie normalerweise `make` eingeben würden, geben Sie `make -jN` (wobei N eine Ganzzahl ist) ein. Der Wert für N variiert je nach Anzahl und der Typen der Computer, die Sie für den Kompilierprozess verwenden. Ich benutzte hierfür `-j5`, um mit einem Dual-P3 und einem K6-2/400 alles für meinen Desktoprechner (ein 1200Mhz Athlon Thunderbird) zu kompilieren. Am besten probieren Sie einige Variationen aus, um den für Sie geeignetsten Wert herauszufinden.

## 3. Cross-Kompilierung

### 3.1 Ein Bemerkung zur Cross-Kompilierung

Cross-Kompilierung benutzt eine Architektur, um für eine andere Architektur Programme zu bauen. Dies kann so einfach sein, wie einen Athlon (i686) zum Bauen eines Programm für einen K6-2 (i586) zu benutzen. Eine andere Möglichkeit wäre, dass man eine Sparc dazu benutzt, Programme für eine PowerPC zu bauen.

### 3.2 Ein persönliche Bemerkung zur Cross-Kompilierung

Ich liebe es im Projekt "Gentoo Cross-Compiling" aushelfen zu können, aber ich habe unglücklicherweise keine nicht-x86 Rechner zur Verfügung. Ich kann Dinge produzieren, die **in der Theorie** funktionieren, aber ich muss andere Leute bemühen dass sie die Dinge testen, die ich schreibe. Dies geht auch bis zu einem gewissen Grad, aber es ist schon schwierig. ;-)

Ich hoffe, dass ich in der nahen Zukunft eine Sparc erstehen kann, um ein Gentoo zu installieren und dann etwas damit Zuhause herumspielen kann.

## 4. Distcc für den Bootstrap-Prozess verwenden

### 4.1 Schritt 1: Setup auf einem neuen Rechner

Als nächstes müssen Sie auf Ihrem zu installierenden System die Gentoo Linux LiveCD booten und allen Schritten bis zum Bootstrapping-Prozess folgen. Dann beginnen Sie mit einigen Vorbereitungen für die Einrichtung.

**Befehlsauflistung 8:** Einrichtung vorbereiten

```
// Hier fügen Sie distcc den FEATURES hinzu
# nano -w /etc/make.conf
// Fügen Sie "distcc" zu FEATURES hinzu.
// Passen Sie MAKEOPTS in /etc/make.conf so an, dass Sie -jN (wie oben beschrieben hinzufügen).
```

Danach fügen Sie der `/etc/passwd` den User distcc hinzu:

**Befehlsauflistung 9:** User distcc in `/etc/passwd` anlegen

```
# echo "distcc:x:240:2:distccd:/dev/null:/bin/false" >>/etc/passwd
// Bitten vergessen Sie nicht die `>>' bei der Eingabe des Kommandos.
```

### 4.2 Schritt 2: Distcc aufsetzen

Den nächsten Schritt führen Sie in einer Chroot-Umgebung auf dem neuen Rechner aus.

**Befehlsauflistung 10:** Distcc auf dem neuen Rechner

```
#emerge --nodeps distcc
```

### 4.3 Schritt 3: Distcc einrichten

Nun richten Sie distcc selbst ein.

**Befehlsauflistung 11:** Letzte Einstellungen zur distcc Einrichtung

```
# /usr/bin/distcc-config --install
# /usr/bin/distcc-config --set-hosts "localhost host1 host2 host3 ..."
```

Distcc sollte nun zum Bootstrap bereit sein! Fahren Sie mit der offiziellen Installationsanweisung fort und vergessen Sie nicht distcc nach dem *emerge system* erneut zu emergem.

## 5. Problembehebung

## 5.1 Mozilla und Xfree

Wenn Sie verschiedene Pakete emergen werden Sie bemerken, dass nicht alle verteilt kompiliert (und vor allem noch nicht einmal parallel gebaut) werden. Der Grund ist, dass die Entwickler der Mozilla und XFree .ebuilds paralleles Kompilieren abgestellt haben, da bekannt ist dass dies Probleme verursachen kann. Es muss also nicht unbedingt ein distcc Problem sein.

Das heisst nicht, dass die Kompilierung mit distcc nicht auch bei dem einen oder anderen Paket fehlschlagen kann.

## 5.2 Eine Mixtur aus hardened-gcc und nicht-hardened-gcc Rechner kann zu Fehlern führen

Mit einer solch langen Überschrift werden weitere Erklärungen fast überflüssig. Wie auch immer. Wenn Sie planen distcc-Rechner mit Rechnern in einem Netzwerk zu verwenden wobei einige den [PaX/hardened-gcc](#) verwenden und einige nicht, werden Sie zwangsläufig auf Probleme stoßen.

Die Lösung verlangt von Ihnen ein wenig Voraussicht; dazu führen Sie *hardened-gcc -R* auf dem Rechner mit PaX/hardened-gcc aus, oder Sie müssen PaX-Protection in ihren Kernel einbauen und danach erst ein *emerge hardened-gcc* ausführen. Beide Pakete sind eine gute Sache, denn zum größten Teil steht ihnen der Schutz beider zur Verfügung und bietet auch dem Benutzer genügend Transparenz.

## 5.3 Verschiedene GCC Versionen

Wenn Sie verschiedene GCC Versionen auf Ihren Rechnern installiert haben, können sehr seltsame Probleme auftreten. Die Lösung ist sicherzustellen, dass auf allen Rechnern die gleiche GCC Version vorhanden ist.

## 6. Distcc Extras

### 6.1 Distcc Überwachung

Distcc wird mit zwei Überwachungstools ausgeliefert. Das textbasierte wird immer mit gebaut und heisst *distccmon-text*. Wenn Sie dieses zum ersten Mal starten, mag es etwas verwirrend wirken, aber im Prinzip ist es sehr einfach zu benutzen. Wenn Sie das Programm ohne Parameter ausführen, startet es sofort. Wenn Sie aber eine Zahl als Parameter übergeben, wird es sich alle N Sekunden aktualisieren. Hierbei ist N der Wert, den Sie beim Aufruf übergeben haben.

Das andere Überwachungstool wird nur gebaut, wenn Sie *gtk* in den *USE*-Flags stehen haben. Dieses basiert auf GTK+ und läuft unter X und ist wirklich sehr schön.

Eine Bemerkung zu den Programmen: Möchten Sie einen emerge überwachen, so müssen Sie die Überwachungstools wie folgt starten:

**Befehlsauflistung 12:** Überwachung starten

```
# DISTCC_DIR=/Verzeichnis/zum/distcc-Verzeichnis portage distccmon-text N
// Oder Sie starten das grafische Tool...
# DISTCC_DIR=/Verzeichnis/zum/distcc-Verzeichnis portage distccmon-gnome
```

### Wichtig

Benutzen Sie Portage und das oben genannte Beispiel, dann gilt: *DISTCC\_DIR=/var/tmp/portage/.distcc*

### Notiz

Sollten Sie Portage nicht verwenden, müssen Sie das *DISTCC\_DIR* nicht explizit angeben. Es genügt dann einfach *distccmon-text N*.

## 7. Zukunftspläne für Distcc und Gentoo

### 7.1 distcc-subnetscan

*distcc-subnetscan* ist ein Perlprogramm welches sich in Entwicklung befindet und ein Netzwerk nach partizipierenden distcc-Diensten durchsucht. Es wird sogar insofern verbessert werden, dass es testet, ob ein Rechner konform der *CHOST* Einstellung ist, um Cross-Kompilieren einfacher zu gestalten.

Das Perl-Script ist [hier](#) zu finden, bis ein passenderer Platz dafür gefunden ist.

## 7.2 distcc-config

*distcc-config*, ist veraltet und wird demnächst neu geschrieben und somit auf den Stand der aktuellen Distcc Version gebracht.



## >> Gentoo Druckereinrichtung

[Bitte Kapitel auswählen]

### 1. Lokales Drucken mit CUPS

#### 1.1 CUPS installieren

Gentoo Linux für das Drucken vorzubereiten ist relativ einfach, da alle benötigten Pakete seit 8. August 2002 im Portage-Tree vorhanden sind. Alles was wir machen müssen, ist den gewünschten Drucker zu konfigurieren und sicherzustellen, dass er Postscript richtig interpretiert. Der erste Schritt ist die Installation der notwendigen Software.

##### **Notiz**

Der nächste Schritt beschreibt die Installation für GNOME2. Das könnte mit KDE funktionieren, da CUPS systemunabhängig ist. TODO: Anleitung für CUPS unter KDE.

##### **Befehlsauflistung 1**

```
# emerge net-print/cups
# emerge gnome-base/gnome-print
# emerge gnome-base/libgnomeprint
# emerge media-gfx/gimp-print
# emerge net-print/gimp-print-cups
```

Jetzt, da die Pakete mit ihren Abhängigkeiten installiert sind, ist es an der Zeit den Drucker selbst zu installieren.

##### **Notiz**

Dieses Dokument baut auf Hewlett-Packard-USB-Drucker auf, allerdings sollte sich die Einrichtung eines Parallel-Port-Druckers nicht sehr davon unterscheiden. Nicht-HP-Drucker sind noch nicht einbezogen, somit wird sich das Besorgen und Installieren zugehöriger Treiber vom hier geschilderten Vorgehen unterscheiden.

Wenn noch nicht geschehen, muss USB-Core- und die USB-Drucker-Unterstützung [[USB-Support -> USB Printer support](#)] in den Kernel einkompiliert werden. Ich habe beide Teile als Module übersetzt; durch *lsm* konnte ich sehen dass die Unterstützung geladen wurde und funktioniert -- allerdings ist dies jedem selbst überlassen.

##### **Wichtig**

Wenn sie einen Parallel-Port-Drucker verwenden, stellen sie sicher, dass die Parallel-Port-Kernel-Optionen anstatt der USB-Optionen aktiviert sind.

##### **Notiz**

Wenn sie nicht sicher sind, wie man den Kernel übersetzt, dann hilft ihnen das [offizielle Kernel-HOWTO](#).

Jetzt, da die notwendigen Kernel-Optionen installiert sind, ist es an der Zeit, zu überprüfen ob der Drucker erkannt wurde und funktioniert. Nachdem sie alles neu gestartet haben, müssen sie nun noch die Drucker-Module laden.

##### **Notiz**

Wenn sie USB in den Kernel eingebaut haben, dürfen sie die USB-Module nicht laden. Wenn sie einen Parallel-Port-Drucker verwenden, ersetzen sie die entsprechenden Module.

##### **Befehlsauflistung 2**

```
# insmod usbcore
# insmod printer
```

Nachdem die Module erfolgreich geladen wurden, stecken sie den Drucker ein und sehen dann in `/var/log/messages` nach, ob er erkannt wurde.

##### **Befehlsauflistung 3**

```
# tail /var/log/messages
```

Sie sollten in etwa folgendes sehen:

##### **Befehlsauflistung 4**

```
hub.c: USB new device connect on bus2/2, assigned device number 2
printer.c: usb/lp0: USB bidirectional printer dev 2 if 0 alt 1 proto 2 vid 0x03F0 pid 0x1104
```

### Notiz

Wenn sie einen Parallel-Port-Drucker verwenden, müssen sie diesen **vor** dem Neustart einstecken. Nachdem das System hochgefahren ist, sollten sie mittels *dmesg* überprüfen, ob der Kernel ihren Drucker erkannt hat.

Ehe sie ihren Drucker für CUPS einrichten, ist es möglich ihn mittels **cat** zu testen.

### Befehlsauflistung 5

```
# touch test.txt
# echo "Hallo Welt" > test.txt
# cat test.txt > /dev/usb/lp0
```

**Alternativ für Parallel-Port-Drucker:**

```
# cat test.txt > /dev/lp0
```

## 1.2 Konfiguration der PPD-Datei

Jetzt ist es an der Zeit, den Drucker und CUPS zu konfigurieren. Wenn sie einen HP-InkJet-Drucker verwenden, ist es notwendig, dass sie den *hpijs* Druckertreiber installieren. Dieser Treiber übernimmt die Interpretation von Postscript, die notwendig ist, um den HP-Drucker zum arbeiten zu bringen. Wenn sie keinen HP-Drucker verwenden, wird es notwendig sein, herstellerspezifische Treiber zu besorgen.

### Notiz

Der *hpijs* Treiber ist **nur** für HP-InkJet-Drucker geeignet. Wenn sie einen HP-LaserJet verwenden, könnte die Unterstützung über das GNOME-Drucksystem möglich sein. Überspringen sie den folgenden Text, wenn sie keinen HP-InkJet-Drucker verwenden.

### Befehlsauflistung 6

```
# emerge net-print/hpijs
```

### Wichtig

Die Dokumentation von HP zum *hpijs*-Treiber findet sich unter [/usr/share/doc/hpijs-1.1](#). Sie sollten diese unbedingt lesen.

Damit ihr Drucker Postscript richtig interpretiert, benötigt CUPS die PPD-Datei (Printer Postscript Definition). Der einfachste Weg eine PPD zu erzeugen ist mit **foomatic**. Die *foomatic* Konfigurationsskripte sind über Link <http://www.linuxprinting.org/download/foomatic> erhältlich. Außerdem ist ein Ebuild für Gentoo in der Entwicklung - dieser ist aber noch nicht im offizielle Portage-Tree. Wenn *foomatic* einmal eingerichtet ist, führen sie folgende Befehle aus, um ihren Drucker zu konfigurieren und verschieben sie dann die PPD an die richtige Stelle. Das folgende Beispiel verwendet die *hpijs*-Treiber für HP-Drucker.

### Notiz

Die Drucker-ID findet sich in der *foomatic*-Datenbank unter [/usr/share/foomatic/db/source/printer](#). Sie müssen ihre Drucker-ID ausfindigen machen und seine genaue Bezeichnung verwenden. Achten sie auf Groß/Kleinschreibung!

### Befehlsauflistung 7

```
# foomatic-configure -s cups -p printer ID -c print_device -n queue name -d hpijs
# mv /etc/foomatic/your.new.ppd /etc/cups/ppd
```

### Wichtig

Wenn sie keinen HP-Drucker verwenden, könnten sie trotzdem *foomatic* zum erstellen ihrer PPD verwenden. Bitte sehen sie in der [foomatic-Dokumentation](#) und bei [linuxprinting.org](http://linuxprinting.org) für mehr Informationen nach.

## 1.3 Konfiguration von CUPS

Jetzt da der Drucker selbst konfiguriert ist, muss CUPS so eingerichtet werden, dass es mit den Druckaufträgen umgehen kann. Praktischerweise kann CUPS über einen Web-Browser auf Port 631 des Drucker-Computers verwaltet werden.

### Wichtig

CUPS bringt eine eigene Konfigurationsdatei für den Daemon mit, die gegebenenfalls angepasst werden muss. Sie sollten die Datei [/etc/cups/cupsd.conf](#) durchsehen und anpassen.

## Befehlsauflistung 8

```
# rc-update add cupsd default
# /etc/init.d/cupsd start
```

Der CUPS-Daemon sollte jetzt laufen; öffnen sie also ihren Lieblings-Browser und setzen sie ihn auf <http://localhost:631>.

Jetzt werden sie hoffentlich von der CUPS-Konfigurations-Startseite begrüßt. Klicken sie auf den Punkt **Administration** am Seitenanfang und geben sie das Passwort von root oder jemanden aus der Gruppe **lp** ein. Nachdem sie angemeldet sind, klicken sie auf **Add Printer** - ein Unterpunkt von **Printers**. Geben sie den benötigten Namen, die Beschreibung und den Ort ihres Druckers ein. Der Name des Druckers muss genau der selbe sein, wie der den sie für foomatic festgelegt hatten! Jetzt müssen sie noch die Schnittstelle angeben, an der ihr Drucker angeschlossen ist. Wählen sie nun zuerst ihren Druckerhersteller und anschließend ihren Druckertyp aus.

Die Einrichtung ist abgeschlossen. Klicken sie nun den Namen ihres Druckers an und versuchen sie eine Testseite zu drucken. Voila, ihre Druckereinrichtung wurde erfolgreich abgeschlossen!

### Notiz

Wenn der Drucker aus unbekanntem Grund nicht funktioniert, sollten sie in der Datei **error\_log** nachschauen, die sich im Verzeichnis für die CUPS-Meldungen (meist `/var/log/cups`) befindet.

## 2. The GIMP, OpenOffice.org 1.0 und Abiword für das Drucken für CUPS einrichten

### 2.1 The GIMP

Als erstes werden wir The GIMP einrichten - einfach weil es cool ist. Starten sie The GIMP und öffnen sie ein existierendes Bild oder erstellen sie einfach ein neues, es ist völlig egal - alles was wir wollen ist das **Datei**-Menü. Klicken sie das Bild mit der rechten Maustaste an und öffnen sie **Datei -> Drucken....** Der Druckdialog sollte erscheinen; wenn ja klicken sie auf **Neuer Drucker**. Wenn ihr Drucker noch nicht aufgeführt ist, geben sie den Namen ihres Druckers ein und wählen sie den Druckertyp aus. Ok, jetzt können sie von The GIMP aus drucken. Probieren sie etwas mit den Einstellungen herum.

### 2.2 OpenOffice.org 1.0 / StarOffice 6

Drucken mit OpenOffice.org-1.0 ist nicht sehr schwer. Gehen sie nach [OpenOffice.org1.0/program](http://OpenOffice.org1.0/program) und führen sie `spadmin` aus. Erstellen sie einen neuen Drucker und wenn **Choose a Driver** (Treiber wählen) erscheint, klicken sie auf **import** und gehen sie zum Ort der PPD ihres Druckers - es sollte die Datei `/etc/cups/ppd` sein. Benutzen sie die voreingestellten Druckerkommandos, geben sie dem Drucker einen Namen, setzen sie ihn als Voreinstellung und sie können losdrucken! Starten sie dafür OpenOffice.org 1.0, öffnen sie das Druckfeld und wählen sie ihren Drucker aus. Drucken sie nun eine Testseite und fertig.

### 2.3 Abiword

#### Warnung

Abiword verursacht mit CUPS-1.1.15-r2 Speicherzugriffsfehler und stürzt ab. Sie sollten zurück zur .14-Serie wechseln, da dies das Problem behebt.

TODO: Abiword-Unterstützung hinzufügen.

Lokales Drucken ist hiermit abgeschlossen; wir fahren fort mit der Netzwerkunterstützung.

## 3. Drucken im Netzwerk mit CUPS und SAMBA

### 3.1 Drucken mit einem Windows-Drucker von einer Linux-Maschine aus

Damit die nächsten beiden Netzwerkdruckoptionen funktionieren, muss zunächst SAMBA installiert werden.

#### Befehlsauflistung 9

```
# emerge net-fs/samba
```

## Notiz

Mehr Informationen zu SAMBA finden sie in der offiziellen [SAMBA HOWTO](#).

Zunächst erstellen sie mit **foomatic** eine PPD-Datei für den Zieldrucker. Vergessen sie nicht in der Datenbank der Seite [linuxprinting.org](http://linuxprinting.org) nachzusehen, ob ihr Drucker mit Linux kompatibel ist oder nicht. Jetzt teilen wir **foomatic** mit, wo der und welcher Drucker es ist. Anschließend verschieben wir die erzeugte PPD an die richtige Stelle.

### Befehlsauflistung 10

```
# foomatic-configure -s cups -p printer ID -c smb://user:password@server/share -n queue name -d
# mv /etc/foomatic/your.new.ppd /etc/cups/ppd
```

Setzen sie jetzt ihren Lieblings-Browser auf <http://localhost:631>, um das CUPS-Konfiguration-Interface zu starten. Führen sie jetzt die oben beschriebenen Schritte aus, um einen Drucker hinzuzufügen - geben sie jedoch diesmal bei der Frage nach der Schnittstelle **smb://user:password@server/share** ein. Dies setzt CUPS auf den Drucker am Windows-Rechner.

## 3.2 Linux als Windows-kompatiblen Drucker-Server verwenden

Linux eignet sich hervorragend als Drucker-Server, eigentlich für jede Art Server. Der erste Schritt ist die Bearbeitung der Datei `/etc/samba/smb.conf`, um die Einstellungen anzupassen. Die Datei **smb.conf** beinhaltet alle Einstellungen, um ihre Linux-Rechner als Windows-NT-Server für alle Windows-Rechner zu tarnen. Im folgenden habe ich ein allgemeines Beispiel von **smb.conf** erstellt. Passen sie dieses auf ihre Netzwerkumgebung an.

### Befehlsauflistung 11

```
[global]
```

```
workgroup = YOUR_WORKGROUP
server string = Gentoo Linux Server
encrypt passwords = True
security = user
smb passwd file = /etc/samba/private/smbpasswd
log file = /var/log/samba/log.%m
socket options = IPTOS_LOWDELAY TCP_NODELAY
domain master = Yes
local master = Yes
preferred master = Yes
os level = 65
dns proxy = No
name resolve order = lmhosts host bcst
bind interfaces only = True
interfaces = eth0
hosts deny = ALL
hosts allow = 192.168.1.4 127.0.0.1 (list of allow hosts here)
debug level = 1
create mask = 0644
directory mask = 0755
level2 oplocks = True
read raw = no
write cache size = 262144
```

```
[printers]
```

```
comment = All Printer
path = /var/spool/samba
browseable = no
```

Jetzt, da SAMBA konfiguriert ist, starten sie es neu, wechseln sie auf ihren Windows-Rechner und installieren sie einen Netzwerkdrucker. Ganz einfach - oder?



## >> Anleitung zur Gentoo Linux Desktop Konfiguration

[Bitte Kapitel auswählen] 

### 1. Voraussetzungen

Eigentlich nicht nötig zu erwähnen: bevor Sie mit dieser Anleitung anfangen, müssen Sie Gentoo Linux [installiert](#) haben. Nach Abschluss dieser recht amüsanten Prozedur brauchen Sie zusätzlich den aktuellen **Portage** tree. Es ist wirklich empfehlenswert, das [Portage User Handbuch](#) zu lesen, aber falls Sie ungeduldig sind oder unter Zeitdruck stehen, schreiben Sie einfach:

**Befehlsauflistung 1:** Den Portage Tree auf den neuesten Stand bringen

```
# emerge sync
```

Noch ein kurzer Hinweis zu den Paketen die ich hierfür ausgewählt habe. Gentoo Linux hat eine riesige Auswahl an Window Managern, E-Mail-Programmen, etc, die einfach installiert werden können. Die Pakete, die ich ausgewählt habe sind nicht speziell für Gentoo Linux gemacht, es sind einfach die Pakete, die der Autor kennt und zu konfigurieren weiß. Mit anderen Worten, dies sind die Pakete des Desktops des Autors.

#### Wichtig

Abhängig davon welchen Desktop Sie installieren werden, sollten Sie die Unterstützung für Desktops, die sie nicht verwenden in der `/etc/make.conf` abschalten. Fügen Sie zur `USE` Zeile `-gnome -gtk` hinzu, wenn Sie KDE verwenden werden oder `-kde -qt`, bei der Verwendung von GNOME.

#### Warnung

Wenn Sie zwischendurch einen Fehler (Bug) finden, melden Sie ihn unter <http://bugs.gentoo.org/>. Melden Sie ihn **nicht** an die Personen, die sich um das Paket kümmern (wie z.B. das KDE-Team); die Entwickler bei Gentoo werden den Fehler nötigenfalls schon weiterreichen.

### 2. XFree86 Installieren

#### 2.1 Installation

Die Installation von X wird selbst auf schnellen Rechnern ein Weilchen dauern. Das System ist recht groß, von daher richten Sie sich darauf ein, ein Buch zu lesen oder ein Nickerchen zu halten...

Um X zu "mergen" brauchen Sie lediglich folgendes zu tippen:

**Befehlsauflistung 2:** XFree86 mergen

```
# emerge xfree
# env-update && source /etc/profile
```

Nach Abschluss dieses recht langen Prozesses beginnt die Konfiguration.

#### 2.2 X Konfigurieren

Um X zu konfigurieren, mag ich das Konsolen-basierte `xf86config`, obwohl die davon generierte Datei `/etc/X11/XF86Config` nicht sehr elegant ist. Bevor Sie starten, sollten Sie die horizontal und vertical refresh rates Ihres Monitors (mein Viewsonic G773 hat horizontal: 30-69kHz, vertical: 55-90Hz), den Typ und die Größe des RAM (in kB) Ihrer Grafikkarte (eine 32768kB GeForce Karte in meinem Fall) und das Modell und Device Ihrer Maus (eine Intellimouse angestöpselt an den PS/2-Port unter `/dev/psaux`) in Erfahrung bringen.

**Befehlsauflistung 3:** `xf86config` ausführen

```
# /usr/X11R6/bin/xf86config
```

Da ich eine Drei-Button-Maus habe, wähle ich nicht die 3-button-emulation. Vergessen Sie nicht die XKB-Unterstützung zu aktivieren, wenn Sie nicht eine US-Tastatur haben (was ich stark annehme) oder mehrere Layouts unterstützt haben möchten. Bei der Auflösung und der Farbtiefe müssen Sie selbst entscheiden, was Sie bevorzugen. Ich lasse 1280x1024 bei 24bpp (bits per pixel, also 16

Millionen Farben) auf meinem Viewsonic laufen, aber nur, da er nicht mit höheren Auflösungen klar kommt. Wenn Sie dieses vorher noch nie getan haben, seien Sie nicht überrascht, wenn Sie mehrere Male `xf86config` aufrufen müssen.

Einen anderen Weg XFree86 zu konfigurieren bietet `XFree86 -configure`. Dieses Kommando erzeugt eine (hoffentlich funktionierende) `XF86Config` Datei im Home-Verzeichnis von root (`/root`). Danach können Sie diese Datei Ihren Anforderungen entsprechend editieren und später nach `/etc/X11/XF86Config` kopieren.

Schauen Sie in [dieses Tutorial](#) für eine tiefergehende X Konfiguration.

#### **Notiz**

Wenn Sie die refresh rates Ihres Monitors nicht kennen, versuchen Sie eine Suche mit der Modellbezeichnung und dem Hersteller bei [Google](#). Die refresh rates werden normalerweise bei den Technischen Spezifikationen mit angegeben.

#### **Notiz**

Benötigen Sie Umlaute oder "dead keys" (z.B. um è, é, ê, ë, ò, ó darzustellen) auf Ihrer Tastatur, müssen Sie Ihre locale angeben bevor Sie X starten. Dazu erstellen Sie die Datei `/etc/env.d/02locale` und fügen die Zeile `LC_CTYPE="xx_XX"` ein, wobei `xx` für Ihre Sprache steht (z.B. "de") und `XX` für Ihr Land (z.B. "AT"). Möchten Sie das EURO-Zeichen verwenden, hängen Sie an die Variable `"@euro"` an (z.B. `LC_CTYPE="de_AT@euro"`). Vergessen Sie nicht `env-update && source /etc/profile` nach diesen Änderungen auszuführen.

Wenn Sie ein Veteran im Konfigurieren von X sind, sind Sie gewohnt einen symbolischen Link von Ihrem Maus-Device zu `/dev/mouse` zu erstellen. Solche Symlinks sind problematisch bei der Verwendung von devfs, da sie ab und an dazu neigen nach einem Neustart verschwunden zu sein. Jedoch geht Gentoo Linux korrekt mit symbolischen Links in `/dev` um. Erstellen Sie einfach den `/dev/mouse` Symlink indem Sie `ln -s` verwenden und er wird den Reboot überleben.

Entscheiden Sie sich zur Verwendung des XFS (den XFontServer), folgen Sie den entsprechenden Anleitungen weiter unten. XFS zu verwenden ist empfohlen, da es die Schriftenkonfiguration um **einiges** einfacher macht. Schauen Sie sich dazu [diesen Artikel von RedHat](#) über XFS an.

#### **Befehlsauflistung 4:** Konfiguration Testen

```
# rc-update add xfs default
# /etc/init.d/xfs start
# startx
```

Haben Sie X ordentlich konfiguriert, sollte `startx` einen erstaunlich hässlichen X Windowmanager starten (twm, um genau zu sein). Startet X, sind Sie auf dem richtigen Weg. Drücken Sie `<ctrl><alt><backspace>` um den X server zu beenden.

`xf86config` wird nur eine Konfigurationsdatei für übliche Komponenten erzeugen. NVIDIA-Karten, USB-Mäuse und viele andere Optionen müssen von Hand eingestellt werden. Ich werde diese Optionen weiter unten erläutern; schauen Sie jedoch ruhig schon mal in die Datei **XF86Config**. Sie ist nicht allzu kryptisch und Sie sollten in der Lage sein, einige interessante Einstellungen vorzunehmen. Folgen Sie dem Link, um einen Einstieg in die [Optimierung des Aussehens](#) zu erhalten.

## **2.3 Eine PS/2-Maus konfigurieren**

Besitzen Sie eine Maus mit Scrollrad werden Sie Ihre XF86Config Datei von Hand ändern müssen:

#### **Befehlsauflistung 5:** Editieren der XF86Config

```
Um auf Nummer sicher zu gehen wird eine Kopie der lauffähigen
Konfigurationsdatei erstellt
# cp /etc/X11/XF86Config /etc/X11/XF86Config.working
# nano -w /etc/X11/XF86Config
```

#### **Befehlsauflistung 6:** Abschnitt Pointer in der XF86Config

```
Section "InputDevice"

#Identifier and driver

Identifier "Mouse1"
Driver "mouse"
```

```
Option "Protocol" "IMPS/2"
Option "Device" "/dev/psaux"
Option "ZAxisMapping" "4 5"
```

Da ich eine IntelliMouse besitze, habe ich das "Protocol" von "PS/2" in "IMPS/2" geändert. Um das Rädchen zum Scrollen zu verwenden muss lediglich die Zeile 'Option "ZAxisMapping" "4 5"' hingefügt werden; 4 und 5 entsprechen Scrollrad hoch und Scrollrad runter.

Besitzen Sie eine Scrollradmaus und *PS/2* funktioniert nicht, versuchen Sie *IMPS/2*, auch wenn Sie keine IntelliMouse besitzen. Zumindest einige Logitech Mäuse benötigen diese Angabe.

### Notiz

Netscape 4.x lässt sich auf diese Weise nicht zum Scrollen bewegen; dazu muss erst die Datei *.Xdefaults* erstellt werden. Schauen Sie sich die entsprechende [Anleitung](#) dazu an. Generell ist es jedoch besser *konqueror*, *mozilla*, *galeon*, *opera* oder einen anderen modernen Browser zu verwenden.

Lassen Sie zur Sicherheit, dass X weiterhin startet, nochmals *startx* laufen.

## 2.4 Eine USB-Maus konfigurieren

Eine USB-Maus ist Ihr Freund bei hochauflösenden Bildschirmen. Der Kernel kümmert sich um die Skalierung, sodass Sie nicht fünf Mal ihre Maus quer über das Mauspad schieben müssen, um einmal quer über den Bildschirm zu kommen.

Das erste was getan werden muss, ist die Installation der Kernel-Module. Für die Verwendung einer USB-Maus werden die Module *usbmouse*, *mousedev*, *hid*, *usbcore*, *usb-uhci* und *input* benötigt. Ist die benötigte Kernelkonfiguration erledigt, laden Sie mit *insmod* die Module.

### Notiz

Verwenden Sie entweder *usbmouse* oder *hid*. Laden Sie beide Module, wird die Maus ihre Tätigkeit einstellen.

### Notiz

Bei der Konfiguration des Input Core Support für die Maus stellen Sie bei der Bildschirmauflösung jene ein, die Sie unter X verwenden wollen. Das ergibt ein schönes und korrektes Scaling.

Stöpseln Sie nun Ihre Maus raus und wieder rein und checken Sie im Kernel-Log, ob dort eine Meldung ähnlich zu der folgenden einschien:

### Befehlsauflistung 7: Kernel Meldung

```
hub.c: USB new device connect on bus1/1, assigned device number 2
input0,hiddev0: USB HID v1.00 Mouse [Microsoft Microsoft IntelliMouse Optical] on usb1:2.0
```

Da die Maus nun entdeckt wurde, überprüfen Sie */dev/input*, ob Ihre Maus dort ist und funktioniert.

### Befehlsauflistung 8: USB-Maus überprüfen

```
# cd /dev/input
# cat mice
```

Bewegen Sie die Maus während Sie auf dem Device lauschen -- Sie sollten nun einen Haufen Müll sehen. Zum Beenden drücken Sie **CTRL-c**.

Sobald die Maus vom System ordentlich entdeckt und installiert wurde, muss X gesagt bekommen, dass wir die USB-Maus verwenden möchten. Alles was benötigt wird, ist eine kleine Änderung des von X verwendeten Geräts für die Maus:

### Befehlsauflistung 9: Editieren der XF86Config

```
Um auf Nummer sicher zu gehen wird eine Kopie der lauffähigen
Konfigurationsdatei erstellt
# cp /etc/X11/XF86Config /etc/X11/XF86Config.working
# nano -w /etc/X11/XF86Config
```

### Befehlsauflistung 10: Abschnitt Pointer in der XF86Config

```

Section "InputDevice"

#Identifier and driver

Identifier "Mouse1"
Driver "mouse"
Option "Protocol" "IMPS/2"
Option "Device" "/dev/input/mice"
Option "ZAxisMapping" "4 5"

```

Okay, starten Sie erneut X und die Maus sollte funktionieren!

Vergessen Sie nicht die benötigten Module zur Datei `/etc/modules.autoload.d/kernel-2.4` hinzuzufügen.

**Befehlsauflistung 11:** Module zur `modules.autoload.d/kernel-2.4` hinzufügen

```

# nano -w /etc/modules.autoload.d/kernel-2.4
Fügen Sie die folgenden Zeilen in die Datei ein:
mousedev
hid
usbcore
input

```

## 2.5 Eine NVIDIA-Grafikkarte konfigurieren

Sollten Sie eine NVIDIA-Grafikkarte besitzen, ist es nun an der Zeit die beschleunigten Treiber von NVIDIA zu installieren.

### Notiz

Wenn Sie später Ihren Kernel neu kompilieren, müssen Sie auch **nvidia-kernel** neu kompilieren. Das kommt daher, da während des Kernel kompilierens die nvidia-Module gelöscht werden.

**Befehlsauflistung 12:** Nur für Besitzer von NVIDIA-Grafikkarten

```

# emerge nvidia-glx
# emerge nvidia-kernel
# nano -w /etc/X11/XF86Config
# modprobe nvidia

```

### Notiz

Verwenden Sie eine Kernel Version anders als 2.4, ersetzen Sie 2.4 mit dieser Version.

**Befehlsauflistung 13:** `/etc/modules.autoload.d/kernel-2.4` (nur für NVidia Besitzer)

```

# nano -w /etc/modules.autoload.d/kernel-2.4
# Fügen Sie die folgende Zeile zu der Datei hinzu
nvidia

```

**Befehlsauflistung 14:** GLX in der XF86Config aktivieren

```

Section "Module"

# This loads the GLX module
Load      "glx" (Nehmen Sie das Kommentarzeichen # am Anfang der Zeile heraus)

```

**Befehlsauflistung 15:** X soll den NVIDIA-Treiber verwenden

```

Section "Device"
Identifier "NVIDIA GeForce"
#Driver    "nv"
Driver    "nvidia"
#VideoRam 32768
# Insert Clocks lines here if appropriate
EndSection

```

Lassen Sie **startx** ein weiteres Mal laufen, um sicherzustellen, dass X immer noch startet (dieses Mal mit einem NVIDIA-Logo, wenn die Treiber korrekt installiert wurden).

## Notiz

Wenn Sie Probleme mit X haben, schauen Sie in die glx-Dokumentation unter </usr/share/doc/nvidia-glx-1.0.XXXX/README.gz>. Dort gibt es eine Menge Konfigurationsvariablen, die sich erheblich auf die Systemperformanz auswirken können.

## 3. Window Manager

### 3.1 Einführung

Im Grunde wird als erstes nach einer X-Installation ein Window Manager installiert. Natürlich können Sie auch bei twm bleiben, wenn Sie einen hässlichen - ich meine funktionalen - Window Manager bevorzugen.

Es gibt viele Window Manager aus denen Sie wählen können, wie beispielsweise KDE, GNOME, Fluxbox, Blackbox, die anderen \*box'en, Windowmaker, Enlightenment, Sawfish, etc. Sie sehen, es gibt eine gute Auswahl an Window Managern in Gentoo Linux.

### 3.2 KDE

Ich gehe davon aus, dass Sie KDE installieren möchten, wenn Sie diesen Abschnitt lesen. KDE wird komplett von Gentoo Linux unterstützt, aber es gibt einige Dinge, die Sie überprüfen sollten, bevor Sie KDE emergen:

- "qt" und "kde" müssen in der USE-Variable in der Datei </etc/make.conf> angegeben sein.
- "-gnome" und "-gtk" sollten gesetzt sein, sind aber nicht zwingend erforderlich

Beide Schritte können folgendermaßen durch Editieren der </etc/make.conf> erledigt werden:

**Befehlsauflistung 16:** Editieren der make.conf

```
# nano -w /etc/make.conf
Innerhalb der make.conf
```

```
USE="qt kde -gnome -gtk"
```

Wenn Sie mehr über USE-Variablen erfahren möchten, schauen Sie sich das [USE HOWTO](#) an.

Es ist immer nützlich mit dem `--pretend` Attribut bei `emerge` zu schauen, welche Abhängigkeiten installiert werden -- und in unserem Fall sind es eine ganze Menge.

**Befehlsauflistung 17:** Ausgeben der KDE-Abhängigkeiten und Installation

```
# emerge --pretend kde
# emerge kde
# env-update && source /etc/profile
```

Schauen Sie in das Verzeichnis </etc/X11/Sessions/> welche KDE-Sessions existieren (z.B. `kde-3.1.2`). Öffnen Sie daraufhin </etc/rc.conf> und editieren Sie diese folgendermaßen:

**Befehlsauflistung 18:** Die XSESSION Variable setzen

```
XSESSION="kde-3.1.2"
```

Wenn Sie KDE nicht mit allen seinen Komponenten installieren wollen (was passiert, wenn Sie `emerge kde` eingeben), aber stattdessen die KDE Pakete installieren, **die** Sie wollen, dann machen Sie es einzeln. Die folgende Tabelle zeigt eine Auswahl der möglichen KDE Pakete. Installieren Sie die gewünschten Pakete mit `emerge <Paketname>`.

| Paket            | Beschreibung  |
|------------------|---|
| kdeaccessibility | Programme rund um Zugänglichkeit                    |
| kdebase          | KDE Basispaket und notwendige Applikationen         |
| kdeaddons        | Diverse ältere oder nicht mehr verwendete KDE Tools |
| kdeadmin         | KDE Administrationstools                            |
| kdeartwork       | Styles für KDE                                      |
| kdeedu           | KDE Lehrprogramme                                   |

|               |  |
|---------------|--|
| kdegames      | KDE Spiele   |
| kdegraphics   | Grafikprogramme für KDE  |
| kdemultimedia | Multimediaprogramme für KDE  |
| kdenetwork    | KDE Netzwerktools  |
| kdepim        | Personal Information Manager, KDE Kalender, Schreibtischnotizen etc. |
| kdesdk        | Entwicklungs Tools   |
| kdetoys       | Unterhaltungstools für den KDE Desktop                               |
| kdeutils      | Grafische Systemprogramme  |
| kde-i18n      | Sprachenpakete, beachten Sie die Anmerkung                           |

#### Notiz

Wenn Sie CUPS zum Drucken verwenden möchten, sollten Sie "cups" zur USE-Variable in der `/etc/make.conf` hinzufügen bevor Sie KDE emergen. Genauso gut kann man es auch per Kommandozeile mittels `USE="cups" emerge kde` machen. Für eine komplette CUPS-Installation schauen Sie bitte im [Gentoo Print Guide](#) nach.

Das kde-i18n Paket nutzt die LINGUAS Umgebungsvariable, um zu prüfen welche Sprachen installiert werden sollen. Zum Beispiel würde `LINGUAS="fr it"` die französische und italienische Sprachunterstützung installieren. Um eine deutsche Sprachunterstützung für KDE zu installieren, sollte die LINGUAS Umgebungsvariable mindestens den Wert `de` erhalten, also `LINGUAS="de"`.

Habe Sie nun ein nettes Nickerchen!

#### Notiz

Sollten Sie eine Gentoo Linux Binary CD besitzen können Sie die Pakete direkt von der CD installieren.

**Befehlsauflistung 19:** Installation der Binärpakete von CD

```
# export PKGDIR=/mnt/cdrom/packages
Für gewöhnlich erwartet portage fertige Pakete in /usr/portage/packages
# emerge --usepkg /mnt/cdrom/packages/All/kdebase-x.y.z.tbz2
Die Angabe -usepkg stellt sicher, dass Binärpakete (falls
vorhanden) bei Abhängigkeiten verwendet werden.
# env-update && source /etc/profile
```

### 3.3 KDM konfigurieren

Ist die Installation von KDE beendet, möchten Sie sicherlich einen netten Login Manager haben, in diesem Fall den KDE Login Manager, oder KDM. KDM ist in den KDE Paketen enthalten und braucht lediglich nur noch aufgerufen werden:

**Befehlsauflistung 20:** KDM starten

```
# kdm
```

Sie sollten sich einloggen können und KDE komplett lauffähig sehen. Okay, Sound klappt noch nicht und Sie haben den nervigen KDE-Start-Sound verpasst, doch das werden wir in wenigen Augenblicken beheben. Öffnen Sie eine Konsole und tippen Sie `ls /usr/bin` (oder irgend etwas anderes, das das Konsolenfenster über den Rand füllt) und schauen Sie, ob Sie mit dem Scrollrad an der Maus im Konsolenfenster hoch- und runterscrollen können. Es sollte wirklich klappen!

Übrig bleibt noch, dass X ebenso startet, wenn Gentoo bootet. Editieren Sie die Datei `/etc/rc.conf` sodass dort `DISPLAYMANAGER="kdm"` steht.

**Befehlsauflistung 21:** `/etc/rc.conf` editieren

```
# nano -w /etc/rc.conf
```

**Befehlsauflistung 22:** Einstellen des Displaymanager in der `/etc/rc.conf`

```
# What display manager do you use ? [ xdm | gdm | kdm ]
DISPLAYMANAGER="kdm"
```

Überprüfen Sie nun, dass das `xdm` Start-Skript beim Booten eingelesen wird.

**Befehlsauflistung 23:** XDM Skript zum default runlevel hinzufügen

```
# rc-update add xdm default
```

Rebooten Sie und schauen Sie ob X/KDE starten!

### **Wichtig**

Das *rc-update* Kommando ist Gentoo Linux' verbesserte Version des ehrwürdigen *chkconfig* Kommandos zum Managen von Init-Skripten. Geben Sie nur *rc-update* auf der Kommandozeile ein und lassen Sie sich die Liste an Optionen ausgeben.

## **4. GNOME**

### **4.1 Installation**

Bevor Sie die GNOME Pakete installieren, sollten Sie Ihre USE-Variable anpassen. Damit wird die GNOME-Unterstützung in Applikationen, die Sie installieren, aktiviert.

- "gtk" und "gnome" müssen in der USE-Variable in der Datei */etc/make.conf* angegeben sein.
- "-kde" und "-qt" sollten gesetzt sein, sind aber nicht zwingend erforderlich

Beide Schritte können folgendermaßen durch Editieren der */etc/make.conf* erledigt werden:

**Befehlsauflistung 24:** Editieren der *make.conf*

```
# nano -w /etc/make.conf
Innerhalb der make.conf

USE="gnome gtk -kde -qt"
```

Wenn Sie mehr über USE-Variablen erfahren möchten, schauen Sie sich das [USE HOWTO](#) an.

Es ist immer nützlich mit dem *--pretend* Attribut bei *emerge* zu schauen, welche Abhängigkeiten installiert werden -- und in unserem Fall sind es eine ganze Menge.

**Befehlsauflistung 25:** Ausgeben der GNOME-Abhängigkeiten und Installation

```
# emerge --pretend gnome
# emerge gnome
# emerge xscreensaver
# env-update && source /etc/profile
```

Das nimmt nun einige Zeit in Anspruch, von daher lehnen Sie sich zurück und schauen Sie sich einen Film an (der Autor empfiehlt "Pulp Fiction").

Öffnen Sie danach die Datei */etc/rc.conf* und ändern Sie die XSESSION Variable folgendermaßen:

**Befehlsauflistung 26:** XSESSION Variable in der */etc/rc.conf* ändern

```
XSESSION="Gnome"
```

Die GNOME-Konfiguration ist ab diesem Punkt annähernd die gleiche wie die von KDE. Es gibt keine GNOME-spezifischen Konfigurationsdateien, jedoch gibt es GDM, den GNOME Login Manager einzurichten.

### **4.2 GDM konfigurieren**

Nachdem GNOME installiert ist, möchten Sie sicherlich einen netten Login Manager, in diesem Fall den GNOME Login Manager, oder GDM verwenden. GDM ist im bereits installierten GDE-Paket enthalten, sodass er zum starten nur noch aufgerufen werden muss:

**Befehlsauflistung 27:** GDM starten

```
# gdm
```

Sie sollten sich einloggen können und GNOME komplett lauffähig sehen. Okay, Sound klappt noch nicht und Sie haben den nervigen GNOME-Start-Sound verpasst, doch das werden wir in wenigen Augenblicken beheben. Öffnen Sie eine Konsole und tippen Sie *ls /usr/bin* (oder irgend etwas anderes,

das das Konsolenfenster über den Rand füllt) und schauen Sie, ob Sie mit dem Scrollrad an der Maus im Konsolenfenster hoch- und runterscrollen können. Es sollte wirklich klappen!

Übrig bleibt noch, dass X ebenso startet, wenn Gentoo bootet. Editieren Sie die Datei `/etc/rc.conf` sodass dort **DISPLAYMANAGER="gdm"** steht.

**Befehlsauflistung 28:** `/etc/rc.conf` editieren

```
# nano -w /etc/rc.conf
```

**Befehlsauflistung 29:** Einstellen des Displaymanager in der `/etc/rc.conf`

```
# What display manager do you use ? [ xdm | gdm | kdm ]
DISPLAYMANAGER="gdm"
```

Überprüfen Sie nun, dass das **xdm** Start-Skript beim Booten eingelesen wird.

**Befehlsauflistung 30:** XDM Skript zum default runlevel hinzufügen

```
# rc-update add xdm default
```

Rebooten Sie und schauen Sie ob X/GNOME starten!

### Wichtig

Das `rc-update` Kommando ist Gentoo Linux' verbesserte Version des ehrwürdigen `chkconfig` Kommandos zum managen von Init-Skripten. Geben Sie nur `rc-update` auf der Kommandozeile ein und lassen Sie sich die Liste an Optionen ausgeben.

## 5. Sound Konfiguration

### 5.1 ALSA

Zur Installation von ALSA, schauen Sie bitte in den [Gentoo Linux ALSA Setup Guide](#).

### 5.2 KDE und Sound

KDE hat einige nette Sound Utensilien, inklusive dem KDE Sound Mixer `kmix`.

#### Notiz

Wenn `kmix` keinen Mixer finden kann, dann denkt es vermutlich, dass Sie keine Soundunterstützung haben. Loggen Sie sich ein und entfernen Sie die Dateien `~/.kde/share/config/kmixrc` und `~/.kde/share/config/kmix:*` und loggen Sie sich erneut ein. Dieses Mal sollte es den Sound Mixer ordentlich finden.

### 5.3 GNOME und Sound

Sie können mit `gnome-alsamixer` Ihre Lautstärken einstellen. Er ist der Teil des `media-sound/gnome-alsamixer` Paket.

## 6. Mail

### 6.1 Postfix -- Installation und Konfiguration

Ich mag `postfix`, da es `sendmail` emuliert, ohne den Alptraum einer `sendmail`-Konfiguration durchleben zu müssen. Wo wir gerade dabei sind, werden wir gleich `mutt` installieren, ein Mail-Programm, sodass wir unsere `postfix`-Konfiguration testen können.

**Befehlsauflistung 31:** `emerge` von `postfix` und `mutt`

```
# emerge postfix mutt
```

Wenn Sie Ihren Mail-Server nur zum Ausliefern von Mail verwenden möchten, dann ist er bereits fertig aufgesetzt. Anderfalls müssen Sie die Datei `/etc/postfix/main.cf` anpassen. Diese Datei ist sehr gut dokumentiert und die Standardeinstellungen sind ausreichend für eine alleinstehende Maschine. Ist Ihr

Rechner direkt an das Internet angeschlossen (z.B. mit einer IP-Adresse aus dem nicht-privaten Bereich und einem DNS-Namen), dann sind die folgenden aufgeführten Änderungen vorzunehmen:

**Befehlsauflistung 32:** Mailempfang in main.cf einschalten

```
# RECEIVING MAIL

# The inet_interfaces parameter specifies the network interface
# addresses that this mail system receives mail on. By default,
# the software claims all active interfaces on the machine. The
# parameter also controls delivery of mail to user@[ip.address].
#
#inet_interfaces = all
#inet_interfaces = $myhostname
#inet_interfaces = $myhostname, localhost
inet_interfaces = $myhostname, localhost
mydestination = $myhostname, localhost.$mydomain
```

## 6.2 Testen von Postfix (mit Mutt)

Schauen wir, ob es klappt:

**Befehlsauflistung 33:** Testen von Postfix

```
# postfix check
# newaliases
# postfix start
postfix-script: starting the Postfix mail system
# mutt -x root
/root/.maildir does not exist. Create it? ([y]/n): [antworten Sie mit y]
To: <root>
Subject: test
(End message with a . on a line by itself)
test
.

# mutt
[Sie sollten nun Ihre Nachricht sehen]
```

Mit *postfix check* stellen wir sicher, dass die *main.cf* fehlerfrei ist und einige wichtige Verzeichnisse erstellt werden. *newaliases* stellt die Aliase bereit und *postfix start* startet letztendlich den Mail Server. Dann habe ich *mutt* (im alten *mailx* oder *mail* Modus) verwendet, um eine Nachricht an mich selbst auf meinem Rechner zu senden. Danach erstellt *mutt* den Mail Spool des Benutzers (in diesem Fall root) im Verzeichnis *~/maildir*. Ein darauf aufgerufenes *mutt* im kompletten interaktiven Modus zeigt die versandte Nachricht, sodass wir sichergehen können, dass lokales Versenden und Zustellen funktioniert. Nicht gezeigt ist das Versenden an eine nicht-lokale E-Mail-Adresse. Führen Sie dieses auch aus, um sicherzustellen, dass dieses ebenso funktioniert.

### Warnung

Versenden Sie von einem lokalen Netzwerk mit einer IP-Adresse aus dem privaten Adressbereich mit der Standard-*main.cf* eine Nachricht an eine externe Adresse, ist es möglich, dass Sie diese niemals sehen werden. Viele Mailserver vernichten automatisch Nachrichten, die von einem Rechner kommen, der nicht über reverse-DNS erreichbar ist.

Angommen alles läuft, sollten Sie postfix zu den Programmen hinzufügen, die beim Systemstart ausgeführt werden:

**Befehlsauflistung 34:** postfix beim Booten mitstarten

```
# postfix stop
postfix-script: stopping the Postfix mail system
# rc-update add postfix default
* postfix added to runlevel default...
* Caching service dependencies... [ ok ]
* rc-update complete.
# /etc/init.d/postfix start
Starting postfix..... [ ok ]
```

## Wichtig

Veteranen in *sendmail* oder *postfix* Junkies können es gewöhnt sein, dass die Nachrichten in `/var/spool/mail` anstelle der Benutzerverzeichnisse und im **mbox**- anstatt des **maildir**-Format gespeichert werden. Es ist nicht allzuschwer, *postfix* und *mutt* so umzukonfigurieren, dass sie das Standard Unix mailbox System (**mbox**) verwenden. Editieren Sie die `/etc/postfix/main.cf` indem Sie **home\_mailbox** kommentieren und **mail\_spool\_directory = /var/spool/mail** einschalten. Erstellen Sie das Mail Spool Verzeichnis mit `mkdir -p /var/spool/mail` und starten Sie *postfix* neu. Das wäre es für den Mail-Server. Für *mutt* müssen Sie die `/etc/mutt/Muttrc` editieren und **set mbox\_type = mbox** für die Verwendung von **mbox** einfügen. Außerdem setzen Sie **set spoolfile=/var/spool/mail**, damit das richtige Verzeichnis verwendet wird.

Bevorzugen Sie ein grafisches E-Mail-Programm, so werden wir später einen Web-basierten Mail-Server, Squirrelmail, installieren. Alternativ empfiehlt Daniel Robbins eine Kombination aus IMAP (was wir später zusammen mit Squirrelmail installieren werden) und KDes *kmail*. Dieser ist Teil von `kde-base/kdenetwork`.

Weiter unten werden wir auch Ximinans **Evolution** als Ihre Personal Information Management (PIM) Software installieren -- komplett mit einer Unterstützung für den Palm pilot. Es kann auch als ein hervorragendes E-Mail-Programm verwendet werden.

## 7. Samba

### 7.1 Installation und Konfiguration von Samba

Samba ist ein ausgezeichnetes Paket um Dateien und Drucker von Ihrem Linux Rechner an Windows Maschinen zur Verfügung zu stellen. Für detaillierte Informationen lesen Sie den Samba Artikel von Daniel Robbins unter <http://www.ibm.com/developerworks?ca=dgr-GentoodWhp>.

#### Befehlsauflistung 35: Samba installieren

```
# emerge samba
# rc-update add samba default
# cd /etc/samba
# cp smb.conf.example smb.conf
# nano -w smb.conf
```

#### Befehlsauflistung 36: smb.conf -- Workgroup einstellen

```
# workgroup = NT-Domain-Name or Workgroup-Name, eg: REDHAT4
workgroup = MYGROUP [Ändern Sie den Gruppennamen entsprechend]
```

#### Befehlsauflistung 37: smb.conf -- Drucken mit CUPS

```
# It should not be necessary to specify the print system type unless
# it is non-standard. Currently supported print systems include:
# bsd, sysv, plp, lprng, aix, hpux, qnx
; printing = bsd
printing = cups
```

#### Befehlsauflistung 38: smb.conf -- Verschlüsselung

```
# You may wish to use password encryption. Please read
# ENCRYPTION.txt, Win95.txt and WinNT.txt in the Samba documentation.
# Do not enable this option unless you have read those documents
encrypt passwords = yes
```

Ich habe recht wenig Verwendung für Samba. Ich möchte nur Dateien zwischen meinen Linux und Windows Rechnern austauschen und meinen Linux Rechner als Print Server (mit *CUPS*) verwenden. Nach der Installation müssen ein paar Änderungen an der Datei `/etc/samba/smb.conf` vorgenommen werden. Im einzelnen muss die **workgroup** Variable den Namen des Windows Netzwerks enthalten (ich verwende normalerweise "OLYMPUS", doch "WORKGROUP" ist der Standard unter Windows, wenn ich mich richtig erinnere). **printing** muss von **bsd** auf **cups** geändert werden. Und vor der Zeile **encrypt password = yes** muss das Kommentarzeichen entfernt werden. Windows Maschinen verwenden generell Verschlüsselung, sodass die letzte Änderung essentiell ist -- außer Sie wollen die Windows Registry auf allen Ihren Windows Rechnern hacken [nicht empfehlenswert].

Unglücklicherweise verwendet Windows für die Passwörter eine andere Verschlüsselungsmethode als Linux, somit müssen Sie eine separate Passwortdatei für Samba verwalten.

**Befehlsauflistung 39:** Benutzer in Samba hinzufügen

```
# smbpasswd -a benutzer [Ersetzen Sie "benutzer" durch Ihren Benutzernamen]
New SMB password:
Retype new SMB password:
Added user benutzer.
```

### Notiz

Bevor Sie Benutzer in die Samba Passwortdatei hinzufügen, müssen diese in `/etc/passwd` gelistet sein.

Für jeden Benutzer, der Samba von einer Windows Maschine benutzen soll, muss `smbpasswd` ausgeführt werden. Samba speichert die Passwörter in `/etc/samba/private/smbpasswd`, die automatisch erstellt wird, wenn Sie das erste Mal `smbpasswd` aufrufen.

**Befehlsauflistung 40:** Samba starten

```
# /etc/init.d/samba start
Starting samba... [ ok ]
```

Starten Sie Samba und Sie sollten nun auf die Gentoo Linux Maschine von jedem Windows Rechner Ihres Netzwerks zugreifen können. Für den Benutzer, mit dem Sie zugreifen, sollte das Home-Verzeichnis sichtbar sein. Ebenso sollten die Drucker angezeigt sein. Wenn Sie nicht auf den Linux Rechner zugreifen können, schauen Sie in [/usr/share/doc/samba-2.2.5-r1/full\\_docs/textdocs/DIAGNOSIS.txt.gz](#) für weitere Hilfe nach.

## 8. Apache und Squirrelmail Webmail

### 8.1 Apache mit SSL und PHP

Angenommen, Sie haben eine feste IP-Adresse (oder benutzen einen Dienst wie dyndns mit Ihrer dynamischen IP-Adresse), dann möchten Sie einen praktischen Weg nutzen, Ihre E-Mails zu lesen, wenn Sie nicht daheim sind. Die Lösung, die ich ausgewählt habe, ist einen Webmail-Server, Squirrelmail, aufzusetzen, den ich von außen über einen beliebigen Webbrowser erreichen kann.

Der erste Schritt ist den Web-Server zu installieren. Die halbe Welt nutzt Apache; also machen wir es auch. Wir werden ebenso PHP (derzeit PHP4) installieren, da Squirrelmail ein PHP-Programm ist.

**Befehlsauflistung 41:** Installation und Konfiguration von Apache

```
# emerge apache mod_php
```

Sie sollten einen ausführlichen Blick in `/etc/apache2/conf/apache2.conf` werden, da Apache sehr vielfältig anpassbar ist. Standardmäßig ist der Apache DocumentRoot (dort, wo zuerst nach html-Dateien geschaut wird, wenn Sie auf `http://www.ihrservername.lan` oder `http://localhost` zugreifen) `/home/httpd/htdocs`; ändern Sie diesen, wenn Sie wollen.

**Befehlsauflistung 42:** `/etc/apache/conf/apache2.conf`

```
# Do NOT add a slash at the end of the directory path.
#
DocumentRoot "/home/httpd/htdocs" (passen Sie diese Zeile ggf. an)
```

**Befehlsauflistung 43:** Apache beim Booten starten

```
# rc-update add apache2 default
* apache2 added to runlevel default...
* Caching service dependencies... [ ok ]
* rc-update complete.
# nano -w /etc/conf.d/apache2
```

Verwenden Sie *rc-update*, um Apache zum default runlevel hinzuzufügen. Dann editieren Sie die Datei `/etc/conf.d/apache2` und fügen die folgenden Zeilen hinzu, sodass Apache automatisch das SSL und PHP-Modul beim Starten lädt.

**Befehlsauflistung 44:** `/etc/conf.d/apache2`

```
APACHE2_OPTS="-D SSL -D PHP4"
```

**Befehlsauflistung 45:** Apache neu starten

```
# /etc/init.d/apache2 start
* Starting Apache... [ OK ]
# lynx localhost
```

Sie können testen, ob Ihr Apache Web-Server läuft, indem Sie mit einem Webbrowser `localhost` aufrufen. Für den Test der SSL-Unterstützung rufen Sie mit einem Webbrowser `https://localhost` auf. Das *https*-Prefix teilt dem Webbrowser mit, eine **sichere** Verbindung zur Seite mittels SSL aufzubauen. Graphische Webbrowser zeigen für eine sichere Verbindung meistens irgendwo ein geschlossenes Vorhängeschloss-Symbol an.

### Notiz

Wenn Sie eine sichere Verbindung zu Ihrem Server aufbauen, werden Sie vermutlich eine Meldung erhalten, dass das Sicherheitszertifikat nicht gültig ist. Das SSL Modul installiert ein selbst-signiertes Sicherheitszertifikat, das für Testzwecke gedacht ist. Für die private Nutzung sind selbst-signierte Zertifikate okay.

Nun müssen wir herausfinden, dass PHP ordentlich funktioniert. Der erste Schritt ist die Erstellung einer einfachen PHP-Datei. Diese wird dann in das DocumentRoot-Verzeichnis kopiert.

**Befehlsauflistung 46:** PHP testen

```
# nano -w /home/httpd/htdocs/test.php
<? phpinfo(=; >
# lynx http://localhost/test.php
```

**Befehlsauflistung 47:** funktionierender Apache + PHP

```

                                                                    phpinfo() (p1 of 26)
                                PHP Logo
                                PHP Version 4.0.6
System Linux my.server.edu 2.4.10 #1 Tue Oct 9 16:33:50 Local
time zone must be set--see zic manuali686 i686 unknown
Build Date Oct 16 2001
```

Wenn Sie etwas wie das obige sehen, dann kann der Apache den PHP-Code ordentlich ausführen.

### Warnung

Wenn Sie sich die komplette Ausgabe anschauen, werden Sie herausfinden, dass das Script `test.php` eine **Menge** Informationen über Ihr System liefert. Sie werden vermutlich genug davon haben, sobald Sie wissen, dass PHP funktioniert. (Oder sie können das tun, was ich mache: `mv test.php test.phps`. Dadurch wird Apache `test.phps` als eine einfache PHP Quelldatei anzeigen anstatt das Script auszuführen.

## 8.2 Courier IMAP

Squirrelmail ist eine PHP-Applikation, aber sie beruht auf einem IMAP-Server als Bindeglied zwischen dem Web-Server und den E-Mails. Da Gentoo Linux üblicherweise **maildir** Mail Verzeichnisse verwendet, ist Courier IMAP eine gute Wahl.

**Befehlsauflistung 48:** emergem und testen von Courier IMAP

```
# emerge courier-imap
# rc-update add courier-imapd default
* courier-imapd added to runlevel default...
```

```

* Caching service dependencies... [ ok ]
* rc-update complete.
# nano -w /etc/courier-imap/authdaemonrc
# /etc/init.d/courier-imapd start
# telnet localhost 143
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.
* OK Courier-IMAP ready. Copyright 1998-2001 Double Precision, Inc.
See COPYING for distribution information.
1 LOGIN benutzer passwd (<-- ersetzen Sie "benutzer" durch Ihren
Benutzernamen und "passwd" durch Ihr Passwort)
1 OK LOGIN OK
(Drücken Sie <ctrl>-"]" und geben "quit" zum Beenden ein)

```

#### **Befehlsauflistung 49:** Korrekte Authorisation in /etc/courier-imap/authdaemonrc

```

##NAME: authmodulelist:0
#
# The authentication modules that are linked into authdaemond. The
# default list is installed. You may selectively disable modules simply
# by removing them from the following list. The available modules you
# can use are: authcustom authcram authuserdb authpam

#authmodulelist="authcustom authcram authuserdb authpam"
authmodulelist="authpam"

```

#### **Wichtig**

Normalerweise wird in der Datei `/usr/lib/courier-imap/imapd.rc` Courier IMAP davon ausgehen, dass die im **maildir**-Format vorliegende Inbox des Benutzers in `$HOME/Maildir` vorzufinden ist. Gentoo Linux hingegen benutzt `$HOME/.maildir`, sodass das courier-imap ebuild das Script `imapd.rc` anpasst, um **\$MAILDIR** zu verwenden, welches vom Gentoo Init Script `/etc/init.d/courier-imap` gesetzt wird. Möchten Sie eine andere Mailbox anstatt `.maildir` verwenden, müssen Sie den Eintrag **MAILDIR=.maildir** im courier-imap Init Script ändern.

Soll Courier IMAP SSL unterstützen, müssen Sie das IMAP SSL Zertifikat erstellen (und Courier IMAP mitteilen, wo es zu finden ist),

#### **Befehlsauflistung 50:** SSL Zertifikat für IMAP erstellen

```
# /usr/sbin/mkimapdcert
```

und dann **imap-ssl** starten.

#### **Befehlsauflistung 51:** Courier IMAP zum default runlevel hinzufügen

```
# rc-update add courier-imap-ssl default
# /etc/init.d/courier-imapd-ssl start
```

Sie können mit einer E-Mail-Applikation mit IMAP-Unterstützung, wie z.B. Mozilla, kmail oder mutt, überprüfen, ob imap funktioniert.

#### **Befehlsauflistung 52:** Testen Ihres IMAP Server

```
# mutt -f imap://localhost
```

#### **Befehlsauflistung 53:** Testen Ihres SSL IMAP Server

```
# mutt -f imaps://localhost
```

## **8.3 Squirrelmail**

Glücklicherweise ist, nachdem nun die anderen Komponenten zur Verfügung stehen, die Installation und Konfiguration von Squirrelmail recht einfach.

#### Befehlsauflistung 54: Squirrelmail mergen

```
# emerge squirrelmail
# cd /home/httpd/htdocs
# cd squirrelmail
# ./configure
```

#### Notiz

Wenn Ihr Apache DocumentRoot nicht `/home/httpd/htdocs` ist, müssen Sie das `net-mail/squirrelmail` ebuild editieren und den Pfad anpassen.

Damit Squirrelmail einwandfrei funktioniert, überprüfen Sie, dass alle seine Dateien dem Benutzer gehören, unter dem Apache läuft (das squirrelmail ebuild nimmt an, dass dieses der Benutzer "apache" ist). Innerhalb von `./configure` wählen Sie die Option 2 aus.

#### Befehlsauflistung 55: Squirrelmail Konfiguration -- Server Settings

```
SquirrelMail Configuration : Read: config.php
-----
Server Settings
1.  Domain                : my.server.edu
2.  IMAP Server           : localhost
3.  IMAP Port              : 143
4.  Use Sendmail/SMTP     : SMTP
6.   SMTP Server          : localhost
7.   SMTP Port            : 25
8.   Authenticated SMTP   : false
9.   Server                : courier
10. Invert Time           : false
11. Delimiter             : detect

R  Return to Main Menu
C.  Turn color off
S  Save data
Q  Quit

Command >>
```

Überprüfen Sie, dass **Server** auf **courier** steht. Gehen Sie zum Hauptmenü zurück und wählen Sie die Option 4, allgemeine Optionen.

#### Befehlsauflistung 56: Squirrelmail Konfiguration -- General options

```
SquirrelMail Configuration : Read: config.php
-----
General Options
1.  Default Charset       : iso-8859-1
2.  Data Directory        : ../data/
3.  Attachment Directory  : /tmp/
4.  Default Left Size     : 150
5.  Usernames in Lowercase : false

R  Return to Main Menu
C.  Turn color off
S  Save data
Q  Quit

Command >>
```

Ich verwende das Data Directory innerhalb des squirrelmail-Pakets, obwohl es sicherer wäre, eines außerhalb des Apache DocumentRoot zu verwenden. Außerdem nutze ich `/tmp` als Verzeichnis zum auslagern von Anhängen.

Nun müssen wir noch Apache neu starten und sind fertig!

#### Befehlsauflistung 57: Apache neustarten

```
# /etc/init.d/apache2 restart
```

Öffnen Sie nun mit einem Webbrowser <https://localhost/squirrelmail/src/login.php>, was klappen sollte.

**Befehlsauflistung 58:** Squirrelmail testen

```
# lynx https://localhost/squirrelmail/
```

**Befehlsauflistung 59:** Squirrelmail Login Seite

```

                                SquirrelMail - Login
                                [sm_logo.jpg]

                                SquirrelMail version 1.2.0 [rc2]
                                By the SquirrelMail Development Team

                                SquirrelMail Login

                                Name: _____
                                Password: _____

                                Login
```

(Ich empfehle nicht wirklich Squirrelmail mit lynx zu bedienen, aber Sie können es tun!)

## 9. Personal Information Management (PIM)

### 9.1 Evolution mit Unterstützung für Palm

Evolution ist eine ganz besonders nette Personal Information Management Software, die mit dem Palm pilot recht gut syncen kann. Der erste Schritt ist Evolution zu mergen. Wenn Sie Evolution mit einem Palm pilot verwenden möchten, schauen Sie nach, dass **pda** in Ihrer USE Variable in `/etc/make.conf` mit angegeben ist, bevor Sie Evolution mergen.

**Befehlsauflistung 60:** Evolution mergen

```
# emerge --pretend evolution
# emerge evolution
```

Unglücklicherweise benötigt das Kompilieren von Evolution eine Ewigkeit (beachten Sie, dass *mozilla* eine Abhängigkeit ist!); halten Sie mehrere Kannen Kaffee/Tee bereit, während Sie warten.

Beim Kompilieren von *evolution* werden die benötigten Palm pilot Tools direkt mit gebaut. Damit normale Benutzer den Palm pilot syncen können, müssen wir ein paar Änderungen an den Rechten für den seriellen Port vornehmen. Ich gehe davon aus, dass Ihre Palm-Station an den ersten seriellen Port angeschlossen ist (`/dev/ttyS0` oder `/dev/tts/0` in der "devfs" Notation).

**Befehlsauflistung 61:** Hinzufügen von seriellen Geräten

```
# groupadd serial
# chown root.serial /dev/tts/0
# chmod g+rw /dev/tts/0
# ls -l /dev/tts/0
crw-rw----  1 root    serial    4,  64 Jan  1 1970 /dev/tts/0
# ln -s /dev/tts/0 /dev/pilot
# nano -w /etc/group (fügen Sie die entsprechenden Benutzer zur Gruppe serial hinzu)
```

Wir haben eine "serial" Gruppe zum System hinzugefügt und Sie sollten alle Benutzer, die den seriellen Port benötigen (jeder Palm pilot Benutzer), zur Gruppe serial hinzufügen. Wir haben der serial Gruppe "rw" Rechte für `/dev/tts/0` vergeben und für die zukünftige Verwendung den symbolischen Link `/dev/pilot` auf `/dev/tts/0` erstellt. Dank der neuesten Änderungen wie devfs unter Gentoo gehandhabt wird, bleiben diese Änderungen nach einem Reboot bestehen.

Um die Gnome pilot Software einzurichten, verwenden Sie das `tools/Pilot settings` Menü in Evolution. Vergessen Sie nicht im Gnome Control Center die "Pilot Conduits" einzurichten als auch den "Pilot Link" zu setzen.



## >> Gentoo Linux ALSA Anleitung

[Bitte Kapitel auswählen] 

### 1. Einführung

#### 1.1 Was ist ALSA?

ALSA (Advanced Linux Sound Architecture) ist ein Projekt, mit dem die Steuerung des Audio Subsystems unter Linux verbessert werden soll, indem viele Bereiche neu programmiert werden. Es ist geplant ALSA in den Linux Kernel 2.6.x (oder 3.x.x -- je nachdem welcher zu erst fertig ist) zu integrieren.

ALSA stellt die Audio and MIDI Funktionen für Linux bereit.

Wir zitieren von der Webseite des <http://www.alsa-project.org> folgende Funktionen:

- Unterstützung für alle Audio Schnittstellen von Standard Soundkarten bis zu professionellen "multichannel audio interfaces".
- Vollkommen modularisierte Sound Treiber.
- SMP und thread-safe design.
- Benutzerschnittstelle (alsa-lib) um das Programmieren von Anwendungen, die auf Ton beruhen, deutlich zu vereinfachen und verbesserte Funktionen bereitzustellen.
- Unterstützung für die ältere OSS (Open Sound System) Benutzerschnittstelle, welches die Abwärtskompatibilität zu älteren OSS Programmen garantiert.

Darüberhinaus kann ALSA noch viel mehr, wie z.B. Full Duplex Aufnahme und Wiedergabe, Unterstützung für mehrere Soundkarten, hardware mixing von Streams, stark erweiterte Mischeigenschaften, um die vielen Features der neuen Soundkarten auch ausnutzen zu können...

#### 1.2 Warum ALSA benutzen?

Wenn Ihre Soundkarte bereits durch den Linux Kernel oder durch das kommerziell erhältliche OSS/4Front Audio Treiber System (welches in allen 2.4.x Linux Kernels vorhanden ist) unterstützt wird, können Sie auch **diese Module** für die Benutzung mit Ihrer Soundkarte kompilieren. Eine Anleitung hierzu finden Sie im [Linux Sound HOWTO](#).

Jedoch haben diese OSS/4Front Treiber einige Einschränkungen -- die kommerzielle Lizenz ist eine davon. ALSA ist ein Ansatz diese Einschränkungen zu umgehen und darüber hinaus Open Source Software mit vollwertiger GPL und LGPL Lizenz einzusetzen, ohne auf professionelle Tonqualität beim Aufnehmen, Abspielen und MIDI Sequenzierung verzichten zu müssen.

#### 1.3 Welche Karten werden von ALSA unterstützt?

ALSA versucht so viele neue Karten wie möglich mit Open-Source Treibern zu unterstützen. Einige Hersteller stellen allerdings nur binäre Pakete zur Verfügung.

Um herauszufinden, ob Ihre Karte von ALSA unterstützt wird, gibt es eine Liste von Soundkarten, die bereits unter ALSA funktionieren: <http://www.alsa-project.org/alsa-doc/>.

### 2. Installation

#### 2.1 Gentoo USE Flags

Um Programme mit ALSA-Unterstützung zu kompilieren, fügen Sie **alsa** zu Ihrer USE-Variablen hinzu. Es gibt dennoch Tools, die Alsa noch nicht unterstützen und OSS benötigen. Alsa bietet eine OSS-Emulation an, wenn Sie **oss** in Ihrer USE-Variable definieren, bevor Sie starten.

#### 2.2 Kernel Module

Da wir zur Zeit noch die 2.4.x Kernel Serie benutzen, müssen die Kernel und ALSA Module separat kompiliert werden. Benutzer der 2.5.x (oder neuere) Serie können dieses aus der Kernelkonfiguration heraus machen, da Alsa in die Kernel-Sourcen aufgenommen wurde und von dort gebaut werden sollte.

Zuerst müssen wir sicherstellen, dass unsere Kernel Konfiguration für den Einsatz von ALSA vorbereitet ist. Dazu ist es notwendig, dass im Kernel der Punkt "Sound Card Support" als Modul (M) einkompiliert wird. Dadurch wird das Modul *soundcore.o* erstellt.

### Notiz

Wahrscheinlich wird es auch funktionieren, wenn "Sound Card Support" direkt in den Kernel (Y) einkompiliert, anstatt es als Modul (M) zu bauen. Jedoch empfiehlt die offizielle ALSA Dokumentation es als Modul (M) zu erstellen, da ALSA beim Start versucht, das Modul zu laden.

Wenn Sie bereits eine funktionsfähige Kernel Konfiguration besitzen, sollten Sie sicherstellen, dass alle anderen Soundkarten Treiber entfernt wurden, mit Ausnahme des bereits erwähnten "Sound Card Support". Um nicht neu booten zu müssen, empfiehlt sich folgendes Vorgehen:

### Befehlsauflistung 1

```
# cd /usr/src/linux
# cp .config ~
# make mrproper
# cp ~/.config .
# make menuconfig
```

Nun sollte **Sound Card Support** als Modul (M) ausgewählt und alle anderen Soundkarten Treiber abgewählt werden. Beenden Sie *menuconfig* und speichern Sie mit Y die Kernel Konfiguration. Danach bauen Sie die Module mit:

### Befehlsauflistung 2

```
# make dep clean
# make modules modules_install
```

Die letzte Zeile stellt sicher, dass vor der Installation neuer Module die alten überschrieben werden, auch die einer älteren ALSA Installation.

### Wichtig

Das bedeutet, dass JEDES MAL wenn der Kernel neu kompiliert wird, *alsa-driver* ebenfalls neu kompiliert werden muss!

### Notiz

Jedoch ist es nicht notwendig *nvidia-kernel* neu zu installieren, da die Nvidia Treiber in einem separatem Verzeichnis abgelegt werden (konkret: */lib/modules/\*/video*) und somit von dem Befehl *make modules modules\_install* nicht betroffen sind.

## 2.3 Das richtige ALSA Modul

Nun wird es Zeit, die passenden ALSA Treiber für Ihre Soundkarte zu installieren. Wenn Ihre Soundkarte auf einem PCI Steckplatz sitzt, kann der output von */proc/pci* den Namen und Typ der Karte offenbaren:

### Befehlsauflistung 3

```
#grep audio /proc/pci
```

### Warnung

Sollten Sie zuvor ein anderes Audio System eingerichtet haben und Nicht-ALSA Module sich noch im Speicher befinden, empfiehlt es sich, diese **jetzt** zu entfernen. Überprüfen Sie mit *lsmod* Ihre geladenen Module und entfernen Sie mit *rmmod* nicht mehr benötigte Treiber auf Ihrem System.

Ein einfaches *emerge alsa-driver* würde **alle** ALSA Treiber kompilieren und in das System einbinden.

Um jedoch Zeit zu sparen, empfiehlt es sich, den **Modul Namen** Ihrer Soundkarte in der [ALSA Soundcard Matrix](#) und folgend **Details** bei **Driver and Docs** in der Spalte des Chipsatzes Ihrer Soundkarte nachzuschlagen. Bei mir handelt es sich um das Modul *snd-emu10k1*, da ich eine SBlive! Soundkarte mit dem **EMU10K!** Chipsatz besitze. Dann setzen Sie die *ALSA\_CARDS* environment Variable, bevor Sie *emerge* ausführen, sodass *emerge* nur die Treiber, die Sie wirklich brauchen auch kompilieren wird. Bei mir sieht das folgendermaßen aus:

### Befehlsauflistung 4

```
# env ALSA_CARDS='emu10k1' emerge alsa-driver
```

### Notiz

Sie können diese Variable auch in Ihrer `/etc/make.conf` speichern, sodass Sie bei einem späteren neuen `emerge` einfach nur `emerge alsa-driver` aufrufen brauchen. Führen Sie z.B. `echo 'ALSA_CARDS="emu10k1"' >> /etc/make.conf` aus.

### Notiz

Wenn Sie vorhaben, ALSA Treiber für mehr als eine Soundkarte zu installieren, können diese durch ein Leerzeichen getrennt an `ALSA_CARDS` übergeben werden, z.B. so: `env ALSA_CARDS='emu10k1 intel8x0 ens1370' emerge alsa-driver`

### Notiz

Sollten Sie OSS Kompatibilität benötigen, so sollte "`emerge alsa-oss`" ausgeführt werden, um den ALSA/OSS compatibility wrapper auf Ihrem System einzurichten.

### Notiz

Wenn Sie planen, Ihren Kernel mehrmals neu zu kompilieren, ist es sinnvoll `alsa-driver` mit der `emerge`-Option `-buildpkg` zu kompilieren. Dadurch erstellen Sie ein Binärpaket. Nachdem Sie den Kernel neu kompiliert haben, können Sie einfach `emerge --usepkg alsa-driver` ausführen, was das Binärpaket installieren wird anstatt den Treiber neu zu kompilieren.

Mit diesen wenigen Handgriffen, sollten die ALSA Module bereits auf Ihrem System installiert sein.

## 2.4 Konfiguration von ALSA

Als nächstes sollten Sie die Konfiguration vornehmen, um ALSA sauber einzurichten. Dazu müssen einige Konfigurationsdateien geändert werden, damit Ihr System von den neu installierten Modulen auch Kenntnis nimmt.

Zuallererst installieren Sie die `alsa-utils` auf Ihrem System:

**Befehlsauflistung 5:** `alsa-utils` emergieren

```
# emerge alsa-utils
```

Danach editieren Sie die Datei `/etc/modules.d/alsa`.

### Warnung

Es ist nicht notwendig `/etc/modules.conf` zu modifizieren. Es sollten immer die Dateien in `/etc/modules.d` bearbeitet werden.

Überprüfen Sie den ALSA Abschnitt am **Ende der Konfigurationsdatei**. Durch Anpassen dieser Zeile können Sie die maximale Nummer der benutzen Soundkarten auf Ihrem System angeben (im Normalfall ist dies nur eine einzige).

**Befehlsauflistung 6:** Am Ende von `/etc/modules.d/alsa`

```
Set this to the correct number of cards.
options snd cards_limit=1
```

Nun geben Sie an, welche Treiber ALSA benutzen soll. Im der gleichen Datei tragen Sie folgende Zeilen ein:

**Befehlsauflistung 7:** Innerhalb von `/etc/modules.d/alsa`

```
## and then run `modules-update' command.
## Read alsa-driver's INSTALL file in /usr/share/doc for more info.
##
## ALSA portion
alias snd-card-0 snd-emu10k1
## If you have more than one, add:
## alias snd-card-1 snd-intel8x0
## alias snd-card-2 snd-ens1370
## OSS/Free portion
## alias sound-slot-0 snd-card-0
## alias sound-slot-1 snd-card-1
##
```

Natürlich sollten Sie hier darauf achten `snd-emu10k1` durch den Namen des richtigen Moduls für Ihre Soundkarte zu ersetzen.

### Notiz

Wenn Sie mehr als eine Soundkarte besitzen, sollte der Wert für `cards_limit` angepasst werden und entsprechend für jede Soundkarte ein `snd-card` alias eingefügt werden. Sollten Sie damit keine Erfahrung haben, so finden Sie in [Kapitel 6](#) des [ALSA Howto](#) weitergehende Informationen.

Als letztes sollten Sie noch sicherstellen, dass folgende Zeilen ganz am Ende der Konfigurationsdatei vorhanden und unkommentiert sind:

**Befehlsauflistung 8:** am Ende von `/etc/modules.d/alsa`

```
alias /dev/mixer snd-mixer-oss
alias /dev/dsp snd-pcm-oss
alias /dev/midi snd-seq-oss
```

Überprüfen Sie noch einmal alle Ihre Eingaben in der Datei `/etc/modules.d/alsa` und wenn Sie sicher sind, dass kein Fehler vorhanden ist, können Sie `modules-update` ausführen:

**Befehlsauflistung 9**

```
# update-modules
```

### Notiz

Durch das Ausführen von `modules-update` werden die Daten von `/etc/modules.d/alsa` in `/etc/modules.conf` geschrieben.

Nun sollten Sie sicherstellen, dass in `/etc/devfsd.conf` die ALSA Geräte korrekt eingetragen und registriert sind.

**Befehlsauflistung 10:** `/etc/devfsd.conf`

```
# ALSA/OSS stuff
# Comment/change these if you want to change the permissions on
# the audio devices
LOOKUP      snd          MODLOAD ACTION snd
LOOKUP      dsp          MODLOAD
LOOKUP      mixer       MODLOAD
LOOKUP      midi         MODLOAD
REGISTER    sound/. *    PERMISSIONS root.audio 660
REGISTER    snd/. *      PERMISSIONS root.audio 660
```

### Notiz

Wie man sehen kann, werden von `devfsd.conf` die Zugriffe auf `/dev/sound` auf `root.audio` beschränkt. Nicht root-user müssen also in der Gruppe `audio` sein, um die Soundkarte auch zu hören.

## 3. ALSA starten

### 3.1 Alsasound zu einem runlevel hinzufügen

Nun sollten Sie daran gehen, ALSA beim booten auch zu starten. Dies geschieht mit einem:

**Befehlsauflistung 11**

```
# rc-update add alsasound boot
```

### Warnung

Das `alsasound` Skript gehört in den "boot" runlevel, und nicht in den "default" runlevel!

### 3.2 Stummschaltung aufheben

Da wir Linux User sind und Rebooten des Systems uncool ist, starten wir das `alsasound` Skript von Hand:

**Befehlsauflistung 12**

```
# /etc/init.d/alsasound start
```

ALSA sollte jetzt laufen. Wenn alles ok ist, sollten beim Ausführen von `lsmod` die ALSA Module angezeigt werden. Allerdings bekommen Sie noch keinen Ton zu hören, da die Kanäle noch stumm geschaltet sind. Um die Einstellungen zu ändern, brauchen wir `amixer`.

**Befehlsauflistung 13:** amixer aufrufen

```
# amixer
```

### Warnung

Normalerweise sollte es gar nicht dazu kommen, aber **wenn** Sie eine Fehlermeldung in der Art von "amixer: Mixer attach default error: No such file or directory" bekommen, sollten Sie *snd-mixer-oss* und *snd-pcm-oss* aus dem Speicher werfen und erneut laden. Danach sollte amixer noch einmal aufgerufen werden.

**Befehlsauflistung 14:** Nur bei Fehlermeldungen durch amixer

```
# modprobe snd-mixer-oss
# modprobe snd-pcm-oss
# amixer
```

Wenn Sie nun so weit gekommen sind, sollten Sie die Stummschaltung der Master und PCM Kanäle aufheben. Manche Hardware verlangt, dass auch bei dem Center- und die Surround-Kanäle die Stummschaltung aufgehoben wird.

**Befehlsauflistung 15:** Stummschaltung aufheben

```
# amixer set Master 100 unmute
# amixer set PCM 100 unmute
Nur, falls das obige zu keinem Erfolg führte:
# amixer set Center 100 unmute
# amixer set Surround 100 unmute
Sound testen:
# aplay $KDEDIR/share/sounds/pop.wav // (pop.wav ist Bestandteil des KDE Paketes)
```

Um zu Überprüfen, ob der Ton funktioniert, benutzen wir den Befehl `aplay` (alsa play). Wenn Sie einen Klang hören, dann arbeitet Ihre Soundkarte. Herzlichen Glückwunsch! Um die Lautstärke an Ihre Bedürfnisse anzupassen, ist das ncurses-basierte Paket *alsamixer* die richtige Wahl.

Sie sind auch sicher dran interessiert *alsa-xmms* zu installieren, da es die ALSA Unterstützung für XMMS bereitstellt.

Wenn Sie Ihr System neu booten, wird das **alsasound** init Skript ihre Lautstärkeeinstellungen sichern und korrekt wieder herstellen.

## 4. Abschließende Bemerkungen

### 4.1 Nach Kernel Aktualisierungen

Wenn Sie Ihren Kernel neu kompilieren oder auf einen aktuelleren Kernel umsteigen, dann müssen Sie die ALSA Module ebenfalls neu kompilieren.

Auch wenn Sie alle Pakete: *alsa-driver*, *alsa-libs* und *alsa-utils* installiert haben, ist es notwendig nur das erste Paket neu zu kompilieren, da es die Module unter `/lib/modules/*/kernel/sound/pci/` ablegt. Ein einfaches

**Befehlsauflistung 16:** Nach jeder Kernel Neuerstellung:

```
# emerge alsa-driver
```

genügt vollständig.

### 4.2 /etc/modules.autoload

Um ALSA zu benutzen, muss diese Datei im Normalfall nicht modifiziert werden. Mit unserem `rc-update add alsasound boot` wird das System die richtigen Module beim Start automatisch laden.

Es ist also nicht notwendig *snd-pcm-oss* oder *snd-mixer-oss* in diese Datei einzufügen. Für weitere Informationen oder häufige Fragen ist [diese FAQ](#) der richtige Anlaufpunkt.

### 4.3 Bekannte Fehler

#### Notiz

Diese Anleitung hängt hinter der Alsa-Entwicklung hinterher. Möglicherweise wurden die Fehler schon längst beseitigt während Sie diese Zeilen hier lesen.

- Wenn Sie **viel Lärm** während der **OSS** emulation hören, fügen Sie **options snd-pcm-oss dsp\_map=1** in `/etc/modules.d/alsa` hinzu.

#### 4.4 Noch mehr Links

Für weitere Informationen rund um ALSA empfehle ich folgende Links:

- [Das Gentoo Linux Desktop Konfigurations Handbuch](#)
- [ALSA Project Homepage](#)
- [ALSA Users Documentation](#)
- [ALSA Howto's and FAQ's](#)
- [Linux Sound HOWTO](#)
- [Sound and MIDI Software For Linux](#)



## >> Gentoo Java Anleitung

[Bitte Kapitel auswählen] 

### 1. Installation eines JDK/JRE

#### 1.1 Überblick

Gentoo Linux unterstützt verschiedene JDKs und JREs. Als Standard ist das Blackdown JDK/JRE installiert, da es frei verfügbar ist und sie keinem Registrierungszwang unterliegen.

##### Notiz

As kaffe becomes a JRE/JDK drop-in replacement, that will most likely become our default.

Allgemein sind die JDKs/JREs von Sun oder IBM schneller, allerdings bedeutet es etwas mehr Arbeit an sie heran zukommen, da es notwendig ist, vor dem Herunterladen die Lizenzbestimmungen zu lesen und zu akzeptieren. IBM verlangt desweiteren eine Registrierung des Nutzers.

Die ebuilds für die Sun und IBM JDKs/JREs werden sie darauf hinweisen, von wo sie die entsprechenden Pakete herunterladen können.

#### 1.2 Installation des Sun/IBM JDKs/JREs

Wenn sie *emerge dev-java/sun-jdk-1.3.1* oder *emerge dev-java/ibm-jdk-1.3.1* ausführen, werden sie darauf hingewiesen, dass sie die entsprechenden TAR-Archive selbst herunterladen müssen. Dies ist auf Grund der Lizenzbestimmungen von Sun bzw. des Registrierungszwangs beim IBM JDK/JRE notwendig.

##### Notiz

Es gibt auch ein *dev-java/sun-jdk-1.4.0* ebuild, allerdings laufen nicht alle Programme fehlerfrei mit Java 1.4. Somit liegt es an ihnen, ob sie Java 1.4 verwenden wollen.

Sie müssen die notwendigen Dateien in das Verzeichnis `/usr/portage/distfiles` herunterladen. Sobald dies getan ist, müssen sie erneut *emerge* ausführen und somit wird das JDK/JRE nach `/opt` installiert.

### 2. Konfiguration des JDK/JRE

#### 2.1 Überblick

Gentoo Linux bietet die Möglichkeit verschiedene JDKs und JREs parallel installiert zu haben, ohne das diese in Konflikt miteinander kommen. Es gibt einige Besonderheiten, die weiter unten erläutert werden.

Mit Hilfe von *java-config* können sie als root die systemweiten Java-Vorgaben festlegen. Außerdem hat jeder Nutzer die Möglichkeit durch *java-config* seine eigenen Vorgaben einzustellen, wenn diese sich von den systemweiten unterscheiden.

#### 2.2 Ein Standard-JDK/JRE festlegen

Durch das Kommando *java-config --list-available-vm*s wird eine Liste der auf ihrem System verfügbaren JDKs/JREs ausgegeben:

##### Befehlsauflistung 1

```
[%1 ~] java-config --list-available-vm
[blackdown-jdk-1.3.1] Blackdown JDK 1.3.1 (/etc/env.d/java/20blackdown-jdk-1.3.1)
[blackdown-jre-1.3.1] Blackdown JRE 1.3.1 (/etc/env.d/java/20blackdown-jre-1.3.1)
[ibm-jdk-1.3.0] IBM JDK 1.3.0 (/etc/env.d/java/20ibm-jdk-1.3.0)
[ibm-jdk-1.3.1] IBM JDK 1.3.1 (/etc/env.d/java/20ibm-jdk-1.3.1)
[ibm-jre-1.3.1] IBM JRE 1.3.1 (/etc/env.d/java/20ibm-jre-1.3.1)
[sun-jdk-1.4.0] Sun JDK 1.4.0 (/etc/env.d/java/20sun-jdk-1.4.0)
```

Der Text in eckigen Klammern "[ ]" ist der Name oder die ID für die jeweilige VM (Virtual Machine). Sie müssen diese ID mit *java-config --set-system-vm* wie folgt verwenden:

##### Befehlsauflistung 2

```
[#1 ~] java-config --set-system-vm=ibm-jdk-1.3.1
Now using IBM JDK 1.3.1 (/etc/env.d/java/20ibm-jdk-1.3.1)
```

### Notiz

Sie müssen root sein, um `--set-system-vm` auszuführen

Sobald sie `java-config --set-system-vm` mit einer entsprechenden VM ID ausgeführt haben, müssen sie ihre `/etc/profile.env` aktualisieren:

### Befehlsauflistung 3

```
[#1 ~] env-update
```

Anschließend müssen sie sich neu am System anmelden oder ihre `/etc/profile` erneut in die Umgebung einbinden (`source /etc/profile`).

Als normaler Nutzer können sie `java-config --set-user-vm` anwenden, um ein eignes JDK/JRE festzulegen. Dabei wird die Datei `$HOME/.gentoo/java-env` mit allen notwendigen Umgebungsvariablen angelegt. Für gewöhnlich wird dies in den Start-Skript der Shell eingebunden (z.B. `$HOME/.zshenv`)

## 2.3 Einen Standard CLASSPATH setzen

`java-config` kann (und sollte) desweiteren dazu verwendet werden, sowohl einen systemweiten Standard CLASSPATH (Klassenpfad), als auch einen Nutzerspezifischen CLASSPATH zu setzen.

Zunächst sollte man sich eine Liste mit allen verfügbaren Bibliotheken ausgeben lassen:

### Befehlsauflistung 4

```
[%1 ~] java-config --list-available-packages
[ant] No description (/usr/share/ant/classpath.env)
[java-gnome] No description (/usr/share/java-gnome/classpath.env)
[java-gtk] No description (/usr/share/java-gtk/classpath.env)
[log4j] " " (/usr/share/log4j/package.env)
```

### Notiz

Derzeit hat keines der Pakete eine passende Beschreibung, dies wird jedoch in naher Zukunft implementiert.

Nochmal sei darauf hingewiesen, dass der Text in eckigen Klammern "[ ]", die IDs sind, die an `java-config --set-system-classpath` übergeben werden müssen:

### Befehlsauflistung 5

```
java-config --set-system-classpath=log4j,java-gtk,java-gnome
```

### Notiz

Das aktuelle Verzeichnis (.) wird kein Teil des Systemweiten Klassenpfades sein. Es sollte in das Login-Profil von root eingetragen werden.

Erneut müssen sie `env-update` ausführen, um die Systemumgebung zu aktualisieren. Außerdem müssen sie sich neu anmelden oder `/etc/profile` neu einbinden (`source /etc/profile`).

Für normale Nutzer erzeugt `java-config --set-user-classpath` die Datei `$HOME/.gentoo/java-env-classpath`, welche automatisch von `$HOME/.gentoo/java-env` eingebunden wird

## 3. Weiterführende Quellen

### 3.1 Offline Quellen

- `java-config` man page
- `java-config --help`
- Das `/usr/bin/java-config` Skript selbst

### 3.2 Online Quellen

- Die Archive der [gentoo-dev](#) und [gentoo-user](#) Mailing-Listen
- #gentoo unter [irc.openprojects.net](#)

## 4. Besonderheiten

### 4.1 Allgemein

Wenn sie eine JRE als ihren vorgegeben Java-Interpreter (Java-VM) eingestellt haben, steht ihnen der Java-Compiler *javac* nicht zur Verfügung, es sei denn sie setzen selbst einen Link auf die Datei *javac*.

Sobald es verfügbar ist wird Jikes verwendet, um einige der größeren Pakete zu übersetzen. Mit der Zeit wird es hoffentlich möglich werden alle Java-abhängigen Pakete mit Jikes zu übersetzen, so dass einige Abhängigkeiten vom JDK zu Abhängigkeiten vom JRE überführt werden können.

Bei einigen Versionen von Portage führt ein *emerge --update world* dazu, dass ein JDK installiert wird, ohne Rücksicht darauf, ob sie es bereits vorher installiert hatten oder nicht. Dies ist ein bekannter Fehler in Portage und wird in einer zukünftigen Version behoben.



# >> Gentoo Linux Anleitung zur deutschen Lokalisierung

[Bitte Kapitel auswählen] ▾

## 1. English, please

This guide is also available in [English](#).

## 2. Uhrzeit

### 2.1 Zeitzone

Um die richtige Zeitzone einzustellen, sollte `/etc/localtime` auf die richtige Zeitzone, deren Daten für Deutschland in `/usr/share/zoneinfo/Europe/Berlin` enthalten sind, zeigen.

**Befehlsauflistung 1:** Zeitzone setzen

```
# ln -sf /usr/share/zoneinfo/Europe/Berlin /etc/localtime
# date
Son Feb 16 08:26:44 CET 2003
```

#### Notiz

Schauen Sie, dass das die Zeitzone durch die drei Buchstaben (in diesem Fall "CET") richtig dargestellt ist.

#### Notiz

Sie können selbst die Variable **TZ** in Ihrer Shell rc-Datei (bei der Bash `.bash_profile`) setzen, die alle Zonendaten aus `/usr/share/zoneinfo` erlaubt. In diesem Fall **TZ="Europe/Berlin"**.

### 2.2 Systemuhr

Wer neben Linux auch noch andere Betriebssysteme installiert, und daher seine Rechneruhr auf die lokale Zeit eingestellt hat, sollte die Umgebungsvariable **CLOCK** in `/etc/rc.conf` richtig stellen. Das hat allerdings den Nachteil, dass nach einer Zeitumstellung die Uhrzeit solange falsch ist, bis man (oder das andere Betriebssystem) die Echtzeituhr entsprechend korrigiert hat. Für reine Linux-Systeme ist es daher empfehlenswert die Echtzeituhr auf UTC (universelle Zeit auf dem 0-Meridian) einzustellen.

**Befehlsauflistung 2:** Lokale bzw. UTC Systemuhr

```
// empfohlen:
CLOCK="UTC"
// oder:
CLOCK="local"
```

## 3. Sprache

### 3.1 Allgemein

Die bevorzugte Sprache wird durch die Umgebungsvariable **LANG** eingestellt, die von der Shell und Window Managern (sowie anderen Applikationen) interpretiert wird. Gültige Werte sind in der Datei `/usr/share/locale` notiert und grundsätzlich von der Form **ab\_CD**, wobei **ab** ein Zwei-Buchstaben-Code für die Sprache und **CD** einer für das Land ist. **\_CD** wird ausgelassen, wenn diese Sprache nur (oder hauptsächlich) in einem Land gesprochen wird. Für eine systemweite Einstellung sollte man **LANG** in `/etc/profile` exportieren. Möchte man systemweit keine Sprache vorgeben, kann man die Variable auch in der `~/bashrc` im eigenen Homeverzeichnis setzen.

**Befehlsauflistung 3:** Setzen der POSIX locale

```
export LANG="de_DE@euro"
```

#### Notiz

Fügen Sie **@euro** zu Ihrer Variable an, wenn Sie das neue Euro-Symbol (€) nutzen wollen.

### 3.2 KDE

Damit man in KDE seine gewünschte Sprache einstellen kann, muss man das entsprechende Sprachpaket installieren.

**Befehlsauflistung 4:** KDE in Deutsch

```
# export LINGUAS="de"
# touch /usr/portage/kde-base/kde-i18n/kde-i18n-3.1.4.ebuild
# emerge kde-i18n
```

### 3.3 OpenOffice

#### Notiz

Angepasste Sprache ist für das openoffice-bin ebuild nicht verfügbar. Die Standardsprache ist dort ENUS (US-English).

Die Standardsparche für OpenOffice sit auf "ENUS"(01) gesetzt. Wenn Sie die Standardsprache für OpenOffice ändern möchten, schauen Sie in das ebuild für den Sprachencode.

**Befehlsauflistung 5:** OpenOffice mit der gewünschten Standardsprache emergent

```
# LANGUAGE="01" emerge openoffice
01 ist der Sprachcode für ENUS bei OpenOffice
```

## 4. Tastatur

### 4.1 Tastatur-Layout für die Konsole

Das Tastatur-Layout für die Konsole wird in der Datei `/etc/rc.conf` durch **KEYMAP** eingestellt. Für eine deutsche Tastatur (qwertz) sind die entsprechenden Tabellen in `/usr/share/keymaps/{arch}/` zu finden. `{arch}` hat weitere Unterscheidungen in verschiedene Layouts (`querty/`, `qwertz/`, `azerty/`, usw.). Manche Sprachen haben verschiedene Optionen. Sie sollten Austesten, welches Layout Sie bevorzugen. Wer nur sehr selten Zeichen mit Tilden (z.B. ñ), Accent grave (z.B. è) o.ä. benutzt, kann die `nodeadkeys`-Varianten benutzen. Diese ersparen einem das, auf das eigentliche Zeichen einzugebende, Leerzeichen, wenn man z.B. nur `~` eingeben möchte.

**Befehlsauflistung 6:** Die Konsolen-Keymap setzen

```
// Eines von
KEYMAP="de"
KEYMAP="de-latin1"
KEYMAP="de-latin1-nodeadkeys"
```

### 4.2 Tastatur-Layout für den X-Server

Das Tastatur-Layout für den X-Server wird in der Datei `/etc/X11/XF86Config` durch die Option **XkbLayout** eingestellt.

**Befehlsauflistung 7:** Die X-Tastaturbelegung setzen

```
Section "InputDevice"
    Identifier "Keyboard1"
    ...
    Option "XkbLayout" "de"
    # Option "XkbVariant" "nodeadkeys"
    ...
```

### 4.3 Eurosymbol für die Konsole

Damit man das Eurosymbol auch in einer Terminalsession ohne X zu sehen bekommt, muss man die Umgebungsvariable **CONSOLEFONT** in `/etc/rc.conf` auf eine Schriftart setzen, die das Eurozeichen beinhaltet. Verfügbare Schriftarten findet man in `/usr/share/consolefonts` (ohne das `.psfu.gz`). *lat0-16 hat das Euro-Symbol.*

**Befehlsauflistung 8:** Setzen der Konsolenschrift

```
CONSOLEFONT="lat0-16"
```

## 4.4 Eurosymbol für den X-Server

Das Euro-Symbol in X zu erhalten ist ein wenig komplizierter. Zuerst sollten Sie die **fixed** und **variable** Definitionen in `/usr/X11R6/lib/X11/fonts/misc/fonts.alias` in Endungen mit **iso8869-15** anstatt **iso8859-1** abändern.

**Befehlsauflistung 9:** Setzen der Default X Schriften

```
fixed      -misc-fixed-medium-r-semicondensed--13-120-75-75-c-60-iso8859-15
variable   -*helvetica-bold-r-normal-*--120-*--*--*--iso8859-15
```

Einige Applikationen verwenden ihre eigene Schriftart und so müssen Sie diesen eigens mitteilen, dass sie eine Schrift mit dem Euro-Zeichen verwenden sollen. Sie können das für jeden Benutzer in der Datei `~/.Xdefaults` (oder auch für neue Benutzer in die entsprechende Datei in `/etc/skel` eintragen) oder global für jede Applikation mit einer Datei in `/usr/X11R6/lib/X11/app-defaults/` (wie z.B. `xterm`) festlegen. In diesen Dateien brauchen Sie meistens nur eine Zeile ändern anstatt eine neue anzulegen. Ändern Sie beispielsweise Ihre `xterm`-Schrift:

**Befehlsauflistung 10:** Schrift für `xterm` setzen

```
// (In Ihrem Home-Directory)
$ echo 'XTerm*font: fixed' >> .Xresources
$ xrdp -merge .Xresources
```

## 4.5 Eurosymbol für (X)Emacs

Für den (X)Emacs definiert man die Schriftart wie folgt in `.Xdefaults`:

**Befehlsauflistung 11:** Schrift für emacs setzen

```
Emacs.default.attributeFont: -*courier-medium-r-*--*--120-*--*--*--iso8859-15
```

Der XEmacs (nicht Emacs) benötigt noch etwas mehr Arbeit. Da er bisher noch nichts mit dem Zeichen **EuroSign** anfangen kann, muss man ihm dies noch per Befehl in der Datei `/home/user/.xemacs/init.el` beibringen:

**Befehlsauflistung 12:** Schrift für `xemacs` setzen

```
(define-key global-map '(EuroSign) '[@])
// Das Zeichen in den eckigen Klammern ist das Eurosymbol.
```



## >> CDs Brennen in Gentoo

[Bitte Kapitel auswählen] 

### 1. Brennen unter Linux

#### 1.1 Einführung

CDs und DVDs brennen unter Linux gilt für viele immer noch als sehr komplizierte Sache. Dies liegt zum einen daran, dass viele Einstellungen distributionsspezifisch sind und somit oft unterschiedliche Vorgehensweisen dargestellt werden, zum anderen auch daran, dass viele Dokumentationen auch veraltet sind. Mit dieser Anleitung sollte es für den Grossteil aller Gentoo Nutzer kein Problem sein, ihren Brenner zum Laufen zu bekommen. Natürlich kann nicht auf alle möglichen Varianten eingegangen werden, sollte dann aber sicherlich den einen oder anderen in die richtige Richtung weisen.

#### 1.2 Die IDE-SCSI Emulation

In der Vergangenheit war CD Brennen nur möglich, wenn man einen SCSI Brenner besass oder den ATAPI-IDE Brenner unter Linux als SCSI Gerät benutzte. Das führte natürlich zur häufig gestellten Frage, warum unter Linux keine Unterstützung für IDE bereitgestellt wurde. Hierzu muss klargestellt werden, dass es IDE Brenner eigentlich nicht gibt, da die IDE Schnittstelle keine solchen Funktionen bereitstellt.

Funktionen wie das Auswerfen einer CD oder das Brennen von Rohlingen kann nur mit SCSI Befehlen durchgeführt werden. Der ATAPI Standard (Attachment Adapter Packet Interface) stellt eine Möglichkeit dar, solche SCSI Befehle über die IDE Schnittstelle zu verschicken. ATAPI-kompatible Geräte mit einer IDE Schnittstelle können also SCSI Befehle nativ verstehen und verarbeiten.

Da die IDE CD-ROM Treiber älterer Linux-Kernel keine Unterstützung für ATAPI anboten, behalf man sich mit einer Emulation, die dem Kernel die IDE CD-ROMS als SCSI Geräte bereitstellte, um so die SCSI-Befehle an die jeweiligen Geräte zu verschicken können. Alternativ entschied man sich gleich zum Kauf eines echten SCSI CD-ROMS, falls man ein entsprechendes System besass.

In der 2.4. Kernel Serie wurde in den IDE-CD Treiber ein sogenanntes CDRom Packet Interface integriert, welches inzwischen auch von cdrecord unterstützt wird. Der Nachteil dieser Lösung ist eine schlechtere Performance, da DMA (Direct Memory Access) nicht möglich ist. Seit der 2.5.45 Kernel Serie gibt es endlich native Unterstützung für ATAPI, die nun auch Brennen mit DMA ohne SCSI ermöglicht. Fast alle der heute auf dem Markt erhältlichen CD oder DVD Brenner sind ATAPI kompatibel und besitzen eine IDE Schnittstelle.

#### **Notiz**

Ohne DMA Unterstützung sind Brenngeschwindigkeiten von 16x oder höher nicht möglich.

#### **Notiz**

IDE/ATAPI Befehle kennen kein Disconnect/Reconnect. Um optimale Performance beim on-the-fly Brennen zu erhalten, dürfen Leser und Brenner nicht an dasselbe IDE Kabel angeschlossen sein.

### 2. Einrichten des Brenners

#### 2.1 ATAPI-IDE Brenner

Auf dem Markt sind inzwischen viele unterschiedliche Typen von Brennern erhältlich. Je nach Anschluss unterscheidet man zwischen ATAPI-IDE, USB, Firewire oder echten SCSI Brennern. Auch in der Fähigkeit mit unterschiedliche Medien, wie CD-ROM, CD-RW, DVD-RW/-R, DVD+RW/+R, DVD+-RW oder DVD/CD-RW Combos macht den Markt nicht gerade übersichtlicher. Die meisten Geräte, egal ob sie CD oder DVDs brennen, sind derzeit mit einem IDE Anschluss ausgestattet.

Die folgenden Schritte dieser Anleitung beziehen sich daher auf ein korrekt angeschlossenes und gejumpertes ATAPI-IDE Gerät. Im Anschluss daran werde ich ab Punkt 2.5 die Eigenheiten von SCSI, USB und Firewire kurz anschnitten.

#### 2.2 Kernel-Optionen

Für Benutzer der 2.4. Kernel Reihe empfiehlt sich aus oben genannten Gründen die IDE-SCSI Emulation. Aber auch für die 2.6. Serie ist die Emulation keine schlechte Idee, da man so unter der

CD-Brennsoftware die größte Auswahl hat. Wenn man aber sicher ist, dass die Software ATAPI Unterstützung anbietet, kann dieser Punkt übersprungen werden. Zurück zum 2.4. Kernel. Zur Benutzung der IDE-SCSI Emulation muss eventuell der Kernel mit den entsprechenden Optionen neu kompiliert werden, falls nicht bereits erfolgt. Sollten Sie bisher noch keinen Kernel neu kompiliert haben, sollten Sie in Abschnitt 16.3. der [Installationsanleitung](#) nachschlagen, wie ein Kernel manuell erstellt wird. Als root führt man in `/usr/src/linux` ein **make menuconfig** aus. Folgende Optionen werden benötigt:

#### **Befehlsauflistung 1:** Kernel Konfiguration

```
ATA/IDE/MFM/RLI support -> IDE, ATA, and ATAPI Block Devices -> <*> SCSI Emulation Support
SCSI support -> <M> SCSI support
SCSI support -> <M> SCSI CDROM support
SCSI support -> <M> SCSI generic support
```

Es ist wichtig, dass die entsprechenden Optionen im Abschnitt "SCSI Support" als Module kompiliert werden, da es möglich ist, dass sie nicht in der eigentlich notwendigen richtigen Reihenfolge geladen werden. Dabei macht es keinen Unterschied ob sie als Module oder direkt im Kernel verwendet werden. Als Module hat man allerdings die Möglichkeit, sie wieder zu entladen und danach in der richtigen Reihenfolge zu laden. Nun sollte der Kernel mit **make dep && make clean bzImage modules modules\_install** neu kompiliert werden und nach dem Mounten der /boot Partition hinüberkopiert werden.

Nun muss der Bootloader entsprechend angepasst werden. In GRUB editiert man dazu `/boot/grub/menu.lst` und fügt die Zeile **hdX=ide-scsi** hinzu. Dabei sollte X durch die entsprechende Bezeichnung des CD-Brenners ersetzt werden. Ist z.B. der Brenner das Master-Gerät am IDE Bus 1, so lautet die Bezeichnung korrekt **hdc=ide-scsi**. Hier ein Beispiel für GRUB:

#### **Befehlsauflistung 2:** /boot/grub/menu.lst

```
title Gentoo 2.4.22
root (hd0,0)
kernel (hd0,0)/bzImage-2422 root=/dev/hda3 hdb=ide-scsi
```

Hier ist die Konfiguration für einen Brenner als Slave Gerät am IDE Bus 0. Benutzer von LILO fügen den Befehl mittels der **append** Zeile hinzu:

#### **Befehlsauflistung 3:** /etc/lilo.conf

```
# Linux bootable partition config begins
image = /boot/bzImage-2422
append= "hdb=ide-scsi"
root = /dev/hdc3
label = Gentoo
read-only # read-only for checking
```

Hier ist der Brenner Master am Bus 0. Das Abspeichern der Änderungen nicht vergessen.

## 2.3 Konfigurationsdateien

Um den Brenner wirklich sauber einzurichten, müssen noch einige Konfigurationsdateien geändert werden. Zum Beispiel sollte dem IDE-CDROM Treiber mitgeteilt werden, dass er den Brenner ignorieren darf. In `/etc/modules.d/scsi` tragen wir folgende Zeilen ein:

#### **Befehlsauflistung 4:** /etc/modules.d/scsi

```
options ide-cd ignore='hdX'
alias scsi_hostadapter ide-scsi
```

Natürlich sollte das X wieder durch die korrekte Bezeichnung des Brenners ersetzt werden, also z.B. hda, hdc ... Sollte die Datei `/etc/modules.d/scsi` nicht existieren, so kann sie ruhig neu angelegt werden. Damit unsere Änderungen auch in `/etc/modules.conf` übernommen werden, führen wir an der Kommandozeile ein **modules-update** aus.

Als nächstes ist `/etc/devfsd.conf` dran. Hier sollte sichergestellt werden, dass das entsprechende Device für den Brenner auch angelegt wird. Die Gentoo Linux Entwickler haben uns netterweise schon einen entsprechenden Eintrag angelegt, der nur noch auskommentiert werden muss. Die Bezeichnung sr0 bezieht sich auf das erste CDROM des simulierten SCSI Bus. Sollte natürlich noch ein anderes CDROM am Bus hängen und der Brenner erst an zweiter Stelle stehen, muss sr0 entsprechend geändert werden, z.B. sr1 für das zweite CDROM am SCSI Bus:

#### Befehlsauflistung 5: /etc/devfsd.conf

```
# Create /dev/cdrw for the first cdrom on the scsi bus
# (change 'sr0' to suite your setup)
LOOKUP      ^cdrw$                CFUNCTION GLOBAL mksymlink sr0 cdrw
REGISTER    ^sr0$                 CFUNCTION GLOBAL mksymlink $devname cdrw
UNREGISTER  ^sr0$                 CFUNCTION GLOBAL unlink cdrw
```

Damit nicht nur root brennen kann, erweitern wir die Schreibrechte um die Gruppe cdrw, in die wir später unser Benutzerkonto aufnehmen werden. Dies legen wir ebenfalls in **/etc/devfsd.conf** fest:

#### Befehlsauflistung 6: /etc/devfsd.conf

```
# Give the cdrw group write permissions to /dev/sg0
# This is done to have non root user use the burner (scan the scsi bus)
REGISTER    ^scsi/host.*/bus.*/target.*/lun.*/generic PERMISSIONS root.cdrw 660
```

Als letzte Datei editieren wir **/etc/modules.autoload**. Da sr\_mod beim Start schon automatisch geladen wird, genügt es, nur das Modul **sg** hinzuzufügen:

#### Befehlsauflistung 7: /etc/modules.autoload

```
# /etc/modules.autoload: kernel modules to load when system boots.
#
# Add the names of modules that you'd like to load when the system
# starts into this file, one per line. Comments begin with # and
# are ignored. Read man modules.autoload for additional details.

# The generic scsi support:
sg
```

Damit der Brenner auch als normales CDROM verwendet werden kann, und wir die Daten auch mounten können, darf ein entsprechender Eintrag in der **/etc/fstab** nicht fehlen.

#### Befehlsauflistung 8: Beispiel für /etc/fstab

```
/dev/sr0          /mnt/cdrw        auto              user,defaults,noauto,ro,exec 0 0
```

Natürlich sollte /dev/sr0 an die eigenen Gegebenheiten angepasst werden.

## 2.4 Kontrolle beim Start

Falls man den Kernel neu kompiliert hat kommt man um einen Neustart nicht herum. Beim Starten sollte man auf einige wichtige Meldungen bezüglich SCSI und dem Laufwerk achten. Als Beispiel hier meine Meldungen mit einem ASUS CD-Writer, der an /dev/hdc angeschlossen ist:

#### Befehlsauflistung 9: Ausgabe von dmesg

```
hdc: ASUS CRW-2410A, ATAPI CD/DVD-ROM drive
ide-cd: passing drive hdc to ide-scsi emulation.

...

SCSI subsystem driver Revision: 1.00
hdc: attached ide-scsi driver.
scsi0 : SCSI host adapter emulation for IDE ATAPI devices
  Vendor: ASUS      Model: CRW-2410A      Rev: 1.0
  Type:   CD-ROM    ANSI SCSI revision: 02
Attached scsi CD-ROM sr0 at scsi0, channel 0, id 0, lun 0
sr0: scsi3-mmc drive: 40x/40x writer cd/rw xa/form2 cdda tray

...

* Loading driver sg...
```

In diesem Fall wurde der Brenner erfolgreich vom IDE-CDROM Treiber ignoriert und an den IDE-SCSI Treiber übergeben, welcher den Brenner als erstes Laufwerk am SCSI Bus (sg0) eingerichtet hat. Sollte die Ausgabe zu schnell vorbei laufen, hilft ein **dmesg** an der Kommandozeile weiter.

Die folgenden Abschnitte gehen genauer auf SCSI, USB und Firewire Geräte ein. Sollte der ATAPI IDE Brenner bereits eingerichtet sein, können Sie gleich zu Punkt 3.0 weiterspringen.

## 2.5 SCSI Brenner

Um einen echten SCSI Brenner in Linux einbinden zu können, muss statt der Unterstützung für SCSI Emulation der passende Treiber für den SCSI Hostadapter geladen werden. Je nach Chipsatz wird der entsprechende SCSI Low-Level Treiber in den Kernel kompiliert. In meinem Beispiel ist dies ein Dawicontrol DC-2974 PCI SCSI Adapter, für den der AM53/79C974 Treiber die richtige Wahl ist.

Ansonsten werden die SCSI Module direkt in den Kernel eingebunden:

### Befehlsauflistung 10: Kernel Konfiguration

```
SCSI support -> <*> SCSI support
SCSI support -> <*> SCSI CDROM support
SCSI support -> <*> SCSI generic support
SCSI low level drivers --->
    <*>AM53/79C974 PCI SCSI support
```

Anschliessend muss wie bereits unter Punkt 2.3 beschrieben, **/etc/devfsd.conf** an die eigenen Bedürfnisse angepasst werden.

Nach einem Neustart sollte der Treiber problemlos geladen werden und die Daten des angeschlossenen SCSI Gerätes ausgegeben werden. In meinem Fall ist dies 0,3,0. Je nach Adapter-Terminierung und SCSI-Anschluss kann das ganz unterschiedlich aussehen.

### Befehlsauflistung 11: Ausgabe von dmesg

```
SCSI subsystem driver Revision: 1.00
PCI: Found IRQ 10 for device 00:0e.0
scsi0 : AM53/79C974 PCscsi driver rev. 0.5; host I/O address: 0x7000; irq: 10
    Vendor: TEAC      Model: CD-R55S      Rev: 1.0R
    Type:   CD-ROM    ANSI SCSI revision: 02
scsil : SCSI host adapter emulation for IDE ATAPI devices
Attached scsi CD-ROM sr0 at scsi0, channel 0, id 3, lun 0
sr0: scsi-l drive
Uniform CD-ROM driver Revision: 3.12
```

Damit wäre das Gerät vollständig eingerichtet und wir können uns in Kapitel 3 der Software-Installation widmen.

## 2.6 USB Brenner

USB Brenner werden über die SCSI Emulation im Kernel angesprochen. Entsprechend konfigurieren wir den Kernel ähnlich wie bei einem ATAPI-IDE Brenner, fügen aber noch die entsprechende USB-Unterstützung (je nach System UHCI oder OHCI) hinzu.

### Befehlsauflistung 12: Kernel Konfiguration

```
ATA/IDE/MFM/RLI support -> IDE, ATA, and ATAPI Block Devices -> <*> SCSI Emulation Support
SCSI support -> <M> SCSI support
SCSI support -> <M> SCSI CDROM support
SCSI support -> <M> SCSI generic support
USB support --->
    <M> Support for USB
    [*]  USB verbose debug messages
    [*]  Preliminary USB device filesystem
    [*]  Enforce USB bandwidth allocation (EXPERIMENTAL)
    --- USB Controllers
    <M>  UHCI (Intel PIIX4, VIA, ...) support
    < > OHCI (Compaq, iMacs, OPTi, SiS, ALi, ...) support
    Den richtigen USB Controller für Ihr System aktivieren!
    <M>  USB Mass Storage support
```

Damit die Geräte beim Einstecken des USB Kabels automatisch erkannt, und die korrekten Treiber geladen werden, benötigen wir noch das Paket **hotplug**.

### Befehlsauflistung 13: Installation von hotplug

```
# emerge -p hotplug
```

These are the packages that I would merge, in order:

```
Calculating dependencies ...done!  
[ebuild N ] sys-apps/hotplug-20030805-r2
```

```
# emerge usbd hotplug
```

Nun müssen wir den hotplug Daemon noch starten und auch in die init Skripte aufnehmen.

#### **Befehlsauflistung 14:** Einrichten von hotplug

```
# /etc/init.d/hotplug start  
* Starting USB and PCI hotplugging... [ ok ]  
# rc-update add hotplug default  
* hotplug added to runlevel default  
* Caching service dependencies... [ ok ]  
* rc-update complete.  
#
```

Nach Anpassung von **/etc/devfsd.conf**, wie in Punkt 2.3 beschrieben, stecken wir den eingeschalteten USB-Brenner ein und schauen mit **dmesg** die Statusmeldungen an:

#### **Befehlsauflistung 15:** Ausgabe von dmesg

```
usb.c: registered new driver usb-storage  
scsi0 : SCSI emulation for USB Mass Storage devices  
scsi : 1 host.  
Vendor: HP Model: CD-Writer+ 8200 Rev: 1.0f  
Type: CD-ROM ANSI SCSI revision: 02  
Detected scsi CD-ROM sr0 at scsi0, channel 0, id 0, lun 0  
sr0: scsi3-mmc drive: 8x/24x writer cd/rw xa/form2 cdda tray  
Uniform CD-ROM driver Revision: 3.11  
WARNING: USB Mass Storage data integrity not assured  
USB Mass Storage device found at 2  
usb.c: usb-storage driver claimed interface c2elf660  
USB Mass Storage support registered.
```

Bei einer ähnlichen Ausgabe wie oben, ist der Brenner korrekt erkannt worden und als **/dev/sr0** eingebunden worden. Perfekt! Nun können wir wie in Kapitel 3 beschrieben, die Software installieren.

## **3. Software Installation**

### **3.1 cdrecord**

Jörg Schillings fantastisches Programm [cdrecord](#) wird von fast allen grafischen Frontends unter KDE und GNOME verwendet. Natürlich ist es als Bestandteil des cdrtools Paketes in Portage zu haben:

#### **Befehlsauflistung 16**

```
# emerge -p cdrtools
```

These are the packages that I would merge, in order:

```
Calculating dependencies ...done!  
[ebuild N ] app-cdr/cdrtools-2.01_alpha14
```

```
# emerge cdrtools
```

Nachdem das Programm und eventuell auch noch seine Abhängigkeiten von Portage kompiliert und installiert wurde, überprüfen wir gleich, ob unser Brenner korrekt erkannt wird. Für Benutzer eines SCSI CD-ROMs oder 2.4. Kernel Nutzer, die die IDE-SCSI Emulation wie oben beschrieben durchgeführt haben, geht dies mit **cdrecord -scanbus**.

#### **Befehlsauflistung 17:** Beispiel für cdrecord

```
# cdrecord -scanbus  
Cdrecord-Clone 2.01a18 (i686-pc-linux-gnu) Copyright (C) 1995-2003 Jörg Schilling
```

```

Linux sg driver version: 3.1.25
Using libscg version 'schily-0.7'
scsibus0:
    0,0,0    0) 'ASUS      ' 'CRW-2410A    ' '1.0 ' Removable CD-ROM
    0,1,0    1) *
    0,2,0    2) *
    0,3,0    3) *
    0,4,0    4) *
    0,5,0    5) *
    0,6,0    6) *
    0,7,0    7) *

```

Mit einer ähnlichen Ausgabe wie oben, ist Ihr Brenner erkannt worden und bereit für das Brennen. Benutzer des 2.4. Kernels, die statt der IDE-SCSI Emulation direkt auf das CDROM Packet Interface zugreifen wollen, müssen den Bus explizit angeben mittels **cdrecord -dev=ATAPI -scanbus**. Benutzer der 2.6. Kernel Serie können das Gerät direkt mit **cdrecord -dev=/dev/hdc** ansprechen.

## 3.2 cdrdao

Als nächstes installieren wir **cdrdao** von Andreas Müller. Es ermöglicht unter anderem das Kopieren von Audio CDs on-the-fly, sowie der Name schon sagt, das Schreiben von CDs im DAO ("Disk-at-once") - Modus.

### Notiz

Cdrecord kann ebenfalls DAO. Die meisten grafischen Frontends verwenden aber weiterhin cdrdao zum Schreiben von Audio CDs.

#### Befehlsauflistung 18: Installation von cdrdao

```

# emerge -p cdrdao

These are the packages that I would merge, in order:

Calculating dependencies ...done!
[ebuild  N  ] app-cdr/cdrdao-1.1.7-r1

# emerge cdrdao

```

Nach der Installation prüfen wir auch noch, ob das Gerät auch korrekt erkannt wird. Sollte die Ausgabe ähnlich wie hier aussehen, dann hat alles geklappt:

#### Befehlsauflistung 19: cdrdao

```

# cdrdao scanbus
Cdrdao version 1.1.7 - (C) Andreas Mueller <andreas*daneb.de>
  SCSI interface library - (C) Joerg Schilling
  Paranoia DAE library - (C) Monty

Check http://cdrdao.sourceforge.net/drives.html#dt for current driver tables.

Using libscg version 'andreas-0.5-UNIXWARE_Patch'

0,0,0: ASUS, CRW-2410A, 1.0

```

Benutzer eines CD-Brenners haben hiermit schon alle für die Kommandozeile notwendigen Tools beisammen und können bedenkenlos den nächsten Punkt überspringen. Besitzer eines DVD Brenners brauchen dagegen noch einige weiteren Programme.

## 3.3 dvd+rw-tools

Die **DVD+RW-Tools** sind eine Sammlung von Programmen für + und - Brenner. Der Name DVD+RW-Tools wurde aus historischen Gründen beibehalten, als nur + Brenner unterstützt wurden. Wir installieren das maskierte Paket wie folgt:

#### Befehlsauflistung 20: Installation von dvd+rw-tools

```

# ACCEPT_KEYWORDS="~x86" emerge -p dvd+rw-tools

These are the packages that I would merge, in order:

```

```
Calculating dependencies ...done!  
[ebuild N ] app-cdr/dvd+rw-tools-5.13.4.7.4
```

```
# ACCEPT_KEYWORDS="~x86" emerge dvd+rw-tools
```

Im Paket ist auch **growisofs** enthalten, ein bequemes Frontend für mkisofs

### Notiz

Um nicht nur Daten DVDs lesen zu können, sollte **udf Unterstützung** in den Kernel einkompiliert sein.

## 4. Brennen auf der Kommandozeile

### 4.1 Daten CDs

Brennen auf der Kommandozeile ist einfacher als man denkt! In diesem Abschnitt werde ich exemplarisch einige Beispiele vorstellen, mit denen man schnell und einfach seine CDs brennen kann, ohne XFree gestartet haben zu müssen. Dies ist sehr praktisch, wenn man z.B. von der Arbeit aus schnell auf dem Heimrechner ein paar Dateien sichern möchte (natürlich sollte im Laufwerk am besten schon ein Rohling bereitliegen). Die Daten, die auf einen Rohling gebrannt werden, müssen zunächst in einem Dateisystem zusammengefasst werden. Diese werden mit Hilfe des Tools **mkisofs** in einem sogenanntem Image zusammengefasst, welches nach einem kurzen Test mittels **cdrecord** auf die Scheibe geschrieben werden.

**Befehlsauflistung 21:** Beispiel für mkisofs

```
$ mkisofs -r -o image.iso bildergalerie/
```

Im obigen Beispiel werden die Daten im Verzeichnis **bildergalerie/** in das Image **image.iso** geschrieben. Die Option **-r** stellt sicher, dass die Daten als Rock-Ridge-Extension gespeichert werden. Dabei werden die Daten in in das DOS-kompatible 8.3 Format umgewandelt (unter Beibehaltung der alten Dateinamen) und die Zugriffsrechte auf öffentlich gesetzt. Möchte man lieber das MS Joliet Dateisystem verwenden, benutzt man stattdessen die Option **-J**.

Als nächstes kann die Integrität des Images überprüft werden. Dazu wird die Datei einfach als eine Partition gemountet. In unserem nächsten Beispiel mounten wir das Image auf **/cdrom** und werfen einen Blick auf das Dateilayout:

**Befehlsauflistung 22:** Beispiel für das Mounten eines Loopback Devices

```
$ mount -t iso9660 -o ro,loop=/dev/loop0 image.iso /cdrom  
$ cd /cdrom  
$ ls -l
```

### Notiz

Um CD Images einsehen zu können, muss die Unterstützung für "loopback" in den Kernel einkompiliert worden sein

Nun wollen wir das Image auf CD brennen. Dazu sollten wir mit **cdrecord -scanbus** vorher herausgefunden haben, an welchem Bus unser Brenner hängt. Dies sowie die ID und LUN werden dabei als dreistellige Zahl, durch Kommata voneinander getrennt angegeben. In meinem obigen Beispiel ist dies 0,0,0. Jeder Benutzer übergibt also mittels **dev=SCSI\_BUS,SCSI\_ID,SCSI\_LUN** die Korrekte Lage des Brenners an cdrecord:

**Befehlsauflistung 23:** Beispiel für cdrecord

```
$ cdrecord -v speed=12 dev=0,0,0 -data cd_image
```

### Wichtig

Sollten der Rohling ein CD-RW Medium sein, muss die Option **blank=fast** hinzugefügt werden.

### Notiz

Mit **cdrecord dev=0,0,0 eject** kann man testen, ob man das richtige Laufwerk ausgewählt hat. Dabei sollte sich die Schublade des entsprechenden Gerätes öffnen. Auch hier bitte 0,0,0 durch die eigenen Werte ersetzen.

Bei langsameren Rechner kann die **-dummy** Option ausprobiert werden. Dabei wird der Brennvorgang simuliert, aber der Laser nicht eingeschaltet. So kann man überprüfen, ob die Übertragungsgeschwindigkeit zum Brenner ausreicht. Moderne scsi3-mmc kompatible Brenner unterstützen auch den treiberspezifischen **driveropts=burnfree** Modus. Weitere Informationen findet man in der Manpage von cdrecord, die mit **man cdrecord** aufgerufen werden kann.

## 4.2 Audio CDs

Im Gegensatz zu einer Daten CD besitzen Audio CDs kein Dateisystem. Stattdessen werden die Musikdaten direkt als 16bit Stereo 44.1 kHz auf die CD gebrannt. Dabei gibt es zwei Möglichkeiten der Datenübertragung DAO ("Disk at once") und TAO ("Track at once"). Letzteres ist recht einfach mit **cdrecord** zu erstellen, hat aber den Nachteil, dass zwischen den einzelnen Stücken kurze Clicks und Pausen auftreten. Für DAO verwendet man das Tool **cdrdao**, welches auch die Möglichkeit, des "on-the-fly" Kopieren unterstützt.

Nun wollen wir zunächst einmal das Brennen im TAO Modus versuchen. Mit dem Tool **cdda2wav** extrahieren wir einen Audio Track von einer Audio CD. In meinem Beispiel habe ich mit der Option **-t 3** das dritte Lied von der CD ausgewählt. Die Daten werden dank **-x** mit 16bit Stereo 44.1kHz Qualität in die Datei **track03.wav** geschrieben. Natürlich muss auch hier wieder **dev=SCSI\_BUS,SCSI\_ID,SCSI\_LUN** an die eigenen Gegebenheiten angepasst werden.

**Befehlsauflistung 24:** Beispiel für **cdda2wav**

```
$ cdda2wav -dev=0,0,0 -t 3 -x track03.wav
```

```
Type: ROM, Vendor 'ASUS' Model 'CRW-2410A' Revision '1.0' MMC+CDDA
724992 bytes buffer memory requested, 4 buffers, 75 sectors
Read TOC CD Text failed (probably not supported).
#Cdda2wav version 2.01a18_linux_2.4.22_i686_amd-athlon-tm--xp-1800+, real time sched., soundcard
AUDIOtrack pre-emphasis copy-permitted tracktype channels
1-12 no no audio 2
Table of Contents: total tracks:12, (total time 52:28.40)
1.( 3:18.62), 2.( 3:27.48), 3.( 4:13.65), 4.( 4:05.35), 5.( 4:47.17),
6.( 4:59.73), 7.( 4:42.32), 8.( 5:14.73), 9.( 4:22.00), 10.( 5:11.65),
11.( 3:32.70), 12.( 4:31.25)

Table of Contents: starting sectors
1.( 0), 2.( 14912), 3.( 30485), 4.( 49525), 5.( 67935),
6.( 89477), 7.( 111975), 8.( 133157), 9.( 156780), 10.( 176430),
11.( 199820), 12.( 215790), lead-out( 236140)
CDINDEX discid: aqqclv93yDV3zqBboWho5FBST0o-
CDDB discid: 0xb00c4c0c
CD-Text: not detected
CD-Extra: not detected
samplefile size will be 44782124 bytes.
recording 253.8666 seconds stereo with 16 bits @ 44100.0 Hz ->'track03'...
cdda2wav: Operation not permitted. cannot set posix realtime scheduling policy
percent_done:
100% track 3 successfully recorded
```

### Notiz

Möchte man alle Dateien aus der CD auf einmal extrahieren, erstellt **cdda2wav** mit der **-B** Option für jeden Track eine eigene Datei

Nun haben wir eine große Datei namens **track03.wav** erstellt. Benutzer eines ALSA-Soundtreibers können mit bei Bedarf mit **aplay track03.wav** probieren. Nun schreiben wir diesen Track auf einen Rohling:

**Befehlsauflistung 25:** Beispiel für TAO mit **cdrecord**

```
$ cdreocrd -v speed=12 dev=0,0,0 -audio track03.wav
```

Natürlich kann man auch mehr als einen Audio Track auf die CD schreiben. Dazu fügt man einfach alle Namen der zu schreibenden Dateien hinter die **-audio** Option an.

### Wichtig

**Cdrecord** schreibt auch \*.cdr und \*.au Daten direkt auf die CD. Allerdings können enkodierte Formate wie z.B: \*.mp3 und \*.ogg nicht ohne weiteres verwendet werden. Diese müssen zunächst in \*.wav oder \*.cdr Dateien umgewandelt werden. Dies kann zum Beispiel mit **mpg123** erfolgen.

Nachdem wir gelernt haben, wie man Daten im TAO-Modus schreiben kann, werfen wir jetzt einen Blick auf **cdrdao**, welches wir ja bereits installiert haben. Leider unterstützt **cdrdao** aufgrund veralteter **libscg** Bibliotheken bislang noch keinen direkten ATAPI Zugriff, sodass Benutzer der 2.6 Kernel Serie sich entweder mit der IDE-SCSI Emulation oder einem echten SCSI Gerät behelfen muss. Alternativ kann man natürlich versuchen, **cdrdao** gegen die neuesten Bibliotheken zu kompilieren. Weitere Informationen dazu [hier](#).

Als erstes lesen wir eine gesamte Audio CD aus und lassen von cdrdao gleich das notwendige TOC erstellen:

**Befehlsauflistung 26:** Beispiel für Datenextraktion mit cdrdao

```
$ cdrdao read-cd --driver generic-mmc --device 0,0,0 image.toc
```

```
Cdrdao version 1.1.7 - (C) Andreas Mueller <andreas*daneb.de>  
  SCSI interface library - (C) Joerg Schilling  
  Paranoia DAE library - (C) Monty
```

```
Check http://cdrdao.sourceforge.net/drives.html#dt for current driver tables.
```

```
Using libscg version 'andreas-0.5-UNIXWARE_Patch'
```

```
0,0,0: ASUS CRW-2410A Rev: 1.0  
Using driver: Generic SCSI-3/MMC - Version 2.0 (options 0x0000)
```

```
Reading toc and track data...
```

| Track   | Mode  | Flags | Start            | Length           |
|---------|-------|-------|------------------|------------------|
| 1       | AUDIO | 0     | 00:00:00( 0)     | 04:32:20( 20420) |
| 2       | AUDIO | 0     | 04:32:20( 20420) | 03:50:32( 17282) |
| 3       | AUDIO | 0     | 08:22:52( 37702) | 04:09:30( 18705) |
| 4       | AUDIO | 0     | 12:32:07( 56407) | 05:20:23( 24023) |
| 5       | AUDIO | 0     | 17:52:30( 80430) | 02:15:50( 10175) |
| 6       | AUDIO | 0     | 20:08:05( 90605) | 04:21:42( 19617) |
| 7       | AUDIO | 0     | 24:29:47(110222) | 03:17:43( 14818) |
| 8       | AUDIO | 0     | 27:47:15(125040) | 04:27:47( 20072) |
| 9       | AUDIO | 0     | 32:14:62(145112) | 03:16:15( 14715) |
| 10      | AUDIO | 0     | 35:31:02(159827) | 05:14:15( 23565) |
| 11      | AUDIO | 0     | 40:45:17(183392) | 04:17:73( 19348) |
| 12      | AUDIO | 0     | 45:03:15(202740) | 03:49:62( 17237) |
| Leadout | AUDIO | 0     | 48:53:02(219977) |                  |

```
PQ sub-channel reading (audio track) is supported, data format is BCD.
```

```
Raw P-W sub-channel reading (audio track) is supported.
```

```
Cooked R-W sub-channel reading (audio track) is supported.
```

```
Copying audio tracks 1-12: start 00:00:00, length 48:53:02 to "data.bin"...
```

```
Track 1...
```

```
Track 2...
```

```
Found pre-gap: 00:00:63
```

```
Track 3...
```

```
Found pre-gap: 00:01:00
```

```
Track 4...
```

```
Found pre-gap: 00:03:20
```

```
Track 5...
```

```
Found pre-gap: 00:03:38
```

```
Track 6...
```

```
Found pre-gap: 00:02:70
```

```
Track 7...
```

```
Found pre-gap: 00:02:27
```

```
Track 8...
```

```
Found pre-gap: 00:00:63
```

```
Track 9...
```

```
Found pre-gap: 00:03:20
```

```
Track 10...
```

```
Found pre-gap: 00:01:07
```

```
Track 11...
```

```
Found pre-gap: 00:01:35
```

```
Track 12...
```

```
Found pre-gap: 00:01:70
```

```
Found 3567 Q sub-channels with CRC errors.
```

```
Found disk catalogue number.
```

```
Reading of toc and track data finished successfully.
```

Die Option **--driver generic-mmc** wird nur benötigt, wenn der Brenner noch nicht in der Treibertabelle enthalten ist. Da cdrdao schon seit längerer Zeit nicht mehr aktualisiert wurde, ist dies bei allen neueren Geräten sicherlich der Fall.

Nun schreiben wir den Inhalt unseres TOC auf einen Rohling:

## **Befehlsauflistung 27:** Beispiel für DAO mit cdrdao

```
$ cdrdao write --driver generic-mmc --device 0,0,0 image.toc

Cdrdao version 1.1.7 - (C) Andreas Mueller <andreas*daneb.de>
  SCSI interface library - (C) Joerg Schilling
  Paranoia DAE library - (C) Monty

Check http://cdrdao.sourceforge.net/drives.html#dt for current driver tables.

Using libscg version 'andreas-0.5-UNIXWARE_Patch'

0,0,0: ASUS CRW-2410A Rev: 1.0
Using driver: Generic SCSI-3/MMC - Version 2.0 (options 0x0000)

Starting write at speed 16...
Pausing 10 seconds - hit CTRL-C to abort.
Process can be aborted with QUIT signal (usually CTRL-\\).
Turning BURN-Proof on
Executing power calibration...
Power calibration successful.
Writing track 01 (mode AUDIO/AUDIO )...
Writing track 02 (mode AUDIO/AUDIO )...
Writing track 03 (mode AUDIO/AUDIO )...
Writing track 04 (mode AUDIO/AUDIO )...
Writing track 05 (mode AUDIO/AUDIO )...
Writing track 06 (mode AUDIO/AUDIO )...
Writing track 07 (mode AUDIO/AUDIO )...
Writing track 08 (mode AUDIO/AUDIO )...
Writing track 09 (mode AUDIO/AUDIO )...
Writing track 10 (mode AUDIO/AUDIO )...
Writing track 11 (mode AUDIO/AUDIO )...
Writing track 12 (mode AUDIO/AUDIO )...
Wrote 493 of 493 MB (Buffer 100%).
Wrote 219977 blocks. Buffer fill min 84%/max 100%.
Flushing cache...
Writing finished successfully.
```

Jetzt haben Sie gelernt, wie man erfolgreich Daten und Audio CDs in Gentoo Linux auf der Kommandozeile gebrannt. Für weitere Details an den Optionen von cdrecord oder cdrdao empfehle ich die jeweiligen Manpages. Viel Spass mit Ihrem System!

## **5. Grafische Frontends zum Brennen**

### **5.1 Kurzübersicht**

Nun sollte man das CD-Brogramm seiner Wahl installieren. Für KDE-User empfiehlt sich k3b, welches inzwischen auch DVDs rippen kann. GNOME-Fans benutzen das Nautilus-Plugin nautilus-cd-burner oder xcdroast, welches dank GTK auch Drag'n drop unterstützt.

## **6. Problembehebung**

### **6.1 Fragen und Antworten**

**Meine CD-Brennsoftware erkennt mein normales CD Laufwerk nicht. Stattdessen wird der Brenner sowohl zum Lesen als auch zum Schreiben verwendet.**

Nicht alle CD-Brennersoftware machen von den cdrecord ATAPI Funktionen Gebrauch. In diesem Fall sollten Sie entweder das CDROM auch als SCSI Gerät emulieren oder ein anderes Programm verwenden.

**Wenn ich K3b als Benutzer verwende, erhalte ich immer die Meldung, dass ich keinen Zugriff auf cdrecord und cdrdao habe, obwohl ich k3b-setup als root ausgeführt habe.**

Zum Brennen erstellt k3b-setup standardmässig die Gruppe **cdrecord**. In Gentoo Linux ist diese allerdings **cdrw**. Auch sollte sichergestellt werden, dass der Benutzer dieser Gruppe zugehörig ist.

**Beim Auführen von cdrecord erhalte ich immer folgende Mitteilung:  
Befehlsauflistung 28**

```
# cdrecord -scanbus
Cdrecord 2.01a14 (i686-pc-linux-gnu) Copyright (C) 1995-2003 Jörg Schilling
cdrecord: No such file or directory. Cannot open '/dev/pg*'. Cannot open SCSI driver.
cdrecord: For possible targets try 'cdrecord -scanbus'. Make sure you are root.
cdrecord: For possible transport specifiers try 'cdrecord dev=help'.
```

Mögliche Ursache: Das Gerät wird als ATAPI-IDE vom Kernel erkannt. Versuchen Sie stattdessen **cdrecord dev=ATAPI:0,0,0 -scanbus**. Ist die IDE-SCSI emulation wirklich korrekt aufgesetzt worden?

**In meiner Anwendung wird mein Brenner 5 oder 6x angezeigt, obwohl ich nur einen einzigen ATAPI-IDE Brenner mit SCSI emulation betreibe**

Wahrscheinlich haben Sie die Option "Check All LUNS" im Kernel für den SCSI Treiber aktiviert. Dabei wird der SCSI Bus auf angeschlossene Geräte untersucht. Da das ATAPI-IDE Gerät keine Ahnung über Logical Unit Numbers hat, antwortet es jedesmal, für jede der sieben Anschlussmöglichkeiten des SCSI Bus.

## 6.2 Weiterführende Dokumentation

[Das CD Writing HOWTO](#)

[Das CD-Recordable FAQ](#)

[USB-CD Writer Howto](#)

Und natürlich die [Gentoo Foren](#) - bitte wie immer erst suchen, dann posten! Gerade zum Thema CDs Brennen gibt es zahlreiche Threads.



## >> Anleitung zur Konfiguration von DSL mit Gentoo Linux

### 1. DSL-Einrichtung

- emerge net-dialup/ppp
- Dateien dekomprimieren
- Datei 't-online' nach etc/ppp/peers kopieren.
- Datei 'net-ppp0' nach etc/init.d kopieren. Evtl. vorher ein Backup der vorhanden Datei erstellen.
- Passwort in etc/ppp/pap-secrets eintragen.
- Benutzername in /etc/ppp/peers/t-online anpassen (bei atm@t-online.de).
- Starten per '/etc/init.d/net.ppp0 start'.
- Ping irgendwohin (zum testen).
- Damit beim Starten des System Gentoo Online geht: 'rc-update add net.ppp0 default'.

[Download 1: t-online.gz](#)

[Download 2: net.ppp0.gz](#)

**Fertig.** Viel Spass...



## >> Fluxbox Desktop in Gentoo

[Bitte Kapitel auswählen] 

### 1. Vorbemerkungen

**Fluxbox** ist einer von vielen Windowmanagern für X mit seinen Wurzeln im Unix Fenstermanagement. Gegenüber seinem Konkurrenten Blackbox und Openbox besitzt er die Möglichkeit, die Titelleisten zu konfigurieren und mehrere Fenster durch Karteireiter (sogenannte "Tabs") miteinander zu Verbinden. Mit einer Scrollmaus können die Arbeitsflächen blitzschnell gewechselt werden. Auch ist er nach dem Start von X sofort einsatzbereit. Neben KDE und Gnome erfreut er sich deshalb insbesondere bei den Gentoo Benutzern grosser Beliebtheit, weshalb ich dieses Dokument geschrieben habe. Die Anwendungen, die ich dabei vorstelle, sind dabei nur eine von den vielen Möglichkeiten, die unter Gentoo Linux angeboten werden.

Dieses Dokument geht davon aus, dass Gentoo auf Ihrem Rechner installiert und die Hardware unter Xfree86 bereits eingerichtet ist. Als erstes sollte man sicherstellen, dass Portage sich auf dem neuesten Stand befindet. Dazu genügt als root ein:

#### Befehlsauflistung 1

```
# emerge rsync
```

### 2. Fluxbox

#### 2.1 Installation

Nun schauen wir doch einmal, was mit Fluxbox installiert wird:

#### Befehlsauflistung 2

```
# emerge -p fluxbox
```

```
These are the packages that I would merge, in order.
```

```
Calculating dependencies ...done!
```

```
[ebuild N ] x11-misc/commonbox-utils-0.1 to /
```

```
[ebuild N ] x11-themes/commonbox-styles-0.1-r1 to /
```

```
[ebuild N ] x11-wm/fluxbox-0.1.10 to /
```

```
Sollte X noch
```

```
nicht installiert sein, so sieht die Liste deutlich länger aus
```

```
# emerge fluxbox
```

Das Kompilieren dieser drei Pakete sollte schneller gehen, als man sich einen Kaffee kochen kann.

#### Notiz

Wer sichergehen will, dass Fluxbox ohne Unterstützung für KDE oder GNOME auskommt, sollte dies vorher mit **export USE="-kde"** bzw. **export USE="-gnome"** setzen.

#### 2.2 Starten

Mit der Installation wurde **/etc/X11/Sessions/fluxbox** angelegt, sodass Fluxbox nun aus den Auswahlmenüs von XDM, GDM oder KDM gestartet werden kann. Um sicherzugehen, dass beim Start von Fluxbox auch die Umgebungsvariablen eingelesen werden können, sollte mit

#### Befehlsauflistung 3

```
# nano -w /etc/X11/Sessions/fluxbox
```

das Skript wie folgt erweitert werden:

**Befehlsauflistung 4:** Erweitertes /etc/X11/Sessions/fluxbox Skript

```
#!/bin/sh
source /etc/profile
startup=$HOME/.xsession
/usr/bin/fluxbox
```

Nun sollte sich Fluxbox starten und mit einem Rechtsklick auf den leeren Hintergrund das Herz von Fluxbox, das eigentlich Startmenu für die einzelnen Anwendungen aufrufen lassen. Mit der mittleren Maustaste kann ein Menu zum Wechseln zwischen den einzelnen Arbeitsflächen aufgerufen werden, aber wie bereits oben beschrieben geht das deutlich schneller mit dem Mausrad.

### Notiz

Benutzer, die Fluxbox nicht mit einem Login-Manager starten wollen, sondern dies direkt über **startx** ausführen wollen, müssen ihre `~/xinitrc` entsprechend anpassen. Am einfachsten geht dies mit einem **echo "exec fluxbox" > ~/.xinitrc**.

## 2.3 Konfiguration

Bei der Installation wird mit **fluxbox-generate\_menu** ein Eintrag aller bereits in Gentoo Linux vorhandenen Anwendungen unter `/usr/share/commonbox/menu` abgelegt. Dieses sollte nun in das eigene Benutzer verzeichnis kopiert werden.

### Befehlsauflistung 5

```
# cp /usr/share/commonbox/menu /home/username/.fluxbox/
```

Mit einem Editor kann dann das Menu beliebig den eigenen Wünschen angepasst werden:

### Befehlsauflistung 6

```
# nano -w /home/username/.fluxbox/menu
```

Man sieht, dass das Prinzip sehr einfach aufgebaut ist. Mit **[exec]** zu Beginn der Zeile wird Fluxbox mitgeteilt, dass es sich um ein ausführbares Programm handelt, dessen Name im Fluxbox Menu als **(name)** erscheint. Der eigentliche Programmaufruf, wie an der Kommandozeile, wird in **{anwendung -option1 -option2}** gepackt. Hier einige Beispiele:

**Befehlsauflistung 7:** Ein Beispiel für `/usr/share/commonbox/menu`

```
[begin] (-- Gentoo Linux --)
    [exec] (xterm) {xterm -fg white -bg black}

[submenu] (Internet)
    [exec] (lynx) {xterm -title lynx -fg white -bg black -e lynx}
    [exec] (xchat) {xchat}
[end]

[submenu] (Editoren)
    [exec] (xedit) {xedit}
[end]

[submenu] (Verschiedenes)
    [exec] (xcalc) {xcalc}
    [exec] (xfonstsel) {xfonstsel}
    [exec] (xman) {xman}
    [exec] (xload) {xload}
    [exec] (xbiff) {xbiff}
    [exec] (editres) {editres}
    [exec] (visehenewres) {viewres}
    [exec] (xclock) {xclock}
    [exec] (xscreensaver-demo)
[end]

[end]

[submenu] (Einstellungen)
    [workspaces] (Arbeitsplatz-Liste)
[submenu] (Styles) {Stil aendern...}
    [stylesdir] (/usr/share/commonbox/styles)
[end]
```

```

    [config] (Konfigurieren)
    [reconfig] (Neu Laden)
[end]

    [restart] (Neu Starten)
    [exit] (Exit)
[end]

```

Die eigentlichen Einstellungen für Fluxbox können in der `~/fluxbox/init` Datei vorgenommen werden. Diese kann ebenfalls mit einem Editor bequem nach eigenen Wünschen angepasst werden:

#### Befehlsauflistung 8

```
# nano -w ~/.fluxbox/init
```

Da ich es liebe, dass geöffnete Fenster gerne ab ca. 10 Pixel Entfernung an den Bildschirmrand "schnappen" und ausserdem die Uhr im 24 Stunden Format sehen möchte, habe ich die **init** wie folgt angepasst:

**Befehlsauflistung 9:** Ein Beispiel für `~/fluxbox/init`

```

Alle anderen Optionen aus Platzgründen ausgeblendet
session.screen0.edge.SnapThreshold:    10
session.screen0.strftimeFormat:       %H:%M

```

Wem das zu umständlich ist, der kann sich mit

#### Befehlsauflistung 10

```
# emerge fluxconf
```

ein grafisches Konfigurationstool für Fluxbox aus dem Netz ziehen, welches mit **fluxconf** gestartet wird.

## 2.4 Der Fluxbox Stil

Nun kommt das grösste Problem der Fluxbox Installation - die Auswahl des geeigneten Themes. Fluxbox Benutzer verwenden unglaublich viel Zeit damit, ihren Desktop so anzupassen, dass er möglichst einzigartig aussieht und darüber hinaus sehr stylish aussieht. Dazu passen natürlich einige besondere Schriften, wie zum Beispiel der [Artwiz-Font](#). Das Mass aller Dinge ist dann natürlich, der mit allem Stolz irgendwo im Web gepostete Screenshot, auf dem der eigene Bildschirm mit sovielen möglichen Applikationen zugesperrt wird um einen guten Eindruck zu erwecken, dass der Benutzer (a) entweder ganz gut beschäftigt ist, um alles noch unter Kontrolle zu haben, oder (b) mal wieder die grandiose Multitasking-überlegenheit von Linux zu demonstrieren. Allem Anschein nach sind transparente und rahmenlose Terminals der letzte Schrei. Anregungen aus den offiziellen Gentoo Foren kann man sich [hier](#) jedenfalls mehr als genug holen, die richtige Wahl zu treffen, ist da schon einiges schwerer.

Man sollte allerdings noch den Unterschied zwischen den Themen, die ein Hintergrundbild einsetzen, kennen. Um in bestimmten Anwendungen, wie zum Beispiel **xhcat** oder **aterm** Transparenz hinzubekommen, bedienen sich diese Anwendungen einem relativ einfachen Trick, indem sie den entsprechenden Ausschnitt aus dem Hintergrund kopieren. Das geht aber natürlich nur bei den Themen, bei denen auch ein Hintergrundbild vorhanden ist. Das ist leider bei den Fluxbox Mustern nicht der Fall. Diese sehen zwar auch sehr schön aus, da sie aber bei Bedarf gerendert werden, haben die Anwendungen keine Möglichkeit, darauf zurück zu greifen.

#### Notiz

Bis zur Fluxbox Version 0.1.9 müssen die Applikationen zum Setzen der Bildhintergründe mit den jeweiligen anderen Paketen installiert werden, so zum Beispiel **emerge Eterm** für **Esetroot**. Das ist nun nicht mehr notwendig. Das paket **commonbox-utils** enthält bereits **Esetroot**, **bsetbg** und einige anderen nützlichen Tools.

Mit der Installation von Fluxbox sollten standardmässig schon einige Themen in `/usr/share/commonbox/styles` abgelegt worden sein. Eine grosse Auswahl gibt es bei [Freshmeat](#); um diese jedem Benutzer (auch von Blackbox, Openbox oder Hackedbox zur Verfügung zu stellen), empfiehlt sich eine Installation in `/usr/share/commonbox/styles` (die Hintergrundbilder in: `/usr/share/commonbox/backgrounds`). Mit dem Anlegen von symbolischen Links

#### Befehlsauflistung 11

```
# ln -s /usr/share/commonbox/styles /home/username/.fluxbox/styles
# ln -s /usr/share/commonbox/backgrounds /home/username/.fluxbox/backgrounds
```

kann nun der Benutzer auf alle installierten Themen zugreifen.

### **Wichtig**

Werden externe Themes installiert, die noch nicht im Portage Tree vorhanden sind, so sollte der Pfad in der jeweiligen **styles** Datei den Gentoo Verhältnissen angepasst werden, also: `/usr/share/commonbox/backgrounds/Bildname.jpg`

## **2.5 Setzen des Hintergrundes**

Wem die gerenderten Hintergründe von Fluxbox nicht zusagen, oder ein einfach spezielles Hintergrundbild gewünscht wird, der kann dies mit:

### **Befehlsauflistung 12**

```
# bsetbg -f /usr/share/commonbox/backkgrounds/meinbild.jpg
```

setzen. Die `-f` Option passt das Bild an die Bilschirmauflösung an. Für gekachelte Bilder bietet sich stattdessen `-t` an. Wer Transparenten Hintergrund in bestimmten Anwednungen, wie zum Beispiel **xchat** oder **Eterm** sehen möchte, sollte den Hintergrund besser mit Esetroot setzen:

### **Befehlsauflistung 13**

```
# Esetroot -f /usr/share/commonbox/backkgrounds/meinbild.jpg
```

### **Notiz**

Natürlich kann der Hintergrund über `~/.xinitrc` bzw. `~/.xsession` bei jedem Start aufgerufen werden, es empfiehlt sich aber, wie bereits oben beschrieben, die Anpassung der entsprechenden Theme-Datei in `/usr/share/commonbox/styles`.

## **3. Dateimanagement mit dem ROX-Filer**

### **3.1 Wozu ein Dateimanager?**

Was für Gnome Nautilus und für KDE Konqueror ist, da kommt Fluxbox zunächst ohne Dateimanager daher. Den Puristen unter uns genügt sowieso die Kommandozeile im Terminal, aber wenn es darum geht, bestimmte Icons auf dem Desktop abzulegen wollen, ein Panel a la Gnome zu haben, oder mal schnell in einem Verzeichnis zu blättern kommt man um ein grafisches Dateimanagement kaum herum. Auswahl gibt es hier wie Sand am Meer, **endeavour**, **gentoo** um nur einige zu nennen. Ich persönlich bevorzuge den ROX-Filer, aber das ist Ansichtssache.

### **Befehlsauflistung 14**

```
# emerge -p rox

These are the packages that I would merge, in order.

Calculating dependencies ...done!
[ebuild N ] x11-misc/shared-mime-info-0.7 to /
[ebuild N ] app-misc/rox-1.3.2 to /

# emerge rox
```

Nun sollte sich dieser mit **rox** aufrufen lassen.

### **3.2 Konfiguration**

Um mit dem ROX Filer Icons auf dem Panel abzulegen können, muss dieser vor dem Start von Fluxbox ausgeführt werden. Damit jeder Benutzer in den Genuss kommt, habe ich die `/etc/X11/Sessions/fluxbox` wie folgt erweitert:

**Befehlsauflistung 15:** Erweitertes `/etc/X11/Sessions/fluxbox` Skript

```
#!/bin/sh
source /etc/profile
rox --bottom=mypanel -o &
/usr/bin/fluxbox
```

Die **-o** Option bewirkt, dass das Panel auf allen Arbeitsflächen angezeigt wird, ansonsten sind die Icons nur auf dem ersten Arbeitsplatz, der beim Start sichtbar wird, zu benutzen. Beim nächsten Neustart von Fluxbox sollte nun unten ein Panel und darin ein Ordner mit dem Namen "Desktop" zu sehen sein. Mit einem Linksklick darauf, öffnet sich der Dateimanager, und die Lieblingsanwendungen können per Drag'n drop auf dem Panel abgelegt werden. Mit Rechtsklick darauf kann ein Kontextmenü geöffnet werden, das benutzerdefinierte Einstellungen (anderes Icon, neuer Name etc...) erlaubt. Noch schneller geht es, wenn man die Einstellungen direkt unter `~/Choices/ROX-Filer/pan_mypanel` vornimmt.

**Befehlsauflistung 16:** Desktop Icons in `~/Choices/ROX-Filer/pan_mypanel`

```
<?xml version="1.0"?>
<panel>
<start/>
<end>
<icon label="xchat">/usr/bin/xchat</icon>
<icon label="xterm">/usr/bin/xterm</icon>
<icon label="Home">/home/username</icon>
</end>
</panel>
```

Im Dateimanager selber kann nun festgelegt werden, welche Anwendung sich beim Klicken auf die Datei öffnet. Für Textdateien bietet sich **xedit** an:

**Befehlsauflistung 17**

```
# nano -w ~/Choices/MIME-types/text-plain
```

und trägt folgende Werte ein:

**Befehlsauflistung 18:** Modifikationen an `~/Choices/MIME-types/text-plain`

```
#!/bin/sh
exec xedit "$1"
```

Dies lässt sich beliebig für alle MIME-Dateitypen vornehmen, die unter `~/Choices/MIME-Info/Standard` eingetragen sind. Weitere MIME-Info Vorlagen befinden sich unter `/usr/share/mime-info`.

## 4. Bildschirmschoner

### 4.1 Konfiguration von xscreensaver

Obwohl bei den heutigen Monitoren die Gefahr des Einbrennens von Bildern schon lange nicht mehr gegeben ist, lohnt sich die Installation von **xscreensaver**, da mit einigen Powermanagement-Optionen aufwarten kann und darüberhinaus viele schöne Motive, wie z.B. den schon lang vermissten "Blue Screen of Death" bereitstellt. Die Installation ist denkbar einfach:

**Befehlsauflistung 19**

```
# emerge -p xscreensaver
```

Das Programm besteht aus zwei Teilen, dem **xscreensaver** Daemon und einer Applikation, die den Zugriff für alle Einstellungen dafür bereitstellt. Damit der Daemon bei jedem Start von Fluxbox geladen wird, legen wir die **.xsession** Datei wie folgt an:

**Befehlsauflistung 20:** Beispiel für `~/xsession`

```
exec xscreensaver &
```

Dann setzen wir noch die korrekten Werte, damit das Skript auch ausgeführt werden kann:

**Befehlsauflistung 21**

```
# chmod +744 .xsession
```

Mit **xscreensaver-demo** passen wir den Bildschirmschoner ganz unseren Wünschen an. Empfehlenswert ist es, unter "Advanced" das Häkchen vor "Display Splash Screen on Startup" zu entfernen, um einen schnelleren Start zu gewährleisten. Natürlich können alle Einstellungen auch direkt in der `~/xscreensaver` Konfigurationsdatei vorgenommen werden.

## 5. "Dockapps" und der mysteriöse "Slit"

### 5.1 Einführung

"Dockapps" sind kleine Anwendungen, deren Aufgabe darin besteht, wichtige Systeminformationen oder allerhand unnützes Zeug dem Benutzer auf dem Bildschirm zu präsentieren. Damit alle diese Anwendungen nicht kreuz und quer auf dem Bildschirm verteilt sind und überdies auch noch ein eigenes Fenster geöffnet haben, gibt es den sogenannten "Slit", eine Art Menüleiste, die alle diese Programme aufnehmen kann und je nach Benutzereinstellung diese immer im Vordergrund oder weniger aufdringlich im Hintergrund auf allen Bildschirmen anzeigt. Da es Dockapps wie Sand am Meer gibt, werde ich hier nur einige wenige exemplarisch vorstellen - ansonsten kann ich nur empfehlen im [Dockapp Warehouse](#) oder [WindowMaker](#) zu stöbern.

### 5.2 Gkrellm

Gkrellm ist so ziemlich das "Schweizer Taschenmesser" unter den ganzen Dockapps. Von Haus aus kann er schon ziemlich viel, zählt man die Plugins dazu dann findet man sicher nichts was es nicht gibt. Mit den entsprechenden Skins passt sich Gkrellm hervorragend in die Fluxbox-Umgebung ein.

#### Befehlsauflistung 22

```
# emerge -p gkrellm-plugins

These are the packages that I would merge, in order.

Calculating dependencies ...done!
[ebuild R ] app-admin/gkrellm-1.2.13 to /
[ebuild N ] x11-misc/gkrellm-mailwatch-0.7.2 to /
[ebuild N ] x11-misc/gkrellmwireless-0.2.2 to /
[ebuild N ] x11-misc/gkrellmmouse-0.0.2 to /
[ebuild N ] x11-misc/gkrellm-gnome-0.1 to /
[ebuild N ] x11-misc/gkrellm-reminder-0.3.5 to /
[ebuild N ] x11-misc/gkrellm-volume-0.8-r1 to /
[ebuild N ] x11-misc/gkrellmwho-0.4 to /
[ebuild N ] x11-misc/gkrellmitime-0.4 to /
[ebuild N ] x11-misc/gkrellm-radio-0.3.3 to /
[ebuild N ] x11-misc/gkrellshoot-0.3.11 to /
[ebuild N ] x11-misc/gkrellmoon-0.3 to /
[ebuild N ] x11-misc/gkrellm-plugins-1.2.11-r1 to /

# emerge gkrellm-plugins
```

Ich persönlich benutze gkrellm nur, um meine POP3 Mailboxen zu überwachen und meine Laufwerke per Mausclick zu mounten. Man sollte sich aber bewusst machen, dass die Mailpasswörter im Klartext in der Datei `~/gkrellm/user_config` gespeichert werden. Nun stellen wir noch sicher, dass Gkrellm bei jedem Start automatisch im Slit landet:

#### Befehlsauflistung 23: Beispiel für `~/xsession`

```
exec xscreensaver &
exec gkrellm -w &
```

### 5.3 wmcpload, wmmemload und wmmemmon

Anzeigen über die Prozessor, Arbeitsspeicher- und Swapauslastung sind immer praktisch. Kommen Sie zusätzlich noch im schicken Digital-Look daher und lassen sich in ihrer farbe anpassen, dann haben sie ihren Platz auf meinem Desktop redlich verdient.

#### Befehlsauflistung 24

```
# emerge -p WMCPULoad wmmemmon wmmemload
```

These are the packages that I would merge, in order.

```
Calculating dependencies ...done!  
[ebuild N ] x11-misc/WMCPULoad-0.8.1 to /  
[ebuild N ] x11-misc/wmmemload-0.1.4 to /  
[ebuild N ] x11-misc/wmmemmon-0.7.0 to /
```

```
# emerge WMCPULoad wmmemmon wmmemload  
emerge
```

### Wichtig

Zur Zeit ist nur wmcpload im Portage Verzeichnisbaum. Um wmmemmon und wmmemload bequem installieren zu können, gibt es die ebuilds [hier](#).

Nun passen wir das Startup-Skript an:

**Befehlsauflistung 25:** Beispiel für ~/.xsession

```
exec xscreensaver &  
exec wmmemload -b -lc rgb:9E/0/0 &  
exec wmmemmon -b -lc rgb:9E/0/0 &  
exec wmcpload -a 90 -lc rgb:9E/0/0 &  
exec gkrellm -w &
```

Die Option -b kümmert sich darum, dass die Programme im Slit laufen. -a 90 sorgt dafür, dass ab 90% Prozessorauslastung eine Warnung in der Farbe -lc rgb.9E/0/0 (hier also ein dunkles Rot) ausgegeben wird.

## 5.4 bbrun

Mit bbrun besitzt der Benutzer eine kleine Kommandozeile. Das ist praktisch, da man ansonsten für auszuführende Programme ein Terminal öffnen muss.

**Befehlsauflistung 26**

```
# emerge -p bbrun
```

These are the packages that I would merge, in order.

```
Calculating dependencies ...done!  
[ebuild N ] x11-misc/bbrun-1.3 to /
```

```
# emerge bbrun
```

Nun noch **bbrun &** in die .xsession eintragen und mit einem Klick auf das kleine Zahnrad kann die Befehlszeile aufgerufen werden.

## 5.5 bbpager

Mit bpager hat der Benutzer eine kleine Fensterübersicht und kann auch einzelne Fenster zwischen den Bildschirmen verschieben.

**Befehlsauflistung 27**

```
# emerge -p bbpager
```

These are the packages that I would merge, in order.

```
Calculating dependencies ...done!  
[ebuild N ] x11-misc/bbpager-0.3.0-r3 to /
```

```
# emerge bbpager
```

Nun noch **bbpager &** in die .xsession eintragen und es startet automatisch.

### Warnung

Leider gibt es Probleme, wenn man bbpager im Slit laufen möchte. Ich rate zur Zeit davon ab.



# >> Gentoo Linux Gnome Upgrade HowTo und FAQ

[Bitte Kapitel auswählen] ▣

## 1. Grundlegendes

### 1.1 Gnome 2.4

11 neue Programme und über 100 Verbesserungen, die auf Benutzer zurückgehen, haben den Weg in Gnome 2.4 gefunden. Mit Epiphany hat Gnome wieder einen eigenen (auf Mozillas Gecko Engine aufbauenden) Browser, der Dateimanager Nautilus wurde überarbeitet und die Tools zur Barrierefreiheit erweitert. Soviel nur als kurzer Vorgeschmack, eine umfassende Übersicht über die wichtigsten Änderungen von Gnome 2.4 finden Sie bei [gnome-de.org](http://gnome-de.org) und bei [gnome.org](http://gnome.org); einen kurzen Überblick verschafft Ihnen ein Blick in die [Release Notes](#).

## 2. Vor dem Upgrade

### 2.1 Portage aktualisieren

... sollten Sie ihr Portage auf den aktuellen Stand bringen.

**Befehlsauflistung 1:** Portage aktualisieren

```
# emerge sync
```

Als nächstes schauen wir uns an, welche Pakete Gnome upgraden möchte.

**Befehlsauflistung 2:** Blockende Abhängigkeiten

```
# emerge -p gnome
```

Wie Sie sehen, blockt das Paket *bonobo-activation* das Upgrade.

## 3. Das Upgrade

### 3.1 Gnome aktualisieren.

Das Problem bezüglich bonobo-activation händeln Sie am besten wie folgt: Aktualisieren Sie Ihr System mit *emerge -u gnome* so weit, bis das Upgrade an bonobo-activation abbricht. Entfernen Sie nun *emerge -C bonobo-activation* und daran anschliessend wieder *emerge -u gnome* um das Upgrade fortzuführen.

Sie können eventuell das gleiche Problem bei gnome-utils und gucharmap durchlaufen, führen Sie an dem Punkt an dem das Upgrade abbricht ein *emerge -u gnome-utils* aus.

## 4. FAQ

### 4.1 Bisher bekannte Probleme und deren Lösungen

#### Ich nutze Opera als Browser, jetzt will Gnome auch Mozilla mergen. Was soll das?

Der neue Gnome eigene Browser Epiphany benötigt die Gecko Engine des Mozilla. Durch ein Anpassen der USE-Flags können Sie Mozilla ohne die IRC-, Composer- und Mail/Newskomponenten emergent.

#### Notiz

Epiphany benötigt einen mit gtk2 kompilierten Mozilla.

**Befehlsauflistung 3:** Mozilla ohne optionale Komponenten emergent

```
USE="moznoirc moznomail moznocompose gtk2"
```

Wenn Sie Mozilla und Epiphany auf keinen Fall installieren möchten, können Sie dem Portage vorgaukeln, dass Epiphany installiert wäre.

**Befehlsauflistung 4:** Epiphany injecten

```
# emerge --inject net-www/epiphany-1.0.1
```

### **Wenn ich mich anmelde, dann bleibt der Gnome Splash-Screen solange stehen, bis ich einmal draufgeklickt habe.**

Eine Lösung ist momentan noch nicht bekannt, bei einigen Gentooists lässt sich der Splash-Screen mit einem Mausklick in die Wüste jagen, bei anderen nicht.

### **Nach dem Upgrade auf Gnome 2.4 funktioniert Evolution nicht mehr. Was kann ich tun?**

Führen Sie ein *revdep-rebuild* durch. *revdep-rebuild* prüft, ob einem Programm Abhängigkeiten fehlen und *emerged* diese bei Bedarf.

### **Was ist mit den Gnome 2.4 ebuids von breakmygentoo.net?**

Die können Sie auch benutzen. Wichtig ist dabei, unbedingt die Anleitung auf der Seite zu beachten und nicht zu vergessen, das USE-Flag 'breakme' zu setzen.

**Befehlsauflistung 5:** zusätzliches USE-Flag für Gnome 2.4 ebuids von breakmygentoo.net

```
USE="breakme"
```

## **5. Lobhudelei**

### **5.1 Dank an:**

ralph für den *revdep-rebuild* Tipp.

ian! aka. Christian Hartmann für seinen XML-Editor, manches hakt noch, aber das Teil wird immer besser!

### **5.2 Sie haben einen Fehler gefunden, oder Vorschläge für Ergänzungen?**

Egal ob Rechtschreibfehler oder eine inhaltliche Korrektur oder Ergänzung, her damit!

[tobias@scherbaum.info](mailto:tobias@scherbaum.info)

### **5.3 Interessante Threads in den Gentoo Foren:**

[1] <http://forums.gentoo.org/viewtopic.php?t=84851>

[2] <http://forums.gentoo.org/viewtopic.php?t=83880>

[3] <http://forums.gentoo.org/viewtopic.php?t=83456>



## >> Anleitung für Hardware 3D-Beschleunigung

[Bitte Kapitel auswählen]

### 1. Einführung

#### 1.1 Was ist 3D Hardware-Beschleunigung und wofür brauche ich es?

Mit 3D Hardware-Beschleunigung wird dreidimensionales Rendern vom Prozessor der Grafikkarte übernommen, anstatt wertvolle Rechenleistung der CPU für das Darstellen von 3D Bildern abzuzweigen. Dies wird auch als "Hardware-Beschleunigung" bezeichnet, im Gegensatz zur "Software Beschleunigung", wo die CPU mit Hilfe der Mesa Software Bibliotheken das Zeichnen übernimmt. Während XFree in der Regel 2D Hardware-Beschleunigung unterstützt, gibt es bei der 3D-Beschleunigung einige Lücken. Spiele, 3D-CAD und Mdoellierungen kommen in der Regel nicht ohne 3D Hardware-Beschleunigung aus.

#### 1.2 Wie erhalte ich 3D Hardware-Beschleunigung?

In den meisten Fällen existieren sowohl binäre als auch Open-Source Treiber. Letzteresind zu bevorzugen, da wir nun einmal Linux verwenden und Open Source eines unserer Hauptprinzipien ist. In machen Fällen sind binäre Treiber oft die einzige Möglichkeit, wie zum Beispiel mit den nVidia Karten. Weitere binäre Treiber sind media-video/mgavideo für Matrox und media-video/ati-drivers für ATI Karten. Weitere Open-Source Projekte sind media-video/kyro-kernel für KyroII Karten und media-video/ati-gatos mit speziellen Augenmerk auf die Video-Möglichkeiten der ATI Grafikkarten.

#### 1.3 Was ist DRI?

Die Direct Rendering Infrastructure ([dri.sourceforge.net](http://dri.sourceforge.net)), besser bekannt unter dem Kürzel DRI ist eine effiziente Schnittstelle, die direkten und dennoch sicheren Zugang zur Grafikhardware ermöglicht. Sie beinhaltet Änderungen am X Server, verschiedenen Client-Bibliotheken und dem Kernel selbst. Hauptverwendungszweck für DRI sind schnelle OpenGL Implementierungen.

#### 1.4 Was ist XFree-DRM und was verbindet es mit dem normalen XFree86?

XFree-DRM ist eine **Erweiterung** zu XFree86, die 3D Hardware-Beschleunigung bereitstellt, die von XFree86 nicht unterstützt werden.

#### 1.5 Zweck dieser Dokumentation?

Diese Dokumentation richtet sich an alle, die Direct Rendering in XFree zum Laufen bekommen wollen. XFree-DRM funktioniert für 3dfx, gamma, i8x0, matrox, rage128, radeon und sis Treiber. Die laufende Entwicklung für mach64 ist bereits im CVS Tree vorhanden -- weitere Infos [hier](#) und im [HOWTO](#). Benutzer einer 2.4. Kernel Reihe benötigen das xfree-drm Paket, da der Direct Rendering Manager XFree 4.3 nicht unterstützt. Dagegen können 2.6. Kernel Nutzer dieses Dokument überspringen, da XFree 4.3 bereits unterstützt wird. Weitere Informationen erhält man auf der [DRI Homepage](#). Weitere Vorschläge, Fragen oder Mails können an Donnie Berkholz gerichtet werden.

## 2. Installation von XFree86 und Kernel Konfiguration

### 2.1 Installation von XFree86

**Befehlsauflistung 1:** Installation von XFree86

```
# emerge x11-base/xfree
```

### 2.2 Kernel Konfiguration

Zunächst sollte der Chipsatz sondiert werden und die Unterstützung für diesen aktiviert werden

**Befehlsauflistung 2:** Prüfen des AGP Chipsatzes

```
[# emerge pciutils; lspci | grep AGP  
# 00:01.0 PCI bridge: Intel Corp. 440BX/ZX/DX - 82443BX/ZX/DX AGP bridge (rev 03)  
// Je nach entsprechender Hardware kann dies anders aussehen.
```

Dies sollte mit allen Kernen funktionieren. In diesem Fall erfolgte die Konfiguration unter gentoo-sources-3.4.20-r5.

### Befehlsauflistung 3

```
[# ls -l /usr/src/linux
lrwxrwxrwx  1 root  root           22 May 29 18:20 /usr/src/linux -> linux-2.4.20-gentoo-r5
// Stellen Sie sicher, dass /usr/src/linux auf Ihren aktuellen Kernel zeigt!
# cd /usr/src/linux
# make menuconfig
```

### Befehlsauflistung 4: Optionen in make menuconfig

```
Processor type and features ---<
 >*< MTRR (Memory Type Range Register) support
Character devices ---<
 >*< /dev/agpgart (AGP Support)
 [*] Intel 440LX/BX/GX and I815/I820/I830M/I830MP/I840/I845/I850/I860 support
// Natürlich sollten Sie die Auswahl an Ihren Chipsatz anpassen!
 [ ] Direct Rendering Manager (XFree86 DRI support)
```

Stellen Sie sicher, dass die Unterstützung für DRM **abgeschaltet** ist. Das XFree-DRM Paket stellt seine eigene Kernel-Unterstützung bereit. Die Kernel-Version ist für XFree 4.2 gedacht.

## 2.3 Den Kernel kompilieren und installieren

### Befehlsauflistung 5

```
# make dep && make clean bzImage modules modules_install
# mount /boot
# cp arch/i386/boot/bzImage /boot
```

Wenn der Kernel einen anderen Namen erhalten soll als bzImage, sollten Sie stattdessen nach /boot/ neuer\_name kopieren. Nicht vergessen grub.conf oder lilo.conf entsprechend anzupassen, und als LILO-Benutzer /sbin/lilo ausführen.

## 3. Installation von XFree-DRM und Konfiguration von Direct Rendering

### 3.1 Installation von XFree-DRM

#### Befehlsauflistung 6

```
# ACCEPT_KEYWORDS="~x86" emerge xfree-drm
```

### 3.2 Konfiguration von XF86Config

Öffnen Sie /etc/X11/XF86Config mit Ihrem Lieblingseditor und fügen Sie die Funktionen für DRI und GLX hinzu.

#### Befehlsauflistung 7: XF86Config

```
...
Section "Module"
    Load "dri"
    Load "glx"
    ...
EndSection
...
Section "Device"
    Driver "radeon"
    ...
EndSection
...
Section "DRI"
    Mode 0666
EndSection
```

Sollten Sie einen anderen Treiber verwenden, müssen Sie radeon durch den entsprechenden Treiber ersetzen

## 4. 3D Beschleunigung testen

### 4.1 Neustart mit dem neuen Kernel

Nachdem Sie ihren Computer neu gestartet haben, wollen wir schauen, ob Direct Rendering aktiviert wurde und wie gut es funktioniert.

#### Befehlsauflistung 8: Rendering testen

```
# startx
// Es ist nicht notwendig den Treiber oder APGART zu laden, auch wenn sie als Modul kompiliert w
//Sie werden automatisch geladen
# glxinfo | grep rendering
direct rendering: Yes
// Kommt hier "No", dann ist die Installation von Direct Rendering fehlgeschlagen
# glxgears
// Prüft die Framerate pro Sekunde (FPS) im Standardfenster. Dies sollte dann durchgeführt werde
```

## 5. Verwenden der CVS Sourcen

### 5.1 Wird CVS benötigt?

Probieren Sie zuerst das xfree-drm Paket aus. Sollte es nicht funktionieren, und Sie haben eine sehr neue Grafikkarte, dann sind die CVS Sourcen ein Versuch wert. Das xfree-drm Paket unterstützt auch die Radeon 9000 Serie.

### 5.2 Wird Ihre Karte von den CVS Sourcen unterstützt?

Schauen Sie, ob Ihre Karte auf der [Liste](#) der ununterstützten Karten steht. Sollte Ihre Karte nicht dabei sein, aber eine ähnliche Karte darunter sein, ist es ein Versuch wert.

#### Warnung

Linux 2.4. unterstützt derzeit kein AGP 8x. Versuchen Sie im BIOS auf AGP 4x herunterzuschalten. Sollte das nicht möglich sein, müssen Sie einen [Patch](#) auf den Vanilla Kernel anwenden.

### 5.3 Installation der CVS Sourcen

Folgen Sie der Anleitung unter "Kompilieren und Installation des Kernels". Dann springen Sie zu Punkt 6 dieser Anleitung und folgen Sie bis zu Punkt 8.3.

### 5.4 Herunterladen der CVS Sourcen

Legen Sie ein Verzeichnis für die CVS Dateien an:

#### Befehlsauflistung 9

```
# cd ~
# mkdir DRI-CVS
```

Nun werden die CVS Sourcen ausgecheckt:

#### Befehlsauflistung 10

```
# cd ~/DRI-CVS
# cvs -d:pserver:anonymous@cvs.dri.sourceforge.net:/cvsroot/dri login
// ( ENTER drücken, wenn ein Passwort verlangt wird)
# cvs -z3 -d:pserver:anonymous@cvs.dri.sourceforge.net:/cvsroot/dri co xc
// Mit der -z3 werden die Daten komprimiert, was eine kürzere Downloadzeit zur Folge hat.
```

### 5.5 Aktualisierung der CVS Sourcen

Um in Zukunft ab und zu den DRI Quellcode mit den neuesten Änderungen zu erhalten, muss ein Update ausgeführt werden:

#### Befehlsauflistung 11

```
# cd ~/DRI-CVS
# cvs -z3 update -dA xc
```

## 5.6 Erstellen eines build Verzeichnisbaumes

Anstatt Objektdateien und Bibliotheken direkt in den Quellcode-Verzeichnisbaum zu legen, können diese in einen parallelen **build** Verzeichnisbaum abgelegt werden. Dieser wird mit dem **Indir** Befehl erzeugt.

### Befehlsauflistung 12

```
# cd ~/DRI-CVS
# ln -s xc XFree40
# mkdir build; cd build
# Indir -silent -ignorelinks ../XFree40
```

Dadurch werden im build Verzeichnisbaum symbolische Links angelegt, welche auf den CVS Verzeichnisbaum verweisen. Fortgeschrittene Benutzer haben oft verschiedene build Verzeichnisbäume zum kompilieren und testen der verschiedenen Optionen.

## 5.7 Anpassung der host.def Datei

Die Datei ~/DRI-CVS/build/xc/config/cf/host.def dient der Steuerung der XFree86 Kompilierung. Diese kann natürlich den eigenen Wünschen oder Systemkonfigurationen angepasst werden. Standardmässig sieht die Datei etwa so aus:

### Befehlsauflistung 13: ~/DRI-CVS/build/xc/config/cf/host.def

```
        #define DefaultCCOptions -Wall
// Für i386:
        #define DefaultGcc2i386Opt -O2
// Für Alpha:
        #define DefaultGcc2AxpOpt -O2 -mcpu=ev6 (or similar)
// Für alle anderen Architekturen
        #define LibraryCDebugFlags -O2
        #define BuildServersOnly YES
        #define XF86CardDrivers vga tdfx mga ati i810
        #define LinuxDistribution LinuxRedHat
        #define DefaultCCOptions -ansi GccWarningOptions -pipe
        #define BuildXF86DRI YES
        /* Optionally turn these on for debugging */
        /* #define GlxBuiltInTdfx YES */
        /* #define GlxBuiltInMga YES */
        /* #define GlxBuiltInR128 YES */
        /* #define GlxBuiltInRadeon YES */
        /* #define DoLoadableServer NO */
        #define SharedLibFont NO
```

Mit der Variable ProjectRoot kann die Installation der XFree86 Dateien angepasst werden. Wir empfehlen die Installation der DRI Dateien in die bestehende XFree86 Installation - dies ist in der regel sicherer und weniger störanfällig, auch wenn wir in anderen Richtlinien nicht so verfahren.

Sollte XFree86 4.x nicht in /usr/X11R6/ installiert sein, muss folgende Zeile zur host.def hinzugefügt werden:

### Befehlsauflistung 14

```
#define ProjectRoot pathToYourXFree86installation
// Note the XF86CardDrivers line to be sure your card's driver is listed.
// Zur Verwendung von 3DNow! Optimierungen in Mesa und DRI Treibern:
#define MesaUse3DNow YES
// Für diese Option ist kein AMD Prozessor erforderlich.
// Der DRI Treiber wird beim Start nach 3DNow! schauen, und bei Bedarf anwenden.
```

Für die Verwendung von SSE Optimierungen in Mesa und DRI muss mindestens ein 2.4.x Kernel vorhanden sein. Mesa überprüft selbstständig, ob SSE sowohl vom Prozessor als auch vom Betriebssystem unterstützt wird. Um Mesa in den DRI Treibern kompilieren zu können, müssen die 2.4.x Kernel Header in /usr/src/linux befinden. Sollten Sie SSE für eine ältere Kernelversion in /usr/src/linux aktivieren, wird die Kompilation von Mesa fehlschlagen. Sagen Sie nicht, wir hätten Sie gewarnt! Als 2.4.x Kernel Benutzer können Sie folgendes hinzufügen:

## Befehlsauflistung 15

```
#define MesaUseKatmai YES
```

## 5.8 Kompilieren des XFree86/DRI Verzeichnisbaumes

Zum Kompilieren genügt folgendes:

### Befehlsauflistung 16

```
# cd ~/DRI-CVS/build/xc/  
# make World >& world.log
```

Manchmal scheint es notwendig zu sein, zusätzlich folgendes auszuführen:

### Befehlsauflistung 17

```
# cd ~/DRI-CVS/build/xc/programs/Xserver/hw/xfree86/os-support/linux/drm/kernel  
# make -f Makefile.linux radeon.o  
// Ersetzen Sie radeon.o durch Ihren eigenen Treiber
```

Es ist normal, dass während der Kompilierung Warnungen auftreten. Da dies eine Weile dauern wird, empfehlen wir solange das Lesen Ihrer Mails oder ein Besuch bei heise.

### Warnung

Die make Option -j sollte nicht verwendet werden. Dies funktioniert nicht unter XFree86/DRI

Mit Ihrem Text Editor können Sie die Log-Datei nach Fehlern durchsuchen, die in der Regel mit \* \* \* beginnen.

## 5.9 Installation des Treibers

Überprüfen Sie, ob die DRI Kernelmodule erfolgreich erstellt wurden:

### Befehlsauflistung 18

```
# cd ~/DRI-CVS/build/xc/programs/Xserver/hw/xfree86/os-support/linux/drm/kernel; ls
```

Für einen 3dfx Voodoo Treiber sollten Sie tdfx.o sehen. Für Matrox G200/G400 ist dies mga.o, für ATI Rage128 r128.o, für Intel i810 ein i810.o, für ATI Radeon ist dies radeon.o. Sollte die Kompilation der DRI Module fehlgeschlagen sein, sollten Sie die Linux Kernel Version überprüfen. Auch werden nicht immer die allerneuesten Kernel unterstützt.

Nun kopieren wir alle Treiberdateien in die XFree86 Installation. Bei Bedarf können die zu überschreibenden Dateien vorher gesichert werden. In diesem Beispiel verwenden wir eine Radeon Karte. Für Ihre eigene Karte müssen Sie die entsprechenden Dateien kopieren. Ein Blick in das entsprechende Verzeichnis hilft hier oft weiter.

### Befehlsauflistung 19

```
# cd ~/DRI-CVS/build/xc/exports/lib/modules/  
# cp dri/r200_dri.so /usr/X11R6/lib/modules/dri/  
# cp drivers/atimisc_drv.o /usr/X11R6/lib/modules/drivers/  
# cp drivers/radeon_drv.o /usr/X11R6/lib/modules/drivers/  
# cp extensions/libdri.a /usr/X11R6/lib/modules/extensions/  
# cp linux/libdrm.a /usr/X11R6/lib/modules/linux/
```

Dann folgen Sie den Anweisungen in Kapitel 3, Abschnitt "Konfiguration von XF86Config".

um das entsprechende DRM Modul in den laufenden KErnel zu laden, sollte das KErnel modul nach /lib/modules/ kopiert werden und mit uname -r `/kernel/drivers/char/drm/ ausgeführt werden. Anschliessend **modules-update** und den X Server neu starten. Sollten Sie den Treiber nicht für den aktuell benutzten Kernel erstellt haben, muss uname -r an den zu verwendenden Kernel angepasst werden.

### Warnung

Stellen Sie sicher, dass ältere DRI Module vorher entladen werden. Manche DRM Module erwarten das apgart vorher geladen wird.

## 6. Performance Tuning

### 6.1 Das meiste aus Direct Rendering herausholen

Einige Optionen können die Performance um mehr als 30% steigern. Diese können in `/etc/X11/XF86Config` gesetzt werden:

#### Befehlsauflistung 20: XF86Config

```
Section "Device"
    Option      "AGPMode" "4"
// Dies steigerte meine FPS von 609 zu 618.
    Option      "AGPFastWrite" "True"
// Dies hatte keinen messbaren Effekt, kann aber die Instabilität des Computers erhöhen.
// Eventuell muss es im BIOS zusätzlich aktiviert werden.
    Option      "EnablePageFlip" "True"
// Dies steigerte FPS von 618 zu 702. Angeblich soll es riskant sein, aber wenige Leute hatten P:
    ...
EndSection
```

Für weitere Tuning Optionen sollte die [DRI Infoseite](#) konsultiert werden.

## 7. Problembehebung

### 7.1 Es funktioniert einfach nicht. Ich habe kein Rendering und ich weiss nicht warum.

Versuchen Sie ein `insmod radeon` vor dem Starten des X server. Versuchen Sie ausserdem `apgart` als Modul zu erstellen, statt direkt in den Kernel zu kompilieren.

### 7.2 Beim Ausführen von `startx` erhalte ich folgenden Fehler: "[drm] failed to load kernel module apgart"

Dies erscheint immer, wenn `apgart` direkt in den Kernel einkompiliert wird. Sie können dies getrost ignorieren.

### 7.3 Direct rendering funktioniert nicht und in `/var/log/XFree86.0.log` erhalte ich eine Fehlermeldung, das die Treiberversion falsch ist.

Sie benutzen nicht den `xfree-drm` Treiber. Sind Sie sicher, dass DRM und der Treiber in den richtigen Kernel kompiliert wurde?

### 7.4 Ich besitze eine Radeon und möchte TV-Out nutzen

Dafür gibt es die `ati-gatos` Treiber. `emerge -s gatos`.

### 7.5 Es funktioniert immer noch nicht. Meine Karte ist so neu und megaübercool, dass sie nicht unterstützt wird.

Probieren Sie die binären Treiber aus. Sollte für Ihre Karte noch keine Treiber geben, probieren Sie VESA. Es ist langsam, funktioniert aber. Halten Sie nach neuen Treibern Ausschau.

### 7.6 I habe eine PCI Karte, die nicht arbeiten will. Hilfe!

Im Abschnitt "Device" muss `ForcePCIMode` eingetragen sein.

#### Befehlsauflistung 21

```
Option "ForcePCIMode" "True"
```

## 8. Referenzen

- <http://forums.gentoo.org/viewtopic.php?t=46681>
- <http://forums.gentoo.org/viewtopic.php?t=29264>

- <http://dri.sourceforge.net/>
- [http://www.retinalburn.net/linux/dri\\_status.html](http://www.retinalburn.net/linux/dri_status.html)



## >> Nano Grundlagen

### 1. Nano Grundlagen

#### 1.1 Zweck

Diese Hilfe wurde geschrieben, um grundlegende Aktionen mit nano zu erläutern. Mehr Informationen über nano finden Sie hier:

<http://www.nano-editor.org>

#### 1.2 Öffnen und Erstellen von Dateien

Das Öffnen und Erstellen von Dateien ist in nano sehr einfach, geben Sie einfach folgendes ein:

##### Befehlsauflistung 1

```
# nano filename
```

Nano ist ein Editor ohne Modus, Sie können also sofort Text einfügen. Wenn Sie eine Konfigurationsdatei wie z.B. `/etc/fstab` bearbeiten, dann benutzen Sie die Option `-w` z.B.

##### Befehlsauflistung 2

```
# nano -w /etc/fstab
```

##### Warnung

Es ist sehr wichtig, dass Sie die Option `-w` benutzen, wenn Sie eine Konfigurationsdatei öffnen. Wenn Sie dies vergessen, könnte das Ihr System davon abhalten zu booten oder andere schädliche Dinge anrichten.

#### 1.3 Ausschneiden und Einfügen

Um eine einzige Zeile auszuschneiden benutzen Sie `^K`. Sie halten einfach die Tasten `CTRL` und `k` zur selben Zeit gedrückt. Die Zeile verschwindet nun. Um Sie wieder einzufügen, bewegen Sie den Cursor einfach dahin, wo Sie sie wieder einfügen wollen und drücken `^U` (`CTRL u`) auf der Tastatur. BAM! Die Zeile ist wieder da. Gehen wir davon aus, Sie müssen einen ganzen Paragraphen bewegen. Um mehrere Zeilen zu bewegen, schneiden Sie einfach jede Zeile einzeln aus und fügen Sie dann wieder ein. BAM! Der komplette Paragraph erscheint an dieser Stelle.

Wenn Sie eine etwas feinere Kontrolle haben wollen, dann müssen Sie den Text markieren. Bewegen Sie den Cursor an den Anfang des Textes wo sie zu schneiden beginnen wollen. Drücken Sie dann `^^` (Das ist `ctrl ^` oder `ctrl 6`). Bewegen Sie nun den Cursor an das Ende des Textes der ausgeschnitten werden soll. Der gewünschte Text sollte nun aufleuchten. Wenn Sie die Markierung rückgängig machen wollen, drücken Sie einfach wieder `^^`. Drücken Sie `^K` um den Text auszuschneiden. Benutzen Sie `^U` um ihn einzufügen. BAM! Nun funktioniert es.

#### 1.4 Nach Text suchen

Nach einem string zu suchen ist einfach, solange Sie an "**WhereIs**" anstatt an "**Search**" denken. Drücken Sie einfach `^W` (`CTRL w`) und geben Sie den string ein. Wie ich sagte, einfach.

#### 1.5 Zusammenfassung

Das ist alles! Danke an kiyose und quazion aus #gentoo. Um mehr Informationen über nano zu bekommen schauen Sie hier:

<http://www.nano-editor.org>



## >> vi lernen -- die "Spickzettel"-Methode

[Bitte Kapitel auswählen] ▾

### 1. Und los geht's

#### 1.1 Einführung

Dieses Tutorial wird Ihnen zeigen, wie man den mächtigen visuellen Editor vi benutzt. Anhand einer speziellen beschleunigten "Spickzettel"-Methode, versucht dieses Tutorial Sie zu einem fortgeschrittenen vi-Benutzer zu machen, und das ohne viel Zeit in Anspruch zu nehmen. In diesem vi-Tutorial werden Sie lernen sich innerhalb von Texten zu bewegen, Text zu bearbeiten, den insert-mode zu benutzen, Text zu kopieren und wieder einzufügen, und natürlich solch wichtige vim-Erweiterungen wie den Visual-Mode und Multi-Window-Editing.

Wenn Sie vi nicht kennen oder Ihnen das Arbeiten damit unbequem erscheint, dann können Sie dieses Tutorial dazu benutzen, um Ihr Arbeiten mit einem der beliebtesten und mächtigsten Editoren im Linux/UNIX-Sektor zu beschleunigen.

#### 1.2 Über den Leitfaden und das vi Lernen beschleunigen

Es gibt eine Sache im Speziellen, die es erschwert, vi zu lernen -- vi hat unglaublich viele Kommandos. Um vi also effektiv nutzen zu können, müssen Sie sich einige davon merken. Dies kann eine lange Zeit in Anspruch nehmen, und eines der Ziele dieses Tutorials ist eben nicht besonders viel Zeit zu beanspruchen. Für uns ist das zu Beginn eine grosse Herausforderung: Wie kann ich Ihnen helfen, diese Kommandos innerhalb kurzer Zeit im Kopf zu haben?

Um dieser Herausforderung entgegenzutreten während wir durch dieses Tutorial voranschreiten, werden wir uns Stückchen für Stückchen einen Spickzettel zusammenstellen. Dieser Zettel wird alle wichtigen vi-Kommandos enthalten. Nachdem Sie dieses Tutorial abgeschlossen haben, wird es Ihnen möglich sein auf diesen Spickzettel zurückzugreifen, wenn Sie ein bestimmtes Kommando vergessen haben. Nach einiger Zeit werden sich die Kommandos in Ihrem Kopf festgesetzt haben, und sie werden immer unabhängiger von diesem Spickzettel werden. Mit der Spickzettel-Methode ist es möglich, vi schneller zu lernen als jemals zuvor.

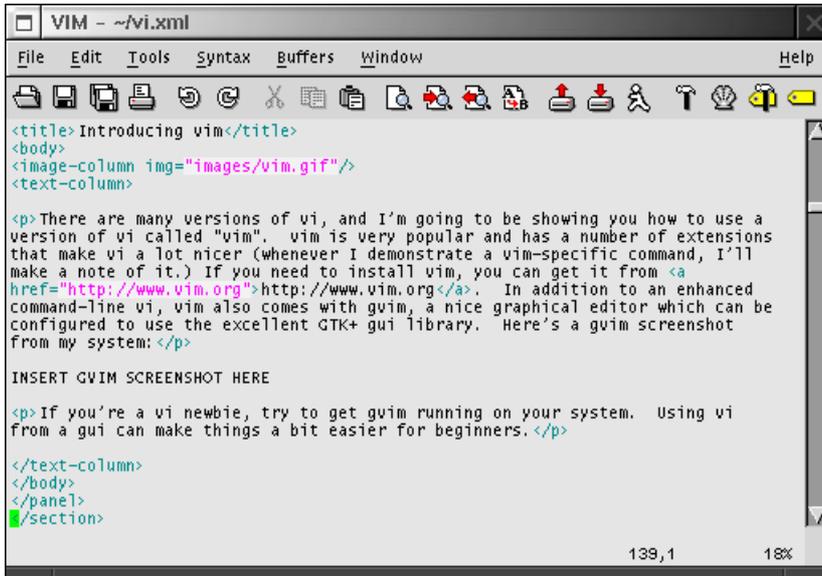
#### 1.3 Der Lernprozess

In diesem Leitfaden werde ich einige Methoden benutzen um Ihnen beim Lernen zu helfen. Zuallererst beschreibe ich, wie ein bestimmtes Kommando funktioniert. Dann werde ich Sie auffordern, einmal selbst in vi dieses Kommando auszuprobieren (zur Übung), und dann werde ich Sie auffordern, das Kommando Ihrem Spickzettel hinzuzufügen. Wenn Sie vi schnell lernen möchten ist es wichtig, dass Sie jeden dieser Schritte auch durchführen. Das Kommando auszuprobieren und dann auf dem Spickzettel zu vermerken wird Ihnen helfen, sich das Kommando zu behalten.

#### 1.4 Einführung in vim

Es gibt viele Versionen von vi, und ich werde Ihnen zeigen, wie man eine Version namens "vim" benutzt. vim ist sehr beliebt und besitzt einige Erweiterungen, die vi viel interessanter machen (wann immer ich ein vim-spezifisches Kommando benutze, werde ich es anzeigen). Wenn Sie vim installieren möchten, dann können Sie es von [www.vim.org](http://www.vim.org) herunterladen. Zusätzlich zum erweiterten Kommandozeilen-vi, kommt auch gvim mit auf den Rechner, ein netter grafischer Editor, der so eingestellt werden kann, dass er die ausgezeichnete GTK+ GUI-Bibliothek benutzt. Hier ein gvim-Screenshot von meinem System:

#### Grafik 1: VIM Screenshot



Wenn Sie ein vi-Neuling sind, dann versuchen Sie, gvim auf Ihrem System zum Laufen zu bekommen. Von einer GUI aus sind die Dinge etwas einfacher für Anfänger.

## 2. Erste Schritte

### 2.1 Eine Datei benutzen

Bevor Sie vi benutzen um Dateien zu bearbeiten, müssen Sie erst einmal wissen, wie Sie sich mit vi innerhalb des Textes bewegen können. vi hat eine Menge von Bewegungs-Kommandos, und wir werden uns viele von diesen ansehen. Für diesen teil des Tutorials suchen Sie sich bitte eine unwichtige Textdatei und öffnen diese mit vi, indem Sie folgendes am Kommandoprompt Ihrer Shell eingeben:

#### Befehlsauflistung 1

```
$ vi meinedatei.txt
```

Wenn Sie vim installiert haben, dann geben sie *vim meinedatei.txt* ein. Wenn Sie die Benutzung von gvim bevorzugen, dann geben Sie bitte *gvim meinedatei.txt* ein. *meinedatei.txt* ist durch den Namen einer einfachen Textdatei auf Ihrem System zu ersetzen.

### 2.2 vi intern

Nachdem vi geladen ist, sollten Sie einen Teil der Textdatei auf dem Bildschirm sehen. Herzlichen Glückwunsch! - Sie befinden sich nun in vi. Im Gegensatz zu anderen Editoren, befindet sich vi nach dem Start in einem speziellen so genannten "Command-Mode". Das heisst, dass wenn Sie nun *l* auf Ihrer Tastatur drücken, können Sie den Cursor einen Zeichen nach rechts bewegen, anstatt den Buchstaben *l* in den Text einzufügen. Im Command-Mode werden die Buchstaben auf Ihrer Tastatur dazu benutzt, Kommandos an vi zu senden anstatt Buchstaben in den Text einzufügen. Die essentiellsten aller Kommandos sind die zur Bewegung des Cursors. Lassen Sie uns einige ansehen.

## 3. Im Text bewegen

### 3.1 Mit vi im Text bewegen, Teil 1

Im Command-Mode können Sie die *h,j,k* und *l* Tasten benutzen, um den Cursor nach links, unten, oben und rechts zu bewegen. Wenn Sie eine moderne Version von vi benutzen, dann können Sie außerdem auch die Pfeiltasten zu diesem Zweck benutzen. Die Tasten *h,j,k* und *l* sind recht nützlich, denn wenn Sie sich an diese gewöhnt haben, dann können Sie sich innerhalb der Datei bewegen ohne Ihre Finger vom Hauptfeld der Tastatur nehmen zu müssen. Versuchen Sie *h,j,k* und *l* (und die Pfeiltasten) zu benutzen, um sich innerhalb des Textes zu bewegen. Versuchen Sie *h* zu benutzen um an den Beginn einer Zeile zu springen. Beachten Sie bitte, dass vi es nicht erlaubt, zum Ende der vorherigen Zeile zu springen, wenn Sie sich am Beginn einer Zeile befinden und dann *h* drücken. Das selbe gilt für das Springen auf die nächste Zeile, wenn Sie sich am Ende einer Zeile befinden.

## 3.2 Mit vi im Text bewegen, Teil 2

vi bietet spezielle Abkürzungen um zum Anfang oder Ende einer Zeile zu springen. Sie können `0` (Null) drücken, um zum ersten Zeichen in einer Zeile zu springen, und sie können `$` benutzen, um zum letzten Zeichen in einer Zeile zu springen. Probieren Sie die Kommandos aus und sehen Sie, was passiert. Da vi so viele nützliche Bewegungs-Kommandos beherrscht, ist er der ideale "Pager" (wie "less" oder "more"). vi als Pager zu benutzen wird Ihnen ausserdem helfen, die Bewegungs-Kommandos schneller zu lernen.

Sie können auch `<Strg>F` und `<Strg>B` benutzen, um sich seitenweise vorwärts und rückwärts zu bewegen. Moderne Versionen von vi (wie vim) erlauben es auch `BildAuf` und `BildAb` für diesen Zweck zu benutzen.

## 3.3 Wortsprünge, Teil 1

vi erlaubt es auch, den Cursor wortweise links oder rechts zu bewegen. Um zum **ersten** Zeichen des nächsten Wortes zu gehen, drücken Sie `w`. Um zum **letzten** Zeichen des nächsten Wortes zu gehen, drücken Sie `e`. Um zum ersten Buchstaben des **vorherigen** Wortes zu gelangen, drücken Sie `b`. Probieren Sie es aus!

## 3.4 Wortsprünge, Teil 2

Nachdem Sie nun ein bisschen mit den Kommandos zur wortweisen Bewegung herumgespielt haben, haben Sie vielleicht festgestellt, dass vi Wörter wie "foo-bar-oni" als fünf separate Wörter ansieht. Dies ist so, weil vi von Hause aus Wörter anhand von Leer- **oder** so genannten Delimiterzeichen trennt. Aus diesem Grund wird foo-bar-oni als fünf Wörter angesehen: "foo", "-", "bar", "-" und "oni".

Manchmal ist es genau das, was man möchte, und manchmal kann man dieses Feature überhaupt nicht gebrauchen. Glücklicherweise versteht vi auch das Konzept von "bigword". vi trennt bigwords anhand von **Leerzeichen oder Zeilenumbrüchen**. Dies bedeutet, dass foo-bar-oni zwar als 5 Wörter angesehen wird aber nur als ein bigword.

## 3.5 Wortsprünge, Teil 3

Um zum nächsten und vorherigen Bigword zu springen, können Sie ein grossgeschriebenes Wortbewegungskommando benutzen. Benutzen Sie `W`, um zum ersten Zeichen des nächsten Bigwords zu springen, `E`, um zum letzten Zeichen des nächsten Bigwords zu springen und `B`, um zum ersten Zeichen des vorhergehenden Bigwords zu springen. Probieren Sie es aus und vergleichen Sie die entsprechenden Wort- und Bigwordbewegungskommandos bis Sie die Unterschiede verstehen.

## 3.6 Grössere Sprünge

Wir müssen noch ein paar Kommandos mehr behandeln, bevor wir unseren Spickzettel zusammenstellen können. Sie können die `(` und `)` benutzen um zum Anfang des vorhergehenden und des nächsten Satzes zu springen. Zusätzlich dazu können Sie `{` oder `}` benutzen, um zum Anfang des aktuellen bzw. des nächsten Absatzes zu springen. Auch hier ist wieder einmal ausprobieren angesagt.

## 4. Beenden

### 4.1 Beenden

Wir haben die grundsätzlichen Bewegungskommandos behandelt, aber es gibt noch weitere Kommandos die Sie kennen müssen. Die Eingabe `:q` beendet vi. Sollte dies nicht funktionieren, dann haben Sie vielleicht die Datei geändert. Um vi nun zu beenden und die evtl. gemachten Änderungen zu verwerfen, müssen Sie `:q!` eingeben. Sie sollten nun wieder zurück auf Ihrer Shell sein.

In vi ist jedes mit einem ":" beginnenden Kommandos ein sogenanntes **ex-mode** Kommando. Der Grund hierfür ist, dass vi einen nicht visuellen Editor mit dem Namen **ex** eingebaut hat. Er kann ähnlich wie sed dazu benutzt werden um zeilenbasierte Bearbeitungsfunktionen durchzuführen. Desweiteren kann er auch dazu benutzt werden um vi zu beenden - wie wir eben gerade gesehen haben. Sollten Sie jemals die Taste `Q` drücken, während Sie sich im Kommandomodus befinden, dann wird vi sofort in den ex Modus schalten. Wenn dies passiert, wird Ihnen ein : Prompt präsentiert, das Drücken von Enter wird den ganzen Bildschirm nach oben rollen. Um wieder in den guten alten vi-Modus zurückzukommen, geben Sie einfach nur vi ein und drücken Enter.

## 5. Der Spickzettel

## 5.1 Die Anfänge des Spickzettels

Wir haben einige Kommandos behandelt und nun ist es Zeit, diese auf unseren Spickzettel zu übertragen. Für den Spickzettel brauchen Sie ein Blatt Papier (wir werden einige Informationen auf diesem Blatt vermerken!). Hier ist ein Bild meines Spickzettels, nachdem ich alle Kommandos aufgeschrieben habe, die wir bis hierhin behandelt haben. Versuchen Sie bitte, meinem Layout zu folgen, sodass alles auf ein Blatt Papier passt.

Grafik 2: Spickzettel

| MOVEMENT | HORIZONTAL                       | OPEN, SAVE + QUIT            |
|----------|----------------------------------|------------------------------|
| →, l     | right one character              | :q quit                      |
| ←, h     | left one character               | :q! quit, throw away changes |
| o        | beginning of line                |                              |
| \$       | end of line                      |                              |
| w/W      | beginning of next word/bigword   |                              |
| e/E      | end of next word/bigword         |                              |
| b/B      | beginning of prev. word/bigword  |                              |
| C/       | beginning of prev./next sentence |                              |
| {/}      | beginning of cur./next PP        |                              |
| ↑, k     | up one line                      |                              |
| ↓, j     | down one line                    |                              |
| P↑, ^B   | up one page                      |                              |
| P↓, ^F   | down one page                    |                              |

## 5.2 Vielseitiger vi

Lassen Sie uns mit unserem Kommandoschnelldurchlauf fortfahren. Im Kommandomodus können Sie zu einer bestimmten Zeile springen, indem Sie G verwenden. Um zur ersten Zeile in einer Datei zu springen, geben Sie 1G ein. Bitte beachten Sie dass grosse G!

Wenn Sie zur nächsten Erscheinung eines bestimmten Textmusters springen möchten, geben Sie /<regex> ein und drücken Sie anschliessend Enter. Ersetzen Sie <regex> mit dem Regulären Ausdruck nach dem Sie suchen. Wenn Sie nicht wissen, wie man reguläre Ausdrücke benutzt, dann haben Sie keine Angst : Die Eingabe von /foo wird zum nächsten Auftauchen von **foo** springen. Das einzige wo Sie aufpassen müssen sind die speziellen Zeichen ^, ., \$ oder \. Setzen Sie diesen Zeichen einen Backslash voran (\) und es sollte funktionieren. Zum Beispiel wird /foo\.gif zum nächsten Auftauchen von von "**foo.gif**" springen.

Um die Suche in Vorwärtsrichtung fortzusetzen, drücken Sie n auf der Tastatur. Um rückwärts zu suchen drücken Sie N. Wie immer sollten Sie diese Kommandos in Ihrer vi-Umgebung testen. Sie können auch // eingeben, um die letzte Suche zu wiederholen.

## 6. Speichern und Bearbeiten

### 6.1 Speichern und Speichern unter...

Wir haben behandelt, wie das **ex** Kommando :q zu benutzen ist, um vi zu beenden. Wenn Sie die gemachten Änderungen Speichern wollen, dann geben Sie :w ein. Wenn Sie die Änderungen in einer anderen Datei speichern wollen, dann geben Sie :w *dateiname.txt* ein, um in der Datei *dateiname.txt* zu speichern. Wenn Sie Speichern und Beenden wollen, so geben Sie :x oder :wq ein.

In vim (und anderen fortgeschrittenen vi Editoren wie elvis) können Sie mehrere Puffer auf einmal geöffnet haben. Um eine Datei in einem neuen Fenster zu öffnen, geben Sie :sp *dateiname.txt* ein. *dateiname.txt* wird in einem neuen geteilten Fenster erscheinen. Um zwischen Fenstern hin und her zu springen, geben Sie <Strg>w<Strg>w (also zweimal Strg-w) ein. Alle :q, :q!, :w und :x -Eingaben, die Sie machen, werden jeweils nur auf das momentan aktive Fenster angewendet.

### 6.2 Einfache Änderungen

Nun wird es Zeit einige der einfachen Bearbeitungskommandos kennenzulernen. Die Kommandos, die wir hier behandeln, werden "einfache" Bearbeitungskommandos genannt, da Sie sie im Kommandomodus verwenden. Die komplexeren Bearbeitungskommandos bringen Sie in den Einfügemodus (insert mode) - ein Modus, der es Ihnen erlaubt Text von der Tastatur ganz normal einzugeben. Wir werden gleich darauf zurück kommen.

Versuchen Sie einmal, den Cursor über einige Zeichen zu stellen und die Eingabe von `x` zu wiederholen. Sie werden sehen, dass dies jeweils den aktuellen Buchstaben am Cursor löschen wird. Nun bewegen Sie sich zur Mitte eines Absatzes irgendwo im Text und geben Sie `J` (Gross!) ein. Sie werden sehen, dass dieses Kommando `vi` anweist, die nächste Zeile an das Ende der aktuellen Zeile anzuhängen. Nun bewegen Sie den Cursor über ein Zeichen und geben `r` ein und dann ein neues Zeichen; Sie werden sehen, dass das alte Zeichen gegen das neu eingegebene ersetzt wurde. Zuletzt bewegen Sie den Cursor zu irgendeiner Zeile im Text und geben Sie `dd` ein. Sie werden sehen, daß `dd` die aktuelle Textzeile löscht.

### 6.3 Wiederholen und löschen

Sie können jedes Bearbeitungskommando wiederholen, indem Sie `.` drücken. Wenn Sie damit experimentieren werden Sie merken, dass `dd...` vier Zeilen löschen wird und `J.....` vier Zeilen aneinanderhängen wird. Wie erwartet, bietet `vi` eine weitere Abkürzung.

Um Text zu löschen können Sie auch das `d` Kommando kombiniert mit Bewegungskommando benutzen. Zum Beispiel wird `dw` alles von der aktuellen Position bis zum Anfang des nächsten Wortes löschen; `d)` wird alles bis zum Beginn des nächsten Satzes löschen und `d}` wird den kompletten Rest des aktuellen Absatzes löschen. Experimentieren Sie mit dem `d` Kommando und den anderen Bearbeitungskommandos bis Sie sich mit ihnen vertraut fühlen.

### 6.4 Rückgängig!

Nun, da wir mit Löschen experimentieren ist es sicherlich genau der richtige Zeitpunkt, auch zu lernen, wie man seine Änderungen wieder rückgängig macht. Durch das Drücken von `u` erlaubte die Originalversion von `vi` nur den letzten Bearbeitungsschritt wieder rückgängig zu machen. Trotzdem erlauben moderne `vi` Versionen wie `vim` mehrere Änderungen durch wiederholtes drücken von `u` auch weitere Änderungen an Ihrer Datei wieder rückgängig zu machen. Versuchen Sie einige der `d` und `u` Kommandos miteinander zu kombinieren.

### 6.5 Aktualisieren des Spickzettels

Es wird Zeit unseren Spickzettel zu aktualisieren! Nachdem Sie alle Kommandos hinzugefügt haben, die wir bisher behandelten, sollte der Spickzettel in etwa so aussehen:

**Grafik 3: Spickzettel mit Bearbeitungskommandos**

THE SEMI-OFFICIAL IBM<sup>®</sup> developerWorks™ vi CHEAT SHEET!

|           |                                  |                  |   |
|-----------|----------------------------------|------------------|---|
| MOVEMENT  |                                  | ←← HORIZONTAL →→ | OPEN, SAVE + QUIT   |
| →, l      | right one character              |                  | :q quit   |
| ←, h      | left one character               |                  | :q! quit, throw away changes                              |
| 0         | beginning of line                |                  | :w filename save as filename                              |
| \$        | end of line                      |                  | :x quit+save  |
| w/W       | beginning of next word/bigword   |                  | WINDOWING (vim, elvis)                                    |
| e/E       | end of next word/bigword         |                  | :sp filename new split-frame window                       |
| b/B       | beginning of prev. word/bigword  |                  | ^W^W goto next window                                     |
| (/)       | beginning of prev./next sentence |                  | BASIC EDITS ↓↓  |
| {/}       | beginning of cur./next ¶         |                  |   |
| ↑, k      | up one line                      | ↑ ↓              | THESE COMMANDS MODIFY TEXT (BUT KEEP YOU IN COMMAND MODE) |
| ↓, j      | down one line                    | ↑ ↓              | x delete character under cursor                           |
| ↑↑, ↑B    | up one page                      | ↑ ↓              | J join next line to end of current line                   |
| ↓↓, ↓F    | down one page                    | ↑ ↓              | r(char) replace char under cursor w/ (char)               |
| (number)G | goto line (number)               | ↑ ↓              | dd delete current line                                    |
| /string   | find string                      |                  | d(move) delete from cursor to (move)                      |
| n/N       | repeat search / backwards        |                  | u undo!   |
|           |                                  |                  | . repeat last edit command                                |

## 7. Einfügemodus

### 7.1 Einfügemodus

Bisher haben wir behandelt, wie man sich mittels vi in Texten bewegt, Datei Ein- und Ausgabe realisiert und ein paar grundlegende Bearbeitungskommandos durchführt. Aber ich habe Ihnen bisher immer nicht gezeigt, wie Sie freie Texteingaben machen können. Dies war Absicht, denn vi's Einfügemodus (insert mode) ist zuerst ein bisschen kompliziert. Nachdem Sie sich mit dem Einfügemodus vertraut gemacht haben, wird Ihnen die Komplexität (und Flexibilität) unverzichtbar erscheinen.

Im vi Einfügemodus haben Sie die Möglichkeit, Text direkt am Bildschirm einzugeben - ganz so wie es auch in vielen anderen Editoren und Textverarbeitungen der Fall ist. Sobald Sie Ihre Änderungen gemacht haben, können Sie Escape drücken, um in den Kommandomodus zurückzukehren. Sie kommen in den Einfügemodus, indem Sie *i* oder *a* eingeben. Wenn Sie *i* eingeben, dann wird Ihr Text vor dem aktuellen Zeichen **eingefügt** werden. Wenn Sie *a* eingeben, dann wird Ihr Text hinter dem aktuellen Zeichen**angehängt** werden. Beachten Sie bitte: nachdem Sie Ihren Text eingegeben haben drücken Sie bitte <ESC> um in den Kommandomodus zurückzukehren.

### 7.2 Vorteile des Einfügemodus

Machen Sie weiter und versuchen Sie, die *a* und *i* Kommandos zu benutzen. Drücken Sie entweder *a* oder *i*, geben Sie ein wenig Text ein und dann drücken Sie Escape, um wieder in den Kommandomodus zurückzukehren. Nachdem Sie *a* oder *i* gedrückt haben, versuchen Sie <ENTER> einzugeben und passen Sie auf, was passiert. Durch die Benutzung der Pfeiltasten und <DEL>-Taste können Sie beachtliche Bearbeitungsschritte machen, ohne ständig in den Eingabemodus und wieder heraus zu springen.

### 7.3 Einfügeooptionen

Hier kommen noch ein paar andere praktische Möglichkeiten, um in den Einfügemodus zu gelangen. Drücken Sie *A* (Gross!) um - unabhängig von der aktuellen Position in der Zeile - mit dem Anhängen am **Ende** der Zeile zu beginnen. Sie können auch *I* (Gross!) drücken, um den Text am **Anfang** der Zeile einzufügen. Oder *o* um eine neue Leerzeile unter der aktuellen Zeile einzufügen, in die Sie dann Text eingeben können, und *O* (Gross!) um eine neue Zeile über der aktuellen einzufügen. Um die aktuelle Zeile mit einer neuen Zeile zu überschreiben geben Sie bitte *cc* als Kommando. Um innerhalb der Zeile Alles von der aktuellen Position bis zum Ende hin zu löschen können Sie *c\$* eingeben. Um umgekehrt alles von der aktuellen Position bis zum Anfang der Zeile zu löschen, geben Sie das Kommando *c0* ein.

Zusätzlich dazu, dass jedes dieser Kommandos eine spezielle Operation durchführt, bringen sie Sie in den Einfügemodus. Nachdem Sie Ihren Text eingegeben haben, drücken Sie <ESC> um in den Kommandomodus zurückzukehren.

### 7.4 Textänderungen

Wir haben das *c* Kommando bisher benutzt, wenn wir *cc*, *c0* und *c\$* angewendet haben. *cc* ist eine spezielle Form des Änderungskommandos, ähnlich zu *dd*. Die *c0* und *c\$* Kommandos sind Beispiele der Anwendung des Änderungskommandos in Verbindung mit einem Bewegungskommandos. In dieser Form arbeitet *c* gleichwertig zu *d*, mit Ausnahme darin bestehend, dass es Sie im Einfügemodus lässt, sodass Sie ersetzenden Text für die gelöschte Region eingeben können. Versuchen Sie einige der Bewegungskommandos mit *c* zu kombinieren und probieren Sie sie mit Ihrer Datei aus (Tipp : *cW*, *ce*, *c()*).

## 8. Verbund-Kommandos

### 8.1 Verbund-Kommandos

vi wird **wirklich** sehr mächtig, wenn Sie damit beginnen Verbund-Kommandos (also Kombinationen) wie z.B. *d{* and *cw* zu benutzen. Zusätzlich zu diesen Kommandos können Sie eine Zahl mit einem Bewegungskommando kombinieren, wie zum Beispiel *3w*, welches vi anweist drei Wörter nach rechts zu springen. Hier ein paar Beispiele zu "Bewegungskombinationen": *12b*, *4j*.

Zusätzlich zu (Zahl)(Bewegungskommando)-Kombinationen erlaubt vi auch *d* oder *c* mit einer Zahl oder einem Bewegungskommando zu kombinieren. So wird zum Beispiel *d3w* die nächsten drei Wörter löschen und *d2j* wird die aktuelle und die nächsten beiden Zeilen löschen. Probieren Sie einige *d* und *c*

Kombinationen aus, um ein Gefühl dafür zu bekommen, wie mächtig und schnell das Editieren mit vi sein kann. Sobald Ihnen diese Kommandos normal erscheinen, werden Sie die Fähigkeit besitzen Dateien in unglaublicher Geschwindigkeit zu bearbeiten.

## 8.2 Aktualisieren des Spickzettels

Es ist wieder einmal Zeit, den Spickzettel zu aktualisieren. Hier sehen Sie, wie er in etwa nun aussehen sollte:

**Grafik 4: Spickzettel mit Verbundkommandos**

THE SEMI-OFFICIAL IBM<sup>®</sup> developerWorks™ vi CHEAT SHEET!

| MOVEMENT  | ◀◀ HORIZONTAL ▶▶                 | OPEN, SAVE + QUIT   |
|---|----------------------------------|---|
| →, l  | right one character              | :% quit   |
| ←, h  | left one character               | :%! quit, throw away changes                              |
| o   | beginning of line                | :%w filename save as filename                             |
| \$  | end of line                      | :%x quit+save   |
| w/W   | beginning of next word/bigword   | WINDOWING (vim, elvis)                                    |
| e/E   | end of next word/bigword         | :%sp filename new split-frame window                      |
| b/B   | beginning of prev. word/bigword  | AW^W goto next window                                     |
| (/)   | beginning of prev./next sentence | BASIC EDITS ↓↓  |
| {/}   | beginning of cur./next ¶         |   |
| ↑, k  | up one line                      | THESE COMMANDS MODIFY TEXT (BUT KEEP YOU IN COMMAND MODE) |
| ↓, j  | down one line                    | x delete character under cursor                           |
| PAUP, ^B  | up one page                      | J join next line to end of current line                   |
| PAWDN, ^F   | down one page                    | r(char) replace char under cursor w/ (char)               |
| (number)G   | goto line (number)               | dd delete current line                                    |
| /string   | find string                      | d(move) delete from cursor to (move)                      |
| n/N   | repeat search f/ backwards       | U undo!   |
|   |                                  | . repeat last edit command                                |
| INSERT MODE   |                                  | i/a insert before/after cursor                            |
| ANY OF THESE COMMANDS => WILL PUT YOU IN INSERT MODE. IN INSERT MODE, YOU CAN TYPE IN TEXT, HIT RETURN FOR A NEW LINE, ←, ↓, →, ↑ TO MOVE AND DELETE. TO RETURN TO COMMAND MODE, HIT [ESC]. |                                  | I/A insert @ beginning/end of line                        |
|   |                                  | o/O new line below/above, then insert                     |
|   |                                  | cc replace current line                                   |
|   |                                  | c(move) replace to (move)                                 |
| COMPOUND COMMANDS - the power of vi   |                                  | CHANGE!   |
| NOTION!   | 3 → 3 chars right                | c) replace rest of sentence                               |
| 4)  | 4 sentences →                    | c\$ replace rest of line                                  |
| 2b  | ← 2 words                        |   |
| 12}   | → 12 paragraphs                  |   |
|   | DELETION!                        |   |
|   | =d3w delete next 3 words         |   |
|   | d} delete remainder of ¶         |   |
|   | d) delete remainder of sentence  |   |

## 8.3 Produktivitätssteigernde Fähigkeiten

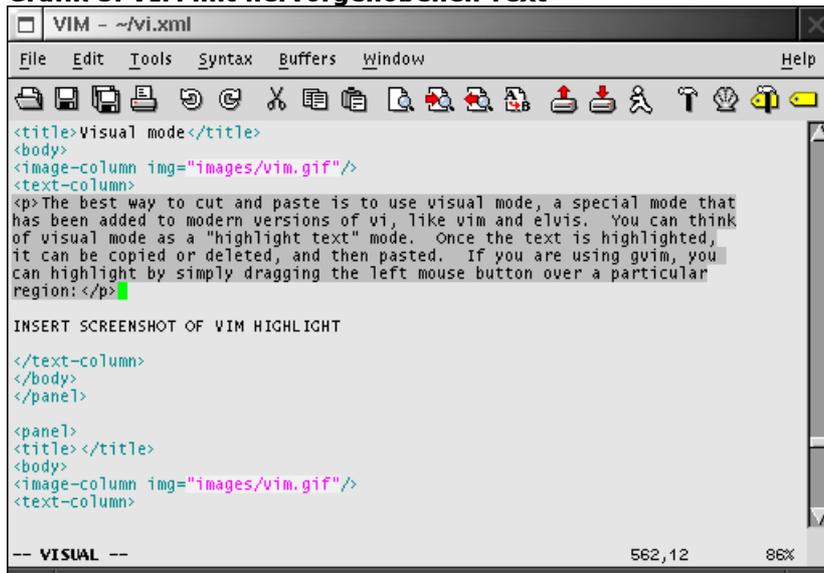
Bisher haben wir behandelt, wie man Bewegungen durchführt, Speichert und Beendet, einfache Änderungen und Löschungen realisiert, und wie man den Einfügemodus benutzt. Mit den Sachen die Sie bisher auf Ihrem Spickzettel haben, sollten Sie schon jegliche Aufgabe erledigen können.

Trotz Alledem: vi hat viele weitere mächtige Kommandos. In diesem Abschnitt werden Sie lernen, wie man **Ausschneidet**, **Kopiert** und **Einfügt**, **Sucht** und **Ersetzt** und **Autoindent** (automatisches Einrücken) benutzt. Diese Kommandos tragen dazu bei, vi spassiger und produktiver zu machen.

## 8.4 Visual mode

Die beste Möglichkeit zum Ausschneiden und Einfügen ist der Visual Mode, ein spezieller Modus der den modernen Versionen von vi (wie vim und elvis) hinzugefügt wurde. Sie können sich den Visual Mode als einen Modus mit Text hervorhebung vorstellen. Sobald der Text hervorgehoben ist, kann er kopiert oder gelöscht und dann eingefügt werden. Wenn Sie gvim benutzen, dann können Sie Text hervorheben, indem Sie einfach mit der linken Maustaste gedrückt über einen beliebigen Bereich fahren.

**Grafik 5: VIM mit hervorgehobenen Text**



Desweiteren können Sie auch in den Visual-Mode gehen indem Sie `v` drücken (dies wird für Sie auch die einzige Möglichkeit darstellen, wenn Sie `vi` aus der Konsole benutzen). Dann können Sie den Cursor bewegen, indem Sie Bewegungskommandos benutzen (normalerweise die Pfeiltasten) und damit einen Textbereich markieren. Sobald er einmal hervorgehoben ist, können wir den Text ausschneiden oder kopieren.

Wenn Sie Text kopieren, dann drücken Sie `y` (das für "yank", also "rausreißen" steht). Wenn Sie Text ausschneiden wollen, dann drücken Sie `d`. Sie werden dann zurück in den Kommandomodus gebracht. Nun bewegen Sie den Cursor zu der Position, an der Sie den ausgeschnittenen oder kopierten Text einfügen wollen und drücken Sie `P` um den Text nach dem Cursor einzufügen, oder drücken Sie `p` um den Text vor dem Cursor einzufügen. Fertig - der Ausschneide-/Kopiervorgang ist schon erledigt! Versuchen Sie ein paar Kopier-, Ausschneide- und Einfügeoperationen bevor Sie zum nächsten Abschnitt fortfahren.

## 8.5 Replacing text

Um Textmuster zu ersetzen, benutzen wir den **ex**-Modus. Wenn Sie das erste Muster auf der aktuellen Zeile ersetzen möchten, geben Sie `:s/<regexp>/<replacement>/` ein und drücken Sie `<ENTER>`, wobei `<regexp>` das Muster ist, das gesucht und ersetzt werden soll und `<replacement>` das Ersetzmuster darstellt. Um alle Entsprechungen auf der aktuellen Zeile zu ersetzen, geben Sie `:s/<regexp>/<replacement>/g` ein. Um jedes Vorkommen innerhalb der Datei zu ersetzen (das will man normalerweise), geben Sie `:%s/<regexp>/<replacement>/g` ein. Wenn Sie global alles ersetzen möchten, aber `vi` vor jeder Ersetzung nachfragen soll, dann geben Sie `:%s/<regexp>/<replacement>/gc` (steht für "bestätigen") ein und drücken Sie `<ENTER>`.

## 8.6 Indentation

`vi` beherrscht Automatisches Einrücken, welches als Hilfe gedacht ist, wenn Sie Quellcode editieren. Die meisten modernen Versionen von `vi` (wie `vim`) werden diesen Modus automatisch aktivieren, wenn Sie eine Quellcodedatei bearbeiten (wie z.B. eine `.c`-Datei). Wenn Automatisches Einrücken aktiviert ist, dann können Sie mittels `<Strg>d` die Einrückung eine Stufe nach links bewegen, und mit `<Strg>t` eine Stufe nach rechts. Wenn Automatisches Einrücken nicht automatisch aktiviert wurde, dann können Sie es manuell anschalten indem Sie `:set autoindent` im **ex** Modus eingeben. Sie können `vi` auch Ihre bevorzugte Tabulatorweite zu benutzen; dies geht mittels des `:set tabstop` Kommandos. `:set tabstop=4` ist recht beliebt.

## 8.7 Der letzte Spickzettel

Nun gut - dies ist das Ende des Tutorials! Nachdem Sie nun all die fortgeschrittenen Editierkommandos auf dem Spickzettel notiert haben, sollte er in etwa so aussehen:

## Grafik 6: Letzter Spickzettel

# THE SEMI-OFFICIAL IBM<sup>®</sup> developerWorks™ vi CHEAT SHEET!

|   |                                  |  |                                       |
|---|----------------------------------|--|---------------------------------------|
| <b>MOVEMENT</b> << HORIZONTAL >>  |                                  | <b>OPEN, SAVE + QUIT</b>   |                                       |
| →, l  | right one character              | :q   | quit                                  |
| ←, h  | left one character               | :q!  | quit, throw away changes              |
| o   | beginning of line                | :w filename  | save as filename                      |
| \$  | end of line                      | :x   | quit+save                             |
| w/W   | beginning of next word/bigword   | <b>WINDOWING (vim, elvis)</b>                                    |                                       |
| e/E   | end of next word/bigword         | :sp filename   | new split-frame window                |
| b/B   | beginning of prev. word/bigword  | ^W^W   | goto next window                      |
| (/)   | beginning of prev./next sentence | <b>BASIC EDITS</b> ↓↓  |                                       |
| {/}   | beginning of cur./next ¶         | <b>THESE COMMANDS MODIFY TEXT (BUT KEEP YOU IN COMMAND MODE)</b> |                                       |
| ↑, k  | up one line                      | x  | delete character under cursor         |
| ↓, j  | down one line                    | J  | join next line to end of current line |
| ↑↑↑, ↑B   | up one page                      | r(char)  | replace char under cursor w/ (char)   |
| ↓↓↓, ↓F   | down one page                    | dd   | delete current line                   |
| (number)G   | goto line (number)               | d(move)  | delete from cursor to (move)          |
| /string   | find string                      | U  | undo!                                 |
| n/N   | repeat search/backwards          | .  | repeat last edit command              |
| <b>INSERT MODE</b>  |                                  | i/a  | insert before/after cursor            |
| ANY OF THESE COMMANDS => WILL PUT YOU IN INSERT MODE. IN INSERT MODE, YOU CAN TYPE IN TEXT, HIT RETURN FOR A NEW LINE, ←, ↓, →, ↑ TO MOVE AND [DELETE]. TO RETURN TO COMMAND MODE, HIT [ESC]. |                                  | I/A  | insert @ beginning/end of line        |
| <b>COMPOUND COMMANDS - the power of vi</b>  |                                  | o/O  | new line below/above, then insert     |
| <b>NOTION!</b>  | 3 → 3 chars right                | cc   | replace current line                  |
| 4 ) 4 sentences →   | 4 ) 4 sentences →                | c(move)  | replace to (move)                     |
| 2 b ← 2 words   | ← 2 words                        | <b>CHANGE!</b>   |                                       |
| 12 § → 12 paragraphs  | → 12 paragraphs                  | c)   | replace rest of sentence              |
| <b>DELETION!</b>  | d3w delete next 3 words          | c#   | replace rest of line                  |
| d § delete remainder of ¶   | d § delete remainder of ¶        |  |                                       |
| d) delete remainder of sentence   | d) delete remainder of sentence  |  |                                       |
| <b>CUT + PASTE (vim, elvis)</b>   |                                  | <b>SEARCH/REPLACE</b>  |                                       |
| 1. PRESS v TO ENTER VISUAL MODE   |                                  | :s/reg/rep/  | 1st match, cur. line                  |
| 2. MOVE CURSOR TO HIGHLIGHT TEXT  |                                  | :s/reg/rep/g   | all matches, cur. line                |
| 3. PRESS d TO CUT, y TO COPY  |                                  | :%s/reg/rep/g  | global replace                        |
| 4. MOVE TO TARGET LOCATION  |                                  | :%s/reg/rep/gc   | global w/ prompt                      |
| 5. HIT P TO PASTE AFTER CURSOR, p TO PASTE BEFORE CURSOR.   |                                  | <b>TABBING</b>   |                                       |
|   |                                  | ^D ← ^T →  | :set autoindent (turn on)             |
|   |                                  |  | :set tabstop=(num) => to set tab size |

Haben Sie Ihren Spickzettel immer bei der Hand und fangen Sie an mit vi Dateien zu editieren und E-Mails zu verfassen. Schauen Sie auf den Spickzettel wenn nötig: Innerhalb einer Woche werden Sie feststellen, dass Sie fast alle Kommandos behalten haben und Ihre Produktivität mit vi wird durch die Decke platzen! ;-)

## 8.8 Ressourcen

Hier sind ein paar Ressourcen, die Ihnen sicherlich weiterhelfen, wenn Sie mehr über vi lernen wollen:

- Die "[vi Lovers Home Page](#)", Eine hervorragende Ressource für alles rund um vi.
- Die "[vim Homepage](#)" ist der Ort für Infos rund um vim.
- Wenn Sie nach einem guten, altmodischen Buch suchen, dann ist "[Learning the vi Editor](#)", 6. Ausgabe sicherlich das Richtige für Sie. Viel Inhalt über vi und vi-Klone.

## 8.9 Über dieses Dokument

Die ursprüngliche Version dieses Artikels wurde zuerst bei IBM developerWorks veröffentlicht und ist Eigentum von Westtech Information Services. Dieses Dokument ist eine erneuerte Version des Original-Artikels und enthält verschiedene Verbesserungen durch das Gentoo Linux Documentation Team.



## >> Gentoo Linux CVS Tutorial

[Bitte Kapitel auswählen] ▀

### 1. Einleitung

#### 1.1 Struktur des Tutorials

Dieses Tutorial besteht aus zwei Teilen. Im ersten Teil werden Sie lernen wie Sie CVS als Nicht-Entwickler benutzen, d.h. wie Sie Quelltexte eines Programms herunterladen und stets aktualisiert halten. Im zweiten Teil werden sie dann in die Benutzung von CVS als Entwickler eingeführt. Dabei werden Sie lernen wie Sie Dateien innerhalb von CVS ändern, hinzufügen oder löschen können. Wenn Sie CVS noch nicht verwendet haben, dann sollten Sie mit dem ersten Teil beginnen. Wenn Sie allerdings schon grundlegende Erfahrungen mit CVS haben, dann können Sie den ersten Teil dieses Tutorials überspringen, wobei Sie den ersten Teil vielleicht überfliegen sollten.

#### 1.2 Was ist CVS und wofür brauche ich es?

CVS ist ein Client-Server-System, das es Entwicklern erlaubt ihre Projekte an einem zentralen Ort zu speichern, dem sogenannten **Repository**. Mit den CVS Client-Programmen können Entwickler nun den Inhalt des Repositories ändern. Das CVS merkt sich jede Änderung am Inhalt der Dateien. Auf diese Weise wird eine komplette Geschichte des Projektes geführt. Entwickler können sich so ältere Versionen einer Datei oder eine Liste von Änderungen ansehen und andere nützliche Anwendungen ausführen.

#### 1.3 Welche Rolle spielt CVS?

Viele Open-Source Projekte haben ihren eigenen CVS Server, die von den Entwicklern des Projekts als zentraler Ablageplatz für ihre gesamte Arbeit genutzt werden. Die Quelltexte werden häufig täglich geändert und verbessert. Dabei sind die Entwickler häufig auf der ganzen Welt verstreut, und doch können sie dank CVS zusammen an einem Projekt arbeiten ohne sich gegenseitig zu behindern.

#### 1.4 CVS -- Die aktuellsten Entwickler-Quelltexte

Wenn die Entwickler mit ihren Änderungen zufrieden sind, dann packen sie ihr Projekt in eine .tar.gz-Datei und veröffentlichen es als neue offizielle Version. Doch aus verschiedenen Gründen sind diese Versionen manchmal nicht aktuell genug. Im ersten Teil dieses Tutorials wird Ihnen nun gezeigt, wie Sie CVS einsetzen, um den neuesten Stand der Entwicklung herunterzuladen und für sich selbst nutzen zu können.

#### 1.5 Habe ich CVS installiert?

Bevor Sie CVS benutzen können, müssen Sie es auf Ihrem System installieren. Der einfachste Weg auszuprobieren, ob CVS schon installiert ist:

##### Befehlsauflistung 1

```
#cvs
```

Wenn ein CVS Befehl gefunden wird, dann ist CVS auf Ihrem System schon installiert. Sonst müssen Sie entweder ein Binär-Paket für Ihre Distribution suchen und installieren, oder CVS selber kompilieren und installieren.

##### Notiz

Anmerkung des Übersetzers: CVS ist natürlich auch im Portage Tree unter [dev-util/cvs](#) zu finden.

#### 1.6 CVS selber kompilieren

Zunächst müssen Sie cvs-1.11.tar.gz von <ftp://ftp.cvshome.org/pub.cvs-1.11/cvs-1.11.tar.gz> herunterladen. Für den Fall, dass auf dem [CVS ftp-Server](#) eine neuere Version existiert, können Sie natürlich auch diese herunterladen. Dann führen Sie diese Befehle aus. (Die Ausgabe wird hier nicht angegeben):

##### Befehlsauflistung 2

```
# tar xzvf cvs-1.11.tar.gz
# cd cvs-1.11
# ./configure
# make
# make install
```

Nun sollte CVS auf Ihrem System installiert sein.

## 1.7 Die CVSROOT

Bevor wir beginnen, müssen Sie über einige Grundlagen Bescheid wissen. Um sich zu einem CVS Repository zu verbinden müssen Sie zunächst die CVSROOT (Das CVS Wurzelverzeichnis) kennen. Die CVSROOT ist eine Zeichenfolge, ähnlich einer URL, das dem cvs-Befehl mitteilt, wo sich das Repository befindet, und wie Sie sich mit diesem verbinden möchten. CVS benutzt verschiedene Formate für die CVSROOT-Zeichenfolge, je nachdem, ob es sich um ein lokales oder entferntes Repository handelt und welche Methode Sie verwenden, um sich mit diesem zu verbinden. Hier sind einige Beispiele für CVSROOTs mit der zugehörigen Erklärung...

## 1.8 Ein lokales Verzeichnis

### Befehlsauflistung 3

```
CVSROOT=/home/cvsroot
```

Diese Einstellung würde man verwenden, wenn man ein lokales CVSROOT-Verzeichnis verwenden möchte. Dabei befindet sich das lokale CVSROOT im Verzeichnis `/home/cvsroot`. Natürlich wäre es auch möglich ein entferntes Verzeichnis mittels NFS an dieser Stelle gemounted zu haben.

## 1.9 Ein entfernter passwortgeschützter Server

### Befehlsauflistung 4

```
CVSROOT=:pserver:cvs@foo.bar.com:/home/cvsroot
```

Dies ist ein Beispiel für eine CVSROOT auf dem entfernten Server `foo.bar.com`, wobei dort das Verzeichnis `/home/cvsroot` verwendet wird. Das vorangestellte `:pserver:` sagt dem client, dass Sie sich über das CVS eigene Passwort-Server-Protokoll anmelden wollen. Diese Methode wird häufig verwendet, um anonymen Benutzern den Zugang zu öffentlichen CVS-Servern zu gestatten.

## 1.10 Ein entfernter Zugang über rsh/ssh

### Befehlsauflistung 5

```
CVSROOT=drobbins@foo.bar.com:/data/cvs
```

Dies ist ein Beispiel für einen Zugang über das RSH oder SSH Protokoll. Dabei wird der CVS Server versuchen, das Repository auf `foo.bar.com` mittels des Benutzerkontos `drobbins` zu erreichen. Wenn die `CVS_RSH` Umgebungsvariable auf `"ssh"` gesetzt wurde, dann wird der CVS-Client die Verbindung über SSH aufbauen, sonst wird RSH verwendet. Der SSH-Zugang ist beliebt, wenn Sicherheit nötig ist. Allerdings geben weder RSH noch SSH anonymen Benutzern eine Möglichkeit sich einzuloggen. Um diese Zugriffsmethode verwenden zu können, brauchen sie ein Benutzerkonto auf dem System `foo.bar.com`.

## 1.11 Einige weitere Dinge...

Zusätzlich zur CVSROOT müssen Sie den Namen des Moduls (eine Sammlung von Quelltextdateien) wissen, den Sie "auschecken" möchten. Weiterhin brauchen Sie noch ein anonymes Passwort, um sich auf dem CVS-Passwort-Server einzuloggen. Es gibt hier kein Standardformat, wie etwa beim anonymen FTP, daher müssen Sie das anonyme Passwort auf der Entwickler-Webseite nachlesen. Wenn Sie all diese Informationen haben, können Sie loslegen.

## 1.12 Erste Schritte mit CVS, Teil 1

Um Quelltexte von einem CVS-Server herunterzuladen sind zwei Schritte notwendig. Zunächst loggen wir uns beim Passwort-Server ein. Dann laden wir die Quelltexte mit dem `checkout` Befehl herunter. Hier ein Beispiel von Befehlen, um die aktuellen Quelltexte des Samba Projektes herunterzuladen:

## Befehlsauflistung 6

```
# export CVSROOT=:pserver:cvs@pserver.samba.org:/cvsroot
```

Dieser erste Befehl setzt die CVSROOT-Umgebungsvariable. Wenn Sie diese Variable nicht setzen, müssen Sie die folgenden zwei Befehle jeweils durch ein `-d:pserver:cvs@pserver.samba.org:/cvsroot` hinter dem `cvs` Befehl erweitern. Die CVSROOT-Umgebungsvariable spart uns also einiges an Tipparbeit.

## 1.13 Erste Schritte mit CVS, Teil 2

Nun besprechen wir die Befehle, die notwendig sind, um eine aktuelle Kopie der Quelltexte zu erhalten. Weiter unten werden wir näher auf die genaue Bedeutung dieser Befehle eingehen.

## Befehlsauflistung 7

```
#cvs login
(Logging in to cvs@pserver.samba.org)
CVS password: (Geben Sie hier ihr Passwort ein)

#cvs -z5 co samba
U samba/COPYING
U samba/Manifesst
U samba/README
U samba/Read-Manifest-Now
U samba/Roadmap
U samba/WHATSNEW.txt
(Dies ist nur ein Ausschnitt aus der eigentlichen cvs co Ausgabe)
```

## 1.14 Erste Schritte mit CVS - Die Erklärung

Der erste Befehl loggt uns auf dem pserver ein. Der zweite Befehl sagt dem CVS Client, dass er das Samba-Modul auschecken soll (`co` - check out). Dabei wird ein gzip-Kompressions-Level von 5 ("-z5") verwendet, so wird die Übertragung über einen langsamen Internetzugang beschleunigt. Für jede Datei die auf dem lokalen System neu erzeugt wird, gibt CVS eine "U [Pfad]"-Zeile aus. Das "U" steht dabei für "update".

## 1.15 Nach dem Checkout

Wenn der checkout-Befehl fertig ist, sehen Sie ein Verzeichnis "samba" in ihrem aktuellen Arbeitsverzeichnis. In allen Unterverzeichnissen werden Sie Unterverzeichnisse mit dem Namen "CVS" finden. In diesen speichert CVS Informationen über die Verzeichnisse; sie können ignoriert werden, da sie für die Benutzung von CVS nicht von Bedeutung sind. Von nun an, brauchen Sie sich keine Sorgen mehr um die CVSROOT-Umgebungsvariable machen. Sie brauchen auch den "-d"-Teil der Kommandozeile nicht mehr. Diese Informationen befinden sich nun alle in den CVS-Unterverzeichnissen. Also, Sie brauchen die CVSROOT-Variablen wirklich nur für den ersten Login und das erste Checkout.

## 1.16 Quelltexte aktualisieren

Jetzt haben Sie die Quelltexte. Sie können diese kompilieren, installieren, einsehen oder was auch immer Sie damit anstellen möchten.

Hin und wieder wird es aber vorkommen, dass Sie diese Quelltexte auf den neusten Stand bringen möchten. Dazu müssen Sie sich nicht wieder am pserver einloggen. Ihre Informationen sind ja noch im CVS Unterverzeichnis gespeichert. Gehen sie also zunächst in das Hauptverzeichnis des Moduls, das Sie ausgecheckt haben, geben Sie dann ein:

## Befehlsauflistung 8

```
# cvs update -dP
```

## 1.17 Ein genauerer Blick auf "cvs update", Teil 1

Wenn es neue Dateien gibt, gibt CVS für jede Datei "U [Pfad]" aus, während die Datei gespeichert wird. Außerdem werden Sie einige Meldungen der Art "? [Pfad]" sehen, wenn Sie das Modul kompiliert haben. Dabei handelt es sich um Objekt-Dateien. CVS erkennt, dass diese Dateien nicht im entfernten CVS-Repository gespeichert sind.

## 1.18 Ein genauerer Blick auf "cvs update", Teil 2

Beachten Sie auch die beiden Befehlszeilenparameter, die wir verwendet haben. "-d" teilt CVS mit, dass neue Verzeichnisse angelegt werden sollen, wenn Sie auf dem entfernten Server eingerichtet worden sind. Dies würde sonst nicht geschehen. "-P" veranlaßt CVS leere Verzeichnisse, die auf dem Server gelöscht wurden, auch im lokalen System zu löschen. CVS neigt sonst dazu viele nicht mehr benutzte Verzeichnisse anzusammeln.

Wenn Sie nur die neuesten Quelltexte herunterladen möchten, ist das eigentlich schon alles was Sie benötigen. Nun werden wir noch einen Blick darauf werfen, was Sie benötigen, wenn Sie mit CVS als Entwickler umgehen müssen.

## 2. CVS für Entwickler

### 2.1 Dateien modifizieren

Als Entwickler werden Sie Dateien ändern wollen, die mittels CVS organisiert sind. Dazu müssen Sie nur wie gewohnt die lokale Datei auf ihrem System bearbeiten. Ihre Änderungen werden nicht auf dem Server geändert bis Sie CVS ausdrücklich den "commit"-Befehl (commit - einreichen) erteilen. Wenn Sie alle Ihre Änderungen sorgfältig getestet haben, und Sie sich sicher sind, dass alles ordentlich funktioniert, brauchen Sie nur diese beiden Schritte zu befolgen. Zunächst gehen Sie sicher, dass Sie die aktuellsten Quelltexte auf Ihrem lokalen System besitzen. Dazu geben Sie folgenden Befehl ein.

#### Befehlsauflistung 9

```
# cvs update -dP
```

### 2.2 CVS berücksichtigt Änderungen Anderer

Wie Sie bereits gesehen haben, bringt "cvs update" Ihre Quellen auf den neuesten Stand. Aber was passiert mit Ihren Änderungen? Keine Sorge, diese gehen nicht verloren. Wenn ein anderer Entwickler eine Änderung an einer Datei vorgenommen hat, deren lokale Version Sie nicht verändert haben, so wird CVS die Änderung einfach übernehmen.

Wenn Sie die Zeilen 1-10 einer Datei geändert haben, ein zweiter Entwickler gleichzeitig die Zeilen 30-40 geändert hat, und diese vor Ihnen ins CVS Repository eingereicht hat, so wird CVS diese Dateien intelligent in Ihre lokale Kopie einbinden. So verlieren Sie Ihre eigenen Änderungen nicht. Durch dieses sogenannte "merging" können zwei oder mehr Entwickler gleichzeitig an einer Datei arbeiten.

### 2.3 Merging ist nicht perfekt!

Haben nämlich zwei Entwickler gleichzeitig **die selbe Region** einer Datei geändert, so wird es etwas komplizierter. CVS wird Sie dann über einen Konflikt informieren. Auch wenn natürlich keine Arbeit verloren geht, ist etwas manuelle Arbeit erforderlich. CVS braucht jetzt Ihre Eingabe um zu entscheiden wie die Änderungen zu "mergen" sind.

### 2.4 "commit" - Übergeben - Oder, wie man seine Arbeit einreicht

Nun werden wir einen genauen Blick darauf werfen, wie die oben erwähnten Konflikte ausgeräumt werden können. Zunächst werden wir allerdings annehmen, dass keine Konflikte vorliegen, als Sie "cvs update -dP" eingegeben haben. Wenn keine Konflikte durch Ihre lokalen Dateien auftreten und Ihre lokalen Dateien auf dem aktuellen Stand sind, sind Sie bereit Ihre Änderungen an das CVS zu übergeben.

#### Befehlsauflistung 10

```
# cvs commit
```

## 2.5 Was "commit" eigentlich macht

"cvs commit" reicht nicht **nur** Ihre Änderungen ein. Bevor die Änderungen eingereicht werden, wird Ihr Editor gestartet. Hier können Sie nun eine Beschreibung Ihrer Änderungen eingeben. Speichern Sie die Datei und verlassen Sie den Editor, werden Ihre Änderungen und die zugehörigen Kommentare den anderen Entwicklern zugänglich gemacht.

## 2.6 Die Log-Datei ansehen

Es ist sehr einfach sich die komplette Änderungsgeschichte einer Datei sowie die entsprechenden Kommentare anzeigen zu lassen. Um diese Informationen zu sehen, geben Sie folgendes ein:

### Befehlsauflistung 11

```
# cvs log meineDatei.c
```

Der "cvs log" Befehl ist rekursiv. Möchten Sie also die gesamten Änderungen eines ganzen Verzeichnisses anzeigen lassen, so wechseln Sie in das entsprechende Verzeichnis, bevor Sie folgenden Befehl eingeben:

### Befehlsauflistung 12

```
# cvs log | less
```

## 2.7 Optionen des "commit"-Befehls

Sie möchten vielleicht einen bestimmten Editor verwenden, um Ihre Log-Einträge zu machen. Dazu können Sie einfach die "EDITOR"-Umgebungsvariable ändern. Es ist eine gute Idee diese Einstellung mittels Ihrer ~/.bashrc-Datei permanent einzurichten.

### Befehlsauflistung 13

```
export EDITOR=jpico
```

Alternativ zum Verwenden eines Editors können Sie CVS den Log-Eintrag auch direkt über die Kommandozeile mitteilen. Verwenden Sie hierzu die -m-Option.

### Befehlsauflistung 14

```
# cvs commit -m 'Ein paar blöde Fehler beseitigt'
```

## 2.8 Die .cvsrc Datei

Bevor wir uns einige weitere CVS-Befehle ansehen, empfehle ich Ihnen sich eine ~/.cvsrc-Datei anzulegen. Mittels dieser Datei können Sie CVS einige standard Befehlszeilenparameter mitteilen, sodass Sie diese nicht jedes Mal eingeben müssen.

### Befehlsauflistung 15

```
cvs -q  
diff -u -b -B  
checkout -P  
update -d -P
```

## 2.9 Die .cvsrc Datei - Die Erklärung

Die erste Zeile der Datei bringt CVS in den "quiet"-(leise)-Modus. So wird die Ausgabe des "cvs update"-Befehls deutlich lesbarer. Außerdem können Sie sobald sie diese Datei eingerichtet haben, einfach "cvs update" anstatt "cvs update -dP" eingeben.

## 2.10 Eine neue Datei in das Repository einfügen

Es ist sehr einfach eine neue Datei in CVS einzubinden. Zunächst erstellen Sie eine neue Datei wie üblich. Dann geben Sie diesen Befehl ein:

#### **Befehlsauflistung 16**

```
# cvs add meineDatei.c
cvs server: use 'cvs commit' to add this file permanently
```

Die Datei wird beim nächsten Ausführen von "cvs commit" dem Repository hinzugefügt werden. Vorher werden andere Entwickler diese Datei nicht sehen!

## **2.11 Ein Verzeichnis in das Repository hinzufügen**

Der Vorgang zum Hinzufügen eines Verzeichnisses ist ähnlich:

#### **Befehlsauflistung 17**

```
# mkdir foo
# cvs add foo
Directory /home/cvsroot/meinCode/foo added to the repository
```

Anders als bei einer Datei, wird ein neues Verzeichnis sofort für andere Entwickler sichtbar. Es ist also kein "cvs commit" erforderlich. Sobald Sie eine lokale Datei in das neue Verzeichnis speichern, werden Sie sehen, dass ein neues Unterverzeichnis namens "CVS" eingerichtet wird. Dieses Verzeichnis dient als Aufbewahrungsort für die Daten, die notwendig sind, um die Dateien zu verwalten. So läßt sich leicht erkennen, ob sich ein bestimmtes Verzeichnis im CVS Repository befindet, indem man nach einem CVS-Unterverzeichnis sucht.

## **2.12 "cvs add" - Notizen**

Wie Sie sich vielleicht gedacht haben, bevor eine Datei CVS hinzugefügt werden kann, müssen Sie sicherstellen, dass das Verzeichnis im CVS Repository ist. Sonst werden Sie diesen Fehler zu sehen bekommen:

#### **Befehlsauflistung 18**

```
# cvs add meineDatei.c
cvs add: cannot open CVS/Entries for reading: No such file or directory
cvs [add aborted]: no repository
```

## **2.13 Ein genauerer Blick auf "cvs update", Teil 1**

Bevor wir uns genauer ansehen, wie Konflikte gelöst werden, sollten wir uns mit der Ausgabe des "cvs update"-Befehls vertraut machen. Wenn Sie eine ~/.cvsrc-Datei mit der Zeile "cvs -q" eingerichtet haben, wird die Ausgabe deutlich einfacher zu lesen sein. "cvs update" informiert Sie detailliert darüber was es gerade unternimmt, indem es einen Buchstaben ausgibt, gefolgt von einem Leerzeichen und dem Namen einer Datei. Zum Beispiel:

#### **Befehlsauflistung 19**

```
# cvs update -dP
? distfiles
? packages
? profiles
```

## **2.14 Ein genauerer Blick auf "cvs update", Teil 2**

"cvs update" verwendet das Fragezeichen, um Ihnen mitzuteilen, dass es nichts über diese Datei weiß. Diese Dateien und Verzeichnisse befinden sich nur in dem lokalen Verzeichnis, und sind nicht Teil des offiziellen Repositories. Und sie wurden auch nicht mit dem "add"-Befehl hinzugefügt. Hier ist eine Liste mit weiteren Buchstaben und deren Bedeutung:

#### **Befehlsauflistung 20**

U [Pfad]

Wird ausgegeben, wenn eine neue Datei oder ein neues Verzeichnis in ihrem lokalen Verzeichnis erzeugt wird, oder wenn eine Datei, die Sie nicht geändert haben, aktualisiert (U wie Update) wird.

#### **Befehlsauflistung 21**

A [Pfad]

Diese Datei ist vorgesehen in das Repository hinzugefügt zu werden, und wird beim nächsten "cvs commit" aufgenommen.

## **2.15 Ein genauerer Blick auf "cvs update", Teil 3**

#### **Befehlsauflistung 22**

R [Pfad]

Wie "A", sagt ein "R" aus, dass diese Datei für die Löschung vorgesehen ist. Also wird die Datei gelöscht, wenn Sie das nächste Mal "cvs commit" ausführen.

#### **Befehlsauflistung 23**

M [Pfad]

Das "M" bedeutet, dass diese Datei von Ihnen modifiziert worden ist. Außerdem kann es bedeuten, dass neue Änderungen vom Repository in die lokale Datei eingebunden wurden.

#### **Befehlsauflistung 24**

C [Pfad]

Der Buchstabe "C" bedeutet, dass ein Konflikt vorliegt. Manuelles einbinden der Änderungen wird nötig sein, bevor Sie "cvs commit" ausführen können.

## **2.16 Konflikte auflösen**

Nun werden wir uns ansehen wie man einen Konflikt ausräumt. Ich bin sehr in das Gentoo Linux Projekt eingebunden, dort haben wir unsere eigenen CVS-Server unter cvs.gentoo.org. Wir Entwickler verbringen unsere Zeit hauptsächlich damit, die Quellen zu hacken, die im "gentoo-x86"-Modul abgelegt sind. In diesem Modul haben wir eine Datei namens "ChangeLog", die (wie Sie bereits erraten haben) eine Beschreibung der wichtigsten Änderungen beinhaltet.

## **2.17 Ein Beispiel-Konflikt**

Da diese Datei fast jedes Mal geändert wird, wenn ein Entwickler eine Änderung vornimmt, ist sie eine der Hauptquellen für Konflikte. Nehmen wir uns diesen Beispielkonflikt vor: Nehmen wir an, dass ich die folgenden Zeilen hinzugefügt habe.

#### **Befehlsauflistung 25**

```
date 25 Feb 2001
```

```
This is the thing I added myself
```

Nehmen wir ferner an, dass bevor ich die Möglichkeit hatte "cvs commit" durchzuführen, jemand anders diese Änderung vorgenommen hat.

#### **Befehlsauflistung 26**

```
date 25 Feb 2001
```

```
This is the part added by another developer
```

## 2.18 Ein Beispielkonflikt - Teil 2

Wenn ich nun "cvs update -dP" ausführe (was man immer tun sollte, vor einem "commit"), so erhalte ich eine Konfliktmeldung, da CVS nicht in der Lage ist meine lokale Änderung selbstständig in das CVS Repository einzufügen.

### Befehlsauflistung 27

```
RCS file: /home/cvsroot/gentoo-x86/ChangeLog,v
retrieving revision 1.362
retrieving revision 1.363
Merging differences between 1.362 and 1.363 into ChangeLog
rcsmerge: warning: conflicts during merge
cvs server: conflicts found in ChangeLog
C ChangeLog
```

## 2.19 Konfliktlösung

Argh, ein Konflikt! Zum Glück ist die Lösung einfach. Ich starte meinen Texteditor, und sehe den Text am Anfang der Datei ChangeLog.

### Befehlsauflistung 28

```
<<<<<<< ChangeLog
date 25 Feb 2001

This is the thing I added myself

=====
date 25 Feb 2001

This is the part added by another developer

>>>>>>> 1.363
```

## 2.20 Konfliktlösung, Teil 2

Anstatt eine der beiden Versionen zu bevorzugen, hat CVS einfach beide Versionen zusammengefügt, und mittels eindeutiger Zeichen erkenntlich gemacht, wo der Konflikt liegt. Nun ist es an mir, die entsprechenden Regionen durch den Text zu ersetzen, der wirklich dort stehen sollte. In diesem Fall, ist der zu ersetzende Text weder der eine noch der andere, beide sollen hintereinander in der Datei stehen.

### Befehlsauflistung 29

```
date 25 Feb 2001

This is the thing I added myself

This is the part added by another developer
```

Nun habe ich die Konfliktregion in der Datei mit dem entsprechenden Text ersetzt (und die "=====" gelöscht). Jetzt kann ich die Änderung mittels "cvs commit" problemlos übertragen.

## 2.21 Einige Tips zur Konfliktlösung

Wann immer Sie eine Datei editieren in der Konflikte aufgetreten sind, gehen Sie die gesamte Datei nach den Konfliktzeichen durch, so dass Sie alle Konfliktregionen bearbeiten! Wenn Sie einen Konflikt übersehen, wird CVS Ihnen nicht erlauben, die Änderungen zu übertragen. Es ist daher offensichtlich sehr wichtig alle Konfliktzeichen ("=====") zu löschen. Falls Sie einen Fehler beim Editieren der Datei machen, und dann Ihre Änderung speichern, können Sie immernoch eine Originalversion Ihrer Datei finden. Diese trägt den Namen ".#Dateiname.Version".

## 2.22 Eine Datei löschen

### Befehlsauflistung 30

```
# rm meine_alte_datei.c
# cvs remove meine_alte_datei.c
```

## 2.23 Eine Datei löschen, weitergeführt

Die Datei wird durch den Befehl fürs Löschen vorgesehen, wenn Sie das nächste Mal ein "cvs commit" durchführen. Sobald Sie ein "commit" durchgeführt haben, wird die Datei offiziell vom CVS Repository gelöscht sein. CVS wird die Datei jedoch nicht einfach "wegwerfen", sondern vielmehr alle Daten im Zusammenhang mit dieser Datei behalten. Sowohl Inhalt, als auch Geschichte der Änderungen. Dies ist wieder ein Beispiel, wie CVS Ihren wertvollen Quelltext beschützt.

"cvs remove" ist rekursiv. Das bedeutet, dass Sie mehrere Dateien gleichzeitig löschen können, und dann "cvs remove" von einem Übergeordneten Verzeichnis aus aufrufen können, mit keinem weiteren Argument. Wenn Sie dies tun, werden alle Dateien als "zu-löschen" gekennzeichnet, bis Sie das nächste Mal "cvs commit" aufrufen. Dann werden alle gelöscht.

## 2.24 Ein Verzeichnis löschen

Wenn Sie ein gesamtes Verzeichnis löschen möchten, empfehle ich diese Vorgehensweise. Zunächst löschen Sie alle Dateien und entfernen Sie sie auch aus dem CVS Repository mittels "cvs remove", wie oben beschrieben.

### Befehlsauflistung 31

```
# rm *.c
# cvs remove
```

## 2.25 Ein Verzeichnis löschen, weitergeführt

Dann übergeben Sie Ihre Änderungen an das CVS Repository.

### Befehlsauflistung 32

```
# cvs commit
```

Nun folgt der Trick. Folgen Sie diesen Schritten, um das Verzeichnis wirklich zu löschen:

### Befehlsauflistung 33

```
# cd ..
# cvs remove mein_Verzeichnis
# rm -rf mein_Verzeichnis
```

Bemerken Sie, dass das Löschen des Verzeichnisses keinen weiteren Aufruf von "commit" benötigt. Verzeichnisse werden in Echtzeit dem Repository hinzugefügt beziehungsweise gelöscht.

## 2.26 Das wars!

Das war Ihre Einführung in CVS. Ich hoffe, dass diese Tutorial Ihnen hilfreich war. Natürlich hat CVS eine Menge mehr Funktionen als die hier vorgestellten, aber dank der umfassenden Ressourcen zum Thema CVS können Sie sich nach Bedarf weiter informieren.

- <http://www.cvshome.org> ist die Heimat der CVS Entwicklung. Dort finden Sie auch eine Menge an Dokumentationen zum Thema CVS, wie etwa [die offizielle CVS-Online-Dokumentation](#)
- Die [CVS VersionControl for Web Site Projects Seite](#) beinhaltet gute Informationen über CVS als Tool für die Entwicklung von Webprojekten.
- Karl Fogel hat ein Buch namens [Open Source Development with CVS](#) geschrieben. Einige Kapitel des Buches sind kostenlos online abrufbar.
- [cvsweb](#) ist ein wirklich gutes CGI Skript, das ein Web-Interface für Ihr CVS Repository zur Verfügung stellt. Sehr gut fürs Browsen.

## 2.27 Über dieses Dokument

Die ursprüngliche Version dieses Artikels wurde zuerst bei IBM developerWorks veröffentlicht und ist Eigentum von Westtech Information Services. Dieses Dokument ist eine erneuerte Version des Original-Artikels und enthält verschiedene Verbesserungen durch das Gentoo Linux Documentation Team.



# >> Gentoo Linux Entwickler HOWTO

[Bitte Kapitel auswählen]

## 1. Der Portage Verzeichnisbaum

### 1.1 Einführung

Der Portage Verzeichnisbaum ist normalerweise unter `/usr/portage` zu finden und ist hierarchisch strukturiert bestehend aus Kategorien gefolgt von Paket-Verzeichnissen. Hier ein Beispiel: Die `util-linux-2.11g.ebuild` Datei ist unter `/usr/portage/sys-apps/util-linux` zu finden. In diesem Verzeichnis können sich mehrere verschiedene Versionen neben dem `util-linux-2.11g.ebuild` befinden. Alle unterschiedlichen Pakete teilen sich unabhängig von ihrer Version dasselbe kategorie/paket Verzeichnis in `/usr/portage`.

### 1.2 Den Portage Verzeichnisbaum mit CVS auschecken

Falls CVS noch unbekannt für Sie ist, gibt es das [CVS Tutorial](#). Der Portage Verzeichnisbaum befindet sich im `gentoo-x86` Paket des Gento Linux Verzeichnisbaumes. Um das ganze (doch recht grosse) Paket auszuchecken, sollte mit Hilfe obiger Anleitung CVS eingerichtet werden und dann der `gentoo-x86` Verzeichnisbaum ausgecheckt werden.

### 1.3 Was gehört nicht in den Portage Verzeichnisbaum?

Befor ein ebuild geschrieben wird, sollte in [bugs.gentoo.org](#) nachgeschaut werden, ob sich nicht jemand anderes bereits die Mühe gemacht hat, das ebuild zu schreiben, welches aber noch nicht in den offiziellen Portage Verzeichnisbaum übernommen wurde. Dabei sollte wie folgt vorgegangen werden: Auf der [bugs.gentoo.org](#) "query" auswählen, als Produkt "Gentoo Linux" markieren, als Component "ebuild" auswählen. Im Suchfeld dann den Namen des ebuilds eingeben, sowie als Status NEW, ASSIGNED, REOPENED und (wichtig!) RESOLVED markieren, dann die Suchabfrage starten.

Generell sollte im Portage Verzeichnis nur die `.ebuild` Dateien und sehr kleine zugehörige Dateien, wie zum Beispiel Patches oder Beispielskonfigurationen gespeichert werden. Diese Dateien sollten immer unter `/usr/portage/kategorie/paket/files` abgespeichert werden, um das kategorie/paket Verzeichnis nicht zu unübersichtlich machen. Generell ist es für Entwickler keine gute Idee Binärdateien (Nicht-ASCII-Dateien) im CVS abzulegen. Wenn dies dennoch notwendig sein sollte (zum Beispiel eine PNG Grafik aus welchem Grund auch immer), dann sollte sie mit der `-kb` option in das CVS eingebracht werden:

**Befehlsauflistung 1:** Code Auflistung 1

```
# cvs add -kb myphoto.png
```

Die `-kb` option teilt CVS mit, dass `myphoto.png` eine Binärdatei ist und einer besonderen Behandlung bedarf. Zum Beispiel das Zusammenführen von zwei verschiedenen Binärdateien wird somit nicht erlaubt, was hier auch naheliegend ist. Wenn wir schon beim Zusammenführen sind, alle Patches die dem Portage Verzeichnisbaum hinzugefügt werden, sollten nicht komprimiert sein. Dies erlaubt dann CVS Veränderungen einzubringen und die Entwickler von dabei möglicherweise entstehenden Konflikten korrekt zu warnen.

Nochmals zur Erinnerung: Die Pakete, die in CVS als stable eingchecked werden, sollten wirklich ohne Einschränkungen funktionieren. Es sollte sichergestellt werden, dass ein guter Satz an Standardeinstellungen gewählt wurde, der auf der Mehrheit aller Systeme laufen wird und somit die Benutzer zufrieden stellt. Wenn das Paket Probleme bereitet, oder Sie nicht sicher sind, wie es zum Laufen gebracht werden kann, hilft sicher ein Blick auf andere Distributionen, welche meistens ihre eigene Fassung des Paketes besitzen. Gute Beispiele sind hier [Mandrake](#) oder [Debian](#).

Beim Einchecken der ebuilds im CVS sollten alle Entwickler `repoman commit` statt `cvs commit` verwenden. Vor Ausführung des `commit` sollte mit **lintool** das Digest, der Changelog und das Ebuild selbst verifiziert werden.

### 1.4 Allgemeine CVS Commit Regeln

#### Warnung

Achtung **lintool** hat einige Probleme. Es sollte stattdessen **repoman** verwendet werden.

- Vor einem commit immer `repoman` ausführen
- Bitte `lintool` vor dem commit ausführen
- Immer testen, ob `package.mask` in Ordnung ist, indem mit `'emerge --pretend glibc'` mögliche Konflikte angezeigt werden
- Immer das `ChangeLog` vor dem commit aktualisieren!
- Um sicherzugehen, dass Konflikte während dem CVS commit keine schwerwiegenden Auswirkungen auf den Portage Verzeichnisbaum auftreten, sollte immer vor dem commit des eigentlichen `ebuilds`, das aktualisierte `package.mask` eincheckt werden.
- Bitte sauber einchecken! Zuerst das aktualisierte `package.mask`, dann das `ebuild`, `ChangeLog` und die Lizenz in einem Zug, da ansonsten die Benutzer bei ihren Installationen Probleme bekommen.

## 1.5 Das files Verzeichnis

Wie vorher bereits beschrieben, existiert in jedem paket Unterverzeichnis das Verzeichnis `files`. Alle Patches, Konfigurationsdateien oder andere Dateien, die von diesem Paket gebraucht werden, gehören hier hinein. Selbst erstellte Patches sollten mit einem Versions-Namen versehen werden, wie zum beispiel `meinpaket-1.0-gentoo.diff`. Die Gentoo Erweiterung im Namen informiert die Benutzer, dass dieser Patch durch uns, den Gentoo Entwicklern, erstellt wurde und nicht aus einer Mailingliste oder sonst wo gezogen wurde. Auch hier sollten die `diffs` nicht komprimiert werden, da CVS mit Binärdateien nicht so gut umgehen kann.

Es sollte immer ein Suffix, wie z.B. `mypkg-1.0` an das Ende jeder Datei gesetzt werden, die in das `files` Verzeichnis gehören, um die einzelnen Dateien unmissverständlich den einzelnen `ebuild` Skripten zuzuordnen können und die Unterschiede zwischen den einzelnen `diffs` sichtbar werden. Dies ist generell eine gute Sache :-). Natürlich kann, wenn genauere Unterscheidung benötigt wird, ein anderes Suffix verwendet werden.

Bei einer grösseren Anzahl von Dateien empfiehlt es sich ein Unterverzeichnis wie z.B. `files/mypkg-1.0` anzulegen und die jeweiligen zugehörigen Dateien hier hineinzukopieren. Bei dieser Methode kann man natürlich auf das Suffix bei den einzelnen Dateien verzichten, da dies bereits aus dem Verzeichnisnamen hervorgeht. Das spart einiges an Arbeit.

## 2. Die ebuild Skripte

### 2.1 Einführung

Die `ebuild` Skripte bilden die Grundlage des gesamten Portage Systems. Sie enthalten die gesamte Information zum Herunterladen, Entpacken, Kompilieren und Installieren des Quellcodes, aber auch eventuelle Änderungen und Konfigurationen, die vor oder nach der Installation oder beim Entfernen durchgeführt werden. Während das meiste von Portage in Python geschrieben ist, sind die `ebuilds` als `bash`-Skripte gehalten, da dies das Aufrufen von Befehlszeilen wie auf der Kommandozeile ermöglicht. Eines der Hauptdesign-Prinzipien der `ebuild` Skripte ist, dass die darin aufgerufenen Befehle denen der Kommandozeile entsprechen, wie wenn der Benutzer das Paket manuell installieren würde. Aus diesem Grund ist die `bash`-Syntax eine gute Wahl.

`Ebuild` Skripte werden durch die `ebuild` und `emerge` Befehle ausgeführt. Man muss sich das `ebuild` Kommando als einfaches Handwerkzeug vorstellen. Es kann ein `ebuild` erstellen und installieren, aber mehr auch nicht. Es kontrolliert, ob Abhängigkeiten erfüllt sind, kann diese aber nicht selbständig auflösen. Auf der anderen Seite ist `emerge` als hochspezialisierte Zentrale für `ebuild`, welches die Möglichkeit besitzt, selbständig abhängige Pakete zu installieren, wenn gewünscht als "was wäre wenn..." dem Benutzer anzeigen, welche `ebuilds` eingeführt **würden** und vieles mehr. Allgemein sticht `emerge` den `ebuild` Befehl in allen Belangen aus, bis auf einen Punkt. Mit `ebuild` kann der Benutzer inkrementell alle unterschiedlichen Bereiche einer Paketinstallation (herunterladen, entpacken, kompilieren, installieren und einführen) Schritt für Schritt ausführen. Für Entwickler ist dies ein unverzichtbares Werkzeug zum debuggen, da nur so Probleme mit dem `ebuild` auf einen kleinen Bereich des Gesamtprozesses eingegrenzt werden können.

### 2.2 Die Benennung von ebuild Dateien

Jeder `Ebuild` Dateiname besteht aus vier Abschnitten:

Der erste Abschnitt ist der Paketname, der nur aus Kleinbuchstaben, den Zahlen 0-9 und dem Bindestrich ('-') enthalten darf. Beispiele sind: `util-linux`, `sysklogd` und `glibc`.

Der zweite Abschnitt ist die Versionsnummer des Paketes, welche normalerweise mit der Versionsnummer des Quellcode Tarballs übereinstimmt. Diese besteht normalerweise aus zwei oder drei Nummern, die durch Punkte von einander getrennt sind, wie zum Beispiel `1.2` oder `4.5.2` (sehr

lange, durch Punkte voneinander getrennte Nummern werden auch unterstützt) und darf von einem einfachen Buchstaben auf die letzte Zahl gefolgt werden, zum Beispiel: *1.4b* oder *2.6h*. Die Paketnummer wird mit der Versionsnummer durch einen Bindestrich verbunden: *foo-1.0*, *bar-2.4.6*, etc.

### Wichtig

Wenn Sie mit dem Gedanken spielen, einen angehängten Buchstaben in Ihrer Versionsnummer zu verwenden, sollten Sie darüber im klaren sein, dass dieser Buchstabe **nicht** dazu verwendet werden sollte, einen eventuellen Alpha oder Beta Status anzuzeigen, da alphas und betas als **preleases** zählen, angehängte Buchstaben jedoch als **neuere Versionen**. Das ist eine grosse Unterscheidung, da Portage die Versionsnummer eines ebuilds dazu verwendet, herauszufinden, ob es neuer oder älter ist als ein Paket mit gleichem Namen aus derselben Kategorie. Es ist sehr wichtig, dass Versionsnummern glaubwürdig die Version eines Paketes repräsentieren, damit Portage seine Abhängigkeits-Prüfungen korrekt ausführt..

Der dritte Abschnitt (optional) enthält eine spezielle Suffix, entweder *\_alpha*, *\_beta*, *\_pre* oder *\_rc*. Alle diese Suffixe werden durch eine Nummer ergänzt, zum Beispiel *linux-2.4.0\_pre10*. Bei identischen Versionsnummern geht Portage davon aus, dass *\_alpha* älter als *\_beta* ist, *\_beta* älter als *\_pre* und *\_pre* älter als *\_rc* ist.

### Notiz

Ein *\_rc* Paket ist älter als ein Paket ohne vorausgehende Suffixe mit Unterstrich (z.B. *linux-2.4.0*) und *linux-2.4.0* ist älter als ein Paket mit einem Buchstaben Präfix, hier *linux-2.4.0b*. Nochmal: Diese Versionsinformationen sind wichtig, da Portage es zur Bestimmung heranzieht, ob ein Paket oder ebuild älter oder jünger ist, als ein Paket aus der selben Kategorie mit gleichem Namen.

Der vierte Abschnitt (ebenfalls optional) ist die Gentoo Linux spezifische **Revisionsnummer**, welche durch *-r#* angegeben wird, wobei *#* ein Integer (Zahlenwert) ist, als Beispiel: *package-4.5.3-r3*. Die Revisionsnummer ist unabhängig von der Versionsnummer des Quellpaketes und kann dazu genutzt werden, die Benutzer davon in Kenntnis zu setzen, dass es sich um ein neues bzw. erweiterte Revision eines ebuilds handelt.

Wenn grössere Veränderungen an einem bestehenden ebuild vorgenommen werden, sollte die Datei vorher kopiert und mit einer um 1 erhöhten Revisionsnummer abgespeichert werden. Die ersten Fassungen besitzen normalerweise keine Revisionsnummer, z.B. *package-4.5.3*, da sie von Portage so betrachtet werden, als ob sie eine Revisionsnummer von Null haben. Das bedeutet für die Zählweise: *1.0* (erste Version), *1.0-r1*, *1.0-r2*, und so weiter.

Und natürlich gehen wir davon aus, dass der **fünfte** Abschnitt der ebuild Bezeichnung nicht vergessen wird, es ist die *.ebuild* Endung.

## 2.3 Aufbau einer ebuild Datei

### 1. Setzen der Variablen:

Der Anfang jeder ebuild Datei besteht aus einigen Variablen, die wie folgt gesetzt werden können:

|                    |  |
|--------------------|--|
| <i>P</i>           | Der Name und die Versionsnummer des Paketes, dies muss normalerweise nicht gesetzt werden, da es von Portage durch den Dateinamen bestimmt wird  |
| <i>A</i>           | Der Name ohne Pfadangabe von dem Haupt-Quellcode des Paketes   |
| <i>S</i>           | Das Quellcode-Verzeichnis für das Paket, im Normalfall <code>\${WORKDIR}/\${P}</code>  |
| <i>DESCRIPTION</i> | Eine kurze Beschreibung des Paketes in einem Satz  |
| <i>SRC_URI</i>     | sind die URIs für jede Quellcode-Datei in diesem Paket, welche durch Leerzeichen getrennt werden. Das Erste ist normalerweise so etwas wie: <b>ftp://ftp.company.com/pub/somepackage/\${A}</b> |
| <i>HOME PAGE</i>   | ist die Homepage des Pakets  |
| <i>DEPEND</i>      | build Abhängigkeiten, siehe dazu den Abschnitt <a href="#">Package Dependencies</a>  |
| <i>RDEPEND</i>     | runtime Abhängigkeiten. mehr dazu im Abschnitt <a href="#">Package Dependencies</a>  |

### 2. ebuild Funktionen

Es gibt verschiedene Funktionen, die ein den ebuild Dateien definiert werden können, um den Prozess der Zusammenstellung und Installation des Paketes steuern zu können.

|                    |  |
|--------------------|--|
| <i>pkg_setup</i>   | mit dieser Funktion können vorbereitende Schritte durchgeführt werden. Dazu gehört die Kontrolle des Benutzerkontos oder die Kontrolle nach einer existierenden Konfigurationsdatei. Diese Funktion muss den Wert 0 zurückgeben, damit das ebuild fortgesetzt werden kann. |
| <i>pkg_nofetch</i> | Informiert den Benutzer über Aktionen oder Downloads, die er selbst ausführen muss (z.B. aus Lizenzierungsgründen).  |

Mit dieser Funktion kann der Quellcode entpackt werden und autoconf/automake/etc. ausgeführt werden. Standardmässig wird das Paket in  $\${A}$  entpackt. Das Standard-Startverzeichnis ist  $\${WORKDIR}$ .

*src\_unpack*  $\${S}$

*src\_compile* Damit wird das Paket konfiguriert und kompiliert. Das Standard-Startverzeichnis ist  $\${S}$ . Mit dieser Funktion wird das Paket in  $\${D}$  installiert. Benutzt das Paket automake, kann dies mit *src\_install* vereinfacht werden. **Es sollte sichergestellt werden, dass die Installation aller Dateien in  $\${D}$  als root ausgeführt wird!**

*pkg\_preinst* Die Befehle in dieser Funktion werden vor dem mergen der Dateien ausgeführt.

*pkg\_postinst* Die Befehle in dieser Funktion werden nach dem mergen der Dateien ausgeführt.

*pkg\_prerm* Die Befehle in dieser Funktion werden vor dem Entfernen der Dateien ausgeführt.

*pkg\_postrm* Die Befehle in dieser Funktion werden nach dem Entfernen der Dateien ausgeführt.

Mit dieser Funktion kann die initiale Konfiguration des Paketes nach der Installation ausgeführt werden. Alle Pfade in dieser Funktion sollen mit dem Prefix  $\${ROOT}$  versehen werden. Diese Funktion wird **nur** ausgeführt, wenn der Benutzer folgenden Befehl ausführt: *ebuild /var/db/pkg/ $\${CATEGORY}$ / $\${PF}$ / $\${PF}$ .ebuild config*.

*pkg\_config*

Folgende Funktionen können ebenfalls im ebuild verwendet werden:

*use* Kontrolle ob, eine oder mehrere gegebenen USE-Flags gesetzt sind. Ist dies der Fall, wird die Funktion die USE-Flag zurückgeben. Zur Kontrolle der Existenz einer solchen Flag kann `[ -z "`use foobar`" ]` ausgeführt werden.

*has\_version* Gibt 1 zurück, wenn auf dem System die geforderte version installiert ist. Als Beispiel: *has\_version >=glibc-2.3.0*.

*best\_version* Gibt **category/package-version** des jeweiligen Pakets zurück. Beispiel: *best\_version x11-libs/gtk+extra*.

*use\_with* Diese Funktion prüft ob eine USE-Flag gesetzt wurde und gibt entsprechend "--with-foobar" oder "--without-foobar" zurück. Bei einem Argument ist dieses sowohl der USE-Flag und der with(out) String. Andernfalls ist das erste Argument der USE-Flag und das zweite Argument der with(out) String. Beispiel: *use\_with truetype freetype*

*use\_enable* Hat die gleiche Funktion wie *use\_with*, gibt jedoch "--enable foobar" oder "--disable foobar" zurück. Prüft ob Portage die Kernel-Versionnummer erkennt. Ist dies nicht der Fall wird eine Fehlermeldung ausgegeben und das ebuild abgebrochen. Wenn im Script die Versionsnummer geprüft werden soll, kann  $\${KV}$  verwendet werden, welche automatisch durch Portage definiert wird.

*check\_KV* Erstellt ein .keep Datei im gegebenen Verzeichnis, so dass dies nicht automatisch von Portage wieder entfernt werden kann.

*keepdir* Führt *./configure* mit den notwendigen Pfadänderungen (Prefix, host, mandir, infodir, datadir, sysconfdir, localstatedir) durch. Optionale Argumente für *./configure* können gesetzt werden.

*econf* Führt ein *make install* durch, sodass Portage weiss, wohin die Dateien installiert werden.

*install* Bricht den aktuellen Prozess ab. Das übergebene Argument wird dem Benutzer als Nachricht dargestellt.

*die* Kann verwendet werden, um den Benutzer über wichtige Dinge zu informieren. Das Argument das an *info* übergeben wird, ist die Nachricht, die der Benutzer erhält.

*info*

## 2.4 Regeln zum Schreiben eines ebuilds

Da ebuilds in Wirklichkeit nur aus einfachen Shell Skripten bestehen, sollte der zu bearbeitende Editor auf diesen Modus eingestellt werden. Korrektes Einrücken sollte beachtet werden, nur Tab-Sprünge und keine Leerzeichen. Am besten den Editor so einstellen, dass die Tabulatorsprünge alle vier Leerzeichen gesetzt sind. Die Klammern um die Umgebungsvariablen sollten nicht vergessen werden; d.h.  $\${P}$  anstatt nur  $P$ .

Längere Zeilen sollten mit '\ ' umgebrochen werden:

**Befehlsauflistung 2:** Code Auflistung 2.1

```
./configure \
--prefix=/usr || die "configure failed"
```

Für weitere Details kann man einen Blick in **skel.ebuild** werfen, dies befindet sich in /usr/portage.

Bei der Benutzung von Vim kann folgender Code am Ende der .vimrc Datei eingetragen werden, damit sind automatisch alle Einstellungen für Gentoo-Dokumente gesetzt.

**Befehlsauflistung 3:** Code Auflistung 2.2

```
if (getcwd() =~ 'gentoo-x86\|gentoo-src\|portage')
set tabstop=4 shiftwidth=4 noexpandtab
endif
```

Bei der Benutzung von emacs kann in das .emacsrc (GNU Emacs) oder in die init.el (XEmacs) eingetragen werden.

**Befehlsauflistung 4:** Code Auflistung 2.3

```
(defun ebuild-mode ()
  (shell-script-mode)
  (sh-set-shell "bash")
  (make-local-variable 'tab-width)
  (setq tab-width 4)
  (setq auto-mode-alist (cons '("\\.ebuild\\\\" . ebuild-mode) auto-mode-alist))
  (setq auto-mode-alist (cons '("\\.eclass\\\\" . ebuild-mode) auto-mode-alist))
```

## 2.5 Die USE Variablen

Die USE-Variablen dienen dem Zweck, Portage global zu konfigurieren und automatisch bestimmte Kompilierungsoptionen an- oder abzuschalten. Hier ist ein Beispiel. Wir gehen nun davon aus, dass Sie ein GNOME-Fan sind, und jedes ebuild, welches eine optionale GNOME Unterstützung anbietet, soll dies automatisch miteinkompilieren. In diesem Fall fügen wir *gnome* in die USE variable von **/etc/make.conf** hinzu und Portage berücksichtigt dies bei jeder Kompilierung eines Paketes mit GNOME Unterstützung. Im umgekehrten Fall, wenn optionale GNOME Funktionen nicht gewünscht sind, editiert man */etc/make.conf* und stellt sicher, dass *gnome* in der USE Variable nicht gesetzt ist. Gentoo Linux besitzt eine überwältigende Anzahl von USE Optionen, die es Ihnen erlauben, das System genau so zu konfigurieren, wie Sie es haben möchten.

### Notiz

Wenn eine USE Variable deaktiviert ist (hier im Beispiel *gnome*) wirkt sich dies nur auf die **optionalen** Kompilierungsoptionen aus. Wenn Sie jedoch ein ebuild *emerge*, welches auf ein GNOME Paket angewiesen ist, wird GNOME automatisch als eine Abhängigkeit hiervon mitinstalliert, falls es nicht bereits erfolgt ist. Das ist auch der Grund, weshalb man vor dem eigentlichen *emerge* ein *emerge --pretend* ausführen sollte. In so einem Fall wissen Sie bereits vorher Bescheid, was alles auf Ihrem System installiert wird.

In Ihren eigenen ebuids kann mit Hilfe der USE Variable geprüft werden, ob die entsprechende Variable vom Besitzer der Installation bereits gesetzt ist. Das *use* Kommando gibt den Namen jeder Variable zurück, die USE und seiner Befehlszeile präsent ist. Der Befehl wird normalerweise so angewendet:

```
if [ "`use X`" ]; then commands; fi
```

USE Variablen können auch dazu verwendet werden, bestimmte Abhängigkeiten zu setzen. In unserem Beispiel wollen wir ein bestimmtes Paket nur dann als erforderlich markieren, wenn eine bestimmte USE Variable gesetzt ist. Dies kann mit folgender Syntax durchgeführt werden: *variable? (mycat/mypackage-1.0-r1)* in der DEPEND Zeile Ihres ebuids. In diesem Fall wird *mycat/mypackage-1.0-r1* nur dann erforderlich, wenn die entsprechende Variable in den USE vorhanden ist. Umgekehrt ist es auch möglich, festzulegen, welche weitere Abhängigkeit verwendet werden soll, wenn eine USE-Flag **nicht** gesetzt ist: *variable? (mycat/mypackage-1.0-r1) : (othercat/otherpackage-1.0-r5)*. In diesem Fall wird dann das Paket *othercat/otherpackage-1.0-r5* anstatt von *mycat/mypackage-1.0-r1* installiert. Soll ein Abhängigkeit nur dann installiert werden, wenn die Variable nicht vorhanden ist, verwendet man *!variable? (mycat/mypackage-1.0-r1)*. Es sollte sichergestellt werden, dass nur die oben genannte Syntax verwendet wird und nicht die Bash eigene "ifs". Bash Bedingungen stören Portages Abhängigkeits-Cache, sodass die Verwendung dieser zu einem funktionsunfähigem ebuild führen wird.

Hier noch ein wichtiger Tipp zur Verwendung von *USE*. In dem meisten Fällen besitzt das Paket bereits ein *./configure* Skript um das Paket zu konfigurieren. Optionale Bestandteile werden durch Übergabe bestimmter Argumente zur Laufzeit mitkompiliert. Zur Übergabe der USE-Flags an das Konfigurationskript geht man am besten wie folgt vor: Zunächst sollte herausgefunden werden, ob eine bestimmte *./configure* Option standardmässig ein- oder abgeschaltet ist.

**Befehlsauflistung 5:** Code Auflistung 2.4

```
DEPEND="X? ( >=x11-base/xfree-4.3 )
        mysql? ( >=dev-db/mysql-3.23.49 )
        apache2? ( >=net-www/apache-2 ) : ( =net-www/apache-1.* )"

src_compile() {
  local myconf
  use X || myconf="--disable-x11"
  use mysql || myconf="${myconf} --disable-mysql"
```

```

    ./configure ${myconf} --prefix=/usr --host=${CHOST} || die
    emake || die
}

```

Im obigen Beispiel prüfen wir, ob die X und mysql USE Variablen abgeschaltet sind. Der Abschnitt *use X* || kontrolliert, ob X in der USE Variable vorhanden ist. Ist dies nicht der Fall, so übergibt er das Argument *myconf="--disable-x11"* an das Konfigurationsskript. Es ist nicht notwendig X11 und mysql explizit zu aktivieren, da diese standardmässig bereits eingeschaltet sind. Wenn jedoch eine bestimmte Option standardmässig abgeschaltet ist kann man wie folgt vorgehen:

**Befehlsauflistung 6:** Code Auflistung 2.5

```

DEPEND="X? ( >=x11-base/xfree-4.3 )
        mysql? ( >=dev-db/mysql-3.23.49 )"

src_compile() {
    local myconf
    use X && myconf="--enable-x11"
    use mysql && myconf="${myconf} --enable-mysql"

    ./configure ${myconf} --prefix=/usr --host=${CHOST} || die
    emake || die
}

```

In diesem Beispiel werden X11 und MySQL-Unterstützung nur dann explizit aktiviert, wenn die entsprechende USE Variable vorhanden ist. Der Abschnitt *use mysql &&* prüft, ob mysql in der USE-Flag gesetzt ist, und übergibt dann *myconf="\${myconf} --enable-mysql"*.

Eine kontinuierlich aktualisierte Liste aller USE Variablen findet man [hier](#).

## 3. Aufbau des Dateisystems

### 3.1 Einführung in FHS

Der Aufbau des Gentoo Linux Dateisystems orientiert sich am am FHS Standard, kurz für: **Filesystem Hierarchy Standard**. Eine kurze Beschreibung dieses Standards wir hier gegeben, eine komplette Spezifikation findet man unter <http://www.pathname.com/fhs/> .

#### Notiz

Der /opt Breich wird in Abschnitt 3.12 der FHS Spezifikation beschrieben. Abschnitt 4.4 behandelt das /usr/X11R6 Verzeichnis. KDE und GNOME werden nicht spezifisch besprochen und sind in der aktuellen Fassung des FHS überhaupt nicht berücksichtigt.

### 3.2 Wohin mit meinem Paket in diesem Dateisystem?

Normalerweise wird bei der Verwendung von autoconf und automake das Paket standardmässig korrekt installiert, allerdings mit einigen wenigen Ausnahmen:

- Wenn das Programm nach /bin, /sbin, /usr/bin oder /usr/sbin installiert wird, sollten die dazugehörigen Man-Pages nach /usr/share/man kopiert werden. Dies kann meistens durch Übergabe des Arguments *./configure --mandir=/usr/share/man* im ebuild Skript bewerkstelligt werden.
- GNU Info Dateien gehören nach /usr/share/info, **auch wenn die Info-Dateien X11, GNOME oder KDE-Programme betreffen**. Nochmals: /usr/share/info ist der **einzige** offizielle Platz für GNU Info-Dateien! Leider instellieren die meisten ./configure Skripte die GNU Dateien nach /usr/info, sodass man mit Hilfe des *--infodir=/usr/share/info* Arguments nachhelfen muss.
- Dokumentationen werden in ein Unterverzeichnis von /usr/share/doc installiert. Dies sollte Namen, Version und Revisionsnummer des betreffenden Programmes beinhalten. Das gilt für alle Programme, GNOME, KDE, X11 wie auch der Konsole. Manchmal wird weitere Dokumentation für spezielle Zwecke in die /usr/shareHierarchy eingebunden.
- X11-spezifische Programme und Bibliotheken sollten immer nach /usr installiert werden und nicht direkt nach /usr/X11R6, welches ausschliesslich für das X Window System, Version 11 Release 6 vorbehalten ist. Dies ist wahrscheinlich einer genauere Interpretation der FHS Spezifikation, als es in anderen Distributionen üblich ist.
- GNOME und KDE Programme gehören ebenfalls nach /usr.

### **Wichtig**

Manche Distributionen installieren KDE und GNOME nach /opt. Zur Zeit existiert kein Standard für die Installation von Dateien der Desktop-Umgebungen. Im Interesse einer einfachen und konsistenten Handhabung haben wir uns entschieden, alle KDE und GNOME Pakete nach /usr zu installieren.

Generell sollte das ebuild seine Dateien in den /usr Verzeichnisbaum installieren. Manche Programme können mit oder ohne GNOME; KDE und X11 Bibliotheken kompiliert werden, was für Verwirrung sorgt. Unsere Lösung, alles nach /usr zu installieren, verhindert doppelte Arbeit und unnötige Komplexität für die ebuild Entwickler. Der Pfad, in den die Programmdateien installiert werden, sollte nicht an das Vorhandensein oder Fehlen von USE Variablen geknüpft werden. Somit installieren die ebuilds im Portage Verzeichnisbaum in fast allen Fällen ihre Dateien nach /usr.

### **Notiz**

Das /opt Verzeichnis in Gentoo Linux ist für binäre Pakete reserviert. Als Beispiele können hier mozilla-bin, acroread und realplayer angeführt werden. Die hierhin installierten Pakete benötigen in der Regel eine /etc/env.d/foo Markierungsdatei. Dies dient der Möglichkeit, Pfade und zusätzliche Variablen der Laufzeitumgebung hinzuzufügen können.



## >> Gentoo Stage-Builder Anleitung

[Bitte Kapitel auswählen] ▾

### 1. Einführung

#### 1.1 Einleitung

Diese Anleitung erklärt anhand unseres Stage-Builder Programms *stager*, wie man aktuelle Stages für Gentoo Linux erzeugen kann.

#### 1.2 Den Quelltext holen

Bevor Sie anfangen irgendetwas zu bauen, sollten Sie überprüfen, ob Sie die benötigten Quelltexte und Tools haben. Zu Beginn ist es wichtig, daß Sie das **gentoo-src** Modul aus CVS ausgecheckt haben und dieses aktuell ist. Die Build-Scripte werden häufig verändert. Daher sollten Sie sich vergewissern, dass Ihnen das aktuelle vorliegt. Falls kein Zugang zu den gentoo-src Modulen besteht, so kann ein Snapshot von *stager* von der Adresse <http://dev.gentoo.org/~avenj/stager/> bezogen werden. Zusätzlich benötigten Sie das aktuelle experimentelle Stage1-Tar-Archiv. Dieses kann üblicherweise unter [experimental/x86/stages](#) auf jedem Gentoo-Mirror gefunden werden. Das Tar-Archiv, nach dem Sie Ausschau halten sollten, hat den Dateinamen **stage1-x86-1.4\_preDATUM.tar.bz2**, wobei **DATUM** das aktuellste Datum sein sollte, welches verfügbar ist.

### 2. Der Build Prozess

#### 2.1 Erste Schritte

Zunächst erstellen Sie ein Verzeichnis auf einer möglichst großen Partition, damit dort die Builddateien untergebracht werden können. Als nächstes kopieren Sie das *stager* Verzeichnis aus **gentoo-src** in das neu angelegte Verzeichnis (welches in diesem Dokument als `/usr/src/build` bezeichnet wird). Alternativ extrahieren Sie das Snapshot-Tar-Archiv in das genannte Verzeichnis. Zu guter Letzt erstellen Sie ein Verzeichnis innerhalb von `/usr/src/build` und nennen dieses **stages**. Dorthin wird das experimentelle Stage1-Tar-Archiv entpackt.

**Befehlsauflistung 1:** Vorbereitungen

```
# mkdir /usr/src/build
// Den oben genannten Pfad können Sie beliebig benennen.

# cp -R /gentoo-src/stager/ /usr/src/build
# cd /usr/src/build/stager ; mkdir stages
# mv stage1-x86-1.4_preDATUM.tar.bz2 /usr/src/build/stager/stages
```

#### 2.2 Erstellung

Zunächst müssen wir ein Stage2-Tar-Archiv mit den üblichen Optimierungen erstellen. Dieses Tar-Archiv wird dann dazu verwendet, um neue Stages inklusive einem frischen Stage1 zu bauen.

**Befehlsauflistung 2:** Erstellung von Stage2

```
# ./stager x86 2 Original Ziel
// Das oben genannte "Original" verweist auf das Datum, welches im Namen des experimentellen Tar-
// "Ziel" verweist auf das Datum des neuen Tar-Archivs. Diese Werte sind dementsprechend zu erse
```

Wenn dieser Prozess abgeschlossen ist, sollten Sie ein frisch erstelltes Stage2-Tar-Archiv neben dem originalen Stage1-Tar-Archiv im *stages* Verzeichnis auffinden. Nun kann mit der Erstellung einer neuen Stage1 mit Hilfe der soeben erstellten Stage2 auf dem gleichen Weg begonnen werden:

**Befehlsauflistung 3:** Erstellung eines neuen Stage1

```
# ./stager x86 1 Ziel Ziel
```

Taataa! - somit haben Sie soeben Ihr erstes Stage1 neben der originalen Stage1 und der neuen Stage2 erstellt. Nun können Sie damit beginnen Tar-Archive zu bauen, die für spezielle CPU-Typen optimiert sind. Bitte beachten Sie, dass diese optimierten Tar-Archive **nicht** dazu benutzt werden können andere Stages zu bauen. Nur die normalen Tar-Archive sollten dazu benutzt werden, um andere Stages zu generieren. Um Stages für andere Prozessortypen zu erstellen, ersetzen Sie "x86" mit dem entsprechendem Typ. Führen Sie stager ohne Parameter für eine Liste der Typen aus.

### 3. Schlussbemerkungen

#### 3.1 Kontakte

| <b>Kontakt</b>               | <b>Funktion</b>              |
|------------------------------|------------------------------|
| <a href="#">Jon Portnoy</a>  | Gentoo Linux Release Manager |
| <a href="#">Jared Hudson</a> | Release Qualitätssicherung   |
| John Davis                   | Autor                        |
| <a href="#">Jon Portnoy</a>  | Autor                        |



## >> Ebuilds einreichen

[Bitte Kapitel auswählen] ▾

### 1. Zu Beginn

#### 1.1 Erstellen eines Bugzilla Accounts

Gehen Sie mit Ihrem Browser nach <http://bugs.gentoo.org>. Dies ist die URL der "Bugzilla Bug Tracking" Datenbank. In der Liste finden Sie einen Link namens "Open a new Bugzilla account". Falls Sie noch keinen Account besitzen sollten (z.B. durch das Einreichen eines Bugs), klicken Sie auf diesen Link und erstellen einen neuen Account.

### 2. Ein Ebuild einreichen

#### 2.1 Erstellen eines neuen Bugs

Unten auf der Hauptseite oder auch der Login Seite finden Sie eine gelbe Box. Wählen Sie den *New* Link. Oder Sie gehen einfach direkt zur URL [http://bugs.gentoo.org/enter\\_bug.cgi](http://bugs.gentoo.org/enter_bug.cgi). Beides bringt Sie zur *Enter Bug* Seite.

##### **Notiz**

Im Bugzilla System werden alle Einträge als Bugs bezeichnet. Dies beinhaltet auch eingereichte Ebuilds. Jedem Bug wird eine Tracking ID zugeordnet. Notieren Sie sich diese ID, besonders falls Sie noch andere Ebuilds haben, die von diesem Ebuild abhängig sind.

Bugzilla kann mehrere Produkte betreuen. Bei Gentoo zum Beispiel können diese anderen Produkte Dokumentation oder Werkzeuge zum Administrieren der Website beinhalten. Ebuilds sollten hier immer beim "Produkt" *Gentoo Linux* eingereicht werden.

Durch das Klicken auf das Produkt *Gentoo Linux* wird ein Formular zum Erstellen eines neuen Bugs geöffnet.

#### 2.2 Ausfüllen des Bug Formulars

Das erste Feld heisst *version*. Wenn Sie die genaue Version wissen, können Sie sie hier setzen. Sollten Sie sie nicht kennen, wählen Sie *unspecified*. Als nächstes wählen Sie die Komponente (component). Alle Ebuild Scripts gehören dem Component Typ *Ebuilds* an.

##### **Wichtig**

Ebuilds sollten **immer** zum Component Typ *Ebuild* zugeordnet werden. Kategorien wie *GNOME*, *Libraries* und *Server* mögen so wirken als würden Sie auch für Ebuilds gelten, aber Sie sind nur für das Berichten von Fehlern dieser Pakete gedacht und **nicht** für das Einreichen von Ebuilds.

Die Kategorien *Platform*, *OS* und *Priority* müssen für Ebuilds nicht gesetzt werden.

Das *Severity* Feld sollte auf *enhancement* gesetzt werden.

Lassen Sie das *Assigned To* Feld leer. Wenn Sie jedoch den Status Ihres Ebuilds überwachen möchten, tragen Sie bei *CC* Ihre Email Adresse ein.

Füllen Sie das *Summary* Feld mit einer Kurzen Notiz aus, in welcher Sie beschrieben welches Paket Sie einreichen und ob es ein Update oder eine ganz neue Version ist. Etwas wie folgend ist großartig: *foo-bar-x.y.z.ebuild (New Package)* oder *foo-x.y.z.ebuild (Update)*

Das *Description* Feld sollte so vollständig wie möglich ausgefüllt werden. Bitte denken Sie an die internationalen Entwickler und User, verfassen Sie die Beschreibung bitte auf Englisch! Hier ist ein Beispiel von Bug Nummer 7:

**Befehlsaufistung 1:** Beispiel eines Kommentar zu enem ebuild auf Bugzilla

Hi!

```
Please find attached gramps-0.7.0.ebuild and a patch so that it will
compile against python-2.2
```

gramps is the Genealogical Research and Analysis Management Programming System. It is a very cool gnome/python app. If you are in the US, you can head over to <http://www.familysearch.com/> and you can probably find a GEDCOM file for your family tree to import into gramps. (Mormons sure do like genealogy)

I suggest app-misc/gramps Also, this ebuild depend on Imaging and ReportLab previously submitted

tod

Wenn die Beschreibung komplett ist, klicken Sie auf *Submit*.

## 2.3 Anhängen des Ebuild Scripts

Nach dem Klicken auf *submit* auf der *Enter Bug* Seite, kommen Sie zur *Posting Bug* Seite. Beachten Sie bitte die Bug Nummer.

Zum Ende der Seite hin finden Sie einen Link *Attach a file to this bug*. Klicken Sie auf diesen Link.

Wählen Sie die Datei durch die Benutzung der Textbox oder des *Browse* Dialogs aus. Im *Description* Feld, geben sie bitte den Namen der Ebuild Datei ein, in etwa den, den Sie im *Summary* Abschnitt auf der letzten Seite benutzt haben.

Bitte stellen Sie sicher, das der "Content" Typ auf "text/plain" gestellt ist und der Knopf für *select from list* markiert wurde.

Es werden keine weitem Felder benötigt, klicken Sie *Submit* und Sie haben es geschafft!

## 3. Mehrere Ebuilds

### 3.1 Abhängigkeiten

Im Fall, dass Sie mehrere Ebuilds einreichen sollten und einige von anderen abhängig sind, ist es sehr hilfreich, wenn diese Abhängigkeiten notiert werden. Durch das erneute Ansehen des Bugs, sollten Sie die Möglichkeit haben, eine weitere Notiz zu ergänzen. Bitte ergänzen Sie hier eventuelle Abhängigkeiten.



## >> Gentoo Linux Dokumentationsleitfaden

[Bitte Kapitel auswählen] ▀

### 1. Grundlagen

#### 1.1 Ziele des Gentoo-Guide-XML-Designs

Die Guide-XML-Syntax ist zugleich minimalistisch als auch ausdrucksstark, so dass sie schnell erlernbar ist, gleichzeitig aber alle notwendigen Eigenschaften bereitstellt, die wir für das Erstellen von Internet-Dokumenten brauchen. Die Anzahl der Tags ist auf ein Minimum beschränkt -- nur die die wir brauchen. Dadurch wird es einfacher Dokumente in andere Formate, wie z.B. DocBook, XML/SGML oder HTML umzuwandeln.

Das Ziel ist, dass **Erstellen** und **Umwandeln** von GuideXML Dokumenten zu vereinfachen.

#### 1.2 Weitere Quellen

Wenn Sie planen Dokumentation zu Gentoo beizutragen oder GuideXML testen wollen, lesen Sie bitte den [Dokumentations Entwickler Leitfaden](#) der Tipps und Tricks zur Erstellung von Dokumentation beinhaltet.

### 2. Guide-XML

#### 2.1 Grundstruktur

Jetzt, da Sie wissen, wie man Guide-XML umwandelt, sind Sie so weit, um die GuideXML Syntax zu erlernen. Wir beginnen mit den einleitenden Tags, die in einem Guide-XML-Dokument verwendet werden:

**Befehlsauflistung 1:** Der einleitende Teil eines Guide-XML-Dokuments

```
<?xml version='1.0' encoding="UTF-8"?>
<guide link="Relativer Link zu Ihrem Guide">
<title>Gentoo Linux Dokumentationsleitfaden</title>
<author title="Chief Architect">
  <mail link="drobbins@gentoo.org">Daniel Robbins</mail>
</author>
<author title="Editor">
  <mail link="thomasfl@gentoo.org">Thomas Flavel</mail>
</author>
<author title="Übersetzung">
  <mail link="kontakt@hendrik-brandt.de">Hendrik Brandt</mail>
</author>

<abstract>
Dieser Leitfaden erklärt Ihnen, wie Sie Dokumente mit Hilfe der neuen
minimalistischen Gentoo GuideXML Syntax erstellen können.
Diese Syntax stellt das offizielle Format für Gentoo-Linux-Dokumente dar -
auch dieses Dokument wurde mit Hilfe der Gentoo's Guide-XML erstellt.
Dieser Leitfaden setzt Grundkenntnisse in XML und HTML voraus.
</abstract>

<license/>

<version>1.0</version>
<date>29. März 2001</date>
```

In der ersten Zeile sehen wir den notwendigen Tag der festlegt, dass das Dokument in XML beschrieben ist. Darauf folgt der Tag `<guide>` -- das gesamte Dokument ist von einem `<guide>` `</guide>` -Paar umschlossen. Schließlich gibt es noch den Tag `<title>`, mit dem der Titel für das Dokument festgelegt wird.

Als nächstes kommen wir zu den `<author>`-Tags, welche Informationen über die unterschiedlichen Autoren eines Dokumentes enthalten. Jeder `<author>`-Tag besitzt ein optionales Element `title=`. Dieses wird benutzt, um die Beziehungen der Autoren zum Dokument (Autor, Co-Autor, Bearbeiter,

Übersetzer usw.) zu beschreiben. Im vorliegenden Beispiel sind die Namen der Autoren in einem weiteren Tag eingeschlossen -- dem Tag `<mail>`. Mit diesem kann eine Email-Adresse mit der betreffenden Person verbunden werden. Der Tag `<mail>` ist optional und kann weggelassen werden. Weiterhin wird nicht mehr als ein Element `<author>` pro Dokument benötigt.

Es folgen die Tags `<abstract>`, `<version>` und `<date>`, die benutzt werden, um die Überschrift, die gegenwärtige Versionsnummer und das gegenwärtige Versionsdatum (im Format TT MMM JJJJ) des Dokuments festzulegen. Damit sind alle Tags, die zu Beginn des Dokuments stehen sollten genannt. Ebenso wie die Tags `<title>` und `<mail>`, sollten diese nur direkt im Tag `<guide>` und nicht irgendwo anders stehen. Weiterhin wird empfohlen (aber nicht gefordert), dass diese Tags vor dem eigentlichen Inhalt im Dokument stehen.

Und schliesslich haben wir das `<license/>` Tag, mit dem wir das Dokument unter der [Creative Commons - Attribution / Share Alike](#) Lizenz veröffentlichen, wie es die [Dokumentations Richtlinie](#) erfordert.

## 2.2 Kapitel und Abschnitte

Sobald Sie die einleitenden Tags spezifiziert haben, können Sie damit beginnen, Strukturelemente des Dokuments hinzuzufügen. Guide-Dokumente sind unterteilt in Kapitel (chapter) und jedes Kapitel kann ein oder mehr Abschnitte (section) enthalten. Jedes Kapitel und jeder Abschnitt hat eine Überschrift. Hier ist ein Beispielkapitel mit einem Abschnitt, der einen Absatz enthält. Wenn Sie dieses XML an das XML im [vorherigen Auszug](#) anhängen und ein `</guide>` am Ende hinzufügen, haben Sie ein (minimales) Guide-Dokument:

### Befehlsauflistung 2

```
<chapter>
<title>Das ist mein Kapitel</title>
<section>
<title>Hier ist Abschnitt eins in meinem Kapitel</title>
<body>

<p>
Dies ist der eigentliche Textinhalt meines Abschnitts.
</p>

</body>
</section>
</chapter>
```

Wie Sie sehen haben ich dem Kapitel durch das hinzufügen eines Kindelements `<title>` in das Element `<chapter>` eine Überschrift gegeben. Anschliessend habe ich einen Abschnitt durch das hinzufügen des Elements `<section>` erzeugt. Wenn Sie sich das Element `<section>` anschauen, werden Sie zwei Kindelemente erkennen -- ein `<title>` und ein `<body>` Element. Da Ihnen das Element `<title>` bereits bekannt ist, sei hier das Element `<body>` erläutert; dieses enthält den eigenlichen Inhalt des derzeitigen Abschnitts. Wir betrachten nun kurz die Tags die innerhalb von `<body>` erlaubt sind.

### Notiz

Das Element `<guide>` kann mehrere Elemente `<chapter>`, und das Element `<chapter>` kann mehrere Elemente `<section>` enthalten. Allerdings kann das Element `<section>` nur einmal das Element `<body>` enthalten.

## 2.3 Ein Beispiel zu `<body>`

Jetzt ist es an der Zeit zu lernen, wie man den eigentlichen Inhalt auszeichnet. Hier ist der XML-Code zu einem Beispiel-`<body>`-Element:

### Befehlsauflistung 3

```
<p>
Das ist ein Absatz. <path>/etc/passwd</path> ist eine Datei.
<uri>http://www.gentoo.de</uri> ist meine Lieblings-Web-Seite.
Geben Sie <c>ls</c> ein - wenn sie wollen. Ich sollte jetzt <e>wirklich</e> schlafen gehen.
</p>

<pre>
Das ist Textausgabe oder Quelltext.
# <i>das sind Benutzereingaben</i>
```

Um HTML/XML lesbarer zu machen, sollten Sie verschiedene Hervorhebungen verwenden:

```
<foo><i>bla</i></foo>
```

```
<codenote>So wird eine Notiz in einen Quelltextabschnitt eingefügt</codenote>
</pre>
<note>
Das ist eine Anmerkung.
</note>
<warn>
Das ist eine Warnung.
</warn>
<impo>
Das ist wichtig.
</impo>
```

Und so wird dieses `<body>`-Element dargestellt:

Das ist ein Absatz. `/etc/passwd` ist eine Datei. <http://www.gentoo.de> ist meine Lieblings-Web-Seite. Geben Sie `ls` ein - wenn Sie wollen. Ich sollte jetzt **wirklich** schlafen gehen.

#### Befehlsauflistung 4

Das ist Textausgabe oder Quelltext.  
# **das sind Benutzereingaben**

Um HTML/XML lesbarer zu machen, sollten Sie verschiedene Hervorhebungen verwenden:  
<foo><i>bla</i></foo>

```
// So wird eine Notiz in einen Quelltextabschnitt eingefügt
```

#### Notiz

Das ist eine Anmerkung.

#### Warnung

Das ist eine Warnung.

#### Wichtig

Das ist wichtig.

## 2.4 Die Tags `<body>`

Im vorherigen Abschnitt wurde eine Menge neuer Tags eingeführt -- jetzt werden sie erläutert. Die Tags `<p>` (Absatz), `<pre>` (Quelltextabschnitt), `<note>` (Anmerkung), `<warn>` (Warnung) und `<impo>` (Wichtig), können alle ein oder mehrere Zeilen Text enthalten. Neben dem Element `<table>` (auf das wir später eingehen), sollten das die einzigen Tags, die innerhalb des Elements `<body>` stehen, sein. Weiterhin sollten diese Tags **nie** verschachtelt werden -- mit anderen Worten, Sie sollten das Element `<note>` nicht innerhalb des Elements `<p>` verwenden! Wie Sie vielleicht bemerken, gibt das Element `<pre>` Leerzeichen und Zeilenumbrüche exakt wieder - dadurch empfiehlt es sich für Quelltextabschnitte. Sie können auch das `<pre>` Tag benutzen:

#### Befehlsauflistung 5: `<pre>`

```
<pre caption = "Ausgabe von uptime">
# <i>uptime</i>
16:50:47 up 164 days,  2:06,  5 users,  load average: 0.23, 0.20, 0.25
</pre>
```

## 2.5 `<path>`, `<c>` und `<e>`

Die Elemente `<path>`, `<c>` und `<e>` können innerhalb jedes `<body>`-Kindelements verwendet werden, ausgenommen ist `<pre>`.

Das Element `<path>` wird verwendet um Text hervorzuheben, der auf eine **Datei auf der Festplatte** verweist -- genauso wie für einen **absoluten oder realtiven Pfad** oder einfach einen **Dateinamen**. Dieses Element wird normalerweise mit einer Maschinentextschrift dargestellt, um es vom Rest des Textes abzugrenzen.

Das Element `<c>` wird verwendet, um **Befehle** oder **Benutzereingaben** zu markieren. Sie sollten sich `<c>` als eine Möglichkeit vorstellen, dem Leser mitzuteilen, dass er etwas eingeben kann, was zu irgendeiner Aktion führt. Zum Beispiel sind alle XML-Tags in diesem Dokument von `<c>` umschlossen, um zu zeigen, dass der Benutzer etwas anderes als einen Pfad eingeben kann. Indem Sie das Element `<c>` verwenden, erleichtern Sie es Ihren Lesern, Befehle die sie eingeben müssen schneller zu erkennen. Desweiteren ist es auf Grund dessen, dass `<c>`-Elemente sich vom regulären Text abheben,

**nicht zwingend notwendig, Benutzereingaben mit doppelten Anführungszeichen zu umgeben.** Verwenden Sie zum Beispiel das Element "<c>" nicht wie in diesem Satz. Vermeiden Sie die Benutzung von nicht-notwendigen doppelten Anführungszeichen, um das Dokument leserlicher und ansprechender zu gestalten.

<e> wird benutzt, um Wörter oder Wortgruppen zu betonen; zum Beispiel: Ich sollte **wirklich** öfters Semikola verwenden. Wie Sie sehen können, hebt sich dieser Text zur Betonung von der normalen Absatzschrift ab. Dadurch wird das von ihnen Gesagte **schlagkräftiger!**

## 2.6 <mail> und <uri>

Wir hatten uns den Tag <mail> schon vorhin angesehen; er wird benutzt, um einen Text mit einer speziellen Email-Adresse zu verbinden; er hat die Form <mail link="mm@bla.de">Herr Martin Mustermann </mail>.

Der Tag <uri> wird verwendet, um auf Dateien oder Seiten im Internet zu verweisen. Es gibt zwei Varianten. Bei der ersten wird die URI direkt im Text angezeigt, so wie dieser Link zu <http://www.gentoo.de>. Um diesen Link zu erzeugen, habe ich <uri>http://www.gentoo.de </uri> eingegeben. Die Alternative wird verwendet, wenn Sie eine URI mit einem abweichenden Text verknüpfen wollen -- z.B. [das deutschsprachige Gentoo Linux Portal](http://www.gentoo.de). Um **diesen** Link zu erzeugen, wurde <uri link="http://www.gentoo.de">das deutschsprachige Gentoo Linux Portal </uri> eingegeben.

## 2.7 Grafiken

Und so können Sie Grafiken in ein Dokument einfügen: <figure link="meinbild.png" short="mein Bild" caption="für immer mein Lieblingsbild"/ >. Das Attribut link= verweist auf die gewünschte Grafikdatei, das Attribut short= legt eine Kurzbeschreibung fest (wird derzeit für das Attribute alt= im HTML-Tag <img> verwendet) und mit caption= wird schließlich eine Bildunterschrift festgelegt -- alles in allem nicht allzu schwer :) Ausserdem wird noch der Tag  aus HTML unterstützt, um Grafiken ohne Unterschrift, Ränder u.a. einzufügen.

## 2.8 Tabellen und Listen

Guide stellt eine vereinfachte Tabellensyntax, vergleichbar mit der von HTML, bereit. Um eine Tabelle zu erzeugen, benutzen Sie den Tag <table>. Eine Reihe wird mit <tr> begonnen. Allerdings unterstützt GuideXML **keinen** Tag <td> wie in HTML, um die eigentlichen Tabelleninhalte einzufügen; verwenden Sie stattdessen <th> für den Tabellenkopf und <ti> wenn Sie normale Informationen einfügen wollen. Sie können <th> überall da verwenden, wo Sie auch <ti> verwenden können -- es gibt also keinen Zwang <th>-Elemente nur in der ersten Reihe zu verwenden. Derzeit stellen diese Tags noch keine Attribute bereit, allerdings werden bald welche hinzugefügt (wie z.B. ein Attribut caption= für <table>).

Um eine sortierte oder unsortierte Liste zu erzeugen, verwenden Sie einfach die gleichwertigen HTML-Tags <ol>, <ul> und <li>. Listen-Tags sollten nur innerhalb von <p>, <ti>, <note>, <warn> und <impo> verwendet werden.

## 2.9 Verweise innerhalb des Dokuments

In Guide ist es wirklich einfach, Verweise zu anderen Teilen des Dokuments mit Hilfe von Hyperlinks zu erstellen. Sie können einen Verweis zu [Kapitel 1](#) durch das Eingeben von <uri link="#doc\_chap1">Kapitel 1</uri> erzeugen. Um auf [Abschnitt 2 in Kapitel 1](#) zu verweisen, geben Sie <uri link="#doc\_chap1\_sect2">Abschnitt 2 in Kapitel 1</uri> ein. Um auf Grafik 3 zu verweisen, geben Sie <uri link="doc\_chap1\_fig3">Grafik 1.3</uri> ein; oder für einen Verweis auf [Quelltextbeispiel 2.2](#) einfach <uri link="doc\_chap2\_pre2">Quelltextbeispiel 2.2</uri>. In Zukunft werden weitere Möglichkeiten, der automatischen Verknüpfung (wie z.B. Unterstützung von Tabellen) hinzukommen.

Allerdings ändern sich viele Anleitungen öfters und solches "Zählen" kann dann zu fehlerhaften Links führen. Um dem abzuhelpen, kann ein Name für ein <chapter> oder <section> mit Hilfe des id Attributes hinzugefügt werden. Nun muss man nur noch auf dieses Attribut verweisen:

**Befehlsauflistung 6:** Verwendung des id Attributes

```
<chapter id="foo">
<title>Das ist foo!</title>
...
```

<p>

Weitere Informationen finden Sie im <uri link="#foo">foo Kapitel</uri>

</p>

## 3. Coding Style

### 3.1 Einleitung

Da jede Gentoo Dokumentation eine große Anstrengung ist und viele Leute die Dokumentation ändern, ist ein Coding Style von Nöten. Ein Coding Style beinhaltet zwei Teile. Der erste (Interner Coding Style) erläutert wie die xml-Tags platziert werden sollen, der zweite beschreibt den Inhalt; wie man die Leser nicht verwirrt.

Beide Teile werden jetzt erläutert.

### 3.2 Interner Coding Style

**Neue Zeilen** müssen direkt nach **jedem** GuideXML-Tag platziert werden (öffnende und schliessende), ausgenommen für <version>, <date>, <title>, <th>, <ti>, <li>, <i>, <e>, <uri>, <path>, <b>, <comment>, <codenote>, <mail>.

**Leere Zeilen** müssen direkt nach **jedem** <body> (nur öffnende Tags) und vor **jedem** <chapter>, <p>, <table>, <author> (Gruppe), <pre>, <ul>, <ol>, <warn>, <note> and <impo> (nur öffnende Tags) platziert werden.

Ein **Zeilenbruch** muss nach 80 Zeilen angewandt werden, ausser im <pre> Tag. Nur wenn es keine andere Wahl gibt kann von dieser Regel abgewichen werden (zum Beispiel wenn eine URL über das Maximum an Zeichen hinausgeht). Der Editor muss dann beim nächsten Leerzeichen umbrechen.

**Einrücken** soll nicht benutzt werden, ausser in XML Konstruktionen bei denen die Parent Tags <tr> (von <table>), <ul>, <ol> und <author> sind. Wenn Einrücken benutzt wird, **müssen** zwei Leerzeichen für jede Einrückung benutzt werden. Das heisst, **keine** Tabs und **nicht** mehr Leerzeichen.

Wenn ein Zeilenbruch in eines der folgenden <ti>, <th> oder <li> Konstrukte fällt, muss Einrückung für den Inhalt benutzt werden.

Ein Beispiel für Einrückung ist:

**Befehlsauflistung 7:** Beispiel: Einrückung

```
<table>
<tr>
  <th>Foo</th>
  <th>Bar</th>
</tr>
<tr>
  <ti>Das ist ein Beispiel für Einrückung.</ti>
  <ti>
    Wennn Text nicht in eine 80 Zeilen lange Zeile passt, müssen
    Sie Einrückung benutzen, wenn das Parent Tag es erlaubt.
  </ti>
</tr>
</table>

<ul>
  <li>Erste Option</li>
  <li>Zweite Option</li>
</ul>
```

**Attribute** sollen keine Leerzeichen zwischen den Attributen haben. "=" Ein Beispiel:

**Befehlsauflistung 8:** Attribute

Falsch : <pre caption = "Attribute">  
Richtig: <pre caption="Attribute">

### 3.3 Externer Coding Style

Innerhalb von Tabellen (`<table>`) und Listings (`<ul>` und `<ol>`), sollten Punkte (".") nur benutzt werden, wenn mehrere Sätze benutzt werden. In diesem Fall sollte jeder Satz mit einem Punkt enden.

Jeder Satz, auch innerhalb von Tabellen und Listings sollte mit einem Großbuchstaben beginnen.

**Befehlsauflistung 9:** Punkte und Großbuchstaben

```
<ul>
  <li>Kein Punkt</li>
  <li>Mit Punkt. Mehrere Sätze, Sie erinnern sich?</li>
</ul>
```

Codelistings sollten **immer** eine *Beschreibung* haben.

Benutzen Sie soweit möglich `<uri>` mit dem *link* Attribut. In anderen Worten, [Gentoo Website](http://www.gentoo.org) wird <http://www.gentoo.org> vorgezogen.

Wenn Sie etwas innerhalb eines `<pre>` Konstruktes kommentieren möchten, benutzen Sie `<codenote>` wenn der Inhalt ein C or C++ Code Schnipsel ist. Andernfalls benutzen Sie `<comment>`. Platzieren Sie den Kommentar **vor** dem Bezug des Kommentars.

**Befehlsauflistung 10:** Kommentar Beispiel

```
(Ersetzen Sie "john" mit Ihrem Benutzer Namen)
# id john
```

## 4. Quellenangaben

### 4.1 Mit dem Schreiben beginnen

Guide wurde speziell dafür entwickelt, es Entwicklern zu ermöglichen, mehr Zeit mit dem Schreiben der Dokumentation und weniger mit dem Erlernen der eigentlichen XML-Syntax zu verbringen. Wir hoffen, dass dies den Entwicklern erlaubt, qualitativ hochwertige Gentoo-Linux-Dokumentation zu schreiben. Wenn Sie uns helfen möchten (oder irgendeine Fragen zu GuideXML haben), dann senden Sie eine Nachricht an die [gentoo-doc Mailing Liste](#) (in Englisch!). Wir wünschen viel Spass;!

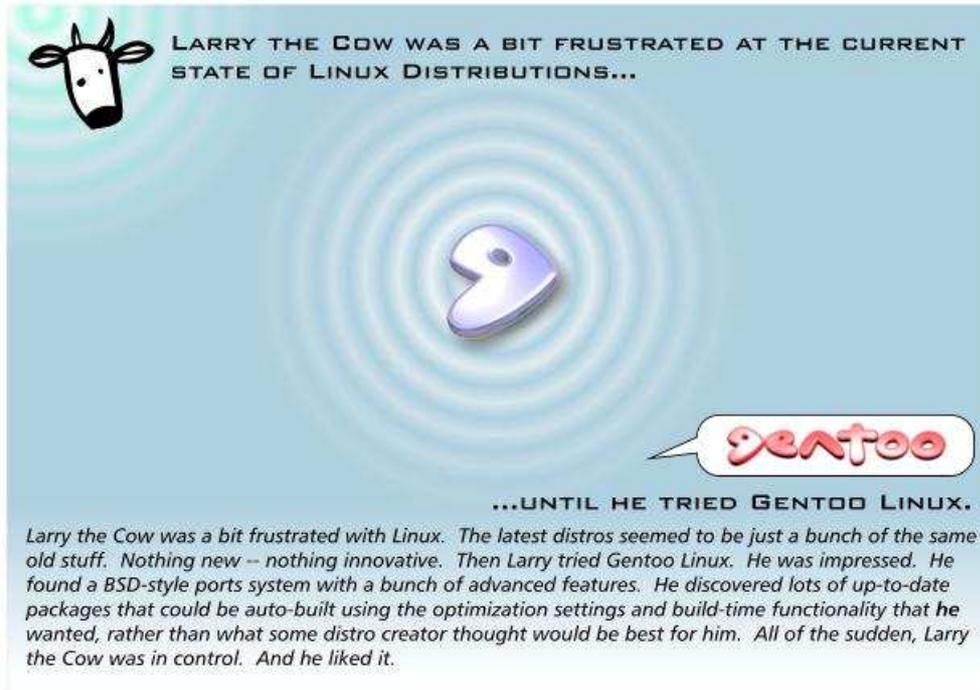




## Über Gentoo Linux

Contents: [1. Gentoo Linux... in einem Poster](#) ■

### 1. Gentoo Linux... in einem Poster



### 2. Gentoo Linux in einem Absatz

Gentoo Linux, eine besondere Art von Linux, die automatisch für nahezu jede Applikation oder jeden Einsatz optimiert und angepasst werden kann. Extreme Leistung, Konfigurierbarkeit und eine erstklassige Benutzer- und Entwicklergemeinde sind die Gütezeichen von Gentoo.

Dank der Technologie namens Portage, kann Gentoo Linux als ein idealer sicherer Server, eine Entwicklungsumgebung, ein professioneller Arbeitsplatzrechner, ein Spielesystem, eine embedded solution oder etwas anderes sein -- wie immer Sie es benötigen. Aufgrund seiner nahezu unbegrenzten Anpassbarkeit, nennen wir Gentoo Linux eine **meta** distribution.

### 3. Was ist Portage?

**Portage** ist das Herz von Gentoo Linux und übt viele Schlüsselfunktionen aus. Zum einen ist Portage das [Software Distributions System](#) für Gentoo Linux. Um die neueste Software von Gentoo Linux zu erhalten, geben Sie einfach das Kommando `emerge sync` ein. Dadurch wird Portage angewiesen Ihren lokalen "Portage tree" über das Internet auf den neuesten Stand zu bringen. Ihr lokaler Portage tree enthält eine komplette Sammlung von Skripten, die von Portage verwendet werden, um die neuesten Gentoo Pakete zu installieren. Derzeit haben wir [mehr als 4000 Pakete](#) in unserem Portage tree, und es werden immer mehr.

Portage ist auch ein [Paketkompilierungs- und -installationssystem](#). Wenn Sie ein Paket installieren möchten, geben Sie `emerge Paketname` ein, wodurch Portage automatisch eine individuelle Version des Paketes installiert -- mit Ihren speziellen Wünschen, Optimierungen für Ihre Hardware und optionellen Zusatzfunktionen, die das Paket bereitstellt.

Portage hält auch Ihr System [up-to-date](#). Allein durch das Kommando `emerge -u world` werden alle von Ihnen gewünschte Pakete automatisch auf den neuesten Stand gebracht.

### 4. Gentoo Linux 1.4\_rc4

Portage hält auf Ihren Wunsch Ihr Gentoo Linux System "up-to-date". Aus diesem Grund spenden erfahrene Gentoo Benutzer keine große Aufmerksamkeit zu "neuen Versionen" von Gentoo Linux -- schließlich ist die neueste und beste Version von Gentoo Linux immer durch den Aufruf des Kommandos `emerge sync` verfügbar. Es gibt keinen Grund mehrere Monate auf eine zu erscheinende "neue Version" von Gentoo Linux zu warten, da Gentoo Linux fortlaufend aufgefrischt wird und diese Verbesserungen unverzüglich für Sie verfügbar sind.

Natürlich stellen wir offizielle CDs von Gentoo Linux zusammen, sodass neue Gentoo Linux Installationen direkt von Beginn an auf dem neuesten Stand wie möglich sind. Hier ist ein Überblick was die aktuelle 1.4\_rc4 Gentoo Linux Version enthält:

Updated 13 Juni 2003

**Daniel Robbins**

Author

**Jorge Paulo**

Editor

**Lars Weiler**

Übersetzer

**Summary:** Gentoo Linux stellt sich vor

**Donate** to support our development efforts.

**Make a Donation**



php|architect is the monthly magazine for PHP professionals, available worldwide in print and electronic format. A percentage of all the sales will be donated back into the Gentoo project.



Tek Alchemy offers dedicated servers and other hosting solutions running Gentoo Linux.



Purchase RAM from **Crucial.com** and a percentage of your sale will go towards further Gentoo Linux development.

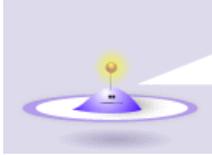
- Unterstützung für x86, PowerPC, UltraSparc und Alpha Prozessoren
- LiveCD-basierte Installation für x86 und PowerPC
- Die neuesten offiziellen Versionen von [KDE](#) und [GNOME](#)
- Verschiedene optimierte Linux Kernel (2.4.20, 2.4.21\_pre)
- Sehr moderne GNU Entwicklungsumgebung (glibc 2.3, gcc 3.2.2)
- Hervorragende Dateisystemunterstützung: ReiserFS, XFS, ext3, EVMS, LVM
- Optimale Hardwareunterstützung: NVIDIA, Creative Labs Live! und Audigy
- Modulares OpenGL und Compiler Sub-System (unterstützt mehrere nebeneinander existierende Versionen)
- Saubere, voneinander abhängige Skripte zur Initiierung des Systems
- Neue "hardened" Gentoo Security Initiative
- 4000+ Pakete mit der neuesten und besten Software
- Erweiterte Portagefähigkeiten

Wenn Sie die Kraft, Geschwindigkeit und Flexibilität von Gentoo Linux anspricht, dann probieren Sie es aus. Wir denken nicht, dass Sie enttäuscht werden.



Win4Lin from  
**NeTraverse** lets you  
 run Windows  
 applications under  
 Gentoo Linux at native  
 speeds.

Copyright 2001-2003 Gentoo Technologies, Inc. Questions, Comments, Corrections? Email [www@gentoo.org](mailto:www@gentoo.org).



**Documentation:**  
[About Gentoo Linux](#)  
[User Docs](#)  
[Developer Docs](#)  
[Other Docs/Translations](#)  
[Philosophy](#)

**Installation:**  
[Gentoo Handbook](#)  
[Installation Instructions](#)

**Older Installation Guides:**  
[Gentoo Linux/x86](#)  
[Gentoo Linux/PowerPC](#)  
[Gentoo Linux/Sparc](#)  
[Gentoo Linux/HPPA](#)  
[Gentoo Linux/Alpha](#)

**Resources:**  
[Mailing lists](#)  
[Performance benchmarks](#)  
[Discussion forums](#)  
[Daily CVS ChangeLog](#)  
[Developer List](#)  
[Bugzilla bug tracker](#)  
[Download Mirrors](#)  
[Official Gentoo IRC channels](#)  
[Online package database](#)  
[View our CVS via the web](#)

**Graphics:**  
[Logos and themes](#)  
[Icons](#)  
[ScreenShots](#)

**Miscellaneous Resources:**  
[Gentoo Linux Store](#)  
[Gentoo-hosted projects](#)  
[IBM dW/Intel article archive](#)

[About](#) | [Projects](#) | [Philosophy](#) | [Docs](#) | [Forums](#) | [Lists](#) | [Bugs](#) | [Store](#) | [GWN](#) | [Get Gentoo!](#) | [Sponsors](#)

## 1. Gentoo Linux Social Contract

### About the Gentoo Linux Social Contract

This social contract is intended to clearly describe the overall development policies and standards of the Gentoo Linux development team. Parts of this document have been derived from the [Debian Social Contract](#), and is generally very similar to it except that certain parts have been clarified and augmented while other parts deemed redundant have been removed. Comments are welcome. Please send them to our [gentoo-dev@gentoo.org](mailto:gentoo-dev@gentoo.org) mailing list.

### Gentoo Linux is and will remain Free Software

We will release our contributions to Gentoo Linux as free software, under the GNU General Public License version 2 (or later, at our discretion.) Any external contributions to Gentoo Linux (in the form of freely-distributable sources or binaries) may be incorporated into Gentoo Linux provided that we are legally entitled to do so. However, Gentoo Linux will never [depend](#) upon a piece of software unless it conforms to the GNU General Public License, GNU Lesser General Public License or some other license approved by the Open Source Initiative ([OSI](#).)

**Note:** We are considering extending the above clause to require that all core Gentoo Linux components must conform to a license approved by the OSI *and* Free Software Foundation ([FSF](#).)

### We will give back to the Free Software Community

We will establish relationships with Free Software authors and collaborate when possible. We will submit bug-fixes, improvements, user requests, etc. to the "upstream" authors of software included in our system. We will also clearly document *our* contributions to Gentoo Linux as well any improvements or changes we make to external sources used by Gentoo Linux (whether in the form of patches, "sed tweaks" or some other form.) We acknowledge that our improvements and patches are much more meaningful to the larger Free Software community if they are clearly documented and explained, since not everyone has the time or ability to understand the literal changes contained in the patches, tweaks, etc. themselves.

### We will not hide problems

We will keep our [bug report database](#) open for public view at all times. Reports that users file on-line will immediately become visible to others.

Updated 19 July 2003

**Donate** to support our development efforts.

**Make a Donation**



[www.phparchitect.com](http://www.phparchitect.com)

php|architect is the monthly magazine for PHP professionals, available worldwide in print and electronic format. A percentage of all the sales will be donated back into the Gentoo project.



Tek Alchemy offers dedicated servers and other hosting solutions running Gentoo Linux.



Purchase RAM from [Crucial.com](#) and a percentage of your sale will go towards further Gentoo Linux development.



Win4Lin from **NeTraverse** lets you run Windows

applications under  
Gentoo Linux at native  
speeds.

.....  
**Copyright 2001-2003 Gentoo Technologies, Inc. Questions, Comments, Corrections? Email [www@gentoo.org](mailto:www@gentoo.org).**