

Aufgabe 1 – *Manpages*

Die so genannten *manpages* bzw. *manual pages*, von denen du schon in der Vorlesung gehört hast, sind Hilfe- und Dokumentationsseiten unter UNIX, wie es sie auch unter anderen Betriebssystemen gibt. Sie helfen dir zum Beispiel, mehr über die Arbeitsweise und Optionen von Kommandos oder den Aufbau von Konfigurationendateien herauszufinden. Mit `man <Kommando>` kannst du sie in der Shell aufrufen.

Gestern hast du dir mit dem Kommando `mkdir Studium` ein Verzeichnis *Studium* in deinem home angelegt. Öffne eine Shell und schau in der manpage von `cp` nach, welche Argumente du brauchst, um dieses Verzeichnis zu kopieren.

In der letzten Übung hast du ebenfalls mit dem Kommando `ls` gearbeitet. Es listet fast alles auf, was in dem Verzeichnis liegt, in dem du dich gerade befindest. `ls` kann jedoch noch vieles mehr. So gibt `ls -l` zum Beispiel mehr Informationen über die Dateien im aktuellen Verzeichnis (*Modifikationszeit, Anzahl, Rechte usw.*) an. Finde mit Hilfe der manpage von `ls` heraus, wie du dir den Verzeichnisinhalt nach Modifikationszeit sortiert ausgeben lassen kannst!

Aufgabe 2 – *ssh und scp*

Oft genug kommt man in die Situation, dass man sich auf einem anderen Rechner einloggen muss um dort zu arbeiten. Entweder weil dieser bestimmte Eigenschaften (*bessere/r Prozessor/installierte Programme/...*) erfüllt oder weil auf diesem Dateien oder Programme liegen, zu denen man sonst keinen Zugriff hat. Das Kommando, mit dem man sich auf einem anderen Rechner anmelden kann, kennst du bereits aus der Vorlesung. Melde dich mit Hilfe von `ssh` (zur Benutzung siehe `man ssh`) auf dem Rechner `fai05` an und schau, was in `/var/tmp/vorkurs` liegt. Lege ein neues Verzeichnis mit deinem Loginnamen im Verzeichnis `/var/tmp` auf **deinem** Rechner an und kopiere (mittels `scp`) das Gefundene in dieses Verzeichnis. (*Tipp: Die Rechnernamen stehen auf den Rechnern*)
Entpacke es mit dem Befehl `tar`. Die richtigen Optionen findest du in den Beispielen der manpage.

Aufgabe 3 – *chmod*

Indem du zwei Optionen hintereinander angibst, wie zum Beispiel bei `ls -la` werden beide bei der Auswertung berücksichtigt. Führe in der Shell einmal `ls` mit den beiden oben angegebenen Parametern aus und noch einmal, jedoch nur mit dem Parameter `-l`. Was fällt dir auf? Informiere dich, was der Parameter `-a` bewirkt.

Schau dich doch mal in den *homes* der anderen Teilnehmer um. Dazu wechselst du mit dem Kommando `cd ..` in das Verzeichnis, in dem dein *home* liegt, lässt

dir anzeigen welche Verzeichnisse noch in diesem Verzeichnis liegen und wechselst wieder mit `cd` in das Verzeichnis deiner Wahl. Betrachte mit den dir mittlerweile bekannten Kommandos (*Tipp: `less`, `cat`*) die Textdateien, die in diesem Verzeichnis liegen. Fällt dir etwas auf, wenn du die Datei `.bash_history` öffnen willst? (*Tipp: Schau dir die Rechte der Dateien mal etwas genauer an*)

Kehre nun mit dem Befehl `cd` ohne weitere Argumente wieder in dein Home-Verzeichnis zurück. In dem vorhin entpackten Archiv (sollte in `/var/tmp/<deinLoginname>` liegen) findest du zwei Dateien, die du nicht betrachten kannst, da du nicht dazu 'berechtigt' bist. Versuche mit dem dir aus der Vorlesung bekannten Kommando `chmod` die Rechte dieser Datei so zu ändern, dass du das Bild anschauen und den Text bearbeiten kannst.

Aufgabe 4 – *Feedback*

Wir würden uns über Feedback zu unserem Einführungskurs freuen. Dazu gibt es unter <http://atlinux.de/eva/> ein kleines Evaluationssystem, in dem jeder anonym seinen Kommentar zur Veranstaltung abgeben kann. Nehmt euch bitte die Zeit und gebt uns Feedback.

Anregungen für Feedback: Wie hat euch die Vorlesung gefallen? Wie waren die Übungen? Welchen Eindruck haben die Übungsleiter auf euch gemacht? Könntet ihr den Ersties des nächsten Jahres den Kurs empfehlen? Und, was euch sonst noch so einfällt...

Zusatzaufgaben – *Part 2*

Da gestern einige Leute sehr schnell fertig waren, gibt es heute mehrere Zusatzaufgaben. Wir hoffen, es ist für jeden Geschmack was dabei.

Alle Zusatzaufgaben kannst du getrennt voneinander bearbeiten!

Z1 – *Befehlsalias*

Nun möchte man evtl. sehr lange Befehle etwas abkürzen. Dafür kann man sogenannte *Aliase* setzen.

Gehe in dein *Home-Verzeichnis* und öffne die Datei `.bashrc` mit einem Editor deiner Wahl. Füge nun `alias la='ls -la'` als neue Zeile hinzu. Nach einem Neustart deiner Shell musst du nur noch `la`, anstatt `ls -la` tippen um alle Dateien zu sehen. Du kannst natürlich noch viel mehr Aliase hinzufügen.

Z2 – E-Mail-Weiterleitung

Mit deinem Login hast du automatisch eine E-Mail-Adresse von der Form *si...@stud.informatik.uni-erlangen.de* erhalten.

Wenn du die E-Mail-Adresse weiterleiten willst, so musst du dir eine Datei namens *.forward* erstellen und dort deine E-Mail-Adresse angeben, zu der alle E-Mails weitergeleitet werden sollen.

Z3 – Eigene Homepage

Jeder Benutzer hat die Möglichkeit für sich selbst eine Homepage über die Uni anzulegen. Weiteres dazu:

<http://wwwcip.informatik.uni-erlangen.de/old/tree/user/richtlinien.html>

Z4 – Latex...aber nicht zum Anziehen

Das CheatSheet wurde mit der Sprache \LaTeX

<http://de.wikipedia.org/wiki/LaTeX>

erstellt. Die Quelldateien zum aktuellen CheatSheet kannst du unter

<http://atlinux.de/vorkurs/CheatSheet.tex>

herunterladen. Schau dir die Datei mit einem Editor deiner Wahl an. Was fällt dir auf?

Im normalen Text sind spezielle Befehle eingebaut, die das Aussehen der Ausgabeseite beeinflussen. Um eine Ausgabeseite zu erstellen verwende folgenden Befehl `pdflatex datei.tex`. Im aktuellen Verzeichnis sollte nun eine Datei namens `datei.pdf` vorhanden sein. Diese kannst du nun z.B. mit `xpdf` anschauen. Wenn du das CheatSheet erweitern willst, verwende folgende Syntax:

```
\cmdbox{Ueberschrift}{
    \cmd{befehl}{argumente}{beschreibung}
    \cmd{befehl}{argumente}{beschreibung}
}
```

`\cmd` kannst du beliebig oft untereinander schreiben.

Sollten wir dein Interesse an dieser mächtigen Setzsprache geweckt haben, so solltest du dich hier

<http://www.uni-giessen.de/hrz/tex/cookbook/cookbook.html>

mal einlesen.

Z5 – Scripting

Falls du die Zusatzaufgaben von gestern noch nicht durchgearbeitet hast, oder noch nicht ganz fertig bist, so tue das jetzt - die nächste Aufgabe knüpft daran an.

- Mit `chmod` kannst du das 'holeUndDrucke.sh' Script als ausführbar markieren: `chmod a+x holeUndDrucke.sh`. Jetzt lässt sich das Script auch mit `./holeUndDrucke.sh <url>` starten.
- Lass uns nun ein etwas praktischeres Script schreiben:

```
#!/bin/bash
w3m -dump \
http://www.studentenwerk.uni-erlangen.de/verpflegung/de/sp-er-sued.shtml\
| grep -C 2 $(date +%d.%m.%y)
```

- Dieses Script öffnet die URL des Studentenwerks mit einem konsolenbasierten Browser und schreibt die Texte aufs Terminal raus. Danach wird es in `grep` gepipet (d.h. die Ausgabe des Programmes 1 dient als Eingabe für das Programm 2) und dort die 2 Zeilen genommen die das aktuelle Datum haben. (genauerer siehe manpages)
- Sinnvollerweise sollte das Script als 'mensa.sh' gespeichert werden und so wie oben ausführbar gemacht werden. Nun könnt ihr euch jeden Tag mittels `./mensa.sh` den aktuellen Mensa-Speiseplan anschauen.

Z6 – Eine Aufgabe für sehr Interessierte

Zu finden unter:

<http://atlinux.de/aufgabe-screen.pdf>

Z7 – Wenn immer noch Interesse besteht...

...dann frag einen Übungsleiter doch mal nach `fg`, `bg` und `jobs`.