



Computer & GNU/Linux Einführung Teil 2

Mario Löscher, Clemens Weißbacher
Basierend auf Folien von
Simon Haller

Befehle von Montag

<code>cd</code>	Verzeichnis wechseln
<code>man</code>	man page anzeigen
<code>pwd</code>	aktuelles Verzeichnis
<code>mkdir</code>	Verzeichnis anlegen
<code>rmdir</code>	leeres Verzeichnis löschen
<code>rm</code>	Dateien oder Verzeichnisse entfernen
<code>ls</code>	Verzeichnis Inhalt auflisten
<code>chmod</code>	Berechtigungen ändern
<code>nano</code> oder <code>vi/vim</code>	Kommandozeilen Editor
<code>sudo</code>	Programm mit anderem User ausführen
<code>apt-get update</code>	Paketliste aktualisieren
<code>apt-get install tldr</code>	das Programm tldr installieren
<code>apt-get upgrade</code>	System updaten

Tipp

Verzeichnisstruktur

Nur als Tipp. Nicht alles in Downloads oder im Temp Ordner liegen lassen.

```
studium/  
+-- 2020w  
  |-- 703000  
  |   +-- AlgProg  
  |   +-- Computer  
  +-- arch  
  |   +-- ps  
  |   +-- vo  
  +-- funk  
  |   +-- ps  
  |   +-- vo  
  +-- la  
  |   +-- ps  
  |   +-- vo  
  +-- prog  
  |   +-- ps  
  |   +-- vo  
  +-- theo  
      +-- sl  
      +-- vo
```

Wichtige Kommandos

echo (*Anzeigen einer Textzeile*)

Optionen:

- n keine neue Zeile
- e aktiviert \ Optionen:
 - \ n neue Zeile
 - \ t horizontaler Tabulator
 - \ v vertikaler Tabulator

Hello World

Hello World in einer Shell kann folgendermaßen realisiert werden:

```
$ echo Hello World
```

Rechtevergabe I

Rechte

- Userrechte
- Gruppenrechte
- Rechte für alle (world, other)

Rechtevergabe II

Anzeigen von Rechten `ls -l Dateiname`

```
- rw-  r--  r--  1  c703101  c703  4387  Jan 08 13:38  Dateiname
- ---  ---  ---  |  -----  -----  -----  -----  -----
|  |    |  |  |  |    |    |          |          Dateiname
|  |    |  |  |  |    |    |          |          Änderungsdatum
|  |    |  |  |  |    |    |    Größe
|  |    |  |  |  |    |    |    Gruppe
|  |    |  |  |  |    |    |    Eigentümer
|  |    |  |  |  |    |    |    Hardlinks
|  |    |  |  |  |    |    |    Rechte aller anderen
|  |    |  |  |  |    |    |    Rechte der Gruppe
|  |    |  |  |  |    |    |    Rechte des Eigentümers (r.. Read, w.. Write, x.. eXecute)
Dateiart (-.. Datei, d.. Directory, l.. Link)
```

Rechtevergabe III

Setzen von Rechte mit `chmod`

```
$ chmod ugo =  $\pm$ rwX [DATEI]  
    u...User, g...Group und o...Other
```

Alternativ: `$ chmod [RECHTE] [DATEI]`

Beispiel:

```
$ chmod 650 [DATEI]  
    Octal jeweils rwx
```

Setzen des Eigentümers mit `chown`

```
$ chown USER.GROUP [DATEI]
```

Skript mit Shebang

Die erste Zeile eines Skriptes sollte den Interpreter definieren.
Zum Beispiel:

```
#!/usr/bin/env bash
#!/bin/bash
#!/bin/sh
#!/usr/bin/env python
#!/usr/bin/python3
```


Skript mit Shebang

Die erste Zeile eines Skriptes sollte den Interpreter definieren.
Zum Beispiel:

```
#!/usr/bin/env bash
#!/bin/bash
#!/bin/sh
#!/usr/bin/env python
#!/usr/bin/python3
```

Hello USER

```
#!/bin/bash
echo "Hello World, how are you $USER"
```

Aufgabe

- Legt ein Shell-Script an, das ein Verzeichnis angelegt und sich selbst in dieses Verzeichnis verschiebt.
- Führt das Script mindestens fünfmal aus.
- Kopiert das Script in euer Home Verzeichnis. (neuer Befehl: cp)

Aufgabe

- Legt ein Shell-Script an, das ein Verzeichnis angelegt und sich selbst in dieses Verzeichnis verschiebt.
- Führt das Script mindestens fünfmal aus.
- Kopiert das Script in euer Home Verzeichnis. (neuer Befehl: cp)
- Löscht den angelegten Verzeichnisbaum mit nur einem Befehl.

Zeichensätze

Die Umgebungsvariablen LC_ALL und LANG

Der Zeichensatz kann mit

LANG=C7-Bit	ASCII d.h. keine Umlaute, kein ß
LANG=en_US8-Bit	ISO Latin 1 (ANSI) Englisch
LANG=de_DE	oder de_AT Deutsch
LANG=de_AT.UTF8	Unicode UTF-8

festgelegt werden.

Designunterschied UNIX/Linux zu Windows

Im Unterschied zu Windows ist die Kodierung nicht Eigenschaft eines Objektes (Datei, Dateisystem etc.). In Windows ist der Zeichensatz durch 2-4 Bytes im Header der Datei kodiert.

Variablen I

Wichtige Umgebungsvariablen

PATH	Legt den Suchpfad für ausführbare Dateien fest
HOME	Homeverzeichnis des Users
USER	Aktuell angemeldeter User
DISPLAY	Legt die IP-Adresse für den X-Server fest
SHELL	Legt die Standard-Shell fest
EDITOR	Legt den Standard-Editor fest
RANDOM	Zufallszahl (Bash interne Variable)

Setzen und ausgeben von Umgebungsvariablen

Umgebungsvariablen werden mit dem Befehl

```
$ export UMGEBUNGSVARIABLE=Wert #je nach Shell auch mit  
$ setenv UMGEBUNGSVARIABLE=Wert
```

gesetzt. Inhalte einer Variable ausgeben:

```
$ echo $UMGEBUNGSVARIABLE
```

Variablen II

Variablen löschen

der Inhalt kann mit

```
$ set Variable
```

```
$ unset Variable1 Variable2
```

gelöscht werden.

```
$ env
```

listet alle gesetzten Variablen.

Anstatt der Umgebungsvariable `$HOME` kann das `~` Zeichen in Kommandos verwendet werden.

Aufgabe

Shebang

- Legt eine Datei an, die euren Usernamen ausgibt.
- Tut so also, ob ihr nicht mitbekommen habt, dass das Shebang am Anfang stehen muss und auch nicht, dass die Datei ausführbar sein soll.
- Gebt *bash Dateiname* im Terminal ein. Was passiert?

Welche Befehle werden wirklich verwendet

Welcher Befehl wird ausgeführt

Da die PATH Umgebungsvariable u.U. ziemlich lang und unübersichtlich wird ist es oft praktisch zu wissen welcher Befehl wirklich ausgeführt wird.

Mit dem Befehl:

```
$ type Befehl
```

kann der verwendete Befehl ausgegeben werden.