

# Eine kleine Einführung in die BASH (Bourne again Shell) Part 1

## Die Grundbefehle einer Shell

Anmerkung: Dieser Vortrag wurde auf dem 2. Anwendertreffen des Ubuntuusers Forum gehalten, der wie bereits das erste Treffen sehr gut besucht war. Ich habe noch einige Anmerkungen die während des haltens gefallen sind hinzugefügt desweiteren wurde der Vortrag nochmal komplett überarbeitet, dass er auch wirklich download tauglich ist ;-) Kursiv geschriebener Text kennzeichnet Ausgaben oder Wege auf dem Desktop. Fett geschriebener Text deutet Befehle die auf der Shell eingegeben werden an. Das ist PART 1 der Shell Einführung die folgenden Shell Vorträge - wenn sie gewollt sind - werden darauf aufbauen... :-)

*[GNOME Desktop]*

So, hier haben wir das typische Ubuntu Desktop, den Window Manager GNOME, jeder kennt ihn, und jeder benutzt ihn. Manche wissen, dass es so was wie eine „Command Line“ wie unter Windows, auch unter Linux gibt. Diese nennt man aber nicht Command Line, sondern BASH, oder Shell. BASH bedeutet Bourne again Shell, sie wurde auf Grundlage der Bourne Shell die von Stephen Bourne 1977 für Unix V7 entwickelt wurde entwickelt. Wer schnell etwas in einem „Terminal“ oder der Shell erledigen möchte, geht einfach über

*[Anwendungen => Zubehör => Terminal]*

und öffnet die Shell.

Dort kann man dann eigentlich noch nicht wirklich viel machen, denn wir sind erstmal User. Wenn man wirklich viele Dinge einzustellen hat, empfiehlt es sich, als erstes ein

**[sudo su]**

zu machen, das root Passwort einzugeben und root zu sein. (Anmerkung: Das empfiehlt sich z.B. wenn man System relevanten Konfigurationsdateien im Übermaß ändern muss; Mehrere Programme installieren muss; usw. Wer als root arbeitet sollte sich immer darüber im klaren sein, dass er - wenn er relativ unerfahren ist - sein komplettes System zerstören kann! Außerdem sollte man sich immer abmelden wenn man den Rechner verlässt oder alle notwendigen Änderungen abgeschlossen hat, das geht mit dem Hotkey STRG+D; Anmerkung: "sudo" steht für superuser do, "su" für substitute - genauere Informationen bitte googlen ;-) ) Dann stehen einem allerhand Befehle zur Verfügung. Auf die wir nun weiter eingehen wollen. Anmerkung vom Wiki Mod: Alle hier ausgeführten Befehle sind auch ohne root Rechte anzuwenden!

Da wir aber mal davon ausgehen, ein reines Textsystem ohne grafische Oberfläche zu haben, wechseln wir mit STRG+ALT + F1 auf TTY1 (Terminal zur Datenein- und -ausgabe). Ergänzung: Um von einem richtigen Terminal zur grafischen Oberfläche zurückzukommen, verwenden wir die Tastenkombination ALT + F7 (Der XServer liegt auf dem Virtuellen Terminal 7)

*[STRG + ALT + F1]*

Nun sehen wir den login Prompt. Wir können scherzhalber versuchen uns als root einzuloggen. Aber hier haben wir keine Chance „we talk ubuntu“! Wir müssen uns erstmal mit unserem normalen User einloggen... Anmerkung: Schon alleine aus Sicherheitsgründen ist es sinnvoll, das direkte Login für den User root zu sperren. (Dazu aber ein anderes Mal mehr...)

*[Login User]*

*[Passwort User]*

Danach machen wir wieder ein...

```
user@laptop:~$ sudo su
```

Jetzt sind wir root... (root ist der Administrator auf Linux Rechnern der alle Rechte hat)

Ergänzung: Die Shell ist folgendermaßen aufgebaut: "user" ist der momentan eingeloggte User; "laptop" der Rechnername; die ~ (Tilde) zeigt hier das aktuelle Verzeichnis an (~ sagt uns, dass wir uns im Home Verzeichnis befinden); das \$ Zeichen zeigt an, dass wir als normaler User eingeloggt sind, wenn wir als root arbeiten (was man aber eigentlich NICHT tut da man unter Umständen viel zerstören kann!) ist nach der Verzeichnisangabe eine # (Raute)

Jetzt wollen wir uns erstmal die Verzeichnisse auflisten lassen...

```
root@laptop:/home/user/# ls
```

Diese Ansicht ist ein bisschen chaotisch, deswegen versuchen wir's mal mit einem...

```
root@laptop:~# ls -l
```

Anmerkung: Die Option -l bedeutet -longversion d.h. wir sehen zusätzlich noch, welche Rechte gesetzt sind, wann die Datei erstellt ist, wer der Besitzer ist, usw.

Momentan befinden wir uns noch im Home Verzeichnis unseres Users, da dass sudo su uns im Home Verzeichnis des davor eingeloggten Users übernommen hat.

Wir wechseln nun ins Home Verzeichnis des Benutzers root, das machen wir,

indem wir

```
root@laptop:/home/user/# cd
```

Das cd ohne weitere Angaben bringt uns direkt ins Home Verzeichnis.

Das Home Dir erkennen wir an der ~ (Tilde). Um anzuzeigen, in welchem Verzeichnis wir uns befinden tippen wir zusätzlich noch

```
root@laptop:~# pwd
```

ein, als Ausgabe erhalten wir... Anmerkung: Dieser Befehl ist nützlich wenn man sich "verirrt" hat; in Mehreren Shells oder SSH Sitzungen gleichzeitig arbeitet und das Verzeichnis nicht angezeigt wird hinter dem Rechnernamen; oder Shellscripte schreibt.

```
/home/user
```

Nun wechseln wir mit dem Befehle cd (Change Directory) der in DOS genauso lautet in das Wurzelverzeichnis, auch root Directory genannt. (Anmerkung: root = Wurzelverzeichnis)

```
root@laptop:~# cd /
```

Nun im root angekommen machen wir ein ls -l. Hier sehen wir nun einige Verzeichnisse aus denen wir allerhand Informationen locken können, dazu wechseln wir ins Verzeichnis..

```
root@laptop:/# cd proc/
```

In diesem Verzeichnis finden wir so ziemlich alles was im Computer verbaut ist, es ist sehr nützlich um genaue Typen Bezeichnungen herauszulesen, dazu machen wir ein...

```
root@laptop:/# cat cpuinfo
```

```
processor: : 0
vendor_id : GenuineIntel
cpu family : 6
model : 8
model name : Pentium III (Coppermine)
stepping : 3
cpu MHz : 497.899
cache size : 256 KB
.
.
.
```

Hier können wir einige nützliche Informationen schnell entnehmen, z.B. wie schnell ist die CPU? Welcher Hersteller? Wieviel Cache? Welche zusätzlichen

Funktionen hat die CPU (mmx, sse...)? Klar, normalerweise weiß man, was man im Laptop hat. Aber da sich Ubuntu auch prima als Live System eignet bietet es sich folgendes an:

"Stellen wir uns folgendes Szenario vor, wir wollen uns einen neuen Laptop kaufen, der Verkäufer ist ein DAU und weiß nicht wirklich, was so drinnen ist, er weiß nur, das ist ein Dell Latitude CPx. Von RAM und CPU hat er noch nie was gehört, also booten wir von der Live CD, gehen in die Shell, rufen schnell... "

```
root@laptop:/# cat /proc/cpufinfo; cat /proc/meminfo
```

Das soll nur als kurzes Beispiel dienen, was man über die Shell alles rausfinden kann. Am besten geht man in ein Terminal, wechselt in das /proc/ Verzeichnis und cat'ed einfach mal alles das da ist. Kaputt machen kann man nicht wirklich was. Und Learning by doing sollte bei Linux Betriebssystemen am Anfang im Vordergrund stehen! Anmerkung: Mit SHIFT + BILD AUF und SHIFT + BILD AB kann man in der Shell nach oben und unten scrollen.

Anmerkung: Den freien Speicher kann man auch einfach mit "free" anzeigen lassen, da diese Übersicht aber in Blocks ausgegeben wird und das nicht wirklich gut lesbar ist, kann man "-b" für Bytes, "-k" für KiloBytes, "-m" für MegaBytes und "-g" für GigaBytes als Option anhängen.

Nun wollen wir eine Datei mal von einem Verzeichnis in ein anderes kopieren. Zuerst wechseln wir das Verzeichnis...

```
root@laptop:/# cd ~
```

Wir erinnern uns, mit cd ~ kommen wir wieder in unser Home Verzeichnis, wir legen mit touch eine neue Datei an, da wir noch einen Zielordner brauchen wo wir die Datei hinkopieren können, legen wir diesen mit dem Befehl mkdir (Make Directory) an. Die Datei die wir mit touch angelegt haben, hat zwar keinen Inhalt, das stört uns aber weiter nicht...

```
root@laptop:~$ mkdir ordner
```

```
root@laptop:~$ touch datei.dat
```

```
root@laptop:~# cp datei.dat ordner/
```

So, nun vergewissern wir uns, dass die Datei auch wirklich im Ordner ist, dazu führen wir folgenden Befehl aus...

```
root@laptop:~# ls -l ordner/
```

Wenn wir die Datei sehen, löschen wir die Datei im Home Dir mit dem Befehl "rm" (Remove)...

```
root@laptop:~# rm datei.dat
```

Dann wechseln wir wieder in das gerade angelegt Verzeichnis...

```
root@laptop:~# cd ordner
```

Danach, wollen wir die Datei umbenennen, da es eine Log Datei ist, die mit der Endung .dat nicht erkenntlich ist. Dazu nutzen wir den Befehl "mv" (move)! Unter Linux gibt es keinen Befehl, der explizit dafür gedacht ist, Dateien umzubennen wie z.b. unter Windows, dort lautet der Befehl "ren"

```
root@laptop:~/ordner# mv datei.dat datei.log
```

Nun wollen wir den Ordner löschen, dazu nutzen wir den Befehl "rm" (remove) den wir vorher schon kennengelernt haben...

```
root@laptop:~/ordner# cd ..
```

```
root@laptop:~/# rm ordner/
```

Das geht aber so nicht, wir erhalten die Fehlermeldung "rm: Entfernen von "ordner" nicht möglich: Is a directory" d.h. im Ordner befindet sich noch etwas, nämlich die Datei die wir gerade angelegt haben, um den Ordner samt Inhalt löschen zu können, müssen wir Rekursiv löschen mit der Option -R... Anmerkung zu "cd ..": Der "." wechselt in das übergeordnete Verzeichnis. Anmerkung zu "rm -R": Vorsicht mit diesem Befehl walten lassen! Lieber dreimal überprüfen, ob die Dateien im zu löschenden Verzeichnis wirklich zu löschen sind, rm -R löscht alles OHNE nachfrage! Ich übernehme keinerlei Haftung für entstehende verlorene Daten und/oder komplett leergefegte Festplatten usw.... ;-) Sagt nicht ich hätte euch nicht ausdrücklich gewarnt!

```
root@laptop:~# rm -R ordner/
```

Nun ist der Order inklusive Inhalt gelöscht....

Falls wir uns anzeigen wollen, wieviele Speicherplatz frei und belegt ist können wir das mit folgendem Befehl ganz einfach machen df -h (df bedeutet diskfree). Dazu geben wir in der Bash einfach:

```
root@laptop:~# df -h
```

ein... Der Schalter "h" gibt das ganze in "human readable" Format aus. Ohne die Option "h" bekommen wir die Ausgabe in Blocks was nicht wirklich gut lesbar ist... Um von einem Verzeichnis die Größe anzeigen zu lassen, gehen wir in das gewünschte Verzeichnis und tippen dort...

```
root@laptop:~# cd /etc/  
root@laptop:/etc/# du -h
```

Ein und bekommen als Ausgabe die Größe des kompletten Verzeichnisses berechnet. Die Option "h" steht in diesem Fall wieder für "human readable"

Format, also passend umgerechnet in KB, MB, GB, TB ... Diese Ausgabe ist aber bei Ordner mit viele Unterordnern unübersichtlich, da wir im Kopf ausrechnen müssten, wieviel Speicherplatz jetzt insgesamt verwendet wird (Tipp am Rande, in der letzten Zeile steht das Ergebnis ;-) ) Um nur die komplette Größe einzeln anzeigen zu lassen, geben wir:

```
root@laptop:/etc/# du -hs
```

Vortrag by Markus (Nasenbär) FISI & Renatus (HellDragon)  
FISI in Ausbildung  
[www.together-technologies.de](http://www.together-technologies.de) | [mirror.together-technologies.de](http://mirror.together-technologies.de)  
18.02.2007 - 04.03.2007