

## Kapitel 2

# Das Betriebssystem Linux

### Allgemeines

Die praktischen Programmierübungen im Computer-Pool werden an PCs stattfinden, auf denen das Betriebssystem Linux installiert ist. Linux ist neben WINDOWS das am meisten verbreitete Betriebssystem. Die Vorteile von Linux sind

- höhere Sicherheit durch strenge Unterteilung der Zugriffsrechte,
- Preis,
- offener Code.

Die Nachteile sind, dass

- manche Dinge nicht so komfortabel wie unter WINDOWS sind,
- WINDOWS-Software auf Linux nicht läuft und man sich umgewöhnen muss, wenn man die entsprechende Linux-Software nutzt.

Bezüglich des ersten Nachteils wurde allerdings im Laufe der vergangenen Jahre viel getan, dass oft kein Unterschied zu WINDOWS mehr vorhanden ist.

Der Name Linux tauchte erstmals 1991 nach der Veröffentlichung des ersten Linux-Kernels durch Linus Torvalds auf. Dieses Betriebssystem wird vor allem auf Servern eingesetzt, so zum Beispiel laufen die Server von Google und Wikipedia unter Linux. Der Einsatz auf Desktops ist vor allem im universitären Bereich zu finden, und auch da vor allem bei Mathematikern und Informatikern. Für den Desktop gibt es viele unterschiedliche Linux-Distributionen:

- SuSe, ist im deutschsprachigen Raum am meisten verbreitet,
- RedHat, ist im amerikanischen Raum am meisten verbreitet,
- Debian, läuft im Computer-Pool.

Informationen zu Unterschieden, Vor- und Nachteilen der einzelnen Distributionen findet man im Internet.

Der Server vom Computer-Pool ist `pyramis.math.uni-sb.de`.

### Zugriffsrechte

Jeder Nutzer (user) ist in Linux einer Gruppe (group) zugeordnet. Die Zugriffsrechte jedes Files und Verzeichnisses in Linux sind lesen (read, r), schreiben (write, w) und ausführen (execute, x). Diese Zugriffsrechte sind gesetzt für den user, die group und alle übrigen. Zum Beispiel, besagt

```
-rw-r----- 1 john users 29444 2006-10-13 12:22 tmp.txt
```

dass das File `tmp.txt` vom Nutzer john gelesen und geschrieben werden kann (Zeichen 2-4), von der Gruppe users nur gelesen werden kann (Zeichen 5-7) und alle übrigen dürfen mit diesem File nichts machen (Zeichen 8-10). Bei einem Verzeichnis

sieht diese Information zum Beispiel wie folgt aus

```
drwxr-xr-x 3 john users 312 2006-10-12 18:26 MOD_PROG
```

Das `d` am Anfang besagt, dass es sich um ein Verzeichnis (directory) handelt, der Nutzer `john` darf im Verzeichnis lesen, schreiben und in das Verzeichnis wechseln, die Gruppe `users` und alle übrigen dürfen nur lesen und in das Verzeichnis wechseln, dort aber nichts ändern (schreiben). Das Setzen und Ändern der Zugriffsrechte geschieht mit dem Befehl `chmod`. Wenn man zum Beispiel das File `tmp.txt` für alle les- und schreibbar machen will, so kann man das mit

```
chmod a+rw tmp.txt
```

und erhält danach die Information

```
-rw-rw-rw- 1 john users 29444 2006-10-13 12:22 tmp.txt
```

Das heißt, die Zugriffsrechte regeln wer was machen darf. Ist der Rechner ordentlich administriert, hat ein Virus oder ein Wurm keine Möglichkeit wichtige Dinge zu verändern, da er dazu nicht die Rechte hat. Die Rechte müssen so gesetzt sein, dass das nur der Administrator, der in Linux `root` heißt, machen darf und um als Administrator zu arbeiten, muss man ein entsprechendes Password eingeben. Aus Sicherheitsgründen sollte ein Password immer so gewählt sein, dass der Nutzernamen nicht ein Teil des Passwords ist und das Password auch Ziffern enthält.

## Werkzeuge

Man hat in Linux die Möglichkeit mit graphischen Benutzeroberflächen (wie in WINDOWS) oder auch auf der Kommandozeilenebene (Shell) zu arbeiten. Die Shell ist ein Kommandointerpreter, der von der Kommandozeile die Anweisungen einliest, diese auf Korrektheit überprüft und ausführt. Eine Liste wichtiger Kommandos wird am Ende des Kapitels gegeben.

Wichtige Werkzeuge, die wir brauchen sind:

- Konquerer (erfüllt etwa die Aufgaben wie der Explorer in Windows)
- Editoren: Kate, emacs, Xemacs (sind selbsterklärend), vi ist ein Kommandozeile-Editor, der dann nützlich ist, wenn man keine graphische Oberflächen öffnen kann (wenn man beispielsweise von außerhalb auf einem Linux-Rechner eingeloggt ist)
- matlab, Software zum Programmieren von Verfahren,
- gcc, der Gnu-C-Compiler,
- latex, Programmiersprache zur Textverarbeitung.

Standardsoftware wie `Firefox`, `acroread` läuft natürlich auch unter Linux. Es gibt natürlich noch viel mehr nützliche Dinge, siehe Literatur, Internet, Handbücher, Übungen.

## Dateinamen, Baumstruktur

Dateinamen:

- Linux unterscheidet Groß- und Kleinschreibung,
- Der Schrägstrich / darf nicht verwendet werden, da er zur Trennung von Verzeichnisnamen dient,
- Sonderzeichen (Leerzeichen, Umlaute, &, ...) sollten vermieden werden, da einige eine spezielle Bedeutung besitzen.

Linux besitzt eine hierarchische, baumstrukturierte Verzeichnisstruktur, siehe Abbildung 2.1. Ausgangspunkt ist die Wurzel / (root). Die Position eines beliebigen (Unter)-Verzeichnisses ist durch den Pfad gegeben. Der Pfad gibt an, wie man von

der Wurzel zu der gewünschten Datei beziehungsweise zum gewünschten Verzeichnis gelangt, zum Beispiel

```
pwd
(pwd – print name of current/working directory) gibt
/home/john
```

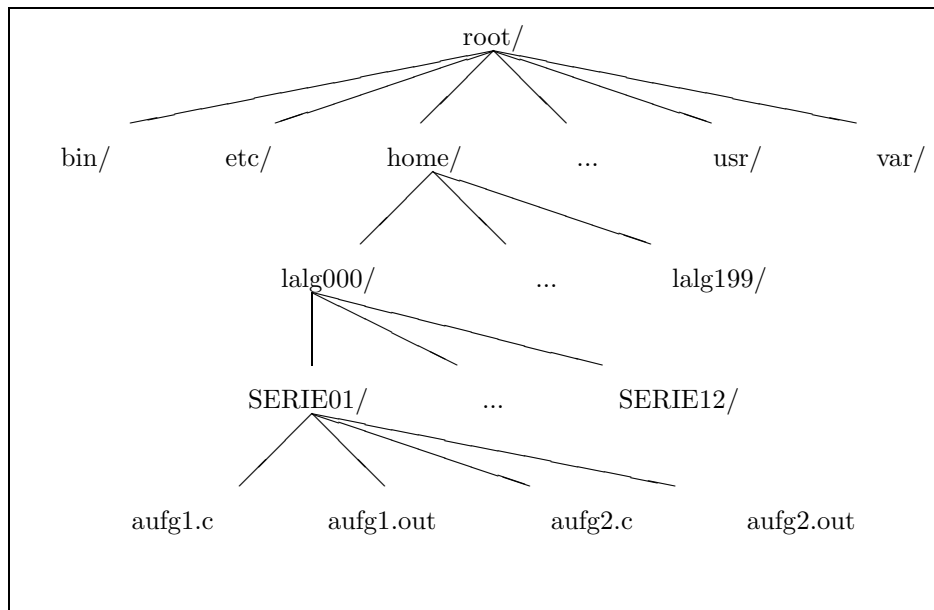


Abbildung 2.1: Beispiel für einen Verzeichnisbaum in Linux.

Pfade, die mit / beginnen, heißen absolute Pfade. Daneben gibt es noch die relativen Pfade, die relativ zur gegenwärtigen Position im Verzeichnisbaum sind. Wichtige relative Pfadbezeichner sind

- . : aktuelle Position im Verzeichnisbaum (Arbeitsverzeichnis),
- .. : das übergeordnete Verzeichnis.

Es ist möglich, mehrere Dateien oder Verzeichnisse gleichzeitig anzusprechen, mit Hilfe sogenannter Wildcards

- \* : ersetzt beliebig viele Zeichen (auch keines)
- ? : ersetzt genau ein Zeichen
- [zeichen1 - zeichen2] : Auswahlsequence; alle Zeichen zwischen zeichen1 und zeichen2 werden ersetzt.

So kann zum Beispiel

```
ls tmp*
```

die Ausgabe

```
tmp tmp.txt
```

geben.

## Liste von Kommandos

Nähere Informationen zu den Kommandos erhält man mit dem man Befehl

```
man Befehlsname
```

Sucht man Befehle, die mit einem Schlüsselbegriff zusammenhängen, so verwende man

```
man -k Schlüsselbegriff
```

Befehl	Bemerkungen
<code>cd</code>	wechsle Verzeichnis; zurück zum vorherigen Verzeichnis : <code>cd -</code> ins Homeverzeichnis: <code>cd</code>
<code>chmod</code>	verändere Zugriffsrechte; z.B. Schreibrecht für alle Nutzer <code>chmod a+w filename</code>
<code>cp</code>	kopiere Datei
<code>df</code>	gibt den belegten und freien Speicherplatz (Harddisc) zurück
<code>env</code>	zeigt die Umgebungsvariablen an
<code>find</code>	findet Dateien, z.B. um all Dateien mit Namen <code>core</code> zu finden <code>find / -name core -print</code>
<code>grep</code>	sucht nach einem Muster in einer Datei, z.B. um alle <code>printf</code> -Befehle in <code>test.c</code> zu finden <code>grep 'printf' test.c   more</code>
<code>gzip</code>	komprimiert Dateien
<code>gunzip</code>	dekomprimiert Dateien <code>filename.gz</code>
<code>kill</code>	beendet Prozesse
<code>ll</code>	zeigt Inhalt von Verzeichnissen
<code>ls</code>	zeigt Inhalt von Verzeichnissen, wichtige Optionen- <code>ali</code>
<code>man</code>	Handbuch, z.B. <code>man man</code>
<code>mkdir</code>	Anlegen von Verzeichnissen
<code>more</code>	Ansehen von Dateien
<code>mv</code>	verschieben von Dateien
<code>passwd</code>	Veränderung des Passwords
<code>ps</code>	zeigt laufende Prozesse an, z.B. vom Nutzer <code>abc</code> <code>ps -u abc</code>
<code>pwd</code>	zeigt Pfadnamen zum gegenwärtigen Verzeichnis
<code>rm</code>	löscht Dateien, <b>nutze besser immer</b> <code>rm -i</code>
<code>rmdir</code>	löscht Verzeichnisse
<code>tail</code>	zeigt die letzten Zeilen einer Datei
<code>tar</code>	Erstellung von Archiven
<code>typeset</code>	setzen von Umgebungsvariablen, z.B. <code>typeset -x PATH=\$PATH:\$HOME/bin</code>
<code>who</code>	zeigt wer im System eingeloggt ist
<code>which</code>	lokalisiert ausführbares Programm

Von einem Hörer ist das Buch [Bar04] zur Einarbeitung in Linux empfohlen wurden.