



Bash

Übung 1

Mario Bischof



Aufgabe 1 - Repetition: Navigieren in Verzeichnissen

1. Wechseln sie mit `cd` in Ihr Heimverzeichnis
2. Wechseln sie ins Verzeichnis `/var/log` mit einer absoluten Pfadangabe
3. Wechseln sie ins Verzeichnis `/etc/udev` mit einer absoluten Pfadangabe
4. Wechseln sie ins Verzeichnis `/etc/` mit einer relativen Pfadangabe
5. Wechseln sie ins Verzeichnis `/etc/network/` mit einer relativen Pfadangabe
6. Wechseln sie ins Verzeichnis `/dev/` mit einer relativen Pfadangabe

Aufgabe 2 - stdout, stdin, stderr:

- a) Die Ausführung von `ls -z` erzeugt einen Fehler (da es die Option `-z` nicht gibt). Starten sie `ls` mit `-z` und leiten sie die Fehler in eine Datei `/root/errorsLs.log`.
- b) Erzeugen sie eine kl. Textdatei und füllen sie diese mit Inhalt. Geben sie die Textdatei mit `cat` aus und leiten sie die Ausgabe wieder in eine neue Datei um. Benutzen sie einmal `>` und einmal `>>` (mehrmals hintereinander). Untersuchen sie die beiden Situationen, indem sie jedesmal den Inhalt der Datei wieder ausgeben. Was passiert wenn sie in dieselbe Datei umleiten wollen?
- c) Leiten sie die Ausgabe von `whoami` in die Datei `info.txt` um
- d) Hängen sie die Ausgabe von `id` an die Datei `info.txt` an
- e) Leiten sie die Datei `info.txt` als Eingabe an das Programm `wc` um und zählen sie damit die Wörter (`-w`)

Aufgabe 3 - grep, cut:

- a) Erzeugen sie eine Textdatei mit folgendem Inhalt:

```
alpha1:1alpha1:alp1ha  
beta2:2beta:be2ta  
gamma3:3gamma:gam3ma  
obelix:belixo:xobeli  
asterix:sterixa:xasteri  
idefix:defixi:ixidef
```

Benutzen sie zur Erzeugung `<<` indem sie Zeile für Zeile an `cat` übergeben, die Ausgabe wird in eine Datei umgeleitet. Benutzen sie das Schlusswort `END`. Durchsuchen sie die Datei mit `grep` nach folgenden Mustern (benutzen sie die Option `--color=auto`):

- Alle Zeilen, welche `obelix` enthalten
- Alle Zeilen, welche `2` enthalten
- Alle Zeilen, welche ein `e` enthalten
- Alle Zeilen, welche **nicht** `gamma` enthalten
- Alle Zeilen, welche `1, 2` oder `3` enthalten (benutzen sie `-E` und eine regex)



b) Gehen sie von derselben Datei aus wie in Aufgabe a). Benutzen sie `cut` und formulieren sie damit einen Befehl um nur folgende Begriffe anzuzeigen:

- Alle Begriffe vor dem ersten `:-` Zeichen
- Alle Begriffe zwischen den beiden `:-` Zeichen
- Alle Begriffe rechts des letzten `:-` Zeichen

Aufgabe 4 - Wildcards:

Lösen sie folgende Aufgaben der Reihe nach (Verwenden sie soviele Wildcards und/oder Braces wie nur irgendwie möglich!):

1. Erzeugen sie ein Verzeichnis `Docs` in ihrem Heimverzeichnis
 2. Erstellen sie die Dateien `file1` bis `file10` mit `touch` im `Docs` Verzeichnis
 3. Löschen sie alle Dateien, welche einer `1` im Dateinamen haben
 4. Löschen sie `file2`, `file4`, `file7` mit einem Befehl
 5. Löschen sie alle restlichen Dateien auf einmal
-
1. Erzeugen sie ein Verzeichnis `Files` in ihrem Heimverzeichnis
 2. Erstellen sie die Dateien `file1` bis `file10` mit `touch` im `Files` Verzeichnis
 3. Kopieren sie das Verzeichnis `Files` mitsamt Inhalt nach `Files2`
 4. Kopieren sie das Verzeichnis `Files` mitsamt Inhalt nach `Files2/Files3`
 5. Benennen sie das Verzeichnis `Files` in `Files1` um
 6. Löschen sie alle erstellten Verzeichnisse und Dateien wieder

Aufgabe 5 - Tilde expansions:

- Führen sie jede der auf der letzten Seite in der Präsentation aufgeführte Erweiterungen der Tilde einmal an Ihrem System aus und stellen sie sicher, dass sie deren Funktionsweisen verstanden haben.

Aufgabe 6 - Für Fortgeschrittene:

- Was macht folgender Ausdruck?

```
dmesg | egrep '[0-9]{4}:[0-9]{2}:[0-9a-f]{2}].[0-9]'
```

- Was macht folgender Ausdruck?

```
grep -oE \  
'((1?[0-9][0-9]?|2[0-4][0-9]|25[0-5])\.)\{3\}(1?[0-9][0-9]?|2[0-4][0-9]|25[0-5])'
```

- Was macht folgender Bash-Befehl?

```
find / -user root -iname "*bash*" -exec cp {} /root/ \;
```