

```

221 Goodbye.
$ ncftp ftp.kde.org
NcFTP 3.1.4 (Jul 02, 2002) by Mike Gleason (ncftp@ncftp.com).
Connecting to 131.220.60.97...
subraum.rhrz.uni-bonn.de FTP server (Version wu-2.6.0(1) Wed Nov 28
19:04:12 GMT 2001) ready.
Logging in...
[...]
ncftp / > quit

You have not saved a bookmark for this site.

Would you like to save a bookmark to:
ftp://ftp.kde.org

Save? (yes/no) no
[...]
$

```

13.2 Wer ist wo online?

Die Fragen in diesem Kapitel dürften fast alle mit den Informationen aus dem Skript zu beantworten sein. Daher: selber probieren! Zumal an dieser Stelle nicht viel mehr als individuelle Beispiele stehen könnten ...

Lösungen und Denkanstöße zu den Aufgaben im Kurs „IF TIF 05 – Linux ist weiblich“

Im Folgenden findet Ihr Lösungen und Lösungsansätze für viele der im Kursskript gestellten Aufgaben. Nicht enthalten sind Fragen, deren Antworten zu stark von den Gegebenheiten im Bremer Rechnerpool abhängen.

Nicht alle Lösungen sind so ausführlich skizziert, dass sie jeder sofort einleuchten müssen. Wenn es Fragen gibt, wenn Ihr andere Antworten, Ergänzungen oder auch Korrekturen habt (nobody is perfect ...), nutzt bitte die Möglichkeit der Diskussion im Kurs und auf der Arbeitsliste tif05@lists.answergirl.de.

Gabriele und Patricia

1 Begriffsverwirrung

Die in Mittel- und Westeuropa zur Zeit am weitesten verbreiteten Linux-Distributionen sind *SuSE Linux* (aus dem Portfolio der Firma Novell), *Fedora Core* (ein freies Projekt, das die Firma Red Hat ins Leben rief, unterstützt und als Basis der eigenen Businessprodukte nutzt), *Mandriva Linux* (die aus der Übernahme des brasilianischen Distributors Conectiva durch die französische Firma Mandrakesoft entstandene kommerzielle Distribution) sowie *Debian GNU/Linux*. Finde die aktuellen Versionen dieser vier Distributionen heraus!

Am Stichtag 12. September 2005 waren aktuell:

- SuSE Linux Professional 9.3 für die Intel/AMD-Plattform (enthält die 32- und 64-Bit-Ausgabe) ist nicht nur als Boxprodukt, sondern auch kostenlos zum Download – sowohl als Installations- als auch als Live-CD – erhältlich. Daneben vertreibt Novell alle möglichen Zusammenstellungen für Firmenkunden, z. B. der „SuSE Linux Enterprise Server 9“ (auch für IBM-Mainframes erhältlich) und die darauf aufbauende der „Novell Linux Small Business Suite“. Für diese gelten andere Lizenzbedingungen als das „auf beliebig vielen Rechnern installierbar“ der Distribution für Endverbraucherinnen.
- Fedora Core 4.
- Mandriva Linux 10.1. Seit der Übernahme des amerikanischen Distributors Lycoris gehört auch dessen letzter „Desktop/LX 1.4“ neben einer Reihe Spezialausgaben und Businessprodukte zum Portfolio.
- Debian GNU/Linux 3.1 („Sarge“) als stabile Version. Bei Debian gibt es daneben immer auch *testing* (meist gut brauchbar für den Einsatz in unkritischen Umgebungen; aktuell: „Etch“) und *unstable* („bleeding edge“; immer „Sid“ genannt).

☞ Suche drei Distributionen heraus, die auf Debian aufsetzen!

- *Knoppix*, <http://www.knopper.net/knoppix/> (das berühmte Live-System)
- *Kanotix*, <http://kanotix.com/> (Knoppix-basierte Live-Distribution, die sich gut als Ausgangsbasis für eine Debian-Installation eignet)
- *Skolelinux*, <http://www.skolelinux.org/> (speziell für Schulen und Bildungseinrichtungen)
- *Ubuntu*, <http://www.ubuntu.com/> (den Shooting-Star unter den Distributionen, den es auch als KDE-Version *Kubuntu* gibt, kann sich frau kostenfrei als CD bestellen)
- *Xandros*, <http://www.xandros.com/> (sehr gute Desktop-Distribution, die es demnächst auch auf Deutsch geben soll)
- ...

☞ Du bekommst eine Support-Mail mit dem Hinweis „Ich benutze Linux 9.3.“ Was könnte die Fragende gemeint haben?

Sie meinte wahrscheinlich die Distribution „SuSE Linux 9.3“, denn einen Linux-Kernel in dieser Version gab und gibt es nicht. Vielleicht verwendet sie aber auch Slackware 9.3 oder Mandrake 9.3 – beide nicht mehr besonders taufisch, aber das ist durchaus nicht unüblich.

2 Komponenten

2.1 Kernel

☞ Welche Linux-Versionen sind momentan aktuell? Beachte, dass zum einen (im Gegensatz zu anderen Betriebssystemen) auch ältere Kernel-Serien weiter gepflegt werden. Zum Anderen unterscheidet frau zwischen *Entwickler-Kerneln* mit ungerader (Minor-) Versionsnummer und für den stabilen Einsatz gedachten „geraden“ Versionen.

Die Antwort findet frau natürlich unter <http://www.kernel.org/>, aber z. B. auch unter <http://www.linux.de/>. Am Stichtag 12. September 2005 waren als stabile Versionen aktuell: Linux 2.6.13, 2.4.31, 2.2.26 und 2.0.40. All diese Kernelserien werden aktiv gepflegt; die aktuelle Entwicklung findet hingegen in der 2.7er Entwicklerkernelversion statt.

☞ Finde einen Kernel-Patch! Für welchen Kernel ist er gedacht?

Weil's wirklich Unmengen von Kernel-Patches gibt, verweisen wir einfach mal auf eine Google-Suche nach den Stichwörtern „Linux“ und „Patches“. Die meisten Kernel-Entwickler veröffentlichen ihre Patches jeweils auf ihren Web-Präsenzen, so z. B. Andrew Morton unter <http://www.zip.com.au/~akpm/linux/patches/>, Rik van Riel unter

```
berta@rechnerb:~$ ssh -v rechnera
berta@rechnera:~$ ls -al ~anna/verz
berta@rechnera:~$ exit
berta@rechnerb:~$ scp -r rechnera:~anna/verz .
```

☞ Log Dich per ftp oder ncftp (am besten zum Vergleich mit beiden) auf dem KDE-FTP-Server <ftp.kde.org> ein und lade zwei Dateien in einem Rutsch herunter.

```
$ ftp ftp.kde.org
Connected to subraum.rhrz.uni-bonn.de.
220 subraum.rhrz.uni-bonn.de FTP server (Version wu-2.6.0(1) Wed Nov 28
19:04:12 GMT 2001) ready.
Name (ftp.kde.org:trish): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password: meine@adres.se
ftp> cd pub/kde
ftp> ls
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 1244
-rw-rw-r-- 1 root root 1107 Aug 18 02:01 .message
drwxrwxr-x 8 root root 4096 Aug 23 14:59 Attic
-rw-rw-r-- 1 root root 1107 Aug 18 02:01 README
-rw-rw-r-- 1 root root 336 Aug 18 01:55 README_UPLOAD
[...]
226 Transfer complete.
ftp> mget README*
mget README? y
200 PORT command successful.
150 Opening BINARY mode data connection for README (1107 bytes).
226 Transfer complete.
1107 bytes received in 0.05 secs (23.0 kB/s)
mget README_UPLOAD? y
200 PORT command successful.
150 Opening BINARY mode data connection for README_UPLOAD (336 bytes).
226 Transfer complete.
336 bytes received in 0.02 secs (18.3 kB/s)
ftp> exit
221-You have transferred 1443 bytes in 2 files.
221-Total traffic for this session was 6847 bytes in 6 transfers.
221-Thank you for using the FTP service on subraum.rhrz.uni-bonn.de.
```

Diese Spezialbehandlung verhindert allerdings effektiv das vermutlich „gemeinte“ Ergebnis, alle Punktdateien und -verzeichnisse und sonst nichts zusammenzupacken. Hierfür muss man zu üblen Tricks greifen: `tar -czvf archiv.tar.gz .?*` packt alle Dateien und Verzeichnisse zusammen, die mit einem Punkt beginnen und anschließend mindestens zwei weitere Zeichen im Namen tragen. Warum zwei Fragezeichen? Weil `.*` auch auf `..` passt, das übergeordnete Verzeichnis. Dass dieser Hack alle Punktdateien ignoriert, die nur aus dem Punkt und einem weiteren Zeichen bestehen, nehmen wir billigend in Kauf – davon gibt es in der Regel nur selbst angelegte...

☞ Pack alle Dateien je zweier Unterverzeichnisse Deines Homeverzeichnisses in ein `tgz`- und ein `tar.bz2`-Archiv. Prüfe die Archive, und pack sie in `/tmp` wieder aus. Pack die Archive anschließend in zwei Schritten in einem neu angelegten Verzeichnis aus: Erst dekomprimieren mit `gunzip` bzw. `bunzip2`, dann auspacken mit `tar`.

```
$ cd
$ tar -czvf test1.tgz verz1 verz2
$ tar -cjvf test2.tar.bz2 verz3 verz4
$ tar -tzvf test1.tgz
$ tar -tjvf test2.tar.bz2
$ cd /tmp
$ tar -xzvf ~/test1.tgz
$ tar -xjvf ~/test2.tar.bz2
$ mkdir test1; cd test1
$ gunzip ~/test1.tgz
$ tar -xvf ~/test1.tar
$ mkdir ../test2; cd ../test2
$ bunzip2 ~/test2.tar.bz2
$ tar -xvf ~/test2.tar
```

Leider ist die `bzip2`-Option von `tar` nicht auf allen Systemen einheitlich: `-I` oder `-j` kommen beide vor. Bei so manch älterem `tar` fehlt sie auch ganz.

13 Netzwerk

13.1 Im Netzwerk bewegen

☞ Bitte Deine Nachbarin, ein Verzeichnis in ihrem Homeverzeichnis mit Rechten zu versehen, die es Dir erlauben, es von ihrer Maschine auf Deine zu kopieren. Logge Dich auf Ihrem Rechner ein und schau nach, ob sie dies getan hat. Kopiere das Verzeichnis auf Deinen Rechner.

```
anna@rechnera:~$ chmod -R a+r verz
anna@rechnera:~$ chmod a+x verz
```

<http://www.surriel.com/patches/>, Rusty Russell unter <http://www.kernel.org/pub/linux/kernel/people/rusty/>...

☞ Sind auf dem Arbeitsrechner die Kernel-Sourcen installiert? Wenn ja: welche Version? Finde einen Punkt heraus, der als Modul konfiguriert ist! Woran hast Du das erkannt?

Wenn Kernel-Sourcen installiert sind, dann befinden sie sich im Verzeichnis `/usr/src/linux` oder in einem anderen, meist um die Versionsnummer erweiterten `/usr/src`-Unterverzeichnis. Die Versionsnummer steht im Zweifelsfall in der Datei `/usr/src/linux/include/linux/version.h`.

Als Modul konfigurierte Kernel-Bestandteile erkennt `fraunhofer` in der Konfigurationsdatei `/usr/src/linux/.config` daran, dass nach der entsprechenden Variablen hinter dem Gleichheitszeichen ein `m` steht. Etwa:

```
CONFIG_IPV6=m
```

☞ Finde heraus, welche Linux-Version auf dem Rechner läuft! Benutze dazu die Befehle `uname -a` und `cat /proc/version` auf der Kommandozeile!

Da die Aufgabe bereits sagt, wie's geht, hier nur ein Beispiel:

```
$ uname -a
Linux lillagroenn 2.4.17 #10 Mon Jun 10 21:49:40 CEST 2002 i686 unknown
unknown GNU/Linux
$ cat /proc/version
Linux version 2.4.17 (root@lillagroenn) (gcc version 2.95.4 20011002
(Debian prerelease)) #10 Mon Jun 10 21:49:40 CEST 2002
```

2.2 Die Shell

☞ In der Datei `/etc/passwd` steht auf jeder Zeile hinter dem letzten Doppelpunkt, welche Shell beim Login für die jeweilige Benutzerin gestartet wird (*Login-Shell*). Was außer `/bin/bash` ist da zu finden?

Zum Anzeigen der Datei kannst Du das Programm `more` verwenden.

Weil das von Installation zu Installation verschieden sein kann, ein Beispiel:

```
postfix:x:103:103::/var/spool/postfix:/bin/false
trish:x:1000:1000:Patricia Jung,,,:/home/trish:/bin/bash
pjung:x:1002:1000::/home/pjung:/bin/bash
nobody:x:65534:65534:nobody:/home:/bin/sh
arnt:x:1001:100::/home/arnt:/usr/bin/zsh
```

Die Z-Shell `zsh` ist wie die Bash eine sehr mächtige, moderne Shell. `sh` steht eigentlich für die Bourne-Shell, beider Ahne. Allerdings haben wir es auf Linux-Rechnern nicht mit der echten Bourne-Shell zu tun; hier ist `/bin/sh` in der Regel ein Link auf die Bash, die in

diesem Fall jedoch Bourne-kompatibel agiert und demzufolge einige Features vermissen lässt.

Der Erklärung der Manpage zu `false`, „tu nichts, und zwar nicht erfolgreich“ ist eigentlich nichts hinzuzufügen: Der entsprechende Pseudo-User `postfix` ist nur dazu da, den Mailserver Postfix auszuführen; niemand kann sich als solcher einloggen, da die „Login-Shell“ `/bin/false` keinerlei Interaktion zulässt und sich sofort beendet.

☞ Finde zwei Beispiele für andere Shells!

Die Korn-Shell `ksh` ist – wie die `Bash` – eine Weiterentwicklung der Bourne-Shell `sh`, allerdings bei weitem nicht so komfortabel wie die `bash`. Noch fortgeschrittener als diese ist die in der vorigen Aufgabe erwähnte Z-Shell `zsh`.

Eine ganz andere Bedienphilosophie verfolgen die Abkömmlinge der (an die Syntax der Programmiersprache C angelehnten) C-Shell `csh`, etwa die `tcsh`.

In eine ganz andere Richtung geht die Perl-Shell `psh` (<http://www.focusresearch.com/gregor/psh/>), deren Syntax der der Skriptsprache Perl folgt.

Nicht zuletzt gibt es Spaß-Shells wie die Adventure-Shell `ash` (<http://www.linux-user.de/ausgabe/2002/04/062-april-shells/april-shells.html>), die die Kommandozeile in eine Art Text-Adventure verwandelt und nicht zu verwechseln ist mit dem leichtgewichtigen Bourne-Shell-Klon `ash`.

2.3 Die Benutzerinnen

☞ Finde Deine UserID mit dem Kommando `id` heraus!

Ein Beispiel:

```
$ id
uid=1002(pjung) gid=1000(trish) groups=1000(trish),100(users)
```

`pjung` hat in diesem Fall die UserID (`uid`) 1002.

☞ Wo steht die UserID in der `/etc/passwd`?

In der dritten Spalte (als Spaltentrenner dient der Doppelpunkt).

☞ Finde mit dem Kommando `groups` heraus, welchen Gruppen Du angehörst!

```
$ groups
trish users
```

Die Userin `pjung` gehört zu den Gruppen `trish` und `users`.

☞ Welchen Gruppen gehört `root` an?

```
for name in `cut -d ":" -f 1 /etc/passwd`; do echo "Hallo $name"!";
done
```

Um das Ausrufezeichen auszugeben ist ein Trick nötig, damit die Shell nicht versucht, es gemäß einer Sonderfunktion zu behandeln: Innerhalb einfacher Anführungszeichen interpretiert die Shell es nicht weiter. `$name` können wir hingegen nicht in einfache Gänsefüßchen setzen, denn wir wollen ja gerade, dass statt der Variablen ihr Inhalt erscheint.

☞ Was macht das Kommando `cut -b 6-17`? Was könnte der Unterschied zu `cut -c 6-17` sein?

`cut -b 6-17` schneidet das sechste bis 17. Byte heraus, die Option `-c` hingegen das sechste bis 17. Zeichen. Das macht keinen Unterschied bei Zeichensätzen, in denen ein Character in einem Byte kodiert ist. Wenn allerdings Unicode o. a. ins Spiel kommen, klappt diese Gleichsetzung nicht mehr.

12 Nützliches Kleinzeug

☞ Drucke eine Textdatei so in eine Datei, dass vier logische Seiten auf einer Druckseite zu liegen kommen. Schau sie mit einem PostScript-Viewer an!

Mit `enscript` gibt es zwei Varianten:

```
enscript -4 -M A4 -r -o bla.ps bla.txt
```

schreibt `bla.txt` im Vierspaltensatz (`-4`) auf ein Blatt A4-Papier. Da das im Hochformat ziemlich eng wird, schaltet `-r` auf Querformat um.

```
enscript -U 4 -M A4 -o bla.ps bla.txt
```

hingegen teilt das Blatt mit `-U A4` in vier Quadranten auf. In beiden Fällen lässt sich die resultierende PostScriptdatei mit `gv bla.ps` anzeigen.

☞ Welche Dateien packt `tar -czvf archiv.tar.gz .*` eigentlich ins Archiv?

Alle Dateien und Verzeichnisse unterhalb des Arbeitsverzeichnisses, auch die, deren Namen mit einem Punkt beginnen. Das Muster `.*` passt nämlich auch auf den einzelnen Punkt, die Abkürzung der Shell für das aktuelle Verzeichnis. `tar -czvf archiv.tar.gz *` ließe die Punktdateien weg – ein Sonderfall, der verhindert, dass diese sonst doppelt eingepackt würden: einmal, weil sie selbst mit Punkt beginnen, ein zweites Mal, weil sie im aktuellen Verzeichnis (also unterhalb von `.`) liegen.

und die runden Klammern und interpretiert sie in der eigenen Sonderbedeutung, und zwar vor der Zuweisung des Wertes zur Variablen.

☞ Schreib ein kleines Skript, das mit Hilfe von `wc` herausfindet, wieviele Zeilen jede einzelne Datei in Deinem Homeverzeichnis lang ist.

```
for datei in `ls -a ~`; do wc -l ~/$datei; done
```

Diese Lösung hat den Nachteil, dass `wc` natürlich nicht die Zeilen eines Verzeichnisses zählen kann und deshalb eine Fehlermeldung der Art `wc: xyz: Is a directory` auf der Standardfehlerausgabe ausgibt. Diese Meldung stört eindeutig. Wir können sie zwar ins Nirvana umleiten ...

```
for datei in `ls -a ~`; do wc -l ~/$datei 2> /dev/null; done
```

..., aber auch jetzt werden Verzeichnisse weiterhin mit der Länge Null aufgelistet. Elegant wäre es, `wc` nur dann von der Leine zu lassen, wenn wir sicher sind, es handelt sich um eine reguläre Datei (`-f`):

```
for datei in `ls -a ~`; do if [ -f ~/$datei ]; then wc -l ~/$datei; fi; done
```

Eine ganz andere Alternative wäre:

```
find ~ -maxdepth 1 -type f -exec wc -l {} \;
```

Die Option `-maxdepth 1` sorgt dafür, dass `find` sich nicht in Unterverzeichnissen des Homeverzeichnisses verliert.

Es gibt übrigens einen Unterschied zwischen beiden Lösungen: Die `for`-Lösung schließt symbolische Links mit ein. Die Hausaufgabe zur Hausaufgabe: Wie verhindere ich das? Zudem sei darauf hingewiesen, dass frau sich die Wildcardmöglichkeiten der Shell zu Nutze machen kann, wenn sie eine `for`-Schleife über den (Teil-)Inhalt eines Verzeichnisses laufen lassen will:

```
for datei in ~/*; do echo $datei; done
```

Beachte dabei aber, dass `*` nicht auf Dateien passt, die mit einem Punkt beginnen.

☞ Schreib ein Shellskript, das für alle Benutzerinnen Deines Rechners eine namentliche Begrüßung ausgibt.

```
$ groups root
root : root
```

Im Beispiel lediglich der Gruppe `root`.

2.4 X-Server

☞ Finde die X-Server-Konfigurationsdatei auf Deinem Rechner mit dem Kommando `locate`.

Ein Beispiel:

```
$ locate XF86Config
/etc/X11/XF86Config-4
```

Hier ist XFree 4 im Spiel. Nach `xorg.conf` sucht frau zum Beispiel mit `locate xorg`.

☞ Schau mit `more` in die Datei hinein. Du solltest darin weitere Aufgabengebiete des X-Servers finden, die wir uns gemeinsam anschauen. Suche mit dem `more`-Befehl / nach den Stichworten `FontPath`, `XkbLayout`, der Section "Pointer" (XFree 3) bzw. Section "InputDevice" (XFree 4/X.org)!

Der `FontPath` (falls spezifiziert) gibt die Adressen von Fontservern und die Pfade zu lokalen Schrift-Datenbanken an. Fehlt dieser Eintrag, kommt ein Default-Pfad zur Anwendung (nähere Informationen gibt die Manpage zur Konfigurationsdatei).

Die Angabe zum `XkbLayout` legt das Tastaturlayout unter X fest. Beachte, das die Belegung auf der Textkonsole und unter X absolut unabhängig voneinander sind.

Im Abschnitt zum `Pointer` bzw. in den `InputDevice`-Sektionen, die sich mit der Maus beschäftigen, spezifiziert frau, was für eine Maus sie hat, an welcher Schnittstelle die angeschlossen ist, welches Mausprotokoll sie spricht, ob ggf. bei einer Zweitastenmaus das gemeinsame Drücken beider Tasten die fehlende mittlere Maustaste simuliert etc.

☞ Logge Dich mit `ssh -X username@hostname` auf dem Rechner Deiner Nachbarin ein und starte dort das Programm `xeyes`. Auf welchem Bildschirm erscheint es? Warum?

Es erscheint auf Deinem eigenen Bildschirm.

Die Option `-X` schaltet das X-Forwarding ein und sorgt so dafür, dass Du Dir grafische Frontends von Programmen, die auf anderen Rechnern laufen, auf Deinem X-Server anzeigen lassen kannst. (Versuch es noch einmal ohne dieses Flag und überprüfe somit, ob X-Forwarding grundsätzlich erlaubt ist.) Das heißt aber nicht, dass Du deswegen auf den X-Server der Nachbarmaschine zugreifen dürftest; dazu bräuchtest Du a) eine entsprechende Berechtigung und müsstest b) die `DISPLAY`-Variable für Dich so umsetzen, dass sie auf den fremden Rechner zeigt.

2.5 Windowmanager

☞ Stell Deinen Kommilitoninnen ein bis zwei Windowmanager kurz (ggf. mit Bild) vor!

Anregungen geben zum Beispiel <http://www.linuxnetmag.com/en/issue5/m5windowmanager1.html>, <http://xwinman.org/> und <http://www.linux.org/apps/all/GUI/WindowManagers.html>.

Achtung: Der Irrtum, dass ein Desktop-Environment ein Windowmanager sei, macht auch vor Webseiten, die Windowmanager auflisten, nicht halt: Selbst dort wird frau KDE oder GNOME gern als „Windowmanager“ untergejubelt bekommen. Allerdings gibt es tatsächlich eine Grauzone: Viele Windowmanager enthalten über ihre eigentliche Aufgabe hinausgehende Funktionalität, viele Mini-Desktop-Umgebungen fühlen sich zu klein und unvollständig an für diesen großen Begriff ...

2.7 Clients, Server und andere Programme

☞ Für Mail (smtp, „Simple Mail Transfer Protocol“) ist ein sogenannter *wellknown port* reserviert, hinter dem der Mailserver lauschen müsste, sofern einer läuft. Suche in der Datei `/etc/services` heraus, welche Portnummer das ist.

Der entsprechende Eintrag lautet

```
smtp          25/tcp      mail
```

☞ Benutze das Kommando `telnet localhost portnummer`, um Dich mit dem SMTP-Port zu verbinden. Läuft ein Mailserver und wenn ja, welcher?

Auf dem folgenden Rechner läuft Postfix:

```
$ telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 lillagroenn.trish.de ESMTP Postfix (Debian/GNU)
quit
221 Bye
Connection closed by foreign host.
```

Mit `quit` beendet frau die Verbindung zum Server.

Bei *wellknown ports* wie Port 25 kann frau statt der Nummer auch das Dienstkürzel als Argument angeben:

```
$ telnet localhost smtp
Trying 127.0.0.1...
telnet: Unable to connect to remote host: Connection refused
```

```
bash      314 trish txt REG      3,2  511400   15414 /bin/bash
[...]
```

`lsdf` („list open files“) zeigt, welche Dateien und Verzeichnisse zur Zeit geöffnet sind und welche Prozesse sich ihrer bedienen. Das Kommando liegt nicht immer in `/usr/sbin`: Wenn es in `/usr/bin` steckt, kann frau es meist ohne Weiteres aufrufen, da `/usr/bin` in der Regel im Pfad enthalten ist.

☞ Suche eine Kombination, die Deinen und den Namen Deiner rechten Nachbarin matcht, aber den Deiner linken Nachbarin nicht.

Auf Stefanie und Veronika würde zum Beispiel `[SV]*i*` passen; eine Kombination, von der Ines, Leena, Andrea, ... ausgeschlossen würden, nicht aber Sabine.

11.2 Shellskripte

☞ Warum reagiert die Shell jetzt auf den Befehl `variable=wert2 $variable` mit `bash: wert2: command not found?`

Weil die Shell Whitespace (hier ein Leerzeichen) als Trennzeichen zwischen Kommandos und Argumenten interpretiert. Im obigen Fall führt sie so zunächst die Zuweisung `variable=wert2` aus und will anschließend den Befehl `$variable` aufrufen. Wegen des Dollarzeichens interpretiert die Shell diesen String zunächst als „Inhalt von `variable`“, also `wert2`. Und ein Kommando dieses Namens gibt es tatsächlich nicht ...

☞ Warum schweigt die Shell auf die Befehlsfolge `unset variable; variable=wert2 $variable` hin, statt die oben genannte Fehlermeldung auszugeben?

`unset variable` tilgt die genannte Variable aus dem „Gedächtnis“ der Shell – sie gibt es nun nicht mehr. `$variable` liefert als Wert demzufolge nichts, und die neue Variable `variable` bekommt daher den String `wert2` plus nichts zugewiesen. Darüber kann sich die Shell nicht beklagen; `echo $variable` gäbe nun `wert2` aus.

☞ Wie ist der Prompt bei Dir definiert? Sieh in der Manpage zu `bash` nach, welche Platzhalter sich darin verwenden lassen. Ändere Deinen Prompt temporär in `aktuelles_verzeichnis(rechnername)#!`

```
trish@lillagroenn:~$ echo $PS1
\u@h:\w\$
$ export PS1="\w(\h)$"
~(lillagroenn)$
```

Das Abschreiben der Manpage erspare ich mir an dieser Stelle :-). Achtet darauf, dass die Gänsefüßchen sehr wichtig sind – ohne sie schnappt sich die Shell die Backslashes

Ruft frau das Kommando im Homeverzeichnis auf, braucht sie `ls` nicht sagen, welches Verzeichnis der Befehl betrachten soll: `~` ist eine Abkürzung für das Homeverzeichnis.

☞ Was tut `cut -c 57-`?

Es schneidet aus der Standardeingabe alles vom 57. Zeichen der Zeile (`-c` steht für „character“) bis zum Schluss heraus.

☞ Überprüfe, ob die Anzahl der Verzeichniseinträge im Verzeichnis `/usr` tatsächlich mit der Zahl im zweiten „`ls -al`“-Block übereinstimmt! Zum Zusammenzählen der Zeilen („lines“) kann das Kommando `wc -l` dienen.

```
$ ls -ald /usr
drwxr-xr-x 13 root root 4096 Jun 25 13:08 /usr
$ ls -al /usr | grep ^d | wc -l
```

13

Durch Weglassen der letzten Pipe sieht frau, dass dabei auch die beiden Verweise `..` und `.` mit gezählt werden. Die Suche nach `d` ist ein Beispiel für einen einfachen regulären Ausdruck im Zusammenhang mit `grep`: Damit sucht das Programm nach `d` am Zeilenanfang.

☞ Sieh Dir Deinen Suchpfad an und erweitere ihn um ein Verzeichnis mit ausführbaren Dateien, das bislang nicht enthalten ist (z.B. `/usr/sbin`). Starte eins der darin enthaltenen Programme (z.B. `/usr/sbin/lsof`) mit und ohne Pfad. Versuche dasselbe in einer anderen Shell.

```
$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/bin/X11:/usr/games
$ lsof
bash: lsof: command not found
$ /usr/sbin/lsof
COMMAND  PID  USER  FD   TYPE    DEVICE    SIZE     NODE NAME
bash     314  trish cwd   DIR     3,6      4096     31873 /home/trish
bash     314  trish rtd   DIR     3,2      4096     2 /
bash     314  trish txt   REG     3,2     511400   15414 /bin/bash
[...]
```

```
$ export PATH=$PATH:/usr/sbin
$ lsof
COMMAND  PID  USER  FD   TYPE    DEVICE    SIZE     NODE NAME
bash     314  trish cwd   DIR     3,6      4096     31873 /home/trish
bash     314  trish rtd   DIR     3,2      4096     2 /
```

Dieses Beispiel zeigt, was passiert, wenn kein Mailserver auf Port 25 lauscht: Der Rechner verweigert die Verbindung.

☞ Finde drei Mailprogramme (Mail User Agents, MUAs) für Linux und prüfe, ob sie auf Deinem Rechner installiert sind! Verwende dazu das Kommando `locate`!

Beispiele für gängige MUAs wären `kmail`, `evolution`, `mutt`, Netscape/Mozilla/Thunderbird, den integrierten Mailclient des Webbrowsers `opera`, `sylpheed`, `pine`, `tkrat` ...

Beim Suchen mit `locate` gilt es auf Groß- und Kleinschreibung zu achten: Die besten Chancen hat frau, wenn sie sich mit durchgängiger Kleinschreibung begnügt, denn einem ungeschriebenen Gesetz zufolge schreiben sich die Dateinamen ausführbarer Dateien (also Programme) in Kleinbuchstaben. Dieser Regel folgt meist auch Software mit grafischer Oberfläche, die sich außerhalb der Kommandozeile gemischt schreibt.

So liefert `locate` KMail in aller Regel keine (oder nicht relevante) Antworten; `locate` `kmail` führt hingegen zum erwünschten Ergebnis.

☞ Was tun `pwd` und `exit`?

`pwd` gibt den Verzeichnispfad des Arbeitsverzeichnisses aus. `exit` schließt die aktuelle Shell. Wenn dies die initiale Login-Shell war, loggt frau sich damit aus.

☞ Was passiert, wenn frau `mkdir` ohne Argument aufruft? Warum?

```
$ mkdir
mkdir: too few arguments
Try 'mkdir --help' for more information.
```

`mkdir` beschwert sich, dass es mit zuwenig Argumenten aufgerufen wurde und ermutigt dazu, sich mit `mkdir --help` eine Kurzanleitung zu holen. Aus der ...

```
Usage: mkdir [OPTION] DIRECTORY...
[...]
```

geht hervor, dass zwar Optionen, nicht aber die Verzeichnisangabe optional sind – ein Verzeichnis anzulegen, das keinen Namen hat, ergibt schließlich keinen Sinn.

3 Hilfe zur Selbsthilfe

3.1 Selbstauskunft

☞ Was tut man `-k`? Wofür könnte das `k` stehen?

`-k` steht für „keyword“. Die Option sucht in allen Kurzbeschreibungen nach dem als Argument angegebenen Schlüsselwort:

```
$ man -k fdisk
cfdisk (8) - Curses based disk partition table manipulator for
```

Linux

```
fdisk (8)      - Partition table manipulator for Linux
sfdisk (8)    - Partition table manipulator for Linux
```

☞ Was sind *cat pages*? Nutze die Manpage zu *man*!

Aus dem Manpage-Code bereits erzeugte, vorformatierte Manpages, die ihren Namen daher haben, dass frau sie sich fein säuberlich „gelayoutet“ direkt mit dem Kommando *cat* (bzw. *zcat*, da Catpages aus Platzgründen meist mit *gzip* komprimiert abgelegt werden) anzeigen lassen kann.

Hier kommt der alte Trick „Tausche Platz gegen Zeitersparnis“ zur Anwendung: Ein einfaches *cat* geht viel schneller als der Formatierungsprozess. Wer hingegen unter Platznot leidet, wird eher in Kauf nehmen, dass sie etwas länger warten muss. Bei modernen PCs spielt dies aber weder in der einen noch in der anderen Rechnung eine nennenswerte Rolle.

☞ *man man* bzw. *man 1 man* beschreibt nicht, welche Sektionen es gibt. Finde mit *man -k man* oder *apropos man* heraus, in welchen anderen Sektionen *man* auftaucht und lies nach, welche Sektionen es gibt.

```
$ apropos man
```

```
[...]
man (1)      - an interface to the on-line reference manuals
man (7)      - macros to format man pages
mandb (8)   - create or update the manual page index caches
manpath (1) - determine search path for manual pages
manpath (5) - format of the /etc/manpath.config file
[...]
```

```
$ man 7 man
[...]
```

The manual sections are traditionally defined as follows:

- 1 Commands

Those commands that can be executed by the user from within a shell.
- 2 System calls

Those functions which must be performed by the kernel.
- 3 Library calls

Most of the libc functions, such as *qsort*(3)

Wir setzen den Befehl *./aufraeumen* ab. *./* ist nichts anderes als eine relative Pfadangabe und besagt, dass sich das eben ausführbar gemachte Skript im aktuellen Arbeitsverzeichnis befindet: Solange *aufraeumen* nicht in einem Verzeichnis im Suchpfad liegt, kann es nicht durch Angabe seines Namens allein gestartet werden.

☞ Wieviele (rpm-)Pakete sind auf Deinem Rechner installiert? Nutze dazu eine Pipe und das Kommando *wc -l*.

Ein Beispiel:

```
$ rpm -qa | wc -l
506
```

☞ Wie ließe sich diese Aufgabe ohne Pipe, aber dafür mit Aus- und Eingabeumlenkung in eine Datei lösen?

```
$ rpm -qa > /tmp/bla
$ wc -l < /tmp/bla
506
```

☞ Worin unterscheidet sich die Ausgabe von *wc -l datei* von der Variante mit Eingabeumlenkung und warum?

```
$ wc -l < /tmp/bla
506
$ wc -l /tmp/bla
506 /tmp/bla
```

wc gibt immer den Namen der Datei mit aus, die es bearbeitet. Im Fall der Eingabeumlenkung hat es jedoch nichts mit einer Datei zu tun, sondern mit der Standardeingabe, die selbst keinen Dateinamen hat. Dass wir den Inhalt der Datei in die Standardeingabe „hineingeschüttet“ haben, weiß *wc* schließlich nicht.

☞ Suche mit einer einzigen Kommandozeile die Namen aller Dateien und Verzeichnisse in Deinem Homeverzeichnis heraus, die heute verändert wurden, und speichere das Ergebnis in einer neuen Datei namens *heute* ab. Benutze dazu die Kommandos *ls* (mit passenden Optionen), *grep* und *cut -c 57-*, Pipes und eine Ausgabeumlenkung.

Am 28. August lautete (eine) Lösung wie folgt:

```
ls -al ~ | grep "Aug 28" | cut -c 57- > heute
```

☞ Finde heraus, welche Init-Skripte im Default-Runlevel Deines Rechners beim Booten *nicht* gestartet werden.

Ist 2 der Default-Runlevel, gilt es zur Beantwortung dieser Frage, im `init.d`-Verzeichnis ein Skript zu finden, zu dem *kein* mit dem Buchstaben `S` beginnender Link im `rc2.d`-Verzeichnis führt.

Um ganz genau zu sein, darf es zudem auch in `/etc/rcS.d` (so vorhanden) keinen entsprechenden Link geben.

Gute Kandidaten sind hier die Initskripte von Datenbank- oder Internetservern.

10 Der Ausschaltknopf

☞ Wie würdest Du lediglich eine Warnung an alle eingeloggten User schicken, ohne den Countdown tatsächlich einzuleiten? Nimm die Manpage zu `shutdown` zu Hilfe!

`man shutdown` listet wie jede Manpage zunächst die SYNOPSIS des Kommandos auf. In eckigen Klammern stehende Teile wie die Flags oder die Warnmeldung sind optional. Hingegen muss die Zeit bis zum Shutdown angegeben werden. Die Warnmeldung kommt nach diesem Argument. Im Abschnitt OPTIONS erklärt die Manpage zudem die passende Option `-k`:

```
/sbin/shutdown [-t sec] [-arkhncfF] time [warning-message]
[...]
-k      Don't really shutdown; only send the warning mes-
sages to everybody.
```

Der (nur `root` erlaubte) Befehl `/sbin/shutdown -k 2 "bla"` erklärt den Usern in einer Broadcast-Message im Abstand von je einer Sekunde dreimal, dass das System demnächst bzw. jetzt in den Wartungsmodus ginge. Außerdem bekommen sie die unglaublich informative Meldung `bla` zu sehen, und zum Schluss passiert dann doch nichts ...

11 Die Bash

11.1 Eingebaute Befehle und Hilfsmittel

☞ Versuche, bei der Eingabe des Befehls `mkdir /tmp/bla` mit so wenigen Buchstaben wie möglich auszukommen!

Auf den meisten Systemen sollte `(m)(k)(Tab)(/)(Tab)(b)(l)(a)(←)` ausreichen. Eventuell muss vor dem Schrägstrich noch ein Leerzeichen eingefügt werden.

☞ Nehmen wir an, der letzte Befehl sei `chmod a+x einladung aufräumen` gewesen. Was passiert, wenn wir nun `(/)(Esc)(←)` eingeben?

```
4 Special files
   Files found in /dev

5 File formats and conventions
   The format for /etc/passwd and other
   human-readable files.

6 Games

7 Macro packages and conventions
   A description of the standard file system
   layout, network protocols, ASCII and
   other character codes, this man page, and
   other things.

8 System management commands
   Commands like mount(8), many of which
   only root can execute.

9 Kernel routines
   This is an obsolete manual section. Once
   it was thought a good idea to document
   the Linux kernel here, but in fact very
   little has been documented, and the docu-
   mentation that exists is outdated
   already. There are better sources of
   information for kernel developers.
```

[...]

Das die Ausgabe von `apropos man` auf vielen Systemen sehr lang wird (die Silbe *man* kommt halt auch in oft in der Kurzbeschreibung benutzten Wörtern wie „command(line)“ oder „manipulation“ vor), wird die erfahrene Nutzerin die Ausgabe auf Passendes beschränken wollen. So zum Beispiel zeigt das Kommando

```
apropos man | grep ^man
```

nur noch die Zeilen aus der `apropos`-Ausgabe an, die mit den drei Buchstaben `man` beginnen (^). `grep -w man` würde tatsächlich nur noch die beiden Manpages zu `man(1)` und `man(7)` finden, denn `-w` überredet `grep` dazu, nur noch auf das ganze Wort `man` am Zeilenanfang anzuspringen.

☞ Vergleiche Man- und Infopage zum Kommando `groups`!

Auf meinem Debian/Testing-System führt mich `info groups` in die verlinkte Dokumentation des `coreutils`-Pakets, dessen Mitglied das Kommando `groups` ist. Während sich

der Inhalt der darin enthaltenen Seite zu `groups` nur sprachlich von dem der Manpage unterscheidet, empfinde ich persönlich die Darstellung der Infopage als unübersichtlich. Aber das ist Geschmackssache...

4 Dateisystem und Dateibaum

4.1 Dateisysteme

☞ Wie groß ist der Arbeitsspeicher Deines Systems?

```
$ ls -al /proc/kcore
-r----- 1 root root 268439552 Aug 24 21:23 /proc/kcore
```

Die Dateigröße von `/proc/kcore` entspricht der Größe des Hauptspeichers, im Beispiel also 268 439 552 Bytes, das sind ($\frac{268\,439\,552}{1024 \cdot 1024} = 256$) MByte.

☞ Schau zuhause in Deinen Arbeitsspeicher rein. Aber Vorsicht, *Pager* wie `more` oder `less` bekommen Probleme mit den nicht druckbaren Zeichen, also lieber vorher mit `strings` filtern:

```
cat /proc/kcore | strings | less
```

Das folgende Beispiel zeigt Ausschnitte aus Patricias RAM, während sie diese Zeilen schreibt:

```
# cat /proc/kcore | strings | less
CORE
CORE
vmlinux
BOOT_IMAGE=Linux2.4 ro root=302
CORE
Unknown interrupt
000707070c0c0c0d0d0d0000000a0a0a050505000000
[...]
~/artikel/informatika/kursskript/skript/unixkurs.tex
    elseif b:netrw_method == 3          " read with ftp + machine, id,
    passwd, and fname
    "      Decho "DBG: read via ftp+mipf (method #3)"
if g:netrw_ftp == 1
\end{document}
diese Zeilen schreibt:
Das folgende Beispiel zeigt Ausschnitte aus Patricias RAM, w
hrend sie
\snt{cat /proc/kcore | strings | less}
druckbaren Zeichen, also lieber vorher mit \snt{strings} filtern:\\
```

```
find / -name init* -type d
```

Dieser sucht unterhalb des Wurzelverzeichnisses `/` nach einem Verzeichnis (`-type d` – „vom Typ *directory*“), dessen Name mit den Buchstaben `init` beginnt.

Ähnlich lässt sich nach allen Links unterhalb des Verzeichnisses `/etc` suchen, die mit dem Buchstaben `S` und einer Ziffer beginnen (und damit prädestiniert dafür sind, auf Initskripte zu zeigen):

```
# find /etc -type l -name S[0-9]*
trish@lillagroenn:~$ find /etc -type l -name S[0-9]*
/etc/rc0.d/S30urandom
/etc/rc0.d/S20sendsigns
[...]
```

will sich frau gleich mit anzeigen zu lassen, wohin diese links führen, geht das zum beispiel so:

```
# find /etc -type l -name S[0-9]* -exec ls -l {} \;
lrwxrwxrwx 1 root root 17 Jan 26 2002 /etc/rc0.d/S30urandom ->
../init.d/urandom
lrwxrwxrwx 1 root root 18 Jan 26 2002 /etc/rc0.d/S20sendsigns ->
../init.d/sendsigns
[...]
```

Die `find`-Option `-exec` führt für jede Fundstelle den dahinterstehenden Befehl aus. Anstelle des geschweiften Klammernpaars setzt `find` den Namen der gefundenen Datei ein. Das Semikolon am Ende wird benötigt, um die damit entstehenden vielen Befehle voneinander zu trennen; und damit die Shell nicht fälschlicherweise meint, dieses Semikolon sei für sie gedacht, muss es mit geschützt werden.

☞ Welches Initskript ist dafür verantwortlich, dass der Cron-Daemon startet?

Gute Kandidaten sind `/etc/init.d/crond` oder `/etc/init.d/cron`. Damit der Daemon beim Booten tatsächlich startet, muss es aber aus dem jeweiligen `rc.d`-Verzeichnis verlinkt sein, zum Beispiel:

```
$ ls -al /etc/rc2.d/S*cron
lrwxrwxrwx 1 root root 14 Jan 26 2002 /etc/rc2.d/S89cron -> ../init.d/cron
```

9 Bootvorgang

☞ Schau in die `/etc/lilo.conf` und versuche herauszufinden, welche Bootmöglichkeiten Dein Rechner bietet.

Gut möglich, dass wir diese Aufgabe im Kurs nicht lösen können, da die Lilo-Konfigurationsdatei oft nur für `root` zugänglich ist.

Im folgenden Beispiel gibt es die Möglichkeit zum Booten zweier verschiedener Linuxkernel (`/boot/vmlinuz-2.4.17` und `/boot/vmlinuz-2.4.17-686`) sowie einer Windows-Installation. Anhand der Labels kann die Benutzerin im Bootmenü aussuchen, was gebootet wird. Trifft sie keine Entscheidung, kommt der Linux-Kernel mit dem Label `Linux2.4` zum Zuge (`default`). Ein `#` leitet eine Kommentarzeile ein. Alle übrigen Parameter sind zum Beispiel in der Manpage zu `lilo.conf` nachzulesen.

```
# Boot up Linux by default.
default=Linux2.4

image=/boot/vmlinuz-2.4.17
label=Linux2.4
read-only
optional

image=/boot/vmlinuz-2.4.17-686
label=Linux2.4deb
read-only
initrd=/initrd.img

other=/dev/hda1
label=win
```

☞ Finde in der `/etc/inittab` heraus, in welchen Runlevel Dein Rechner von sich aus startet. Welche Zeile könnte das sein?

Der folgende Beispielrechner bootet ins Multiuser-Runlevel 2:

```
# The default runlevel.
id:2:initdefault:
[...]
# Runlevels 2-5 are multi-user.
```

☞ Wo liegen die Initkripte und die jeweiligen Links auf Deinem Rechner?

Wenn unterhalb des Verzeichnisses `etc` tatsächlich kein Ordner zu finden sein sollte, der sich `init.d` o. ä. nennt, lässt sich der `find`-Befehl zur Hilfe nehmen:

[...]

4.2 Mounten

☞ Schau Dir die Datei `/etc/fstab` an und ermittle, welche Arten von Dateisystemen im Rechenzentrum verwendet werden und was damit gemacht werden kann.

```
$ cat /etc/fstab
/dev/hda2 / ext2 defaults,errors=remount-ro 0 1
/dev/hda3 none swap sw 0 0
proc /proc proc defaults 0 0
/dev/fd0 /floppy auto defaults,user,noauto 0 0
/dev/cdrom /cdrom iso9660 defaults,ro,user,noauto 0 0
/dev/hda5 /usr ext2 defaults 0 2
/dev/hda6 /home ext2 defaults 0 2
/dev/hda7 /opt ext2 defaults 0 2
```

Auf diesem Beispielrechner gibt es eine IDE-Platte `/dev/hda`, auf der mehrere Partitionen vorhanden sind. Die vom vorliegenden System benutzte Swap-Partition ist `/dev/hda3`. Zusätzlich gibt es diverse Datenpartitionen (`/`, `/usr`, `/home` und `/opt`, auf denen das Ext2-Dateisystem zum Einsatz kommt. Für alle werden die Default-Einstellungen¹ benutzt. Tritt beim Mounten der Root-Partition ein Fehler auf, ist für diesen Fall festgelegt, dass die Partition „read only“, also unbeschreibbar, remountet wird (`errors=remount-ro`).

Für Wechseldatenträger sieht diese `fstab` vor, dass sie – selbst wenn ein Datenträger vorhanden ist – nicht automatisch beim Booten gemountet werden (`noauto`), aber dafür auch von unprivilegierten Userinnen an passenden Stellen (`/floppy` für Disketten aus „Laufwerk A:“, `/cdrom` für CD-ROMs²) eingehängt werden dürfen (`user`). Bei den Disketten soll `mount` das Dateisystem selbst herausfinden (`auto`); CD-ROMs werden nur lesbar eingehängt.

Das `proc`-Filesystem wird on-the-fly erzeugt, nimmt keinen Platz auf Festplatten weg und braucht deshalb keine Partitionsangabe.

☞ Wie voll sind die Platten momentan?

```
$ df
Filesystem      1k-blocks    Used Available Use% Mounted on
/dev/hda2          719452    271196   411712   40% /
```

¹Das sind `rw`, `suid`, `dev`, `exec`, `auto`, `nouser`, `async`; da die Erklärungen in der Manpage zu `mount` (8) wirklich gut sind, sei darauf verwiesen.

²Audio-CDs werden nicht gemountet!

```

/dev/hda5          1921156  1446072  377492  80% /usr
/dev/hda6          2885780  2595176  144016  95% /home
/dev/hda7          1438888  1233520  132276  91% /opt

```

Im Beispiel sind von den ca. 703 MByte (719 452/1024) der Root-Partition etwa 40 Prozent belegt: ca. 402 MByte (411 712/1024) haben noch Platz. Die Ausgaben zu den übrigen Partitionen interpretieren sich analog.

4.5 Bewegen im Verzeichnisbaum

☞ Handelt es sich bei der Datei `xmag` im Verzeichnis `/usr/X11R6/bin/` um ein Verzeichnis, einen Link, ein Programm oder eine normale Datei?

```

$ cd /usr/X11R6/bin/
$ ls -F xmag
xmag*

```

Also um eine ausführbare Datei. Unterhalb von `/usr/X11R6` befindet sich das X-Window-System und eine Menge zugehöriger Programme, darunter die Lupe `xmag`.

5 Benutzerinnen und ihre Rechte

5.1 Dateirechte

☞ Worauf beziehen sich die letzten zwei Beispielzeilen zu `.` und `..`?

Auf das Arbeitsverzeichnis bzw. dessen Oberverzeichnis.

5.2 Dateirechte ändern

☞ Das Verzeichnis `informatica` besitzt aktuell die Zugriffsrechte `rwrx-xr--`. Wie ändern sich diese mit dem Kommando `chmod u=rwx,g+w,o=xt informatica`? Welche Auswirkungen hätte ein anschließendes `chmod a=rwx,o+t informatica`?

Nach dem ersten `chmod`-Befehl lauten die Rechte `rwrxwx--t`. Wegen des Gleichsetzungsoperators resultiert `chmod a=rwx` in Lese-, Schreib- und Wechselrechten für alle (`rwrxwxrwx`). Um das Sticky-Bit zu erhalten, muss es noch einmal (`o+t`) neu vergeben werden. Anschließend lauten die Rechte `rwrxwxrwt`.

☞ Erzeuge mit dem Befehl `touch rechte` eine Datei `rechte` in Deinem Homeverzeichnis. Welche Zugriffsrechte hat sie? Benutze die oktale Version, um niemanden außer Dich auf die Datei zugreifen zu lassen. Du selbst verleiht Dir Lese- und Schreibrechte.

```

[1]+  Stopped                  mozilla
$ xeyes &
$ fg
mozilla

```

(Strg)+C

```

$ ps -auxw | grep xeyes
Bad syntax, perhaps a bogus '-'?
trish   3837  0.0  0.5  2936 1388 pts/9    S    00:59   0:00 xeyes
trish   3841  0.0  0.1  1336  436 pts/9    S    01:00   0:00 grep xeyes
$ kill 3837

```

☞ Starte drei `xeyes` und beende sie alle gleichzeitig.

```

$ xeyes & xeyes & xeyes &
[1] 3864
[2] 3865
[3] 3866
$ killall xeyes
[1] Terminated          xeyes
[2]- Terminated         xeyes
[3]+ Terminated         xeyes

```

8.4 Die Zeit unter Kontrolle

☞ Wann wird der folgende Eintrag abgearbeitet?

```
0 13 * 9 5-7 echo 'Der Kurs geht weiter!'
```

An Freitagen, Samstagen und Sonntagen im September jeweils um 13 Uhr.

☞ Lass Dir in 10 Minuten von Cron eine Übersicht über die Plattenbelegung (`df`) schicken. Die aktuelle Zeit findest Du mit dem Kommando `date` heraus.

Angenommen, „in 10 Minuten“ wäre 20.15 Uhr am 14. September, so lautete der Eintrag

```
15 20 14 9 * df
```

Wenn nach `crontab -e` der `vi` auftaucht, nicht vergessen, mit `i` in den Editiermodus zu schalten und zum Speichern `(Esc):w(q)` zu benutzen.

```
[...]
$ cd stow-1.3.3
$ ./configure --prefix=$HOME
$ make
make: Nothing to be done for 'all'.
$ make install
```

Sorry – das war ein extrem unspektakuläres Beispiel: `stow` ist nur ein Perlskript, muss also nicht kompiliert werden. Aus diesem Grund bekommen wir nach dem `make`-Aufruf keinen Compiler-Output zu sehen, sondern die lapidare Meldung, dass es kein entsprechendes Make-Target gibt.

☞ Welche Version des C-Compilers `gcc` käme auf dem Pool-Rechner zum Einsatz? Finde den passenden Aufruf mit der `gcc`-Option `--help` heraus!

```
$ gcc --help
Usage: gcc [options] file...
Options:
  --help                Display this information
                        (Use '-v --help' to display command line options of sub-processes)
  -dumpspecs            Display all of the built in spec strings
  -dumpversion          Display the version of the compiler
[...]
$ gcc -dumpversion
2.95.4
```

Mit der Option `--help` lässt sich den meisten Kommandozeilenprogrammen eine Kurzhilfe entlocken: Sollten sie über diese Option nicht verfügen, sorgt diese Fehlbedienung in der Regel dafür, dass die Hilfeseite trotzdem erscheint.

8.3 Prozesse im Griff

☞ Starte `mozilla` im Vordergrund einer Shell. Schick den Webbrowser manuell in den Hintergrund. Starte das Programm `xeyes` in derselben Shell im Hintergrund. Hole `mozilla` wieder in den Vordergrund. Beende Mozilla, ohne auf sein Menü zuzugreifen. Suche nach dem `xeyes`-Prozess in der Prozesstabelle und beende ihn mit dem `kill`-Befehl.

```
$ mozilla
```

(Strg)+(z)

```
$ touch rechte
$ ls -al rechte
-rw-r--r--  1 trish  trish          0 Aug 24 22:33 rechte
$ chmod 600 rechte
$ ls -al rechte
-rw-----  1 trish  trish          0 Aug 24 22:33 rechte
```

☞ Welches Ergebnis hat der Befehl `chmod 5771 datei`?

Er vergibt das Sticky- und das SUID-Bit (1+4=5), alle Rechte für Eigentümerin und Gruppe sowie Ausführbarkeitsrecht für die Übrigen (`-rwsrwx--t`).

☞ Probiere aus, was passiert, wenn Du Dir von einer Dir gehörenden Datei und einem Verzeichnis alle Rechte entziehst!

```
$ chmod 0000 datei
$ ls -al datei
-----  1 trish  trish          0 Aug 24 22:33 datei
$ cat datei #Leseversuch
cat: datei: Permission denied
$ vi datei #Schreibversuch

~
"datei" [Permission Denied]
$ ./datei #Versuch, die Datei auszuführen
bash: ./datei: Permission denied
$ rm datei #Löschversuch
rm: remove write-protected file 'datei'? y
$ ls -al datei
ls: datei: No such file or directory
```

Außer dem Lösversuch ist alles gescheitert – und jener klappte auch nur auf Nachfrage, weil wir die Datei besitzen. Den Editor `vi` kann frau übrigens mit `:w!` dazu überreden, eine eigene schreibgeschützte Datei doch zu schreiben.

Ab dem `#` interpretiert die Shell den Rest der Zeile als Kommentar – diese Erklärungen braucht Ihr natürlich nicht eingeben.

```
$ mkdir verzeichnis
$ chmod 0000 verzeichnis
$ touch verzeichnis/bla #Schreibversuch
touch: creating 'verzeichnis/bla': Permission denied
$ cd verzeichnis #Wechselversuch
```

```

bash: cd: verzeichnis: Permission denied
$ ls -al verzeichnis/      #Leseversuch
ls: verzeichnis/: Permission denied
$ rmdir verzeichnis
$ ls -al verzeichnis
ls: verzeichnis: No such file or directory

```

Ändere Deine umask so, dass Deine neuen Dateien niemand mehr außer Dir lesen kann und dass in neue Verzeichnisse nur Du selbst wechseln darfst.

umask 077 überlässt Dir Lese- und Schreibrechte für neue Dateien und Verzeichnisse sowie das x-Bit für letztere. Die Gruppen-Rechte und die der anderen werden jeweils auf nichts gesetzt.

Das Umsetzen der umask hat natürlich nur Einfluss auf die aktuelle Shell und ihre Kindprozesse. Soll sie immer gelten, gehört sie in eine der Startdateien der Shell.

5.3 Besitzerinnen ändern

chmod, chgrp und chown lassen sich auch auf alle Dateien und Unterverzeichnisse in einem Ordner gleichzeitig anwenden. Um welche Option muss frau den Befehl dann jeweils erweitern? Nutze diese Option, um jeweils allen anderen das Leserecht auf alle Dateien eines Verzeichnisses zu entziehen!

man chmod, man chgrp und chown geben hier übereinstimmend die Option -R für „rekursiv“ an.

```

$ ls -alR bildchen/
total 12832
drwxr-xr-x  4 trish trish   4096 Aug 11 18:43 .
drwxr-xr-x 85 trish trish   8192 Aug 31 17:44 ..
drwxr-xr-x  2 trish trish   4096 Jul 14 13:02 .xvpics
-rw-r--r--  1 trish trish  867389 Jul  7  2003 CIMG0012.JPG
[...]
bildchen/.xvpics:
total 20
drwxr-xr-x  2 trish trish   4096 Jul 14 13:02 .
drwxr-xr-x  4 trish trish   4096 Aug 11 18:43 ..
-rw-r--r--  1 trish trish   4557 Jul 14 13:13 new_passport.jpg
$ chmod -R o-r bildchen
$ ls -al bildchen/
total 12832
drwxr-x--x  4 trish trish   4096 Aug 11 18:43 .
drwxr-xr-x 85 trish trish   8192 Aug 31 17:44 ..
drwxr-x--x  2 trish trish   4096 Jul 14 13:02 .xvpics

```

von dieser Userin (etwa auf einer anderen Maschine) benutzten Passwort anlegen, ohne das Passwort kennen zu müssen.

Wie würdest Du als root Billes Homeverzeichnis löschen?

```
# rm -rf /home/bille
```

8.2 Installieren von Software

Zu welchem Paket gehört das Kommando ls? Nenne fünf weitere Programme, die zu diesem Paket gehören! Lies die Paketbeschreibung.

```

$ rpm -qf /bin/ls
fileutils-4.0-3
$ rpm -qi fileutils
Name       : fileutils                Vendor: (none)
Version    : 4.0                    Distribution: (none)
Release    : 3                     Build Host: bra.gulbrandsen.priv.no
Install Date: 2001-06-12T14:16:51Z Build Date: 2001-06-12T14:16:03Z
Size       : 1220799              Source RPM: fileutils-4.0-3.src.rpm
Group      : System/Base
Copyright  : GPL
Packager   : Stefan Probst <sp@caldera.de>
Summary    : GNU File utilities.
Description:
These are the GNU file management utilities. Most of these programs
have significant advantages over their Unix counterparts, such as
greater speed, additional options, and fewer arbitrary limits.

The programs that come with this package are: chgrp, chown, chmod, cp,
dd, df, du, install, ln, dir, vdir, ls, mkdir, mkfifo, mknod, mv, rm,
rmdir, sync and touch.

```

Lade das Paket stow-Paket http://ftp.debian.org/debian/pool/main/s/stow/stow_1.3.3.orig.tar.gz herunter und installiere es in Deinem Homeverzeichnis!

```

$ ncftppget http://ftp.debian.org/debian/pool/main/s/stow/stow_1.3.3.orig.
tar.gz
$ tar -xzvf stow_1.3.3.orig.tar.gz
stow-1.3.3/
stow-1.3.3/README

```

Die Superuserin (die theoretisch auch anders heißen kann als `root`) hat immer die UserID 0. Im Beispiel ist `trish` der Gruppe `trish` zugeordnet.

☞ Finde mit Hilfe der `manpage` heraus, wofür die `useradd`-Optionen `-c`, `-d`, `-g`, `-p`, `-s` und `-u` stehen!

```
$ man useradd
[...]
-c comment
The new user's password file comment field.
-d home_dir
The new user will be created using home_dir as the
value for the user's login directory. The default
is to append the login name to default_home and use
that as the login directory name.
[...]
-g initial_group
The group name or number of the user's initial
login group. The group name must exist. A group
number must refer to an already existing group.
The default group number is 1.
[...]
-p passwd
The encrypted password, as returned by crypt(3).
The default is to disable the account.

-s shell
The name of the user's login shell. The default is
to leave this field blank, which causes the system
to select the default login shell.

-u uid The numerical value of the user's ID. This value
must be unique, unless the -o option is used. The
value must be non-negative. The default is to use
the smallest ID value greater than 99 and greater
than every other user. Values between 0 and 99 are
typically reserved for system accounts.
[...]
```

Achtung: Mit `-p` wird wirklich das bereits verschlüsselte Passwort als Argument angegeben. Dieser Parameter taugt also nicht zum initialen Setzen eines Passworts (das macht `root` mit `passwd`), sondern zum Übertragen bereits verschlüsselter Passwörter auf einen neuen Account. Somit kann die Systemadministratorin einen neuen Account mit einem

```
-rw-r----- 1 trish trish 867389 Jul  7  2003 CIMG0012.JPG
[...]
bildchen/.xvpics:
total 20
drwxr-x--x 2 trish trish  4096 Jul 14 13:02 .
drwxr-x--x 4 trish trish  4096 Aug 11 18:43 ..
-rw-r----- 1 trish trish  4557 Jul 14 13:13 new_passport.jpg
```

Die Option `-R` gibt es, wie hier zu sehen, auch für das Kommando `ls` und sorgt dafür, dass auch der Inhalt der Unterordner – wie hier `.xvpics` – gleich mit angezeigt wird.

6 Weitere Befehle zur Dateiverwaltung

6.1 Dateien und Verzeichnisse löschen

☞ Lege ein Verzeichnis namens `loeschtest` an und wechsele hinein. Füll es mit den Dateien `nichtwichtig.txt`, `nichtwichtig`, `blafasel`, `.bla` und dem Verzeichnis `ordner`. Überprüfe die obengenannten Aussagen mit dem Befehl `rm nichtwichtig.*!` Was passiert mit `ordner`? Warum?

```
$ mkdir loeschtest
cd trish@lillagroenn:/tmp$ cd loeschtest
$ touch nichtwichtig.txt nichtwichtig blafasel .bla
$ mkdir ordner
$ ls -aF
./ ../ .bla blafasel nichtwichtig nichtwichtig.txt ordner/
$ rm nichtwichtig.*
rm: cannot remove 'nichtwichtig.': No such file or directory
rm: cannot remove 'ordner': Is a directory
$ ls -aF
./ ../ .bla ordner/
```

`rm` kann ohne Zusatzoption keine Verzeichnisse löschen, weshalb `ordner` verschont bleibt. Eine Datei namens `nichtwichtig` haben wir nicht angelegt, weshalb sie nicht weggeworfen werden kann. Das Muster `*` passt nicht auf Dateien, die mit einem Punkt beginnen, weshalb auch `.bla` übrig bleibt.

☞ Probiere das rekursive Löschen am Verzeichnis `loeschtest` aus der vorherigen Aufgabe aus! Ist das Unterverzeichnis `loeschtest/ordner` anschließend immer noch vorhanden?

```
$ pwd
```

```

/tmp/loeschtest
$ cd ..
$ rm -r loeschtest
$ ls -al loeschtest/ordner
ls: loeschtest/ordner: No such file or directory
$ ls -al loeschtest
ls: loeschtest: No such file or directory

```

Mit `rm -r` verschwinden auch alle in einem Verzeichnis enthaltenen Punktdateien und Unterverzeichnisse.

☞ Schau Dir mit dem Kommando `alias` an, welche Aliase Du derzeit benutzen kannst!

Beispiel:

```

$ alias
alias ls='ls --color=auto'

```

6.3 Verzeichnisse manipulieren

☞ Lege ein Verzeichnis `temporaer` an (Du kannst es gern mit einigen anderweitig herkopierten Dateien füllen). Wieviele Kopien davon gibt es nach den drei Befehlen `cp temporaer TMP`, `cp -r temporaer temp` und `mv temporaer tmp`? Warum?

```

$ mkdir temporaer
$ cp temporaer TMP
cp: omitting directory 'temporaer'
$ cp -r temporaer temp
$ mv temporaer tmp
$ ls -F
[...]
temp/  tmp/
[...]

```

Zwei Stück, da nur eine Kopieraktion, `cp -r`, erfolgreich war und `mv` das Verzeichnis nicht kopiert, nur umbenennt. `cp` ohne Optionen möchte Dateien kopieren und lässt Verzeichnisse weg („omitting directory“). Selbst wenn `temporaer` Dateien enthält, werden die von einem einfachen `cp` nicht kopiert, denn `cp` arbeitet zunächst einmal nicht rekursiv.

☞ Gibt es die Option `-p` auch für `rmdir`? Wenn ja: Kann sie als Ersatz für `rm -r` dienen?

Ja und nein: `rmdir -p` fängt zwar von „hinten“ an zu löschen, sobald aber eines der Verzeichnisse nicht leer ist, stellt der Befehl seine Arbeit ein:

- Adie (<http://www.fox-toolkit.org/adie.html>)
- *ee* und *ae* (<http://mahon.cwx.net/>)
- CE (<http://jolomo.net/ce/>)
- Crisp (<http://www.vital.com/>)
- Glimmer (<http://glimmer.sourceforge.net/>)
- Jed (<http://www.jedsoft.org/jed/>)
- ...⁵

8 Crashkurs für Hobby-Sysadminen

8.1 Benutzerinnenverwaltung

☞ Werden auf Deinem Rechner Shadow-Passwörter verwendet? Wenn ja, versuch einen Blick in die `/etc/shadow` zu werfen!

```

$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
[...]

```

Auf dem Beispielrechner offensichtlich ja: In der zweiten Spalte steht ein `x` anstelle eines verschlüsselten Passworts.

```

$ cat /etc/shadow
cat: /etc/shadow: Permission denied

```

☞ Welche UserID hat `root`, welche hast Du? Welchen Gruppen bist Du zugeordnet?

```

$ id root
uid=0(root) gid=0(root) groups=0(root)
$ id
uid=1000(trish) gid=1000(trish) groups=1000(trish)

```

⁵Es dürfte deutlich geworden sein, dass Texteditoren wirklich in rauen Mengen zur Auswahl stehen.

☞ Wechsle ins Verzeichnis `/tmp` und suche mit `find` nach dem in einer der letzten Aufgabe erstellten `temp`-Directory (oder alternativ nach einer anderen Datei irgendwo unterhalb Deines Homeverzeichnisses).

```
/tmp$ find ~ -name temp
/home/trish/staroffice6.0/user/temp
```

Statt `~`, einer Kurzform für das eigene Homeverzeichnis, darf natürlich auch `/home/ifnnnn` stehen.

☞ In welchen Verzeichnissen liegt der „Quellcode“ der verschiedenen Manpages zu `man`?

Die komprimierten englischen Fassungen liegen (in der Regel) in `/usr/share/man/man1/man.1.gz` und `/usr/share/man/man7/man.7.gz`. Sie lassen sich zum Beispiel mit dem Befehl `zcat` anzeigen.

Zur Lösung der Aufgabe kann frau zum Beispiel folgendermaßen herangehen: Zunächst findet sie mit `which man` heraus, dass das `man`-Kommando selbst im Verzeichnis `/usr/bin/man` liegt. Demnach werden sich auch die Manpages unterhalb von `usr` befinden – in einem Oberverzeichnis namens `man`,

```
$ find /usr -type d -name man
/usr/share/doc/libncurses5-dev/html/man
/usr/share/doc/python2.3-xml/man
/usr/share/man
/usr/X11R6/man
```

Die `find`-Option `-type d` sucht nach einem Verzeichnis. Da `man` weder etwas mit der Bibliothek `libcurses`, noch mit der Programmiersprache Python zu tun hat, bleiben nur die letzten beiden Verzeichnisse übrig (abgesehen davon, dass Manpages per Konvention nicht in `doc`-Verzeichnissen abgelegt werden)³. `/usr/X11R6` ist unwahrscheinlich, da es sich bei `man` nicht um grafisches Programm für das X-Window-System handelt.

```
$ find /usr/share/man -type f -name man*
/usr/share/man/man8/mandb.8.gz
/usr/share/man/man1/man.1.gz
/usr/share/man/man1/manpath.1.gz
/usr/share/man/ja/man1/man.1.gz
[...]
/usr/share/man/man7/man.7.gz
[...]
```

³Auf Euren Rechnern kann es gut sein, dass Ihr diese beiden Fremdverzeichnisse gar nicht findet.

sucht dann nach Dateien (`-f`), die `man` beginnen. Hier sehen wir, dass die Namen der Manpage-Dateien alle mit dem Namen des beschriebenen Programms/der beschriebenen Datei, gefolgt von einem Punkt beginnen. Dadurch lässt sich die Suche auf Manpages zu `man` einschränken:

```
$ find /usr/share/man -type f -name man.*
/usr/share/man/man1/man.1.gz
/usr/share/man/ja/man1/man.1.gz
/usr/share/man/man7/man.7.gz
[...]
```

Hieran sieht frau auch, ob nicht-englische Dokumentation installiert ist: bei `/usr/share/man/ja/man1/man.1.gz` handelt es sich um die japanische Fassung der Manpage zu `man(1)`.

6.6 Extrawurst: die Drucker

☞ Wie würdest Du den Druckjob Nr. 10 auf dem Drucker `davinci` aus der Queue entfernen?

```
lprm -Pdavinci 10
```

7 Editoren

7.1 Vi

☞ Schreibe eine etwa fünf- bis zehnzeilige Datei, in der das Wort *Informatica* vorkommt. Kopiere eine *Informatica*-Zeile ans Ende der Datei. Lösche in einer *Informatica*-Zeile alle Wörter vor und hinter *Informatica*. Suche nach *Informatica* und ersetze alle Vorkommen durch *Informatica Feminale*. Speichere die Datei. Mach die Suchen-und-Ersetzen-Aktion rückgängig. Verlass die Datei, ohne sie zu speichern.

Mögliche Arbeitsschritte zur Lösung dieser Aufgabe:

1. `vi dateiname`
2. Umschalten in den Editiermodus z. B. mit `i`
3. Schreiben des Textes
4. Umschalten in den Kommandomodus mit `Esc`
5. Zum Anfang des Textes mit `↑+g`