

Rechenzentrum Universität Kiel

<http://www.uni-kiel.de/rz/>

Unterlagen zum Kurs

# Linux – grundlegende Administration

17.-18. 9. 2008

Birgit Ohlenbusch, Tel. (0431) 880-1394, Email [ohlenbusch@rz.uni-kiel.de](mailto:ohlenbusch@rz.uni-kiel.de)

---

Ich habe mich bemüht, die Kursunterlagen verständlich und inhaltlich richtig zu gestalten. Es kann sein, daß sie Fehler und auch Lücken beinhalten. Ich freue mich über jeden Hinweis darauf.

Birgit Ohlenbusch

# Inhalt

|   |           |
|---|-----------|
| <b>1 Allgemeines.....</b>   | <b>1</b>  |
| 1.1 Kursnotebooks.....  | 1         |
| 1.2 Kurzüberblick über Unix-Befehle.....  | 1         |
| 1.3 Umgang mit Prozessen.....   | 2         |
| 1.4 Der Root-Account.....   | 3         |
| 1.5 Allgemeines zum Konfigurieren von System und Software.....                      | 5         |
| 1.6 virtuelle Konsolen.....   | 6         |
| 1.7 Logging.....  | 6         |
| <b>2 Grundlegende Netzkonfiguration.....</b>  | <b>8</b>  |
| 2.1 Grundlegendes.....  | 8         |
| 2.2 benötigte Angaben für Rechner im LAN.....                                       | 9         |
| 2.3 Konfiguration mit Yast.....   | 9         |
| 2.4 betroffene Systemdateien.....   | 11        |
| <b>3 Software und Patche.....</b>   | <b>12</b> |
| 3.1 Wo gibt es Software?.....   | 12        |
| 3.2 Installation von Software mit YaST.....   | 13        |
| 3.3 1-Klick-Installationen (One-Click-Install).....                                 | 14        |
| 3.4 Softwarekorrekturen(Patche).....  | 14        |
| 3.5 Update auf eine neue SUSE-Version.....  | 16        |
| 3.6 Software-Verwaltung mit zypper-Befehlen.....                                    | 17        |
| 3.7 rpm-Software-Pakete.....  | 17        |
| <b>4 Dateien, Plattenpartitionen, Mounen.....</b>                                   | <b>19</b> |
| 4.1 Dateizugriffsrechte.....  | 19        |
| 4.2 Plattenpartitionen.....   | 21        |
| 4.3 Mounen von Platten und anderen Datenträgern.....                                | 22        |
| 4.4 Zugriff auf Windows Laufwerke.....  | 23        |
| <b>5 Booten von Linux.....</b>  | <b>24</b> |
| 5.1 Der Bootvorgang beim PC.....  | 24        |
| 5.2 Die Initial Ramdisk.....  | 24        |
| 5.3 Der Master-Boot Record (MBR) mit der Partitionstabelle, Bootsektoren.....       | 25        |
| 5.4 Bootloader.....   | 26        |
| 5.5 Neustart / Herunterfahren des Rechners / Der Init-Prozeß.....                   | 28        |
| <b>6 Dies und Das.....</b>  | <b>30</b> |
| 6.1 X-Konfiguration: Bildschirm / Grafikkarte / Tastatur / Maus / Zeichensätze..... | 30        |
| 6.2 Spracheinstellung, Lokalisation.....  | 31        |
| 6.3 Stromsparfunktionen.....  | 32        |
| 6.4 Druckerkonfiguration.....   | 32        |
| 6.5 Benutzerverwaltung.....   | 33        |
| 6.6 Cronjobs.....   | 35        |
| 6.7 SSH Zugang zum eigenen Rechner.....   | 35        |
| <b>7 Mit Problemen fertig werden.....</b>   | <b>36</b> |
| 7.1 Log-Dateien.....  | 36        |
| 7.2 Die grafische Oberfläche reagiert nicht mehr.....                               | 36        |
| 7.3 Mein Rechner bootet nicht mehr - was kann ich tun?.....                         | 36        |
| 7.4 Grub startet nicht mehr – das Bootmenü erscheint nicht.....                     | 36        |
| 7.5 Ich habe das root-Passwort vergessen.....                                       | 37        |

# 1 Allgemeines

## 1.1 Kursnotebooks

Zur Verfügung stehen Notebooks mit installiertem openSUSE 11.0

- deutsch
- Desktop KDE 3.5

die Release Notes stehen unter `/usr/share/doc/release-notes`

## 1.2 Kurzüberblick über Unix-Befehle

das erste Wort einer Befehlszeile wird als Befehl/Programm interpretiert

- entweder ein Shell-Befehl oder ein Programm
- Syntax oft: `Befehl -Optionen Argumente` Beispiel: `ls -alt *.conf`
- `$PATH`: Suchliste für Programme

`<Tab>` für Vervollständigung von Befehlen und Dateien

`* ?` Platzhalter bei Dateien

**Variablen** in der Bash

- Zuweisung: `Variable=Wert`
  - kein Leerzeichen vor und hinter =
  - Beispiele:
    - `Datei=core`
    - Alle Dateien im Verzeichnis oder Unterverzeichnissen von /home, die mit txt enden:
      - `Dateien=`find /home -name '*.txt'``
  - Inhalt: `$Variable`; Beispiel: `echo $PATH`
- eine Variable wird durch `export Variable` zur **Environment Variable**
  - Beispiel: `PATH=$HOME/bin:$PATH; export PATH`

`&` Ausführung im Hintergrund; sinnvoll insbesondere bei grafischen Anwendungen

Beispiel: `kwriteln /etc/motd&`

`>` Ausgabeumleitung; Beispiel: `ls -l > listing.txt`

`>>` Ausgabeumleitung - Anhängen an bestehende Datei; Beispiel: `date >> listing.txt`

`|` Pipe; Beispiel: `ls -l | more`

Online Hilfe:

- meistens: `man Befehl`
- manchmal `info Befehl` oder auch `Befehl --help`
- Dokumentation unterhalb von `/usr/share/doc/packages`
- Im openSUSE Start-Handbuch: Grundlagen – Shell-Grundlagen

### 1.2.1 Einige oft benutzte Befehle:

| Programme                |                                  |                         |
|--------------------------|----------------------------------|-------------------------|
| <code>which</code>       | Pfadangabe des Befehls/Programms | <code>which ls</code>   |
| <code>whereis</code>     | wo ist das Programm?             | <code>whereis ls</code> |
| <code>man Befehl</code>  | Manpage anzeigen                 | <code>man ls</code>     |
| <code>info Befehl</code> | Infos zum Befehl anzeigen        | <code>info ls</code>    |

| Dateioperationen   |  |  |
|--|--|--|
| <code>ls -Optionen Dateien</code>  | Auflisten von Dateien  | <code>ls -al</code>                    |
| <code>pwd</code>   | Ausgabe des aktuellen Verzeichnisses   |  |
| <code>cat -Optionen Dateien</code>   | Dateiinhalte ausgeben  |  |
| <code>more -Optionen Dateien</code><br>oder<br><code>less -Optionen Dateien</code> | Dateiinhalte seitenweise ausgeben<br><Return>: zeilenweise<br><Leertaste>: seitenweise<br><code>/Suchbegriff</code> : Suche ( <code>n</code> wiederholt) | <code>less /etc/passwd</code>          |
| <code>head -Optionen Dateien</code>  | die ersten Zeilen ausgeben   | <code>head -20 /etc/inittab</code>     |
| <code>tail -Optionen Dateien</code>  | die letzten Zeilen ausgeben  | <code>tail -f /var/log/messages</code> |
| <code>diff Datei1 Datei2</code>  | Vergleich zweier Dateien   |  |
| <code>cd Verzeichnis</code>  | Verzeichniswechsel   |  |
| <code>find</code>  | Suche von Dateien  | <code>find . -name '*.odt'</code>      |
| <code>grep Ausdruck Dateien</code>   | Suche von Zeichen in Dateien   | <code>grep root /etc/passwd</code>     |
| <code>du -Optionen Dateien</code>  | Platzverbrauch von Verzeichnissen/Dateien  | <code>du -s *</code>                   |

| Partitionen        |  |                                |
|--------------------|--|--------------------------------|
| <code>df</code>    | Größe+Verbrauch der gemounteten Datenträger                  |                                |
| <code>fdisk</code> | Plattenpartitionierung                                       | <code>fdisk -l /dev/hda</code> |
| <code>fsck</code>  | Check und Reparatur von Dateisystemen<br>(file system check) |                                |

| Prozesse                                   |   |                              |
|--|---|------------------------------|
| <code>ps [-]Optionen</code>                | Prozesse anzeigen                           | <code>ps aux</code>          |
| <code>kill-Signal ProzessIDs</code>        | Prozesse beenden                            | <code>kill -TERM 1024</code> |
| <code>killall-Optionen ProzessNamen</code> | Prozesse beenden                            |                              |
| <code>top</code>                           | Prozesse nach Ressourcenverbrauch auflisten | <code>top</code>             |

### 1.3 Umgang mit Prozessen

jedem Prozess ist insbesondere eine process id (pid) und ein Eigentümer zugeordnet

#### Auflisten der Prozesse mit ps

- Optionen mit vorgestelltem -: Posix-Syntax, ohne BSD-Syntax

Beispiel: `ps aux`

`a` alle Benutzer  
`u` Ausgabe des Prozesseigentümers  
`x` auch im Hintergrund laufende Prozesse

Viele weitere nützliche Optionen in der Manpage

## Auflisten der Spitzenverbraucher mit top

- Sortierkriterium nachträglich einstellbar, insbesondere
  - M** Speicherverbrauch
  - P** CPU Verbrauch
  - q** Verlassen des Programms
- sehr mächtiges Werkzeug, s. Manpage

## Beenden von Prozessen

### kill, killall

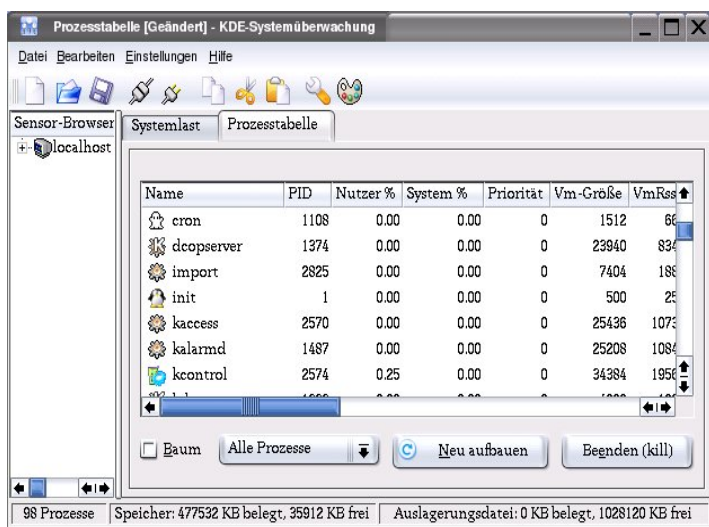
- Senden eines Signals an den Prozess (Signale s.a. **man 7 signal**)

<Strg>**C** Beenden eines interaktiven Prozesses

<Strg>**Z** Stoppen eines interaktiven Prozesses

- daraufhin:
  - bg** läßt den Prozess im Hintergrund weiter laufen
  - fg** läßt den Prozess im Vordergrund weiter laufen (Ausgabe/Eingabe vom Konsolfenster)

## ksysguard, eine grafische Alternative



## 1.4 Der Root-Account

**ACHTUNG: Root darf alles!**

### 1.4.1 Verhaltensregeln für root

- nur wenn nötig als root arbeiten, lieber als unprivilegierter Benutzer
  - Schutz sowohl vor sich selber als auch für andere und das System
  - bei SUSE wird bei Aufruf von YaST nach dem Root-Passwort gefragt, wenn unprivilegiert eingeloggt
- Login nur über sichere Verbindungen: lokales Terminal oder verschlüsselte Verbindung (ssh)
- Login niemals direkt als root (Ausnahme evtl. von der Konsole), sondern zunächst als unprivilegierter Benutzer, dann mit Hilfe von **su** oder **sudo** für einzelne Kommandos
  - Vorteile:
    - es ist nachvollziehbar, wer als root gearbeitet hat
    - erhöhter Schutz durch
      - einen zweiten (nicht unbedingt bekannten) Benutzernamen
      - 2 Passwörter statt nur ein Passwort

- erzwingen durch - abhängig vom Dienst (grundsätzlich sowieso nur benötigte Dienste aktivieren!):
  - ftp: Zeile `root` in `/etc/ftpusers` (SUSE Standard)
  - ssh: Eintrag `PermitRootLogin no` in `/etc/ssh/sshd_config`
- nur sichere Programme/Scripte aufrufen, vertrauenswürdige Software installieren
- Nur „sichere Verzeichnisse“ in der PATH-Variablen! Insb. der Punkt `.` (das aktuelle Verzeichnis) darf nicht enthalten sein - sonst Gefahr von trojanischen Pferden

### 1.4.2 Vorübergehend als root arbeiten

- su, sudo
- Konsole: Sitzung *Befehlsfenster (Systemverwaltungsmodus)*
- kdesu, Fenster *Befehl ausführen*

### 1.4.3 su

Mit dem `su`-Befehl wird eine Shell für einen anderen Benutzer (Standard: root) gestartet. Es ist das Passwort des anderen Benutzers anzugeben.

```
user@host:~ su -
Password:
root@host:~
```

immer `su -`  
statt `su`

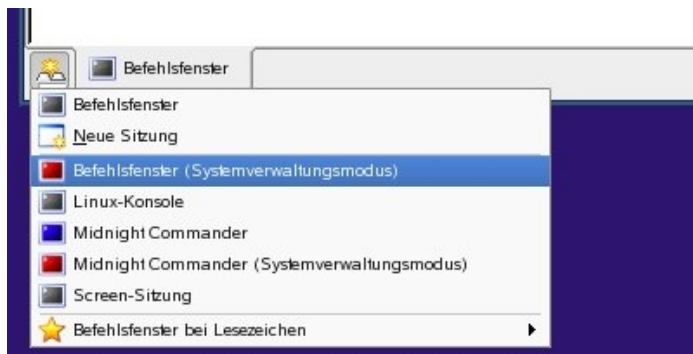
die Konsole-Sitzung *Befehlsfenster (Systemverwaltungsmodus)* benutzt das `su`-Kommando

falls grafische Anwendungen unter einer anderen Kennung gestartet werden, bieten sich folgende Alternativen an:

- `kdesu Programm`
- Fenster *Befehl ausführen* (z.B. über `<Alt> <F2>`)

### 1.4.4 root-Zugriff mit Hilfe der Konsole-Sitzung

- im Konsole-Fenster den *Neue Sitzung* – Button (links unten) gedrückt halten, Sitzung *Befehlsfenster (Systemverwaltungsmodus)* auswählen
- Definition der Sitzung unter *Einstellungen – Konsole einrichten... - Sitzung*



### 1.4.5 sudo

Ausführen eines Kommandos als ein anderer Benutzer (Standard: root)

- Logging-Funktionalität
- Konfigurationsdatei `/etc/sudoers` zum Eintragen der Rechte (und Konfigurationseinstellungen)
  - legt insbesondere fest, wer welche Kommandos (als root) ausführen darf

zwei Authorisierungsarten:

- der Benutzer muß sich mit dem root-Passwort authorisieren (Standard-Einstellung bei SUSE)
  - Eintrag `Defaults targetpw` in Konfigurationsdatei
- der Benutzer muß sich nicht mit dem root-Passwort, sondern seinem Passwort authorisieren

wer darf sudo benutzen

- wird in sudos Konfigurationsdatei festgelegt
  - SUSE Standardeinstellung: jeder darf (`%users ALL=(ALL) ALL`)  
Achtung: i.a. nur sinnvoll zusammen mit `Defaults targetpw` !!!!

### Ändern der Konfigurationsdatei `/etc/sudoers`

nur mit Hilfe des Befehls `visudo` - nicht direkt editieren!!

- mit der Environment-Variablen `EDITOR` läßt sich ein anderer Editor (Standard: `vi`) einstellen
  - z.B. `EDITOR=joe visudo`
- Locking-Funktion
- Beim Abspeichern Fehlermeldung, falls syntaktisch falsche Einträge

### Beispieleintrag in der Datei `/etc/sudoers`:

Die Benutzer `adminuser1`, `adminuser2` sollen die Kommandos `/bin/ls`, `/bin/cat`, `/usr/bin/du`, `/usr/bin/file` als `root` ausführen dürfen, Authorisierung über ihr Kennwort:

```
# Host alias specification

# User alias specification
User_Alias  BERATUNG=adminuser1,adminuser2

# Cmnd alias specification
Cmnd_Alias  UEBERPRUEFE=/bin/ls,/bin/cat,/usr/bin/du,/usr/bin/file

# User privilege specification
root      ALL=(ALL) ALL
BERATUNG  ALL=UEBERPRUEFE
```

**Achtung:** zunächst die Zeilen

```
Defaults targetpw
%users ALL=(ALL) ALL
```

auskommentieren

### sudo-Beispiel:

```
adminuser1@host:~ sudo ls -l /var/log/messages

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these two things:

    #1) Respect the privacy of others.
    #2) Think before you type.

Password:
-rw-r-----  1 root    root      3885024 Aug  9 11:33 /var/log/messages
adminuser1@host:~ sudo cat /var/log/messages | tail -1
Aug  9 11:34:48 host sudo: adminuser1 : TTY=pts/4 ; PWD=/var/log ; USER=root ;
COMMAND=/bin/cat /var/log/messages
```

## 1.5 Allgemeines zum Konfigurieren von System und Software

Die Konfiguration des Linux-Systems ist i.a. in Konfigurationsdateien (Textdateien) festgelegt

- i.a. unter `/etc` (s.a. FHS <http://www.pathname.com/fhs/>)

Möglichkeiten zum Konfigurieren

- durch Editieren der Konfigurationsdatei
  - danach muß der Prozeß oft neu gestartet werden oder aber ihm ein `HANGUP`-Signal geschickt werden, z.B.:

```
root@host:~ ps x | grep inetd
 547 ?        S          0:00 /usr/sbin/inetd
 7727 pts/4  S          0:00 grep inetd
root@host:~ kill -HUP 547
```

- mit Hilfe von Kommandos

- oder aber mit Hilfe von Konfigurationstools
  - bei SUSE sind das:
    - YaST für fast alles (YaST=Yet another System Tool)
    - SAX für die Grafik (s. 6.1 X-Konfiguration: Bildschirm / Grafikkarte / Tastatur / Maus / Zeichensätze) - ist auch in YaST integriert

### ACHTUNG – SUSE-Besonderheit:

- bei SUSE werden die meisten Konfigurationseinstellungen zentral in Dateien im Verzeichnis und Unterverzeichnissen von `/etc/sysconfig` eingetragen und durch Aufruf von `/sbin/SuSEconfig` in die eigentlichen Konfigurationsdateien verteilt.

Die Variablen in diesen Dateien werden beim Booten von den verschiedensten Diensten ausgewertet (s.a. 5.5 Neustart / Herunterfahren des Rechners / Der Init-Prozeß)

- Dokumentation in `/usr/share/doc/packages/sysconfig/`

**d.h. Vorsicht: per Hand gemachte Änderungen von Konfigurationsdateien außerhalb von `/etc/sysconfig` gehen nach einem Aufruf von YaST evtl. verloren!!**

- wer per Hand Dateien im Verzeichnis `/etc/sysconfig` editiert, sollte anschließend `/sbin/SUSEconfig` aufrufen
- die Parameter in `/etc/sysconfig/*` lassen sich auch bequem mit *YaST - System - Sysconfig Editor* ändern.
  - Beispiel: Parameter `DISPLAYMANAGER_AUTOLOGIN`

## 1.6 virtuelle Konsolen

erreichbar über die Tastenkombinationen `<Strg><Alt><F1>`, `<Strg><Alt><F2>`, ...

üblich wie folgt belegt:

|   |   |
|---|---|
| <code>&lt;F1&gt;</code> , ... <code>&lt;F6&gt;</code> | Textkonsolen, dabei <code>&lt;F1&gt;</code> : Bootmeldungen |
| <code>&lt;F7&gt;</code>                               | grafische Umgebung  |
| <code>&lt;F10&gt;</code>                              | Kernelmeldungen (s.a. 1.7 Logging)                          |

## 1.7 Logging

i.a. in Dateien unter `/var/log`; insbesondere

- Datei `messages` für fast alles
- Datei `boot.msg` für Boot-Meldungen
- bei Problemen am besten verfolgen mit `tail`, z.B. `tail -f /var/log/messages`
  - auch: Yast – Andere – Systemprotokoll anzeigen

Logging wird i.a. vom **syslogd** übernommen, bei SUSE ab 9.3: **syslog-ng**

```

What is syslog-ng?
=====
Syslog-ng tries to fill the gaps original syslogd's were lacking:
* powerful configurability
* filtering based on message content
* message integrity, message encryption (not yet implemented in 1.4.x)
* portability
* better network forwarding

Aus: /usr/share/doc/packages/syslog-ng/README
  
```

### syslogd Konfigurationsdatei `/etc/syslog.conf`

Zeilen: `facility.priority action`

- `facility = auth | authpriv | cron | daemon | kern | lpr | mail | mark | news | syslog | user | uucp | local0 | ... local7`
- `priority = debug | info | notice | warning | err | crit | alert | emerg`
- `action =`
  - eine Log-Datei (beginnend mit `/`)



- Named Pipe (beginnend mit `|`)
- Terminal (`/dev/...`)
- entfernter Rechner (`@Rechnername`)
- eingeloggte Benutzer (`User1, User2, ...` jeder: `*`)

viele erklärte Beispiele in der Manpage von `syslog.conf`

**syslog-ng Konfigurationsdatei:** `/etc/syslog-ng/syslog-ng.conf`

andere Syntax

zunächst **Definitionszeilen** – beginnend mit **source** | **filter** | **destination**

- meist nur eine source-Zeile

dann mit **log** beginnende Zeilen, die bestimmen,

- was (sources | filter)
- wohin (destination)

protokolliert werden soll

ein Beispiel:

```
source      src      { internal(); unix-dgram("/dev/log"); };
filter      f_mail   { facility(mail); };
destination mail    { file("/var/log/mail"); };
log         { source(src); filter(f_mail); destination(mail); };
```

Einführung in `syslog-ng`, von Winfried Mueller: <http://www.wikidorf.de/reintechnisch/Inhalt/SyslogNGEinfuehrung>

**logger**: Kommandointerface zum `syslog`

**logrotate**: Kommando zum Rotieren, Komprimieren von Logdateien

- wird meist regelmäßig aufgerufen (openSUSE: `/etc/cron.daily/logrotate`)

## 2 Grundlegende Netzkonfiguration

### 2.1 Grundlegendes

#### IP-Adressen

- in einem IP-Netzwerk ist jedem Rechner (genauer: jeder Netzwerkkarte) eine **IP-Adresse** zugeordnet
  - die IP-Adresse ist bei IP, Version 4 (heutiger Standard) a.b.c.d (a,..d=0...255), also 4x8=32 Bit

#### Subnetze, Router (Gateway), Netzwerkmasken

- eine **Netzwerkmaske (netmask)** bestimmt, zu welchem **Subnetz** eine IP-Adresse gehört
  - dabei werden IP-Adresse und **Netzwerkmaske** binär miteinander kombiniert (AND), z.B.:  
IP-Adresse 134.245.123.40 und Netzwerkmaske 255.255.0.0 ergibt das Subnetz beginnend mit 134.245

|               | Dezimal        | Binär                                     |
|---------------|----------------|---|
| IP-Adresse    | 134.245.123.40 | 10000110 . 11110101 . 01111011 . 00101000 |
| Netzwerkmaske | 255.255.0.0    | 11111111 . 11111111 . 00000000 . 00000000 |
| Subnetz       | 134.245.0.0    | 10000110 . 11110101 . xxxxxxxx . xxxxxxxx |

- Schreibweise für das Netz: 134.245.123.40/16 (die ersten 16 Bits sind fest)

ein anderes, etwas komplizierteres Beispiel:

IP-Adresse 134.245.123.40 und Netzwerkmaske 255.255.255.192 ergibt das Subnetz aller IP-Adressen zwischen 134.245.123.0 und 134.245.123.63:

|               | Dezimal                          | Binär                                     |
|---------------|----------------------------------|---|
| IP-Adresse    | 134.245.123.40                   | 10000110 . 11110101 . 01111011 . 00101000 |
| Netzwerkmaske | 255.255.255.192                  | 11111111 . 11111111 . 11111111 . 11000000 |
| Subnetz       | 134.245.123.0-<br>134.245.123.63 | 10000110 . 11110101 . 01111011 . 00xxxxxx |

- Schreibweise für das Netz: 134.245.123.40/26 (die ersten 26 Bits sind fest)

- Rechner in einem Subnetz können sich gegenseitig erreichen
- eine Routingtabelle legt fest, wie andere Rechner erreicht werden können
  - meist ein Eintrag für alle Rechner außerhalb des eigenen Subnetzes (Angabe des Standardgateways)
    - wichtig: der Gateway muß im eigenen Subnetz sein!

#### spezielle IP-Adressen:

- 127.0.0.1/8: beziehen sich auf lokalen Rechner, meistens 127.0.0.1 für localhost (Loopback-Interface)
- Adressbereiche für private Netze:
  - 10.0.0.0 - 10.255.255.255
  - 172.16.0.0 - 172.31.255.255
  - 192.168.0.0 – 192.168.255.255

mehr: <http://de.wikipedia.org/wiki/IP-Adresse>

## 2.2 benötigte Angaben für Rechner im LAN

|                    | für Uni-Rechner   | zu Hause   |
|--------------------|---|--|
| <b>Rechnername</b> | Absprache mit Netzbeauftragten des Instituts  | beliebig   |
| <b>IP-Adresse</b>  | 134.245.x.y - Absprache mit Netzbeauftragten des Instituts<br>Klinikum: 194.94.160.x ... 194.94.191.x | - (wird vom ISP dynamisch zugewiesen)<br>private IP-Adressen für vernetzte PCs,<br>z.B. 10.x.y.z |
| <b>Domainname</b>  | xxx.uni-kiel.de - Nachfrage beim Netzbeauftragten des Instituts oder RZ                               | -  |
| <b>Nameserver</b>  | 134.245.1.36<br>134.245.1.2<br>134.245.10.7   | Vom ISP<br>falls übers RZ: 134.245.1.36, 134.245.1.2,<br>134.245.10.7                            |
| <b>Gateway</b>     | i.a. 134.245.1.200<br>Ausnahmen: Techn. Fak., Klinikum,<br>Weltwirtschaft,...                         | Vom ISP<br>falls übers RZ: 134.245.1.200   |
| <b>Netmask</b>     | i.a. 255.255.0.0<br>Ausnahmen wie beim Gateway  | -  |

s.a. [www.uni-kiel.de/rz/netz/adressen/](http://www.uni-kiel.de/rz/netz/adressen/)

## 2.3 Konfiguration mit Yast

Konfiguration einer Netzwerkkarte:

- [YaST - Netzwerkgeräte – Netzwerkeinstellungen](#) (2.3.1)

Konfiguration für Modem / ISDN / DSL über [YaST – Netzwerkgeräte – ...](#)

- Beschreibung im SUSE Online-Handbuch

Probleme mit WLAN-Karten unter openSUSE:

- [http://de.opensuse.org/WLAN\\_Probleme](http://de.opensuse.org/WLAN_Probleme)
- [http://en.opensuse.org/HCL/Network\\_Adapters\\_\(Wireless\)](http://en.opensuse.org/HCL/Network_Adapters_(Wireless))



### 2.3.1 Konfiguration einer Netzwerkkarte:

[YaST - Netzwerkgeräte – Netzwerkeinstellungen](#)

#### Reiter Globale Optionen



SUSE unterscheidet zwischen zwei verschiedenen Methoden für den Netzwerkaufbau

- *Benutzergesteuert mithilfe von NetworkManager*
  - Applet auf der Kontrollleiste (   )
  - sinnvoll bei Laptops mit wechselnden Netzwerken

- *Traditionelle Methode mit ifup*
  - hier im Kurs auswählen

## Reiter Übersicht

erkannte Netzwerkkarten werden aufgelistet, können über den Schalter *Bearbeiten* konfiguriert werden; hier:

- *PRO/Wireless LAN2100 3B Mini PCIAdapter*
- *NetXtreme BCM5705M Gigabit Ethernet*

hier im Kurs:

- *PRO/Wireless LAN2100 3B Mini PCIAdapter* über Schalter *Löschen* deaktivieren
- *NetXtreme BCM5705M Gigabit Ethernet* über Schalter *LBearbeiten* konfigurieren:

Schalter *Bearbeiten* ruft das Fenster *Netzwerkkarten-Einrichtung* auf; im Reiter *Adresse* eintragen:

- **Statisch zugewiesene IP-Adresse** auswählen
- **IP-Adresse:** 134.245.123.NummerDesNotebooks
- **Subnetzmaske:** 255.255.0.0
- **Hostname:** notebookNummerDesNotebook

**Netzwerkkarten-Einrichtung**

Allgemein | **Adresse** | Hardware

Gerätetyp: Ethernet | Konfigurationsname: eth2

Keine IP Adresse (Für verbundene Geräte)  
 Dynamische Adresse (DHCP)  
 **Statisch zugewiesene IP-Adresse**

IP-Adresse: 134.245.123.25 | Subnetzmaske: 255.255.0.0 | Hostname: notebook25

Zusätzliche Adressen

- Damit haben wir:

**Netzwerkeinstellungen**

Globale Optionen | **Übersicht** | Hostname/DNS | Routing

| Name                                    | IP-Adresse         |
|---|--------------------|
| PROWireless LAN 2100 3B Mini PCIAdapter | Nicht konfiguriert |
| NetXtreme BCM5705M Gigabit Ethernet     | 134.245.123.25     |

**NetXtreme BCM5705M Gigabit Ethernet**  
 MAC : 00:0d:56:3c:f9:60

- Geräteiname: eth0
- Beim Booten automatisch gestartet
- IP-Adresse: 134.245.123.25, Subnetzmaske 255.255.0.0

Hinzufügen | Bearbeiten | Löschen

## Reiter Hostname/DNS

- *Hostname*: notebookNummerDesNotebooks, *Domainname*: kurs.uni-kiel.de
- *Hostnamen über DHCP ändern* deaktivieren
- *Ändere /etc/resolv.conf manuell* auswählen
- *Nameserver*: 134.245.1.2 134.245.1.36 134.245.10.7
- *Domainsuche*: kurs.uni-kiel.de

**Netzwerkeinstellungen**

Globale Optionen    Übersicht    **Hostname/DNS**    Routing

Hostname und Domänenname

Hostname:     Domänenname:

Hostnamen über DHCP ändern    Keine Schnittstelle mit DHCP

Hostname in /etc/hosts schreiben

Ändere /etc/resolv.conf manuell

Nameserver und Domänensuchliste

Nameserver 1:     Domänensuche:

Nameserver 2:

Nameserver 3:

DNS Daten über DHCP aktualisieren    Keine Schnittstelle mit DHCP

## Reiter Routing

*Standardgateway*: 134.245.1.200

## 2.4 betroffene Systemdateien

- Dateien unter /etc/sysconfig/network (SUSE-spezifisch)
- /etc/hosts
  - lokale „Zuordnungs-Tabelle“ für IP-Namen und IP-Adressen  
insbesondere Eintrag für den Rechner wie z.B.

```
134.245.123.40 notebook40.kurs.uni-kiel.de notebook40
```

- /etc/HOSTNAME

- Rechnername, z.B.

```
notebook40.kurs.uni-kiel.de
```

- /etc/resolv.conf

- IP-Namensauflösung, insbesondere DNS Server, z.B.

```
domain kurs.uni-kiel.de
nameserver 134.245.1.2
nameserver 134.245.1.36
nameserver 134.245.10.7
search kurs.uni-kiel.de
```

## 3 Software und Patche

Software wird als gebündeltes Softwarepaket angeboten:

- **RPM-Format** - Standard bei SUSE und den meisten anderen Distributionen
- **Debian-Format** - bei Debian, Ubuntu,..
- **tgz-Archiv** (komprimiertes tar-Archiv)

fremde Softwarepakete lassen sich mit der Software Alien ins rpm-Format umwandeln

- Vorteil insbesondere für tar-Pakete: die Software läßt sich leicht wieder deinstallieren

RPM- und Debian Pakete enthalten auch

- Informationen zur Software
- Softwareabhängigkeiten
- Skripte, die vor/nach der Installation ablaufen

nicht aber tar-Pakete

Software gibt es als ablauffähiges Paket oder aber als Sourcepaket, das noch zu kompilieren ist

### Installation/Deinstallation/Update

mit Hilfe einer Utility wie **YaST** bei SUSE

es gibt auch noch weitere Softwareverwaltung-Tools wie z.B

- Kpackage, der KDE Paketmanager,
- GnoRPM - das GNOME Äquivalent
- Synaptic bei Debian

oder bei SUSE mit dem Befehl **zypper** (Debian: apt-..., aptitude)

oder direkt mit Hilfe des Befehls **rpm** (Debian: dpkg)

- bei SUSE danach /sbin/SUSEconfig aufrufen

neuerdings 1-Klick-Installationen möglich

### Installationsquellen (Software Repositories / Paketdepots / Paketquellen)

... sind „Softwarelager“ - hier gibt es Softwarepakete

... können sein

- CD/DVD
- ISO-Abbild einer CD/DVD
- Verzeichnis auf einem ftp- / Web- Server
- ein Verzeichnis auf dem Rechner
- ...

... beinhalten i.a. nicht nur Pakete, sondern auch

- Kurzinformationen zur Software und
- öffentlichen Schlüssel zum Verifizieren der Pakete

... die Softwareverwaltung (SUSE: YaST/zypper) legt fest, welche benutzt werden

### 3.1 Wo gibt es Software?

beim Anbieter der Distribution; SUSE LINUX (auch Verweise auf Externe):

- *Zusätzliche\_Paketquellen: [http://de.opensuse.org/Zusätzliche\\_Paketquellen](http://de.opensuse.org/Zusätzliche_Paketquellen)*
- *[http://de.opensuse.org/Zusätzliche\\_Paketquellen/Externe\\_Quellen](http://de.opensuse.org/Zusätzliche_Paketquellen/Externe_Quellen)*
- *[http://en.opensuse.org/Additional\\_YaST\\_Package\\_Repositories](http://en.opensuse.org/Additional_YaST_Package_Repositories)*
- Liste der Spiegelservers der aktuellen stabilen SUSE Version:  
*[http://de.opensuse.org/Spiegelservers\\_der\\_stabilen\\_Version](http://de.opensuse.org/Spiegelservers_der_stabilen_Version)*
- *Additional YaST Package Repositories: [http://en.opensuse.org/YaST\\_package\\_repository](http://en.opensuse.org/YaST_package_repository)*

beim Hersteller der Software wie z.B.

- Linux-Kernel: [www.kernel.org/](http://www.kernel.org/)
- KDE Software: [www.kde.org/](http://www.kde.org/)

- GNOME: [www.gnome.org](http://www.gnome.org)

Suchmöglichkeiten:

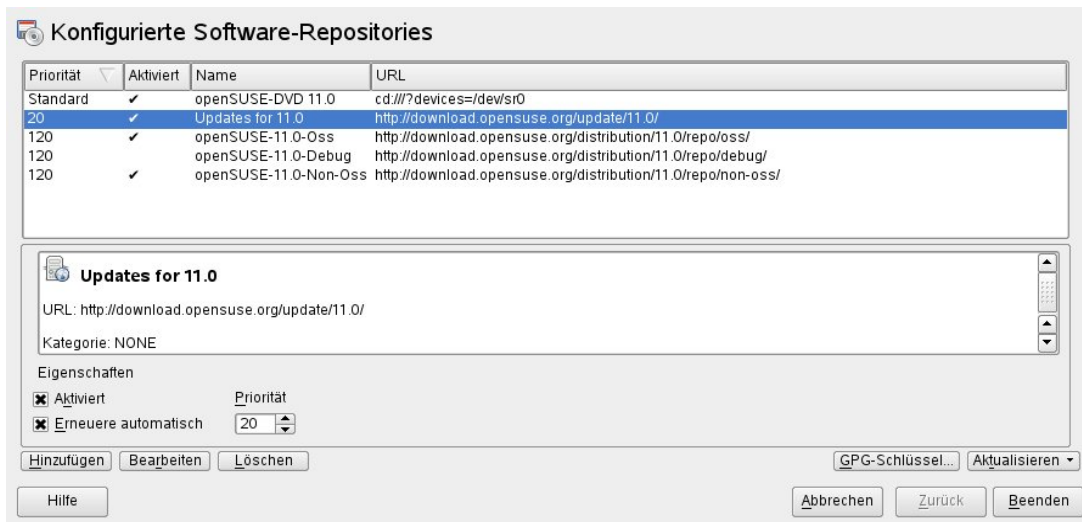
- openSUSE Software Suche: <http://packages.opensuse-community.org/>
- <http://sourceforge.net/> großes Software-Archiv
- <http://www.rpmseek.com> hier kann man auch nach bestimmten Programmen oder fehlenden Bibliotheken suchen
- <http://www.rpmfind.net> ebenfalls für Suche geeignet
- oder aber - wie oft - bei google ([www.google.de/linux](http://www.google.de/linux))

### 3.2 Installation von Software mit YaST

- Auflösungen von Software-Abhängigkeiten
- Überprüfung von Signaturen
- Protokoll nach `/var/log/YaST2/`

#### 3.2.1 Anpassung der Installationsquellen

YaST - Software - Software-Repositories



Installationsquellen haben den Status

- aktiviert – nicht aktiviert
- automatisch erneuern – nicht automatisch erneuern
  - die SUSE-Quellen oss,non-oss,debug ändern sich nicht – nicht automatisch erneuern

Sinnvoll:

- CD/DVD als Installationsquelle deaktivieren (ist ja nicht immer im Laufwerk)
- evtl: ISO-Image der openSUSE DVD hinzufügen
- bei den Servern für die SUSE-Quellen einen Spiegelserver auswählen, z.B. die TU Chemnitz:

|                           |   |
|---------------------------|---|
| Updates                   | <a href="ftp://ftp.tu-chemnitz.de/pub/linux/opensuse/update/11.0">ftp://ftp.tu-chemnitz.de/pub/linux/opensuse/update/11.0</a>                                       |
| Open Source Pakete (oss)  | <a href="ftp://ftp.tu-chemnitz.de/pub/linux/opensuse/distribution/11.0/repo/oss">ftp://ftp.tu-chemnitz.de/pub/linux/opensuse/distribution/11.0/repo/oss</a>         |
| Add On Software (non-oss) | <a href="ftp://ftp.tu-chemnitz.de/pub/linux/opensuse/distribution/11.0/repo/non-oss">ftp://ftp.tu-chemnitz.de/pub/linux/opensuse/distribution/11.0/repo/non-oss</a> |

Hinzufügen-Button (entspricht YaST - Software – Zusatz-Produkt):



- Community/Gemeinschafts-Repositories: schon vordefinierte Software-Repositories zur Auswahl

### 3.2.2 Software Installation / Deinstallation mit YaST

YaST - Software - Software installieren oder löschen

- Auswahl verschiedener Ansichten übers Listenfeld *Filter*
- Symbole für Paketstatus (s.a. Hilfe-Symbole)
  - Status änderbar durch Mausklick oder Kontextmenü

### Übung ~ Softwareverwaltung mit Yast

#### 3.3 1-Klick-Installationen (One-Click-Install)

... ermöglichen Installationen mit einem Mausklick über einen Webbrowser

eine sogenannte ymp-Datei enthält Anweisungen,

- welche Software-Quellen benutzt werden sollen
- welche Software installiert werden soll

Installation geschieht letztendlich mit YaST

- ändert man nichts, sind die Software-Quellen auch nach der Installation aktiviert

Auch per Kommandozeile möglich:

```
/sbin/OCICLI ymp-Datei
```

Beispiel: wer den Nvidia Treiber installieren möchte ([http://de.opensuse.org/Proprietäre\\_NVIDIA-Grafiktreiber](http://de.opensuse.org/Proprietäre_NVIDIA-Grafiktreiber)):

```
/sbin/OCICLI http://opensuse-community.org/nvidia.ymp bzw.
```

```
/sbin/OCICLI http://opensuse-community.org/nvidia-legacy.ymp
```

Multimedia unter Linux

- aus patentrechtlichen Gründen bietet ein frisch installiertes openSUSE nicht viele Multimediämöglichkeiten
- um Multimedia nutzen zu können, müssen viele Softwarepakete nachinstalliert werden
  - Möglichkeit, diese per *OneClick Install* nachzuinstallieren:  
<http://opensuse-community.org/Multimedia>

#### 3.4 Softwarekorrekturen(Patche)

SUSE stellt korrigierte Softwarepakete zur Verfügung (update Repository)

- der Softwareverwaltung muss also zunächst das update Repository bekannt gemacht werden (3.2.1)
  - Tipp: falls kein oder nur ein langsamer Netzanschluß: Patch-CD erstellen und als Updatequelle angeben (<http://>



*de.opensuse.org/SDB:Patch\_CD\_erstellen)*

- s.a. 3.4.4 Grundsätzliches zu Software-Updates (Patches)

Unterscheidung in

- Sicherheit - beseitigen ernsthafte Sicherheitsgefahren - auf jeden Fall installieren
- empfohlen - beheben Probleme, die zu Schäden an Ihrem Computer führen können
- optional - beheben Probleme ohne Sicherheitsrelevanz, oder bieten Verbesserungen

Installation mit








- dem openSUSE Updater-Miniprogramm (3.4.1) oder
- YOU (3.4.2) oder
- mit dem zypper Befehl (3.6) oder
- automatisches Online-Update (3.4.3)

**Die Pakete werden mit Hilfe des SUSE GnuPG-Schlüssels verifiziert.**

Information über sicherheitsrelevante Korrekturen über Mailingliste *SUSE-security-announce*

### 3.4.1 *openSUSE Updater* Applet auf der Kontrollleiste:

zeigt den Aktualisierungs-Stand anhand unterschiedlicher Symbole an:

|   |   |  |
|---|---|--|
|    | Grüner Geeko-Kopf mit grünen Pfeilen                                | Keine Patches oder neue Versionen verfügbar  |
|    | Grüner Geeko-Kopf mit grünen Pfeilen (KDE)/<br>Grauer Geeko (GNOME) | openSUSE Updater ist ausgelastet (es wird z. B. nach Updates gesucht oder Software installiert)  |
|  |   |  |
|  | Rotes Dreieck mit Ausrufezeichen                                    | Sicherheits-Patches verfügbar  |
|  | Orangefarbener Stern mit Pfeil                                      | Empfohlene/optionale Patches und/oder neue Versionen sind verfügbar  |
|  | Gelbes Dreieck mit Ausrufezeichen                                   | Ein Fehler ist aufgetreten.  |
|  | Blauer Kreis mit Fragezeichen                                       | Es ist kein Aktualisierungs-Repository definiert. Wenn Sie in dieser Phase auf den openSUSE Updater klicken, werden Sie gefragt, ob nach Aktualisierungen gesucht werden soll. Wenn Sie zustimmen, wird das YaST-Modul Konfiguration für Online-Aktualisierung gestartet |

Meldung, wenn neue Updates vorhanden

konfigurierbar übers Kontextmenü – insbesondere wie oft geprüft wird

sinnvoll für Arbeitsplatzrechner

falls Applet nicht vorhanden: es wird mit dem Befehl `opensuseupdater-kde` gestartet

### 3.4.2 YaST-Online-Update (YOU)

Yast - Software – Online-Aktualisierung

Farbe kennzeichnet Art des Updates:

- rot: sicherheitsrelevanter Update
- blau: empfohlener Update

### 3.4.3 automatisches Online-Update

Yast - Software – Automatisches Online-Update

erzeugt einen Cron-Job (6.6 Cronjobs), der das Kommando

```
zypper --quiet up -y -t patch --skip-interactive
```

ausführt

### 3.4.4 Grundsätzliches zu Software-Updates (Patches)

- Wichtig: vorm Update immer zumindest die Konfigurationsdateien unter /etc und /var/lib sichern!
- Nach einem Update Konfigurationsdateien überprüfen: ist neue Mail für root da?

#### Was ist eigentlich ein Update eines rpm-Paketes?

= Ersetzen einer alten Softwareversion durch eine neue,

- wobei Konfigurationsdateien beibehalten oder gesichert werden:  
eine angepasste Konfigurationsdatei wird
  - **beibehalten**,
    - falls sie in den Softwarepaketen nicht verändert wurde
    - oder aber im Paket ausdrücklich dieses festgelegt wurde  
dann neue Default-Konfigurationsdatei mit Endung *.rpmnew*

rpm-Pakete:  
nach Updates immer nach Dateien mit Endungen  
*.rpmorig*, *.rpmsave* und *.rpmnew*  
suchen, Konfigurationsdateien bei Bedarf ändern,  
anschließend *.rpmxxx*-Dateien löschen

- sonst **gesichert** als *x.rpmsave* oder *x.rpmorig* und die neue Standardkonfigurationsdatei wird benutzt  
mit Hilfe des Scripts `/etc/init.d/rpmconfigcheck` können solche Konfigurationsdateien leicht aufgespürt werden

### 3.5 Update auf eine neue SUSE-Version

- Den PC mit der SUSE CD/DVD booten
- Update des bestehenden Systems auswählen

## 3.6 Software-Verwaltung mit zypper-Befehlen

Doku: zypper Manpage, Zypper Anleitung: <http://de.opensuse.org/Zypper/Anleitung>

### Installationsquellen

```
zypper repos
zypper service-add --no-refresh --type YaST \
ftp://ftp.hs.uni-hamburg.de\
/pub/mirrors/opensuse/opensuse/distribution//10.2/repo/oss/ "main repository"
zypper service-delete http://ftp.chg.ru/pub/Linux/SuSE/suse/update/10.2
zypper service-rename "main repository" oss
```

### Software

```
zypper search kernel
zypper search -details kernel      # Angabe sämtlicher Versionen
zypper info kernel
zypper install kernel-source      # install zum Installieren, aber auch zum Updaten
zypper remove linux-kernel-headers-2.6.18.2-3
```

### Patche

```
zypper patch-check      #Anzahl und Art
zypper patches          #Liste aller Patche
zypper pch |grep Needed|grep -v "Not Needed"  # Nur die zu installierenden Patche
zypper patch-info xorg-x11-server
zypper list-updates     # Listet benötigte Patche --type package: auch neuere Versionen (z.B. von Packman)
zypper update
```

### Übung ~ Softwareverwaltung mit zypper

## 3.7 rpm-Software-Pakete

RPM = Red Hat Package Manager

- in einem rpm-Paket stehen
  - die eigentliche Software
  - Informationen zur Software
  - Scripte, die vor oder nach der Installation ausgeführt werden
  - Paketabhängigkeiten
- rpm-Pakete
  - werden in einer eigenen Datenbank verwaltet
  - sind leicht wieder zu deinstallieren
  - Updatemöglichkeit

### 3.7.1 Überprüfung der Authentizität von Paketen

- Ein signiertes rpm-Paket lässt sich leicht überprüfen durch

```
rpm --checksig rpmPaket
```

Voraussetzung: der zum Signieren verwendete Schlüssel ist bekannt (s. 3.7.2)
- Die md5-Summe eines Softwarepaketes lässt sich mit dem Befehl

```
md5sum SoftwarePaket
```

anzeigen. Sie sollte mit der auf dem FTP-Server,... übereinstimmen.

### 3.7.2 Authentifizierung von Paketen und Mails durch Signaturen

**gpg = GNU Privacy Guard** - Software zum Verschlüsseln und Signieren

- Pakete und Mails können gpg/pgp-Signaturen zur Authentifizierung enthalten
- gpg/pgp-Schlüssel einer Benutzer-ID:
  - geheimer Schlüssel zum Signieren
  - öffentlicher Schlüssel zum Authentifizieren
- zur Authentifizierung ist es notwendig, den öffentlichen Schlüssel der zum Signieren verwendeten Benutzer-ID zu besitzen
  - SUSE:

|                               | Benutzer-ID der Signatur                 |
|-------------------------------|--|
| Pakete                        | SUSE Package Signing Key <build@SUSE.de> |
| Mail - Security Announcements | SUSE Security Team <security@SUSE.de>    |

s.a. [www.SUSE.de/de/support/security/index.html](http://www.SUSE.de/de/support/security/index.html)

- öffentliche Schlüssel
  - im Verzeichnis `$HOME/.gnupg`
  - auflisten: `gpg --list-keys`
  - einfügen: `gpg --import SchlüsselDatei`
  - als Textdatei exportieren: `gpg --armor --export --output SchlüsselDatei Benutzer-ID`  
z.B.: `gpg --armor --export --output SUSE-gpg.txt build@SUSE.de`
- gpg-Schlüssel können auch in die rpm-Datenbank importiert werden
  - einfügen: `rpm --import SchlüsselDatei`
  - auflisten: `rpm -qa gpg-pubkey*`
  - ansehen: `rpm -qi gpg-pubkey-xyz`
  - löschen: `rpm -e gpg-pubkey-xyz`
- mehr zu gpg: [www.gnupg.de](http://www.gnupg.de), insb. dort: *Das GNU-Handbuch zum Schutze der Privatsphäre*

### 3.7.3 Einige Optionen des rpm-Befehls

| Einige Optionen des rpm-Befehls   |   |
|---|---|
| <b>Installieren / Löschen</b>   |   |
| <code>-i rpm-Paket</code>   | Installieren  |
| <code>-U rpm-Paket</code>   | Installieren, falls Software noch nicht vorhanden, sonst Updaten (s. 3.4.4) |
| <code>-F rpm-Paket</code>   | Updaten (s. 3.4.4), falls Software schon installiert                        |
| <code>-e Software</code>  | Deinstallieren  |
| <code>--import Schlüsseldatei</code>  | Importieren eines öffentlichen gpg Schlüssels                               |
| <b>Abfragen</b>   |   |
| <code>-qa</code>  | Liste der installierten Software  |
| <code>-qa gpg-pubkey*</code>  | Liste der importierten gpg Schlüssel  |
| <code>-qf Dateiname</code>  | Softwareliste, die die Datei enthält  |
| <code>-q --whatrequires Eigenschaft</code>  | Softwareliste, die die Eigenschaft (Library ...) benötigt                   |
| <code>-q --whatprovides Eigenschaft</code>  | Softwareliste, die die Eigenschaft liefert                                  |
| <b>Informationen</b>  |   |
| <ul style="list-style-type: none"> <li>• zum rpm-Paket <code>-q Optionen -p rpm-Paket</code></li> <li>• zur installierten Software <code>-q Optionen SoftwareName</code></li> </ul> |   |

|                                   |  |
|-----------------------------------|--|
| <code>... -i</code>               | Informationen  |
| <code>... -l</code>               | Dateiliste   |
| <code>... -s</code>               | Dateiliste mit Status  |
| <code>... -d</code>               | Liste der Dokumentationsdateien                                  |
| <code>... -c</code>               | Liste der Konfigurationsdateien                                  |
| <code>... --requires</code>       | benötigte Eigenschaften  |
| <code>... --provides</code>       | Eigenschaften, die die Software zur Verfügung stellt             |
| <code>... --scripts</code>        | Scripte, die bei Installation / Deinstallation ausgeführt werden |
| <b>Sonstiges</b>                  |  |
| <code>--checksig rpm-Paket</code> | Authentifizieren des Paketes                                     |
| <code>--rebuilddb</code>          | Neuaufbau der rpm Datenbank (bei Problemen benutzen)             |

### Übung ~ Informationen zu installierter Software mit rpm anzeigen

## 4 Dateien, Plattenpartitionen, Mounten

- Dateien sind in einem Dateibaum, beginnend bei / (root) angeordnet
- Datenträger wie Plattenpartitionen, CDrom-Laufwerke, Disketten werden durch *Mounten* in den Dateibaum eingehängt
- für den Benutzer ist so verborgen, auf welchem Datenträger sich eine Datei befindet
- Dateien sind Attribute wie insbesondere Eigentümer, Zugriffsrechte und Änderungsdatum zugeordnet

### 4.1 Dateizugriffsrechte

einer Datei sind

- dem Eigentümer der Datei
- der zugeordneten Gruppe
- allen anderen Kennungen

Zugriffsrechte zugeordnet (Lesen/Schreiben/Ausführen)

| Befehle            |                                       |
|--------------------|---------------------------------------|
| <code>chmod</code> | Zugriffsrecht ändern                  |
| <code>chown</code> | Eigentümer der Datei ändern           |
| <code>chgrp</code> | Gruppenzugehörigkeit der Datei ändern |

| Zugriffsbit                   | Bedeutung   |
|-------------------------------|---|
| 4 (r)                         | Lesen   |
| 2 (w)                         | Schreiben   |
| 1 (x)                         | Ausführen (bei Verzeichnissen: ins Verzeichnis wechseln)  |
| 1*** (t)                      | Sticky-Bit, nur bei Verzeichnissen: jeder darf nur seine eigenen Dateien im Verzeichnis löschen |
| 4***<br>(s S beim Eigentümer) | SUID-Bit: Programm wird unter der UID des Dateieigentümers ausgeführt                           |
| 2***<br>(s S bei der Gruppe)  | SGID-Bit: Programm wird unter der GID der Gruppe ausgeführt                                     |

## ein paar Beispiele

|          |          |          |
|----------|----------|----------|
| <b>u</b> | <b>g</b> | <b>o</b> |
|----------|----------|----------|

u: Besitzer

g: Gruppe, o: Rest

## Verzeichnis

|                                  |              |           |
|----------------------------------|--------------|-----------|
| Besitzer darf alles              | drwx r-x --- |           |
| Gruppe darf Inhalt auflisten, cd | 111 101 000  |           |
| der Rest darf nichts             | 7 5 0        | chmod 755 |

## Verzeichnis mit Sticky Bit

|  |                 |            |
|--|-----------------|------------|
| Besitzer darf alles  | drwx rwx -T     |            |
| Gruppe darf alles, bis auf fremde Dateien löschen, Rest nichts | 001 111 101 000 |            |
|  | 1 7 5 0         | chmod 1750 |

## Datei

|                               |             |           |
|-------------------------------|-------------|-----------|
| Besitzer darf lesen,schreiben | rw- r-- r-- |           |
| alle anderen dürfen lesen     | 110 100 100 |           |
|                               | 6 4 4       | chmod 644 |

## Programm

|                               |             |           |
|-------------------------------|-------------|-----------|
| Besitzer darf alles,          | rwX r-x r-x |           |
| alle anderen dürfen ausführen | 111 101 101 |           |
|                               | 7 5 5       | chmod 755 |

## Programm mit suid Bit

|   |                 |            |
|---|-----------------|------------|
| alle dürfen ausführen,                          | rws r-x r-x     |            |
| Besitzer zusätzlich schreiben.                  | 100 111 101 101 |            |
| Programm läuft im Kontext des Dateibesitzers ab | 4 7 5 5         | chmod 4755 |

## Anzeigen aller SUID- und SGID-Programme:

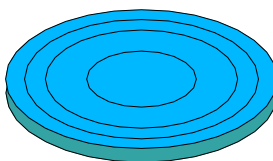
```
find / -type f \( -perm -4000 -o -perm -2000 \) -exec ls -lg {} \;
```

## Weltweit beschreibbare Dateien, Dateien ohne gültigen Eigentümer finden

|                                      |  |
|--------------------------------------|--|
| weltweit beschreibbare Dateien       | <code>find / -type f -perm -2 -exec ls -l {} \;</code>     |
| weltweit beschreibbare Verzeichnisse | <code>find / -type d -perm -2 -exec ls -ld {} \;</code>    |
| Dateien ohne Eigentümer              | <code>find / -nouser -o -nogroup -exec ls -ld {} \;</code> |

## 4.2 Plattenpartitionen

**Partition**=zusammenhängender Plattenbereich zwischen 2 Zylindern, dem durch Formatieren ein Dateisystem aufgesetzt wird (dann auch als **Laufwerk**, **Datenträger** bezeichnet)

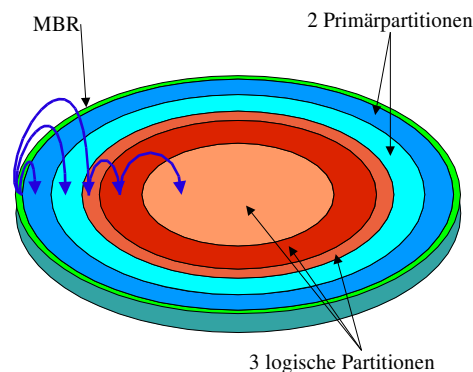


- Aufteilung der Festplatte in kleinere Einheiten.
- Jedes Betriebssystem braucht seine eigene Partition(en)
- Partitionen dürfen sich nicht überschneiden.
- Eine Festplatte muß nicht notwendigerweise vollständig partitioniert werden, Partitionen können auch später noch hinzugefügt werden.

### Partitionstypen

Eine Platte wird in 1 bis 4 **Hauptpartitionen** unterteilt:

- bis zu 4 **Primärpartitionen** und
- maximal eine **erweiterte Partition**
  - kann in weitere **logische Partitionen** unterteilt werden



### 4.2.1 Bezeichnungen für Platten, Partitionen bei Linux

| IDE-Platten: (*) |  | /dev/hda, /dev/hdb, /dev/hdc, /dev/hdd   |
|------------------|--|--|
|                  | Primärpartitionen/erweiterte Partition | /dev/hda1, /dev/hda2, /dev/hda3, /dev/hda4,<br>/dev/hdb1, /dev/hdb2, /dev/hdb3, /dev/hdb4,<br>/dev/hdc1, ... |
|                  | logische Partitionen                   | /dev/hda5, /dev/hda6, ...<br>/dev/hdb5, ...  |
| SCSI-Platten:    |  | /dev/sda, /dev/sdb, ...  |
|                  | Partitionen                            | /dev/sda1, /dev/sda2, ... /dev/sdb1, ...   |

(\*) Bezeichnungen für IDE-Platten bei openSUSE ab Version 10.3: **sd** statt **hd** (also wie SCSI-Platten)

### 4.2.2 Partitionen für Linux

| Minimum            | häufig             | Bedeutung   |
|--------------------|--------------------|---|
| / (Root-Partition) | / (Root-Partition) | Das System  |
| swap-Partition     | swap-Partition     | virtueller Speicher                                 |
|                    | /boot (*)          | Linux Kernel, LILO-Dateien                          |
|                    | /usr               | Standard-Programme                                  |
|                    | /usr/local         | zusätzliche Programme                               |
|                    | /opt/              | zusätzliche Programme (z.B. KDE)                    |
|                    | /var               | insb. temporäre Dateien, Log-Dateien                |
|                    | /var/mail          | (temp.) Mailboxen der Benutzer                      |
|                    | /home              | Benutzerbereiche                                    |
|                    | /tmp               | für temporäre Dateien                               |
|                    | ...                | weitere Benutzer-Bereiche, für Spezialsoftware, ... |

### 4.2.3 Partitionen anlegen, löschen

- bei SUSE leicht mit *YaST - System – Partitionieren*
- sonst z.B. mit `fdisk`:

```

root@host:~ fdisk /dev/hda
Befehl (m für Hilfe): m
Befehl  Bedeutung
a      (De)Aktivieren des bootfähig-Flags
b      »bsd disklabel« bearbeiten
c      (De)Aktivieren des DOS Kompatibilitätsflags
d      Eine Partition löschen
l      Die bekannten Dateisystemtypen anzeigen
m      Dieses Menü anzeigen
n      Eine neue Partition anlegen
o      Eine neue leere DOS Partitionstabelle anlegen
p      Die Partitionstabelle anzeigen
q      Ende ohne Speichern der Änderungen
s      Einen neuen leeren »Sun disklabel« anlegen
t      Den Dateisystemtyp einer Partition ändern
u      Die Einheit für die Anzeige/Eingabe ändern
v      Die Partitionstabelle überprüfen
w      Die Tabelle auf die Festplatte schreiben und das Programm beenden
x      Zusätzliche Funktionen (nur für Experten)

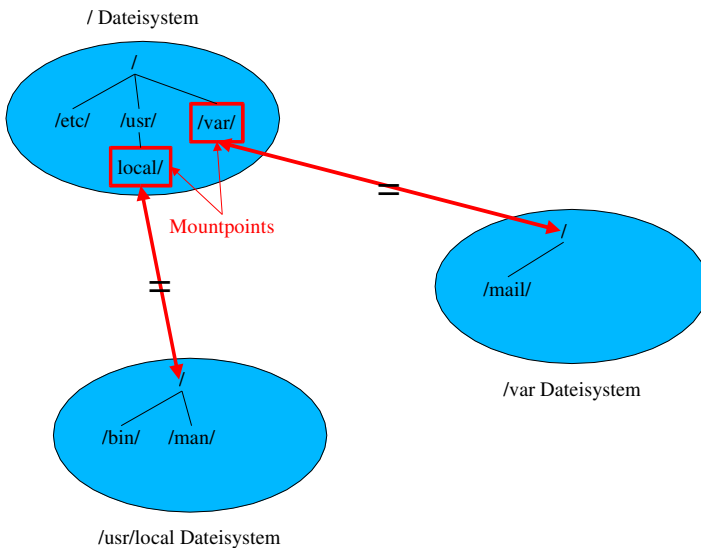
```

`fdisk -l Platte` listet die Partitionen einer Platte auf

- ein Dateisystem anlegen mit `mkfs`, überprüfen/reparieren mit `fscck`
- Alternativen: `cfdisk`, `parted`

### 4.3 Mounten von Platten und anderen Datenträgern

- Um auf Partitionen zugreifen zu können, muß die Partition zunächst **gemountet** werden.



- ein Datenträger wie ein Dateisystem / CDROM-Laufwerk / Diskette wird in ein Verzeichnis (=Mountpoint) gemountet
- die zu mountenden Datenträger können auch auf einem entfernten Rechner liegen
  - nfs-Mounts bei Linux-/Unix (NFS=Network File System)
  - smb-Mounts bei Windows-Freigaben (SMB=Server Message Block System)
- die Mountpoints sollten keine Dateien / Verzeichnisse enthalten
- `/etc/fstab`: Einträge der Datenträger mit Mountpoints
  - alles, was beim Booten gemountet werden soll

#### Mounten/Unmounten:

```
mount [Optionen] [Gerät] [Mountpoint]
```

```
umount Mountpoint
```

Beispiele:

|                                     |  |
|-------------------------------------|--|
| <code>mount -a</code>               | alles aus <code>/etc/fstab</code> mounten                              |
| <code>mount /dev/sda6 /home1</code> | die Partition <code>/dev/sda6</code> unter <code>/home1</code> mounten |
| <code>mount /home</code>            | <code>/home</code> wie in <code>/etc/fstab</code> beschrieben mounten  |

- `mount` ohne alles zeigt die gemounteten Dateisysteme an



- nur root darf mounten/unmounten
  - Ausnahme: Option `user` in `/etc/fstab` für das Gerät

fstab-Beispielzeile:

```
/dev/hda1 /windows/C ntfs ro,noauto,user,gid=users,umask=0002 0 0
```

|                               |  |
|-------------------------------|--|
| <code>/dev/hda1</code>        | der zu mountende Datenträger, hier:<br>Partition |
| <code>/windows/C</code>       | Mountpoint                                       |
| <code>ntfs</code>             | Dateisystemtyp                                   |
| <code>ro,noauto,user</code> , | Optionen   |
| ...                           |  |
| <code>0</code>                | für Backup mit <code>dump</code>                 |
| <code>0</code>                | kein <code>fsck</code> beim Booten des Rechners  |

- ein unmount klappt nur, wenn es keine *offenen* Dateien auf dem Datenträger gibt – evtl. auch mit der Option `-l`).
  - offene Dateien lassen sich mit Hilfe des Befehls `lsdf` aufspüren.

### Mounten eines iso-Images

mit Hilfe der Option `-o loop`, z.B. `mount -o loop cd.iso /mnt`

### Dateisysteme können mit der Option `nosuid` gemountet werden

- das heißt: SUID-Bits bei Programmen haben keine Bedeutung
- geringere Gefahr, daß Programme in dem Bereich als SUID-Root ablaufen können
- Insb. für `/home`, `/var`, `/tmp` /boot sinnvoll

## 4.4 Zugriff auf Windows Laufwerke

Windows-Dateisysteme:

- FAT
  - Standard bei Windows 95/98
  - unterstützt keine Zugriffsrechte
- NTFS
  - Standard bei Windows NT/2000/XP/2003
  - detaillierte Zugriffsrechte
  - kein offener Standard, Spezifikation nicht freigegeben

Linux kann

- FAT Partitionen lesen und schreiben
- NTFS Partitionen lesen, schreiben mittlerweile ausgereift!? - bei openSUSE 11.0 werden vorhandene NTFS-Partitionen automatisch mit Schreibrechten eingebunden.

Möglichkeiten des Datenaustausches zwischen Linux und Windows:

- FAT32-Partition (Nachteil: max. Dateigröße 4 Gbyte)
- ext2/ext3-Partition erfordert unter Windows Installation eines Treibers
  - Explore2fs (<http://www.chrysocome.net/explore2fs>) nur lesend
  - Virtual Volumes (<http://www.chrysocome.net/virtualvolumes>)
    - noch Beta, Nachfolger von Expore2fs, auch Schreibzugriff
  - Ext2 Installable File System Driver (<http://www.fs-driver.org>)
    - Ext2 File System Driver for Windows (<http://sourceforge.net/projects/ext2fsd>)

Projekte/Software für ntfs-Schreibzugriff unter Linux:

(Auszug aus der Wikipedia <http://en.wikipedia.org/wiki/NTFS>)

- Linux kernel 2.6: contains a driver written by Anton Altaparmakov (Cambridge University) and Richard Russon. It supports file read, overwrite and resize, in some cases.

- NTFSMount: A userspace driver with limited file and directory read/write support is available using ntfsmount,[8]
- NTFS-3G: A userspace driver with full read/write support based on ntfsmount and included in many Linux distributions. **Wird bei openSUSE 10.3 benutzt.**
- NTFS for Linux: A commercial driver with full read/write support available from Paragon.
- Captive NTFS: A 'wrapping' driver which uses Windows's own driver, ntfs.sys. This method is slower than the native drivers.

## 5 Booten von Linux

### 5.1 Der Bootvorgang beim PC

PC wird eingeschaltet

1. das BIOS (Basic Input Output System):
  - a. Initialisierung von Bildschirm und Tastatur
  - b. Hauptspeichertest
  - c. Infos aus den CMOS Werten - Datum, Zeit, wichtigste Peripherie-Geräte (insb. Festplatten) werden ausgelesen
2. Laden des Betriebssystems
  - a. 1. Festplatte (bzw. Diskette / CDROM / DVD), 1. Datensektor von 512 Byte (=MBR=Master Boot Record) wird in den Speicher geladen
  - b. Programm (Boot Loader) zu Beginn dieses Sektors wird aufgerufen

- Bootvorgang bislang unabhängig vom installierten Betriebssystem
- Gerätezugriff übers BIOS

falls es sich um einen Linux Bootloader handelt, der ein Linux System starten möchte:

- c. Dieser „first stage Boot Loader“ ruft den eigentlichen Bootloader auf
- d. Dieser „second stage Boot Loader“ lädt den Linux Kernel und bei Bedarf eine sogenannte Initial Ramdisk in den Speicher, startet den Kernel und informiert ihn dabei über diese Initial Ramdisk
- e. der Kernel mountet das Root Dateisystem und überläßt dem Init-Prozeß (s. 5.5 Neustart / Herunterfahren des Rechners / Der Init-Prozeß) das Starten des Systems

### 5.2 Die Initial Ramdisk

**Initial Ramdisk** (auch **initdisk** / **initrd**) = virtuelle Platte im Speicher, die u. U. beim Bootvorgang verwendet wird

**Problem:**

- es gibt viele verschiedene Plattentreiber, insbesondere für SCSI-Platten, die nicht alle im Kernel eingebunden sind
- der Kernel mountet mit als erstes / (das Rootdevice) und braucht so den Plattentreiber der „Rootplatte“

**eine Lösung ist die initrd:**

1. initrd wird vom Bootmanager geladen
2. vom Kernel vorübergehend als / gemountet
3. das Programm **linuxrc** auf der initrd lädt die benötigten Module in den Kernel
4. der Kernel unmountet die initrd und mountet die eigentliche Rootpartition

Dieses Verfahren wird auch schon bei der Installation verwendet

bei der Installation wird bei Bedarf automatisch eine initrd für das neue System angelegt

**Manuelles Erstellen einer Initrd bei SUSE 8.2:**

1. in `/etc/sysconfig/kernel` (früher `/etc/rc.conf`) unter `INITRD_MODULES` Module eintragen (wird bei Installation automatisch erledigt)
2. Erstellen der Initrd mit `/sbin/mk_initrd`
3. Bei LILO muß dem Bootloader mit dem `lilo`-Befehl eine veränderte Initrd mitgeteilt werden

### 5.3 Der Master-Boot Record (MBR) mit der Partitionstabelle, Bootsektoren

Im ersten Sektor (512 byte) einer Festplatte steht der sogenannte **Master-Boot Record (MBR)**

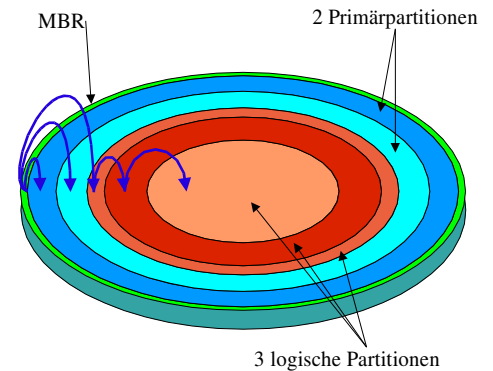
- die ersten 446 Byte: Programmcode  
das Programm, das beim Starten des PCs als erstes ausgeführt wird und durch das ein Bootmanager oder aber ein Betriebssystem gestartet wird
- dann 64 Byte **Partitionstabelle**, die angibt, wo sich die Hauptpartitionen befinden.

letzten 2 Byte der Partitionstabelle=AA55

#### Die Partitionstabelle

In der Partitionstabelle des MBRs gibt es für jede Hauptpartition einen Eintrag der Art:

```
struct partition {
char active; /* 0x80: bootable, 0: not bootable */
char begin[3]; /* CHS for first sector */
char type;
char end[3]; /* CHS for last sector */
int start; /* 32 bit sector number (counting from 0) */
int length; /* 32 bit number of sectors */
};
```



Die Informationen begin/end und start/length sind redundant und werden benutzt von:

| Betriebssystem                             | benutzte Eintäge                        | Plattenzugriff                   | max. adressierbar               |
|--|---|----------------------------------|---------------------------------|
| Linux                                      | start/length                            | direkt                           | 2 <sup>32</sup> Sektoren~2,2 TB |
| DOS, Windows 3.11,<br>WfWG, Windows NT 3.* | begin/end                               | BIOS INT13 call                  | 8,4 GB                          |
| Windows95, Windows98,<br>OSR2              | start/length<br>(begin/end=1023/255/63) | Extended BIOS INT13<br>Interface | 2 <sup>32</sup> Sektoren~2,2 TB |

#### Der Programmcode im MBR

kann sein

- **DOS-Programmcode** (first stage boot loader) (Standard bei DOS, Windows95/98)
  - ruft den Bootsektor der aktiven Partition auf (secondary stage boot loader)
  - kann mit dem DOS-Befehl `fdisk /mbr` neu geschrieben werden
    - fdisk gibt es u.a. auf der Windows98 Startdiskette
    - Achtung: eine Partition sollte dann aktiv sein - ansonsten wird der Rechner nicht booten
- **GRUB- oder LILO-Bootsektor**
  - startet den Bootsektor einer Partition
  - oder den Linux-Kernel - Möglichkeit, dem Kernel eine Kommandozeile mitzugeben
- Bootsektor eines anderen Bootmanagers

**Bootsektor**= 1. Sektor einer Festplatten-Partition

- für Code zum Booten des Betriebssystems auf dieser Partition
- beim Formatieren als DOS- / Windows oder OS/2 Partitionen wird der Code automatisch geschrieben
- allerdings nicht beim Formatieren als Linux-Partition - erst mal leer - also nicht von selbst startbar
  - kann nachträglich mit dem GRUB/LILO-Bootsektor gefüllt werden - Vorsicht: nur bei Linuxpartitionen!
- letzte 2 Bytes wie beim MBR, falls gültiger Systemstart-Code

#### Sichen des MBRs und eines Bootsektors

unter Linux kann mit dem Befehl

```
dd if=/dev/xxx of=LiloBootsektor bs=512 count=1
```

der Bootsektor der Platte bzw. Partition `/dev/xxx` in die Datei `LiloBootsektor` gesichert werden.

Zurückschreiben eines in der Datei `LiloBootsektor` gesicherten Sektors:

1. in den MBR der Platte `/dev/xxx` (Partitionstabelle bleibt erhalten):

```
dd if=LiloBootsektor of=/dev/xxx bs=446 count=1
```

2. in den Bootsektor der Partition `/dev/xxx`:

```
dd if=LiloBootsektor of=/dev/xxx
```

## 5.4 Bootloader

die beiden Linux Bootloader heißen

- **LILLO (Linux BootLOader)**
  - früher der Standard Linux Bootloader
- **GRUB (GRand Unified Bootloader)**
  - mittlerweile der Standard Linux Bootloader, obgleich Version erst 0.97
  - Vorteile gegenüber LILO:
    - GRUB kennt die gängigsten Dateisysteme (ext2, reiserfs,...), muß daher nicht Hardwareadressierung benutzen.
    - komfortable grub-Shell mit automatischer Namensvervollständigung per Tab-Taste
    - die beim Booten auswählbaren Menüs sind editierbar (Passwort-Schutz einstellbar)
    - Booten übers Netz möglich

Beide können Microsoft Betriebssysteme per „Chain Loading“ (d.h. durch Aufruf des Programmes im Bootsektor der Windows-Partition) booten

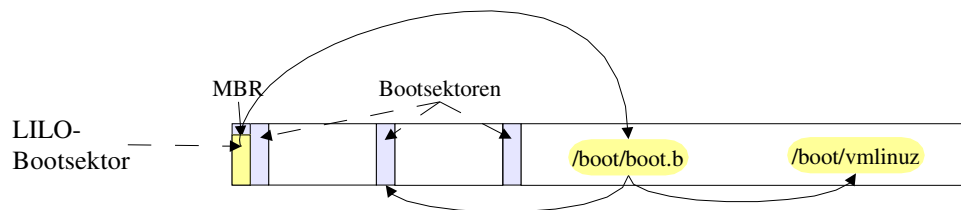
Linux läßt sich auch durch andere Bootloader booten.

- es ist dabei sinnvoll, daß ein solcher Bootloader zunächst GRUB oder LILO aufruft, der dann den Linux Kernel startet

### 5.4.1 Vorgehen der Linux Bootloader im Überblick

Ablauf beim Booten mit LILO oder GRUB:

1. Der LILO/GRUB-Bootsektor („first stage bootloader“) ruft den eigentlichen Bootloader („Second stage bootloader“) auf



2. Der „Second stage bootloader“ (LILO: `boot.b`, GRUB: `stage2 (stage1.5)`)
  - gibt eine Auswahl der zu bootenden Betriebssysteme aus und startet
    - einen Linux-Kernel. Dabei können Parameter übergeben werden
    - oder den Programmcode eines Bootsektors einer beliebigen Partition (und damit z.B. Windows98)
      - ... chain-loading
  - oder startet ein Betriebssystem direkt ohne Menüanzeige

Der LILO/GRUB-Bootsektor kann geschrieben werden

- in den MBR (damit wird LILO/GRUB beim Booten von Platte gestartet)
- in den Bootsektor einer Partition (sinnvoll, wenn ein anderer Bootmanager benutzt werden soll)
- auf CD ...

## 5.4.2 Der Linux Bootloader GRUB

Ausschnitt aus /usr/share/doc/packages/grub/README von grub-0.93-106:

```
- provides fully-featured command line and graphical interfaces
- recognizes fdisk partitions and BSD disklabels
- can dynamically read Linux ext2fs, ReiserFS, JFS and XFS, BSD ufs,
  MS-DOS FAT16 and FAT32, Minix fs, and VSTa fs filesystems, plus
  hardcoded blocklists
- can boot Multiboot-compliant kernels (such as GNU Mach), as well
  as standard Linux and *BSD kernels
```

### GRUBs Bezeichnungen für Partitionen,...

GRUB verwendet etwas andere Bezeichnungen für Partitionen, Disketten:

- sie stehen immer in runden Klammern
- Diskettenlaufwerk: (fd0)
- Platten heißen
  - (hd0), (hd1),...
 es wird nicht zwischen ide- und scsi-Platten differenziert, die Numerierung beginnt bei 0
- Plattenpartitionen:
  - (hd0,0), (hd0,1), ... Partitionen der ersten Platte
  - (hd1,0), (hd1,1), ... Partitionen der zweiten Platte
  - ...

Achtung: Numerierung beginnt auch hier bei 0! i.a. entspricht daher z.B. /dev/hda1 (hd0,0)

Dateien werden entsprechend wie z.B. (hd0,0)/boot/vmlinuz benannt.

grub benutzt für die Zuordnung seiner Bezeichnungen zu den Linux-Devicenamen die Datei

- device.map (i.a. /boot/grub/device.map)

die bei der Installation des Grub-Bootsektors erzeugt wird.

### GRUBs Dateien und Programme

die Grub Shell `grub`

die Utility `grub-install` für den vereinfachten Aufruf von grub

Dateien in dem Verzeichnis `/boot/grub`:

- Bootloader *\*stage\** (werden beim Booten ausgeführt)
- Konfigurationsdatei `menu.lst` (bestimmt das Verhalten des Bootloaders)
- `device.map`: von grub erzeugte Datei für die von grub benutzte Device-Zuordnung. Beispiel:

```
ukzrz-c107:~ # cat /boot/grub/device.map
(hd0)  /dev/hda
(fd0)  /dev/fd0
```

### Die GRUB Konfigurationsdatei `menu.lst`

wird bei der SUSE Standard-Installation automatisch angelegt (`/boot/grub/menu.lst`)

nach einem allgemeinen Abschnitt, in dem insbesondere festgelegt werden kann

- wie lange beim Hochfahren auf eine Benutzereingabe gewartet wird (`timeout 8`: 8 Sekunden)
- welches System ohne zutun gebootet wird (`default 0`: das erste erwähnte)

folgen mit `title` beginnende Abschnitte für die zur Auswahl stehenden Systeme

die in der Konfigurationsdatei stehenden Zeilen sind grub-Befehle, die beim Booten ausgeführt werden.

## Absichern der Parametereingabe und Editiermöglichkeit beim Booten

Zeile wie

```
passwd geheim
```

im allgemeinen Abschnitt der Datei menu.lst

Hier ein Beispiel:

```
# Modified by YaST2. Last modification on Fri Jun 27 08:29:36 2003

color white/blue black/light-gray
default 0
gfxmenu (hd0,6)/boot/message
timeout 8

title linux
    kernel (hd0,6)/boot/vmlinuz root=/dev/hda7 vga=0x305 splash=silent showopts
    initrd (hd0,6)/boot/initrd

title windows
    root (hd0,1)
    chainloader +1

title failsafe
    kernel (hd0,6)/boot/vmlinuz.shipped root=/dev/hda7 showopts ide=nodma apm=off
    acpi=off vga=normal nosmp noapic maxcpus=0 3
    initrd (hd0,6)/boot/initrd.shipped

title SUSE 8.1
    root (hd0,5)
    initrd /boot/initrd
    kernel (hd0,5)/boot/vmlinuz root=/dev/hda6
```

## 5.5 Neustart / Herunterfahren des Rechners / Der Init-Prozeß

**Init** = Prozeß, der beim Booten durch den Kernel gestartet wird und die Kontrolle über den weiteren Ablauf, das Starten weiterer Prozesse übernimmt.

die von init benutzten Scripte zum Starten von Programmen sind unter **/etc/init.d** zu finden

- Beim Systemstart wird zunächst **/etc/init.d/boot** ausgeführt, später die Startscripte des Verzeichnisses **/etc/init.d/rcn.d**, wobei n der „Standard Runlevel“ ist; i.d.R. **/etc/init.d/rc5.d**

Konfigurationsdatei: **/etc/inittab**

- Falls der Rechner nicht mit der Tastenkombination <Control>-<Alt>-<Del> heruntergefahren werden soll, die Zeile

```
ca::ctrlaltdel:/sbin/shutdown -r -t 4 now
```

durch # am Zeilenanfang auskommentieren.

- Auch im Single-User-Mode ist immer ein Passwort anzugeben:

Eintrag wie

```
# what to do in single-user mode
ls:S:wait:/etc/init.d/rc S
~~:S:respawn:/sbin/sulogin
```

- Änderungen werden durch den Befehl **telinit q** sofort wirksam.

Doku: **/etc/init.d/README**, Man-Pages **inittab**, **init**

### 5.5.1 Runlevel, Rechner runterfahren/neu starten

Der Rechner befindet sich immer in einem sogenannten **Runlevel**

ein Runlevel bestimmt, welche Prozesse gestartet werden

- die mit S beginnenden Scripte im Verzeichnis **/etc/init.d/rcRunlevel.d**

Auszug aus `/etc/inittab`:

```
# runlevel 0 is System halt (Do never use this for initdefault)
# runlevel 1 is Single user mode
# runlevel 2 is Local multiuser without remote network (e.g. NFS)
# runlevel 3 is Full multiuser with network
# runlevel 4 is Not used
# runlevel 5 is Full multiuser with network and xdm
# runlevel 6 is System reboot (Do never use this for initdefault)
```

Der Rechner wird beim Booten in den in `/etc/inittab` definierten Default Runlevel hochgefahren:

```
# default runlevel
id:5:initdefault:
```

Runlevelwechsel mit dem `init`-Befehl:

```
init Runlevel
```

beim Neustart oder Runterfahren des Rechners besser mit Hilfe des `shutdown`-Befehls, z.B.

```
shutdown -r +5 Achtung! - Neustart in 5 Minuten
```

```
shutdown -h now
```

- Die Scripte in den Verzeichnissen `/etc/init.d/rc0.d`, `rc1.d`,...

sind immer Verweise auf Scripte im übergeordneten Verzeichnis `/etc/init.d`

die Scriptnamen beginnen mit einem S oder K

- S: Script wird beim Start des Runlevels mit der Option `start` ausgeführt
- K: Script wird beim Verlassen des Runlevels mit der Option `stop` ausgeführt, falls es im neuen Runlevel kein entsprechendes Startscript gibt

die Scripte werden entsprechend ihrer Durchnummerierung nacheinander ausgeführt

- mittlerweile bei SUSE nicht mehr der Fall

die Scripte müssen die Argumente `start`, `stop`,... verarbeiten können

SUSE Besonderheit: die Scripte lesen zu Beginn die in den Dateien unter `/etc/sysconfig` stehenden Parameter ein.

`/etc/init.d/skeleton`: ein Beispielsgerüst für ein solches Script

Konfiguration mit `chkconfig`

- bei SUSE auch mit `Yast` möglich: `Yast – System – Runlevel-Editor`

ist ein Neustart / das Stoppen/... eines Dienstes notwendig, so kann dies gut durch Aufruf eines Scriptes in `/etc/init.d` geschehen

Beispiel: `/etc/init.d/xdm restart`  
einfacher: `rcxdm restart`

Bei SUSE erleichtert der `Runlevel-Editor` unter `YaST – System` die Verwaltung

## 6 Dies und Das

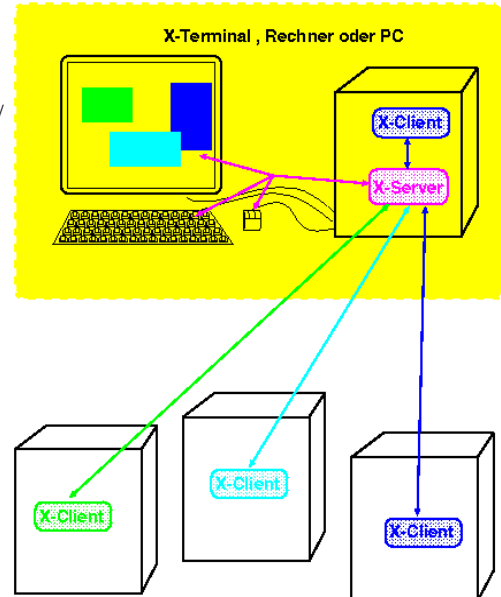
### 6.1 X-Konfiguration: Bildschirm / Grafikkarte / Tastatur / Maus / Zeichensätze

für Bildschirm / Grafikkarte / Tastatur / Maus ist der X-Server zuständig (Programm mit dem Namen X)

#### 6.1.1 Das X-Server / X-Client - Konzept

##### X-Server:

- Programm, das auf dem Rechner läuft, an dem Bildschirm, Tastatur und Maus angeschlossen ist
- Vermittler zwischen grafischer Anwendung und Bildschirm/Tastatur/Maus
- Der X-Server weiß,
  - welche Anwendung die Tastatur und Mauseingaben bekommt
  - welches Fenster einer Anwendung zugeordnet ist
- Kommunikation zwischen X-Server und X-Anwendungen über das sogenannte X-Protokoll



##### X-Display:

- An einen Rechner angeschlossener Bildschirm/Tastatur/Maus
- Bezeichnung i.a. Rechnername:0.0

##### X-Client (X-Anwendung):

Programm, das auf ein X-Display zugreift

**X-Protokoll:** Protokoll, mit Hilfe dem sich X-Server und X-Client verständigen – **unverschlüsselte** Datenübertragung

#### 6.1.2 X-Server, XFree86, X.Org

**XFree86** vom XFree86 Projekt ([www.xfree86.org](http://www.xfree86.org)) war lange Zeit (bei SUSE bis 9.1) der Standard X-Server für Linux.

**X.Org** von der X.Org-Foundation ([www.x.org](http://www.x.org)) ist der X-Server bei openSUSE 10.2

Konfigurationstool bei SUSE: **SaX2** (SAX=SUSE Advanced X Configuration Tool)

- mittlerweile in YaST integriert
  - Konfigurationsdatei:
- `/etc/X11/xorg.conf` bei X.Org
- `/etc/X11/XF86Config` bei Xfree86; die Syntax ist identisch

`X -version` zeigt die benutzte X-Version an

##### Log-Datei

- von X.Org: `/var/log/Xorg*`
- von XFree86: `/var/log/XFree*`

Dokumentation: insbesondere Man-Page `xorg.conf`

#### 6.1.3 Probleme mit dem grafischen Modus

Grafikkarte bestimmen mit Hilfe des Befehls

- `lspci` - listet alle Geräte am PCI Bus auf
- `sync; SuperProbe` - falls Grafikkarte nicht mit `lspci` erkannt wurde

Falls es keinen passenden X-Server gibt, können als Treiber (Driver in der Device-Section der Konfigurationsdatei) `vga` oder `vesa` oder `fbdev` ausprobiert werden

bei Problemen Rechner ohne grafische Oberfläche booten (z.B.beim Booten den Wert 3 als Parameter angeben), oder mit `init 3` den Rechner in den Runlevel 3 bringen - s.a. 5.5 Neustart / Herunterfahren des Rechners / Der Init-Prozeß)

- anschließend kann die Konfigurationsdatei editiert werden, mit Sax konfiguriert werden,...



Grafik-Treiber für Nvidia-Karten:

- falls 3D-Unterstützung erwünscht oder Beamer angeschlossen werden soll: kommerziellen (kostenlosen) Treiber von Nvidia installieren. Doku für openSUSE unter <http://en.opensuse.org/Nvidia>

### 6.1.4 Tastaturbelegungen

Abschnitt *InputDevice* – *Keyboard* in der Datei *XF86Config*, z.B:

```
Section "InputDevice"
    Driver      "Keyboard"
    Identifier  "Keyboard[0]"
    Option      "MapName" "Standard Keyboard [ pc105 ]"
    Option      "Protocol" "Standard"
    Option      "XkbLayout" "de"
    Option      "XkbModel" "pc105"
    Option      "XkbRules" "xfree86"
    Option      "XkbVariant" "nodeadkeys"
EndSection
```

bei KDE z.B auch benutzerspezifisch über das *KDE Kontrollzentrum – Regional-Einstellungen - Tastaturlayout* einstellbar

- werden mehrere Tastaturlayouts (z.B deutsch/englisch) eingestellt, können Sie über die Kontrollleiste ausgewählt werden

### 6.1.5 Schriftarten (Fonts)

Schriftarten bestimmen, welche Zeichen angezeigt werden können. Früher waren iso8859-1 Schriftarten Standard. Sie enthalten jedoch nicht das Euro-Symbol. Dazu wird eine iso8859-15 oder utf-8 Schriftart benötigt. Neuerdings scheint sich der utf-8 Zeichensatz durchzusetzen. Standard ab SUSE 9.1

Grund-Konfiguration über *xorg.conf* – Datei

Schriftarten können auch mit Hilfe eines sogenannten Fontservers zur Verfügung gestellt werden

Befehle, die mit Schriftarten zu tun haben:

- `xlsfonts`, `xfd`, `xfontsel`, `xset -q`
- `fc-list`, `fc-cache`

## 6.2 Spracheinstellung, Lokalisation

**i18n** = Internationalization (I + 18 Zeichen + n)

**l10n** = Localization (L + 10 Zeichen + n)

*Internationalisierung bedeutet in der Informatik bzw. in der Softwareentwicklung, ein Programm so zu gestalten, dass es leicht (ohne den Quellcode ändern zu müssen) an andere Sprachen und Kulturen angepasst werden kann; dieser zweite Schritt wird dann als Lokalisierung (engl. localization, abgekürzt L10N) bezeichnet.*

aus <http://de.wikipedia.org/wiki/I18n>

### 6.2.1 Zeichensätze

**ASCII**: 7 Bit Zeichensatz (-> 127 Zeichen)

- i.d.R stimmen die anderen Zeichensätze in diesen Bits überein

**ISO-\***: 8 Bit Zeichensätze, insbesondere

- 8859-1 (=Latin-1): Westeuropa, umfasst u.a. Umlaute und Akzente
- 8859-15 (=Latin-9): quasi Latin-1 inklusiv Euro-Zeichen

**Unicode (UTF-8/UTF-16)**: 16 Bit Zeichensatz

- Unicode löst allmählich die ISO-Zeichensätze ab, bei SUSE ab 9.1 ist utf-8 Standard
- Problem: Unicode umfasst zwar alle ISO\_\*-Zeichen, die Kodierung für 'nicht ASCII-Zeichen', insbesondere auch für deutsche Umlaute, ist nicht identisch

Konvertieren von Dateinamen mit `convmv`

z.B. Latin-9 nach UTF-8: `convmv --notest -f iso-8859-15 -t utf8 datei`

Konvertieren des Dateiinhalts mit `iconv` oder `recode`

z.B. Latin-9 nach UTF-8: `iconv -f iso-8859-15 -t utf8 -o utf8Datei latin1Datei`

## 6.2.2 Variablen zum Einstellen der gewünschten Lokalisierung

für Spracheinstellungen sind die Variablen LANG und LC\_\* von Bedeutung. Der Befehl `locale` listet die Variablen mit eingestelltem Wert auf; z.B. bei SUSE 9.2 deutsch:

```
user@host:~> locale
LANG=de_DE.UTF-8
LC_CTYPE="de_DE.UTF-8"
LC_NUMERIC="de_DE.UTF-8"
LC_TIME="de_DE.UTF-8"
LC_COLLATE="de_DE.UTF-8"
LC_MONETARY="de_DE.UTF-8"
LC_MESSAGES="de_DE.UTF-8"
LC_PAPER="de_DE.UTF-8"
LC_NAME="de_DE.UTF-8"
LC_ADDRESS="de_DE.UTF-8"
LC_TELEPHONE="de_DE.UTF-8"
LC_MEASUREMENT="de_DE.UTF-8"
LC_IDENTIFICATION="de_DE.UTF-8"
LC_ALL=
```

die Variable LC\_ALL überschreibt alle LC\_\* Variablen,

die Variable LANG gibt einen Wert für alle LC\_\* Variablen vor

i.a. reicht es somit aus, entweder LANG oder LC\_ALL zu setzen.

der Wert hat die Syntax `Sprache_Land.Zeichensatz` (mögliche Werte: `locale -a`)

Einstellungen bei SUSE: `/etc/sysconfig/language`

- Benutzer können durch Zuweisen der Variablen andere Einstellungen für sich vornehmen

## 6.3 Stromsparfunktionen

es gibt 2 verschiedene Techniken, die vom BIOS unterstützt werden:

- APM (Advanced Powermanagement)
  - Stromversorgungszustand ablesbar, Standby-Zustand möglich
  - einige Funktionen wie Standby oder Suspend werden vom BIOS eventuell selbst ausgeführt (z.B. beim Schließen des Notebookdeckels)
  - wird von SUSE 9.2 nur verwendet, wenn ACPI vom Bios nicht unterstützt (Standard-Einstellung)
- ACPI (Advanced Configuration and Power Interface)
  - nur bei neuerer Hardware
  - bietet zusätzlich Konfiguration und Steuerung einzelner Hardwarekomponenten
  - wird bei SUSE 9.2 wenn möglich beim Booten aktiviert (Standard-Einstellung)
  - Doku: <http://acpi.sourceforge.net/>

es gibt Kernel-Parameter, die die Standardeinstellungen verändern (z.B. `acpi=off` oder `apm=off`)

bei SUSE

- Daemon **powersaved**, wird bei Booten automatisch gestartet (`/etc/init.d/powersaved`)
- Konfiguration über *YaST2 – System – Power-Management* möglich
- Konfigurationsdateien unter `/etc/sysconfig/powersave`

ausführliche Info im *SUSE Administrationshandbuch, Power-Management*,

außerdem unter `/usr/share/doc/packages/powersave`

der KDE-Desktop bietet das Applet **kpowersave** für die Kontrollleiste

## 6.4 Druckerkonfiguration

bei SUSE leicht einzurichten mit

- `YaST - Hardware - Printer`

Linux kennt unterschiedliche Printsysteme

- **LPD** - älter, wohl noch am weitesten verbreitet
- **CUPS** (Common UNIX Printing System): *www.cups.org* – Standard bei SUSE LINUX 11.0

Die Druckersprache bei Linux ist **PostScript**

Für nicht-PostScript-Drucker werden **Ghostscript-Treiber** verwendet, die Postscript in die entsprechende Druckersprache übersetzen

- insbesondere bei sogenannten Windows-Druckern (GDI-Drucker) problematisch, da Druckersprache nicht Standard
- Soll ein Drucker neu beschafft werden, unbedingt überprüfen (*www.linuxprinting.org*), ob von Linux unterstützt

Ein Druckerfilter (Dokumentfilter) kann das Dateiformat erkennen, wandelt die Datei in PostScript um, die Konvertierung in die Druckersprache wird Ghostscript übertragen.

**Dateien, Verzeichnisse, Befehle ums Drucken bei LPD:**

|                             |                                   |
|-----------------------------|-----------------------------------|
| <code>/etc/printcap</code>  | Konfigurationsdatei für Drucker   |
| <code>/var/spool/lpd</code> | Spoolverzeichnisse der Drucker    |
| <code>lpq</code>            | Anzeigen der Druckerwarteschlange |
| <code>lpr</code>            | Drucken einer Datei               |
| <code>lprm</code>           | Löschen eines Druckauftrags       |
| <code>lpc</code>            | Druckersteuerung                  |

bei CUPS ist die Administration über eine Webseite möglich:

- `http://localhost:631` lokal
- `http://CupsServer:631` für einen entfernten CUPS Server (Vorsicht: die Verbindung ist unverschlüsselt!)
- vor der ersten Benutzung ist mit dem Befehl `lppasswd -g sys -a root` ein Administrationsaccount einzurichten

Konfigurationsdateien für CUPS stehen im Verzeichnis `/etc/cups`

- **cupsd.conf** Konfigurationsdatei für einen CUPS-Server
- **client.conf** Konfigurationsdatei für einen CUPS-Client

**Achtung – wenn der Drucker für CUPS mit YaST eingerichtet wurde:**

- der Drucker ist für andere Rechner benutzbar und
- von jedem Rechner kann die Web-Administrationsseite aufgerufen werden!

änderbar durch Konfiguration des cupsd (`/etc/cups/cupsd.conf`) oder mit Hilfe eines Firewalls den Port 631 sperren

**Infos:**

*Printing HOWTO*

<http://www.linuxprinting.org>

SUSE Administrationshandbuch

## 6.5 Benutzerverwaltung

### 6.5.1 Grundsätzliches

Benutzer und Gruppen verwalten:

- entweder mit *YaST - Sicherheit&Benutzer* - ...
- oder mit Kommandos
  - sinnvoll bei vielen Benutzern, können auch in Skripten verwendet werden

Jedem Benutzer ist zugeordnet (über die Datei `/etc/passwd`)

- eine Benutzerkennung
- eine UID (user identifier; eine Zahl) vom System wird immer die UID, nicht die Benutzerkennung benutzt -z.B. für

Zugiffsrechte auf Dateien,

- eine Gruppe. Dieser Gruppe werden insbesondere neu erstellte Dateien zugeordnet
- ein Homeverzeichnis, insbesondere mit den für ihn relevanten Konfigurationseinstellungen
- eine Shell, die beim Einloggen für ihn gestartet wird
- ein Passwort für die Authorisierung des Benutzers

nicht benötigte Benutzerkennungen entfernen

Kennungen, die sich nicht interaktiv einloggen: `/bin/false` als LoginShell zuweisen

evtl. Default-Dateien, die beim Anlegen eines Benutzers in sein Homeverzeichnis kopiert werden, unter `/etc/skel` anlegen

Default-Werte in Datei `/etc/login.defs` überprüfen

## 6.5.2 Dateien für die Benutzerverwaltung

`/etc/passwd`: Benutzerkennungen; ein Auszug:

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/bash
user:x:48141:444:user:/home/user:/bin/bash
```

- s. `man 5 passwd`

`/etc/shadow`: „verschlüsselte“ (genauer: gehashte) Passwörter; ein Auszug:

```
root:Yq7w6gqT29xYc:9473:0:10000::::
bin:*:8902:0:10000::::
user:X9BOZkZoXYK9s:11031:0:99999:7:0:11062:
```

- s. `man 5 shadow`
- Verschlüsselung (bei SUSE bei der Installation auswählbar) durch
  - crypt-Funktion (max 8 Zeichen pro fürs Passwort) oder
  - MD5-Hash-Algorithmus (sicherer, und max 256 Zeichen fürs Passwort)
  - blowfish (ebenfalls sicherer, Standard bei openSUSE 11.0)

`/etc/group`: Gruppen; ein Auszug:

```
root:x:0:root
bin:x:1:root,bin,daemon
users:x:100:kurs
kurse:x:444:
```

- s. `man group`

`/etc/skel`: Dateien, die beim Anlegen eines Benutzers in sein Homeverzeichnis kopiert werden; Standard bei SUSE 8.0:

```
ls -a /etc/skel
.          .Xresources  .emacs     .profile   .xim       .xtalkrc
..         .bash_history .exrc      .urlview   .xinitrc   Documents
.Xdefaults .bashrc      .kermrc    .xcoralrc  .xserverrc.secure public_html
.Xmodmap   .dvipsrc    .muttrc    .xemacs    .xsession
```

`/etc/login.defs`: Default-Werte (auch einstellbar mit YaST); ein Auszug:

```
#
# Delay in seconds before being allowed another attempt after a login failure
#
FAIL_DELAY          3

#
# The default PATH settings.
#
ENV_PATH            /usr/local/bin:/usr/bin:/bin

#
# The default PATH settings for root:
#
ENV_ROOTPATH       /sbin:/bin:/usr/sbin:/usr/bin
```

### 6.5.3 Befehle für die Benutzerverwaltung

|                       |   |
|-----------------------|---|
| <code>useradd</code>  | Benutzer anlegen (-m -k inklusive Erstellung des Homeverzeichnisses und Kopieren der Default-Dateien) |
| <code>passwd</code>   | Passwort setzen   |
| <code>userdel</code>  | Benutzer löschen (userdel -r samt Dateien)  |
| <code>groupadd</code> | Gruppe anlegen  |
| <code>groupdel</code> | Gruppe löschen  |
| <code>usermod</code>  | Benutzereinstellungen ändern  |
| <code>chage</code>    | Benutzereinstellungen der shadow-Datei ändern   |
| <code>groupmod</code> | Gruppenzugehörigkeiten festlegen  |

### Übung ~ Benutzerverwaltung

## 6.6 Cronjobs

= Programme, die regelmäßig ausgeführt werden

Crontabs (Definition der Cronjobs) stehen unter

- `/var/spool/cron/tabs`
  - Name der Datei ist gleich dem Benutzer, unter dessen Kennung die Jobs ablaufen
  - Auflisten mit: `crontab -l`
  - Einfügen: `crontab -e`
  - Beschränkung auf gewisse Benutzer durch Dateien `deny` und `allow` im Verzeichnis `/var/spool/cron`
  - eine Zeile in so einer Crontab-Datei hat die Form  
`Minute Stunde Tag-des-Monats Monat Wochentag Befehl`
- `/etc/cron.d`
  - eine Zeile in so einer Crontab-Datei hat die Form  
`Minute Stunde Tag-des-Monats Monat Wochentag Benutzer Befehl`
- oder in `/etc/crontab`  
hierüber werden Scripte in den Verzeichnissen `/etc/cron.daily`, `/etc/cron.hourly`, `/etc/cron.monthly`, `/etc/cron.weekly` aufgerufen

Beispiele (aus der Manpage zu den crontab-Einträgen - `man 5 crontab`) für Crontabeinträge in `/var/spool/cron/tabs/Benutzerkennung`

```
# run five minutes after midnight, every day
5 0 * * *      $HOME/bin/daily.job >> $HOME/tmp/out 2>&1

# run at 2:15pm on the first of every month -- output mailed to paul
15 14 1 * *    $HOME/bin/monthly

# run at 10 pm on weekdays, annoy Joe
0 22 * * 1-5   mail -s "It's 10pm" joe%Joe,%%Where are your kids?%

23 0-23/2 * * * echo "run 23 minutes after midn, 2am, 4am ..., everyday"
5 4 * * sun    echo "run at 5 after 4 every sunday"
```

## 6.7 SSH Zugang zum eigenen Rechner

bei openSUSE ist der ssh Server (sshd) nach der Installation gestartet

die Firewall unterbindet jedoch den ssh Zugang von außen

### ssh Zugang ermöglichen

sshd bei der Firewall erlauben `YaST – Sicherheit und Benutzer – Firewall – erlaubte Dienste`

### ssh-Zugang einschränken

in `/etc/hosts.deny` Zeile

```
sshd : ALL EXCEPT LOCAL
```

in `/etc/hosts.allow` den ssh Zugriff einzelnen Rechnern erlauben. Zeile wie z.B.

```
sshd: 134.245.123.33 134.245.123.42
```

oder

```
sshd: 134.245.123.
```

## 7 Mit Problemen fertig werden

### 7.1 Log-Dateien

zunächst überprüfen, ob es Fehlermeldungen gibt (i.a. unter `/var/log`, s.a. 1.7 Logging)

### 7.2 Die grafische Oberfläche reagiert nicht mehr

über virtuelle Konsole (`<Strg><Alt><F1>`,... `<Strg><Alt><F6>`) einloggen, oder aber per ssh auf den Rechner zugreifen

- der grafischen Sitzung zugeordnete Prozesse beenden
  - `ps x`
  - `kill -HUP ...` oder
  - `KILL -TERM ...` oder
  - `KILL -9 ...`
  - oder grafische Oberfläche neu starten
- `/etc/init.d/xdm stop`
  - `/etc/init.d/xdm start`

Eine harte Alternative: `<Strg><Alt><Backspace>` beendet abrupt den X-Server und damit die auf der Konsole laufenden X-Anwendungen

### 7.3 Mein Rechner bootet nicht mehr - was kann ich tun?

Fehlermeldungen beim Hochfahren beachten

Linux von einer Bootdiskette/InstallationsCD ... booten

- eventuell nur in den Single User Mode (Kernel-Option `s` beim Hochfahren mit LILO/GRUB)
- das eigene System falls möglich booten
- Rettungssystem sonst

Bei Bedarf - falls Rettungssystem - notwendige Dateisysteme mounten, z.B.

```
mount /dev/sda5 /mnt
```

Log-Dateien (s.7.1) überprüfen

dem Fehlerfall entsprechend handeln, z.B.

- Konfigurationsdatei ändern
- Lilosektor in den MBR schreiben
- Backup einspielen
- ...

Falls der Bootloader Grub nicht mehr startet: s. 7.4

### 7.4 Grub startet nicht mehr – das Bootmenü erscheint nicht

Falls nur der Bootloader im dem MBR fehlt, die Partitionen aber sonst noch okay sind, kann der Bootloader wieder hergestellt werden:

- von einer Linux Life CD/DVD (oder auch der openSUSE InstallationsCD – Rettungssystem) booten
- die **grub Shell** mit dem Kommando `grub` starten

- Die Partition, die das Grub-Verzeichnis enthält, angeben, falls es die erste Partitionen auf der ersten Platte (/dev/hda0) ist:  
grub> `root (hd0,0)`  
ist unklar, auf welcher Partition, so hilft der Befehl  
grub> `find /boot/grub/stage1`  
weiter
- den GRUB Bootsektor in den MBR schreiben:  
grub> `setup (hd0)` MBR der ersten Platte
- Grub mit `quit` beenden und den Rechner mit `shutdown -r now` neu starten (nicht vergessen: CD/DVD vor dem Neustart entfernen!)
  
- auch möglich: von openSUSE Installations CD/DVD booten, *Repair installed System* auswählen

## 7.5 Ich habe das root-Passwort vergessen

1. Ein Rettungs-System (z.B. mit Installations-CD) booten
2. Die Root-Partition mounten, z.B.  
`mount /dev/sda7 /mnt`
3. in der shadow-Datei das Passwort löschen, z.B.  
`vi /mnt/etc/shadow ...`
4. `umount /dev/sda7 /mnt`
5. Den Rechner im Single User Mode starten (Option 1 beim Start mit Grub)
6. das root-Passwort setzen
7. den Rechner neu starten