

Vorlesung Linux-Praktikum

4. Aliase und Shellskripte

Dirk Frettlöh

Folien: Carsten Gnörlich, DF

Technische Fakultät
Universität Bielefeld

Willkommen zur fünften Vorlesung

Was gab es beim letzten Mal?

- ▶ Prozesse
- ▶ Dateiberechtigungen

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Willkommen zur fünften Vorlesung

Was machen wir heute?

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Dateien finden

Dateiverwaltung

Dateien suchen

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

find (find files and directories)

```
$ find . -name zwei.txt  
./verzeichnis/zwei.txt  
./zwei.txt
```

Ohne `-name` gibt find *alle* Dateien dort aus.

Beachte:

- ▶ `$ grep xxx` sucht `xxx` *in* Dateien
- ▶ `$ find ordner -name xxx` sucht Dateien, die `xxx` *heißen*

Also:

- ▶ `$ find . | grep xxx` ist gleich
- ▶ `$ find . -name *xxx*`

Dateiverwaltung

Dateien suchen

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Wildcards gehen auch:

```
$ find . -name *.jpg  
./Fotos/img_38.jpg  
./Fotos/img_39.jpg
```

Aber: Wenn es im aktuellen Verzeichnis bild.jpg gibt: Dann ersetzt die shell *.jpg durch bild.jpg und setzt diesen Namen hier ein. Besser:

```
find . -name "*.jpg"
```

Das mit den " ist ein wichtiges Prinzip: **Quoting**

Exkurs: Funktion der Kommandozeile verstehen

echo zum Untersuchen von Aufrufen

Wdh.: echo gibt seine Parameter aus

```
$ echo eins zwei drei  
eins zwei drei
```

Nützlicher Seiteneffekt:

- ▶ echo zeigt, wie Befehle ihre Argumente "sehen":

```
$ echo find . -name *.jpg  
find . -name bild.jpg
```

```
$ echo find . -name "*.jpg"  
find . -name *.jpg
```

Reihenfolge: die Shell ersetzt *zuerst* wildcards (falls möglich), gibt *dann* Parameter an Befehl weiter.

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Aliase

Aliase

Motivation

alias: Ein Programm unter einem anderen Namen verwenden

Motivation: Immer `ls -l` tippen ist lästig

```
$ alias ll="ls -l"
```

```
$ ll
```

- ▶ `ls -l` wird als neuer Befehl mit dem Namen `ll` definiert
- ▶ in Zukunft kann man einfach `ll` tippen

Aliase

Genauere Betrachtung

```
$ alias s2="sort -k 2 -n"
```



```
$ s2 planeten.txt
```

- ▶ Shell ersetzt linke Seite der Gleichung (`s2`) durch rechte Seite (`sort -k 2 -n`)
- ▶ alias kann weitere Aufrufwerte haben (`planeten.txt`)

Shellskripte

Exkurs: verschiedene Shells

Es gibt verschiedene Kommandozeileninterprete, also shells:

- ▶ bash: *Bourne-again shell* heute DIE shell, auf fast allen Linux- und MacOS-Systemen. Von GNU, seit 1989
- ▶ sh: *Thompson shell* (1971), *Bourne shell* (1975)
- ▶ ksh: *Korn shell*, pdksh: public domain ksh
- ▶ ash: *Almquist shell*, dash: Debian ash
- ▶ zsh: *Z shell*, csh: *C shell*,... moderne Erweiterungen

Anzeigen mit `$ echo $SHELL`

Shellskripte

Exkurs: verschiedene Shells

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

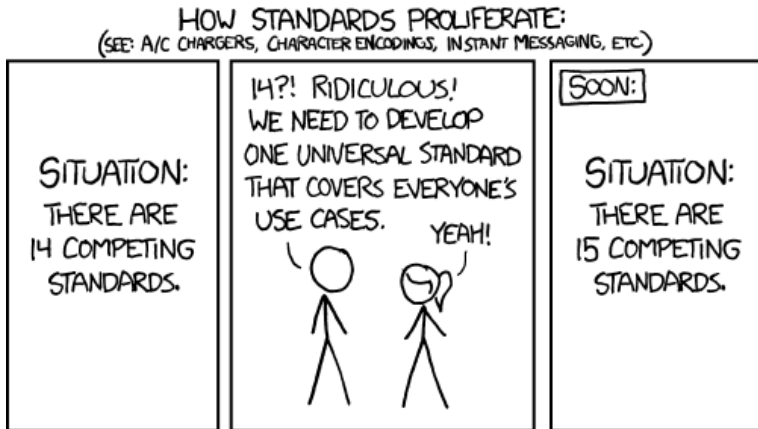
Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick



...bei shells zum Glück gerade nicht so: bash!

Aliase

Die .bashrc

.bashrc: “bash resources” - Konfigurationsdatei

- ▶ wird beim Starten der Kommandozeile ausgeführt
- ▶ was ihr in die .bashrc hineinschreibt, gilt, als hättet ihr es direkt eingegeben

Eine .bashrc: (bzw auf Mac: .bash_profile)

```
PATH="$PATH:$HOME/meins/ps"  
alias ll='ls -l'  
alias evicne='evince'  
alias pine='alpine'
```

Vorsicht: Falsche .bashrc → Account klappt nicht mehr!

Aufruf der Konfigurationsdateien

Noch bevor man in einer bash etwas eingeben darf, werden verschiedene Konfigurationsdateien gestartet.

Ein Szenario (bei Mac)

- ▶ `/etc/profile` (globale bash-Konfig.)
- ▶ `/etc/bashrc` (globale bash-Konfig.)
- ▶ Falls vorhanden: `~/.bash_profile`
- ▶ Sonst `~/.bash_login`
- ▶ Sonst `~/.profile`

Das genaue Vorgehen ist bei verschiedenen Unixen immer leicht unterschiedlich.

Es kann auch sein, dass `~/.bash_profile` auch noch die `~/.bashrc` aufruft.

Umgebungsvariablen

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Später sehen wir Variablen (z.B. `i=1`)

Falls `i` eine Variable ist, liefert `echo i` einfach `i`.

```
$ echo $i
```

 liefert den Inhalt von `i`.

Es gibt in der `bash` ganz spezielle Variablen, die **Umgebungsvariablen**. Z.B.

```
$ echo $SHELL
```

```
bash
```

```
$ echo $USER
```

```
frettlöe
```

```
$ echo $PWD
```

```
/homes/frettlöe    (vgl pwd!)
```

Umgebungsvariablen

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

```
$ echo $LANG
```

```
de_DE.UTF-8
```

```
$ echo $PS1
```

```
\u@\h:\w$
```

PS1 regelt die Gestalt der Eingabeaufforderung, hier:

```
frettløe@hopf57:~$      (user, host, working directory)
```

...und viele mehr.

Alle Anzeigen mit **env**, bzw besser

```
~ $ env | more
```

```
TERM_PROGRAM=Apple_Terminal
```

```
SHELL=/bin/bash
```

```
TERM=xterm-256color
```

```
TERM_PROGRAM_VERSION=421.2
```

```
USER=frettløe
```

```
....
```


Umgebungsvariablen

Suchpfade

which Welchen Pfad hat Programm...

Programme wie `ls`, `grep`, `find`... liegen irgendwo als Dateien auf dem Rechner. Anzeige wo mittels:

```
$ which find  
/usr/bin/find
```

```
$ which ls  
/bin/ls
```

`PATH` sagt Linux, wo es `ls` suchen soll.

```
$ echo $PATH  
/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/homes/f
```

Aliase

Grenzen von Aliasen

- ▶ nur für einzelne Befehle/Pipes praktikabel
- ▶ nur ein variabler Eingabewert/-datei pro Alias

Gegenbeispiel:

```
head -2 planeten2.txt; tail -n +3 planeten2.txt | sort
```

- ▶ Eingabewert/-datei muss am Ende stehen
- ▶ Programmiermöglichkeiten sehr eingeschränkt
(Nutzung von Variablen, Fallunterscheidungen, Schleifen)

Shellskripte

Shellskripte

Wdh: Aufgaben der Kommandozeile

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

1. Programm ausführen
2. Programme zu mächtigeren Werkzeugen kombinieren (siehe Pipes!)
3. Kommandozeilen-Skripte
 - ▶ (1) und (2) abstrahieren und in Datei speichern
 - ▶ wiederverwenden statt erneut eintippen

Shellskripte

Aufbau

Prinzipieller Aufbau eines Shellskriptes

- ▶ Textdatei mit folgendem Inhalt:

```
#!/bin/bash
```

Shell zum Ausführen des Skriptes

```
echo Hallo  
echo ich bin ein  
echo Shellskript
```

Aufrufe, wie Ihr sie auch
direkt eintippen würdet

Shellskripte

Aufrufen

Ausführbarkeits-Bit setzen; aufrufen (mit ./):

```
$ chmod u+x skript.sh
```

```
$ ./skript.sh
```

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Shellskripte

Quellen für Programme und Skripte

Shellskripte verhalten sich wie “echte” Programme

- ▶ warum dann `$./skript.sh`
und nicht einfach `$ skript.sh`?

Die Shell hat folgende “Quellen” für Programme:

- ▶ direkt angegebene Pfade (z.B. `/bin/`, oder `./`)
- ▶ voreingestellte “Suchpfade” in `$PATH`

Shellskripte

Suchpfade erweitern

Vorsicht: vermurkster Suchpfad → alle Programme “weg”

(Programme sind noch da, aber die Shell findet sie nicht mehr)

```
> PATH=$PATH:~/shell-skripte
```

← nicht vergessen (beliebte Falle ;-)

Erste Hilfe: absolute Pfade benutzen, z.B.

```
$ /bin/ls
```

```
$ /usr/bin/emacs ~/.bash_aliases
```

```
$ which ls
```

```
/bin/ls
```


Shellskripte

Keine Sicherheitslücken aufmachen!

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Bitte **nicht** nachmachen:

~~den Punkt `.` in den Suchpfad aufnehmen, also
`PATH=.:$PATH` oder `PATH=$PATH:.`~~

Im Verzeichnis `/tmp` gebe es folgendes Skript:

```
#!/bin/bash
```

```
rm -rf ~/* # löscht das Benutzerverzeichnis
```

und zwar mit dem Namen "`ls`".

Würdet Ihr dort `./ls` aufrufen? Nein?

Dann nehmt auch `.` nicht in Euren Suchpfad auf!

Shellskripte

Parameterübergabe

Beispiel zur Übergabe von Parametern:

```
#!/bin/bash
```

```
echo "Erstes : $1"
```

```
echo "Zweites: $2"
```

```
echo "Drittes: $3"
```

```
echo "Anzahl : $#"
```

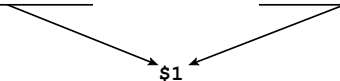
```
echo "Alle   : $*"
```

Shellskripte

Beispiel

Zur Erinnerung an Vorlesung 3:

```
> head -2 planeten2.txt; tail -n +3 planeten2.txt | sort
```



\$1

Abstrahieren und zusammenfassen:

```
#!/bin/bash
```

```
head -2 $1 ; tail -n +3 $1 | sort
```

```
$ ./hsort.sh planeten2.txt
```

Shellskripte

zu sortierende Spalte mit angeben

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

```
#!/bin/bash
```

```
head -2 $1 ; tail -n +3 $1 | sort -k $2 -n
```

```
$ ./hsort2.sh planeten2.txt 2
```

Shellskripte

Nützliche Eigenschaften von echo

echo **-n**: unterdrückt Zeilenvorschub

- ▶ Ausgabezeile mit mehreren echo-Befehlen erzeugen (nur sinnvoll innerhalb von Skripten)
-

```
#!/bin/bash
```

```
echo -n Mehrere Echo-Befehle
```

```
echo -n bauen eine
```

```
echo Zeile auf
```

```
$ ./test.sh
```

```
Mehrere Echo-Befehle bauen eine Zeile auf
```

Shellskripte

echo gibt Variablen aus

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

```
$ echo $PATH  
/usr/local/bin:/usr/bin:/bin
```

```
$ echo "Mein login ist $USER"  
Mein login ist frettloe
```

Oft ist es egal, ob man mit '...' quotet oder mit "...".

Hier ist es wichtig! Nur "... " zeigt den Inhalt der Variable.

Shellskripte

echo als Anfang einer Pipe

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

```
$ echo "Linux" | wc -m
```

```
6
```

```
$ echo -n "Linux" | wc -m
```

```
5
```

(wc -m: Zählt die Buchstaben)

Shellskripte

Parameter in Eingaben umwandeln

In Shellskripten:

```
#!/bin/bash
```

```
wc -m $1          # Fall A  
echo $1 | wc -m   # Fall B
```

```
$ ./skript.sh hallo
```

Was ist der Unterschied zwischen Fall A und Fall B?

- ▶ Fall A: wc zählt Zeichen in der Datei "hallo"
- ▶ Fall B: wc zählt Zeichen im ersten Parameter (hier "hallo")

Shellskripte

Ergebnisse von Programmaufrufen ausgeben

`$(...Aufruf ...)`: liefert Ausgabe des Aufrufs

Beispiel (date gibt Datum aus):

```
$ date "+%d. %B %Y"  
11. November 2020
```

```
$ echo "Log vom $(date "+%d. %B %Y") für $USER:"  
Log vom 11. November 2020 für frettloe:
```

Es geht beliebig komplex (mit Pipes):

```
$ echo "Die Sonne hat $(tail -n +3 planeten2.txt | wc -l)  
Planeten."  
Die Sonne hat 8 Planeten.
```

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Shellskripte

grep: Suchtext am Zeilenanfang/-ende verankern

"`^text`" ⇒ *text* muss am Zeilenanfang stehen

"`text$`" ⇒ *text* muss am Zeilenende stehen

```
$ grep Text text.txt
```

Der Text steht in der Mitte

Text muss am Anfang stehen

Am Ende steht der Text

```
$ grep "^Text" text.txt
```

Text muss am Anfang stehen

```
$ grep "Text$" text.txt
```

Am Ende steht der Text

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Überblick

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

| | |
|----------------------|-----------------------------------|
| <code>find</code> | Finden von Dateien |
| <code>echo</code> | echot Text |
| <code>alias</code> | Alias setzen |
| <code>.bashrc</code> | Konfiguration der bash |
| <code>which</code> | Anzeige des Pfads eines Programms |

| | |
|------------------------|------------------------|
| Umgebungsvariablen | (PATH, SHELL, USER...) |
| Skripte | skript.sh |
| <code>\$(...)</code> | Ausgabe des Befehls |

Ausblick

Nächste Woche machen wir...

- ▶ Versionskontrollsystem git
 - ▶ Funktionsweise
 - ▶ Lokal benutzen

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Ende der heutigen Vorlesung

Unix-
Praktikum

Dirk Frettlöh

Dateiverwaltung

Finden von Dateien

Aliase

speichern

\$PATH

Shellskripte

Parameter

echo

grep (mehr dazu)

Ausblick

Vielen Dank fürs Zusehen!

Bis nächste Woche!