

# Institut für Informatik

der Ludwig-Maximilians-Universität München

## Systempraktikum – Wintersemester 2008/2009

*Prof. Dr. Heinz-Gerd Hegering, Prof. Dr. Dieter Kranzlmüller  
Thomas Schaaf*

### Einstieg in Linux (zum Selbststudium)

Diese Praxis-Einführung soll Ihnen den Einstieg in den Umgang mit der Linux-Shell erleichtern, falls Sie bisher nur wenig Erfahrung damit gesammelt haben. Führen Sie die nachfolgenden Übungen direkt an einem CIP-Rechner oder per Remote-Login von zuhause durch (siehe unten).

#### Nach dem Anmelden am CIP-Rechner:

- Sie sehen die grafische Oberfläche KDE.
- Starten Sie eine Konsole (Shell), indem Sie in der Kontrollleiste am unteren Bildschirmrand auf das entsprechende Icon klicken.
- Die Konsole zeigt Ihnen an, in welchem Verzeichnis Sie sich gerade befinden (im Normalfall ist das zu Beginn Ihr Home-Verzeichnis). Sie können jetzt Text/Befehle eingeben.

#### Erste Schritte mit der Linux-Shell:

1. Geben Sie den Befehl `ls` ein (und bestätigen Sie mit Enter). Sie sehen eine Auflistung der Dateien und Ordner in Ihrem Home-Verzeichnis. Selbst wenn das Verzeichnis leer ist, sehen Sie dort immer die Einträge `.` und `..`, bei denen es sich um Links auf das aktuelle Verzeichnis selbst und auf das Oberverzeichnis handelt.
2. Geben Sie `mkdir neu` ein, um ein Unterverzeichnis mit dem Namen `neu` zu erstellen. Wechseln Sie mit `cd neu` in dieses Verzeichnis und mit `cd ..` von dort aus wieder zurück in Ihr Home-Verzeichnis.
3. Mit dem Befehl `echo` lassen sich beliebige Zeichenketten auf der Konsole ausgeben. Testen Sie das mit `echo Text`.
4. Geben Sie nun `echo Text > test.txt` ein. Die Ausgabe der Zeichenkette erfolgt jetzt nicht mehr direkt in der Konsole, sondern wird in eine Datei umgeleitet. Es wird also im aktuellen Verzeichnis eine neue Datei mit dem Namen `test.txt` angelegt, die den Text „Text“ enthält.
5. Verwenden Sie den Befehl `ls` erneut, um sich davon zu überzeugen, dass sich die Datei `test.txt` nun tatsächlich in Ihrem Home-Verzeichnis befindet.
6. Zu nahezu jedem Befehl lassen sich spezielle zusätzliche Parameter angeben, die Einfluss auf die Wirkung des Befehls haben. Solche Parameter werden Flags genannt und werden durch ein Minus-Zeichen eingeleitet. Geben Sie folgendes ein: `ls -l`. Der Befehl `ls` wird nun also mit dem Parameter (Flag) `-l` aufgerufen. Wie Sie sehen, wird die Ausgabe wesentlich ausführlicher als zuvor.

---

<sup>0</sup>Stand: 22. September 2008

7. Welche weiteren Flags gibt es zum `ls`-Befehl? Zu jedem Shell-Befehl lässt sich eine Hilfe-Seite (Manual) anzeigen. Rufen Sie die Hilfe-Seite zum Befehl `ls` auf, indem Sie `man ls` eingeben. Mit den Pfeil-Tasten und der Bild-auf- und Bild-ab-Taste können Sie im Text scrollen. Suchen Sie die Erklärung zum `-l`-Flag. Drücken Sie die Taste `q`, um die Hilfe-Seite wieder auszublenden.
8. Rufen Sie die Hilfe-Seite zum Befehl `cat` auf und finden Sie heraus, was dieser Befehl macht. Testen Sie mit `cat test.txt`.
9. Geben Sie folgendes ein: `ls -l >> test.txt`. Anschließend rufen Sie erneut `cat test.txt` auf. Was ist passiert? Worin besteht der Unterschied zwischen `>` und `>>`? (Betrachten Sie dazu vor allem den Anfang der Datei `test.txt`.)
10. In Ausführung befindliche Programme werden durch Prozesse abgebildet, die vom Betriebssystem verwaltet werden. In der Shell können Sie sich die aktuell laufenden Prozesse zum Beispiel so anzeigen lassen: `ps -e -f -u`. Finden Sie heraus, wofür die Flags `-e`, `-f` und `-u` stehen (`man ps`).
11. Verwenden Sie den Befehl `wc` und einen geeigneten Parameter, um die Zeichen in der Datei `test.txt` zu zählen.
12. Ein sehr wichtiges Konzept (nicht nur von Linux) sind die Umgebungsvariablen. Dabei handelt es sich um Behälter für Daten, die das Betriebssystem gelegentlich benötigt. Sie können sich jede Umgebungsvariable als eine Art Spickzettel des Betriebssystems vorstellen. Geben Sie `env` ein, um sich alle Umgebungsvariablen und deren aktuelle Werte anzeigen zu lassen. Die Darstellung hat das Format `VARIABLENNAME=Wert`.
13. Lassen Sie sich nur den Wert der Umgebungsvariablen `SHELL` anzeigen, indem Sie `echo $SHELL` eingeben. Sie sehen jetzt zum Beispiel `bin/zsh` (CIP-Pool) oder `bin/bash`, was einfach dem Namen der Shell entspricht, die als Standard definiert wurde und die Sie demnach gerade nutzen. (Ähnlich wie es verschiedene Internet-Browser wie Firefox, Mozilla oder Internet Explorer gibt, gibt es auch verschiedene Shells.)

### **Remote-Login von zuhause aus:**

Sie können auch von zuhause aus auf die Rechner im CIP-Pool zugreifen, ohne Linux dafür bei sich installieren zu müssen. Verwenden Sie dazu unter Windows einfach einen SSH-Client (z.B. PuTTY). Der Rechnername für den Remote-Login lautet `remote.cip.ifi.lmu.de`. Wenn Sie zuhause ohnehin Linux verwenden, nutzen Sie einfach den Befehl `ssh`. Weitere nützliche Informationen rund um dieses Thema finden Sie auf den Seiten der Rechnerbetriebsgruppe (RBG) unter <http://www.rz.ifi.lmu.de/FAQ/>. Hier finden Sie im Abschnitt „Von zuhause/remot aus“ alle Informationen und Schritt-für-Schritt-Anleitungen für den Remote-Login am CIP-Pool.

### **Dateien von zuhause ins CIP-Home-Verzeichnis kopieren:**

Mit dem Remote-Zugriff können Sie von zuhause Shell-Befehle absetzen und in Ihrem Home-Verzeichnis arbeiten als wären Sie im CIP-Pool. Um unter Windows Dateien von Ihrer Festplatte zuhause in das entfernte Home-Verzeichnis zu kopieren oder umgekehrt Dateien aus dem Home-Verzeichnis auf Ihre lokale Festplatte zu übertragen, benötigen Sie zusätzlich einen SFTP-Client wie z.B. WinSCP (<http://www.winscp.net>). Auch zu diesem Thema geben die Webseiten der RBG Hilfestellung.