

Unix/Linux-Einführung

Dr. Walter Ebner
Dr. Albert Weichselbraun

Wirtschaftsuniversität Wien

Betriebssystem: Unix

Kennzeichen:

- Multi-User
- Multi-Tasking
- Leichte Portierbarkeit
- von SW-Entwicklern für SW-Entwickler geschrieben
- **case-sensitive** (Unterscheidung zwischen Klein- und Großbuchstaben)

Unix - Varianten

- IBM: AIX
- SUN: SUN-OS
- HP: HP-UX
- OSS: GNU/Linux
*BSD

Login Powernet-Rechner

An den Powernet Rechnern (Unix-Server) mittels telnet anmelden :

- In Windows: Start-Button → Ausführen → „telnet **gutemine.wu-wien.ac.at**“ eingeben.
- Login mit h und Matrikelnummer
z. B.: h0551756
- Passwort eingeben (Bei der Eingabe sieht man keine Sternchen)

Abmelden: **exit** eingeben.

Achtung: statt Telnet sollte man aus Sicherheitsgründen immer ssh verwenden.

ssh-Login am balrog


Am balrog (Linux-Server) ist telnet nicht erlaubt, da es ein unsicheres Protokoll ist. Stattdessen verwendet man ssh.

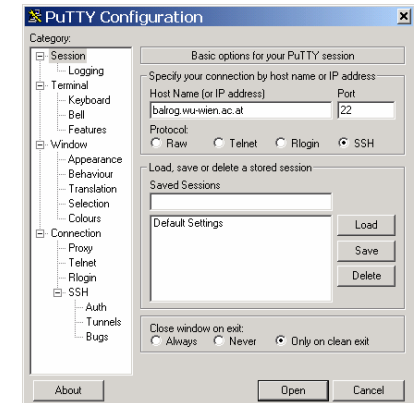
- In Linux/Unix: Terminal öffnen und diese Zeile eingeben (Das login wird mit -l übergeben):
ssh -l j9551756 balrog.wu-wien.ac.at oder
ssh j9551756@balrog.wu-wien.ac.at
- Passwort eingeben (Passwort wird nicht maskiert)

Abmelden: **exit** eingeben.

ssh-Login am balrog

- Unter Windows ist standardmäßig kein ssh-Client installiert.
PuTTY ist ein freier ssh-Client für Windows. Download unter:
<http://www.putty.nl/download.html>
- Vorgehensweise:

 en und bei Host
`putty.exe` balrog.wu-wien.ac.at
eingeben → Open anklicken →
Beim ersten Mal Fingerprint
bestätigen → Login und PW
eingeben.



Command-Line/Shell

- Nicht graphisches Benutzerinterface
- Befehlszeileninterpreter
- Syntax:

ls **-l** **file.txt ...**
└─┬ └─┬ └──────────┬──┘
Befehlsname Flag Argumente

Help

- man-Pages
- Syntax:

man ls
└─┬

Gesuchter Hilfsbegriff

Manual-Pages werden im less-Pager angezeigt.

Befehl - less

- Gibt eine Datei oder den Standardinput
Seitenweise aus.
- Syntax: **less file.txt**
- Befehle:
 - Blättern: Space (Leertaste), Pfeiltasten gehen nicht immer
 - Beenden: q
 - Suchen: / dann Suchbegriff eingeben
 - Mit n kann der letzte Suchbegriff erneut gesucht werden

Internet

9

Commandozeile

- Wichtige Shortcuts:
 - Tabulator: autocompletion
 - In der History blättern: "Cursor up/down"
 - Strg + „r“: in der History suchen
 - Strg + „a“ / „e“: Zeilenanfang/-ende
 - Strg + „d“: **Delete Char**
 - Strg/Alt + „f“: **Forward Char/Word**
 - Strg/Alt + „b“: **Backward Char/Word**
- weitere Informationen: man readline

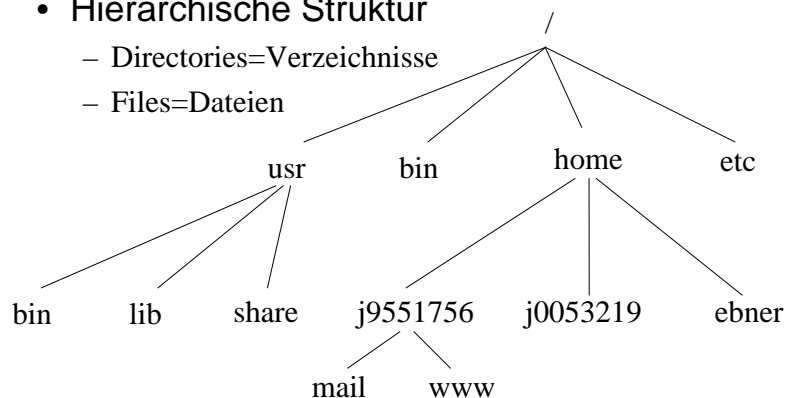
Internet

10

Dateisystem

• Hierarchische Struktur

- Directories=Verzeichnisse
- Files=Dateien



Internet

11

Datei-System: Pfad

- Pfad = Weg durchs Dateisystem zu einer Datei.
- Pfad und Dateiname identifizieren Datei eindeutig.
- Trennzeichen zwischen einzelnen Verzeichnissen und Dateien: /
Beispiel: /home/j9551756/datei.txt
- Absolute Pfadangaben beginnen immer mit dem /
- Relative Pfadangaben werden relativ zum aktuellen Verzeichnis angegeben.

Internet

12

Homedirectory

- Normalerweise ist das Homedir in **/home** zu finden. z.B **/home/albert** oder **/home/j0525050**
- Am balrog ist das Homedir folgendermaßen strukturiert: **/ai/balrog/home/j05/j0525050**
- Mit **echo \$HOME** oder **echo ~** wird dieser Pfad ausgegeben.

Navigation im File-System

- Bei textbasiertem Interface kann man nur ein Directory überschauen.
- Befehle zur Navigation:
 - **ls** zeigt den Inhalt des aktuellen Verzeichnisses (**list**)
 - **pwd** zeigt den absoluten Pfad zum aktuellen Verzeichnis (**print working directory**)
 - **cd *dirname*** wechselt in das entsprechende Verzeichnis (**change directory**)
 - **cd ..** wechselt in das darüberliegende Verzeichnis**Achtung:** zwischen **cd** und **..** muss eine Leerstelle sein.

Tips + Tricks

- **cd** ohne Argumente führt zurück in das *Homedirectory*.
- **~** steht für das *Homedirectory*.
- **.** ist ein Symbol für das aktuelle Verzeichnis.
- **..** steht für das übergeordnete Verzeichnis.

Tips + Tricks

- ***** ersetzt irgendeine Zeichenkette.
z.B.: **ls w*** listet alle Files auf, die mit w beginnen.
- **?** ersetzt genau ein Zeichen
z.B.: **ls dokument?.pdf**
- **Groß/Kleinschreibung beachten.**

File-System verändern

- Verzeichnisse anlegen (**make directory**):
mkdir *Verzeichnisname*
- Verzeichnisse löschen (**remove directory**):
rmdir *Verzeichnisname*
Achtung. Um ein Verzeichnis mit rmdir löschen zu können, muss es leer sein!
- Verzeichnisse + Inhalt löschen:
rm -r *Verzeichnisname*

pico – Ein simpler Editor

Mit einem Editor kann man Textdateien erstellen und verändern.

- Start: **pico** *filename*
- Beenden: Strg-x
- File lesen: Strg-r
- File sichern: Strg-o

Andere Editoren: vi, joe, emacs, xemacs

Achtung: Manchmal heißt der pico auch **nano**

Xemacs

- Xemacs ist ein grafischer Texteditor mit vielfältigen Funktionen zur Darstellung von Programmcode wie den von Perl, C, Java, Python, HTML, etc. unter anderem durch Syntax Highlighting.
- Mit **xemacs filename &** in einer shell (z.B. xterm) starten.
Anm.: Um unter MS Windows grafische Programme verwenden zu können, wird der X-Server *Exceed* benötigt.

Dateien kopieren (copy)

cp *file1 file2*

Kopiert die Datei file1 nach Datei file2 im selben Verzeichnis.

cp *test.html www*

Kopiert die Datei test.html in das Unterverzeichnis www. Annahme: www ist ein Verzeichnis, das bereits existiert.

cp *test.html ..*

Kopiert die Datei test.html in das darüberliegende Verzeichnis.

cp *test.html ../index.html*

Kopiert die Datei test.html in das darüberliegende Verzeichnis unter dem neuen Namen index.html.

cp -r *dir1 dir2*

Kopiert das Verzeichnis dir1 mit seinem gesamten Inhalt nach dir2

Dateien verschieben (move)

mv file1 file2

Der Dateiname file1 wird in file2 umbenannt. Die Datei wird nicht verschoben, da sie im selben Verzeichnis bleibt.

Genauso können auch **Verzeichnisse** umbenannt werden.

mv test.html www

Verschiebt die Datei test.html in das darunterliegende Verzeichnis www. Das Verzeichnis muss existieren. Ansonsten wird die Datei lediglich umbenannt.

mv test.html www/index.html

Verschiebt die Datei test.html in das darunterliegende Verzeichnis www unter dem neuen Namen index.html.

mv test.html ..

Verschiebt die Datei test.html in das darüberliegende Verzeichnis.

mv ../../www/index.html ~/daten/privat

Dateien löschen (remove)

rm file1 file2 file3 ...

Löscht die genannten Dateien im aktuellen Verzeichnis.

rm *

Löscht alle Dateien im Verzeichnis.

rm -r WWW

Löscht das Verzeichnis www samt allen darin enthaltenen Dateien und Verzeichnissen.

VORSICHT!!!

Alias

- Festlegung von Kurzformen für Kommandoaufrufe.

- Beispiele:

```
alias dir='ls -l'
```

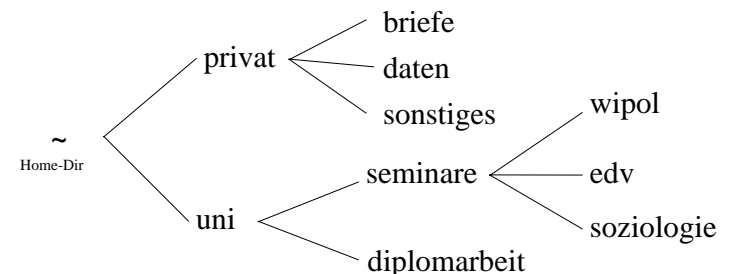
```
alias md='mkdir'
```

```
alias rd='rm -r'
```

Hinweis: Werden diese Aliase in die Datei ~/.bashrc eingetragen, stehen sie automatisch bei jeder Sitzung zur Verfügung.

Übung

- Legen Sie in ihrem Home-Directory folgende Verzeichnisstruktur an.



Übung

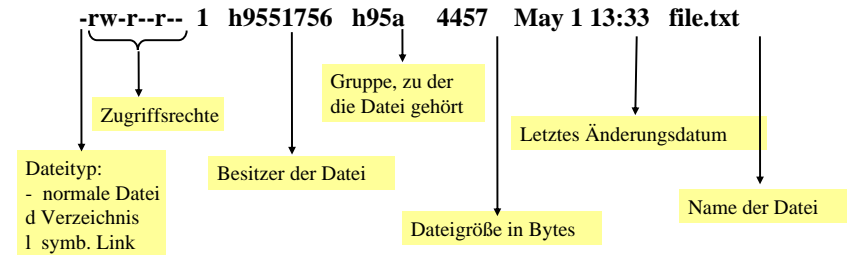
- Legen Sie im Verzeichnis `edv` mit dem Editor `pico` die Datei `seminararbeit.txt` an.
- Kopieren Sie diese Datei ins Verzeichnis `diplomarbeit`
- Duplizieren Sie dort diese Datei. Geben Sie ihr den Namen `dipl.txt`
- Löschen Sie das Verzeichnis `privat` samt Unterverzeichnissen.
- Ändern Sie den Namen des Verzeichnis `diplomarbeit` zu `dipl`.

Datei-Informationen

Durch die Eingabe von

ls -l

lassen sich die Dateiinformationen anzeigen.



User-Management

Es gibt drei Benutzer-Typen:

user *Besitzer eines Files.*

group *eine Gruppe von Benutzern.*

others *alle anderen Benutzer.*

all : Summe aus **user**,
group und **others**

User-Management

- Jedes File gehört einem User.
- Jedes File gehört einer Gruppe.
- Jeder User kann mehreren Gruppen angehören.

Ändern des Besitzers einer Datei (**change owner**):

Befehl: **chown** *Besitzer* file1 [file2 file3 ...]

Bsp: `chown j9551756 file.txt`

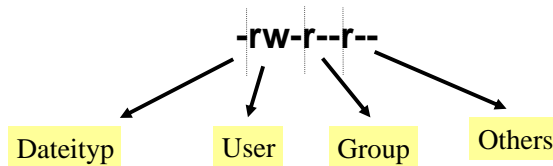
Ändern der Gruppenzugehörigkeit einer Datei (**change group**):

Befehl: **chgrp** *Besitzer* file1 [file2 file3 ...]

Hinweis: funktioniert nur, wenn man auch die entsprechenden Rechte hat.

Filepermissions

- Durch Eingabe von **ls -l** lassen sich die Zugriffsrechte (Filepermissions) anzeigen.



Filepermissions

- Dateityp: **-** Einfache Datei
d Verzeichnis (Directory)
l symbolischer Link
- Permissions: **r** read (4)
w write (2)
x executable (1)

Filepermissions hinzufügen

chmod *usertype+permission file*

Beispiel:

- **chmod g+w file** Gruppenmitglieder dürfen das File verändern.
- **chmod o+r file** Andere Benutzer dürfen das File lesen.

Filepermissions entfernen

chmod *usertype-permission file*

Bsp:

- **chmod u-w file** Der User selbst darf das File nicht mehr verändern.
- **chmod g-w file** Andere User der selben Gruppe dürfen das File nicht mehr verändern.
- **chmod a+rxw file** Alle erhalten Lese- Schreib- und Ausführungsrechte.

Filepermissions (Übung)

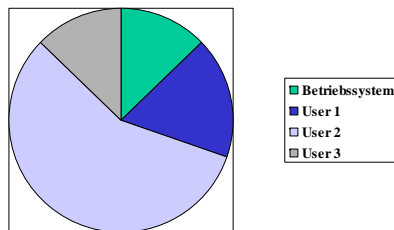
- Legen Sie mit einem Editor ein File an.
- Lassen Sie sich die Filepermissions anzeigen.
- Verändern Sie die Permissions so, dass Sie kein Recht mehr zum Verändern dieses Files haben.
- Versuchen Sie, das File mit dem Editor zu verändern.

Filepermissions (Übung)

- Verändern Sie die Permissions so, dass sie nun auch kein Recht zum Lesen des Files haben, die Gruppe soll weiterhin Leserechte besitzen.
- Überzeugen Sie sich bei Ihrem Nachbarn, dass Sie dessen File lesen können.

Prozess-Management

- Prozesse sind Aufgaben die von einem Computer (quasi) parallel ausgeführt werden.



Prozess-Management

- Um die laufenden Prozesse anzuzeigen verwendet man den Befehl **ps**. Jeder Prozess hat eine eindeutige Nummer (PID=Prozess ID).

Beispiele:

ps u zeigt die eigenen Prozesse an.

ps au zeigt auch die Prozesse von anderen Usern.

Prozess-Management

kill PID

beendet den Prozess mit der entsprechenden Nummer (PID).

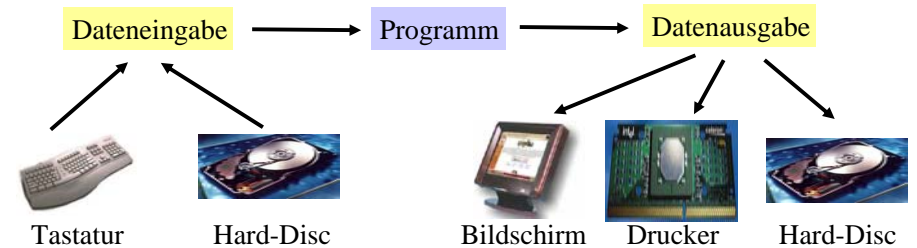
kill -9 PID

Der Signalcode -9 bedeutet auf jeden Fall terminieren.

top

Zeigt eine Tabelle der top-CPU Prozesse. Wird mit q beendet.

Eingabe/Ausgabe



Standardoutput/Umlenkung

- Standardmäßig ist der Bildschirm zur Ausgabe vorgesehen. Die Ausgabe lässt sich jedoch auch zu anderen Geräten umlenken:

echo eins 1>zahlen.txt 2>fehler.txt

echo zwei>>zahlen.txt

> Erstellt oder überschreibt die Datei

>> Erstellt oder hängt an bestehende Datei an

Hilfsprogramme - echo

echo text

Beispiele:

echo Guten Tag gibt "Guten Tag" am
Bildschirm aus.

echo hallo>gruss erzeugt eine Datei mit Inhalt
"hallo".

Hilfsprogramm - echo

- Umgebungsvariable dienen zur Speicherung von Informationen für das Betriebssystem. Der Inhalt kann mit *echo* angezeigt werden.

echo \$DISPLAY

Hilfsprogramm - cat

- *cat* (**concatenate**) hängt mehrere Files aneinander und gibt den Inhalt am Bildschirm aus:

cat file1 file2...

Beispiele:

cat vorname name *gibt den Inhalt der Files vorname und name am Bildschirm aus.*

Hilfsprogramm - cat

cat vorname name>anrede *schreibt den Inhalt der Files vorname und name in eine gemeinsame*

Datei anrede.

cat anrede *gibt den Inhalt von anrede am Bildschirm aus.*

Hilfsprogramm - grep

grep suchbegriff file

sucht im *file* nach dem *Suchbegriff* und gibt dann jene Zeilen auf dem Bildschirm aus, in denen der Suchbegriff gefungen wurde.

Kommen im Suchbegriff Leerzeichen vor, ist der Suchbegriff in Anführungszeichen einzuschließen.

Beispiel:

grep HTML datei.txt *gibt Zeilen aus file datei.txt, in denen die Buchstabenfolge HTML vorkommt aus.*

grep "HTML 4.0" datei.txt

Hilfsprogramm - wc

wc -flags file

Zählt bestimmte Dinge in einem File:

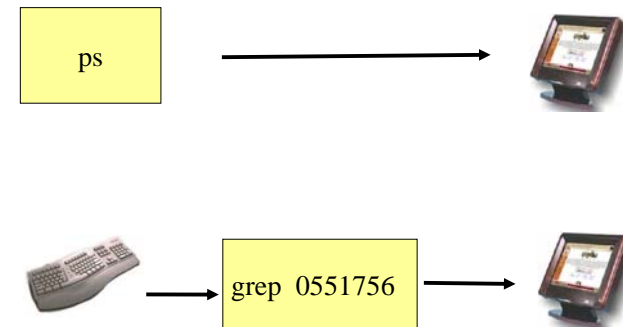
Beispiele:

wc -l file.txt gibt Anzahl der Zeilen in file.txt aus.

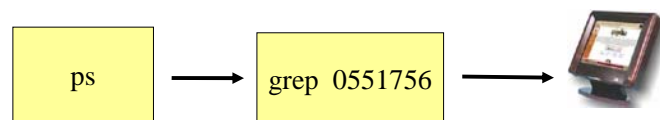
wc -w file.txt gibt Anzahl der Wörter aus.

wc -c file.txt gibt die Anzahl der Bytes aus.

PIPES



PIPES



ps au | grep 0551756

Übungsbeispiele

- Wenn Sie auf dem Server balrog arbeiten, gehen Sie in das folgende Verzeichnis:
cd ~aweichse/public_html/internet
- Geben Sie die Datei html40.txt auf dem Bildschirm aus.
- Suchen in html40.txt nach dem Wort 'HTML' mittels grep und verwenden Sie dazu eine PIPE.
- Zählen Sie die Zeilen in denen das Wort 'HTML' vorkommt.

Übungsbeispiele

- Kopieren Sie die Datei `html40.txt` in ihr Homedirectory. Wechseln Sie dorthin.
- Suchen Sie in derselben Datei nach dem Begriff 'ISO8879' (wieder mittels einer PIPE).
- Leiten Sie die Ausgabe des vorigen Beispiels in eine Datei mit dem Namen `iso8879.txt` um.

Sonstige Unix-Befehle

- **head `datei.txt`** gibt die ersten 10 Zeilen einer Datei aus.
- **tail `datei.txt`** zeigt die letzten 10 Zeilen einer Datei.
- **sort `datei.txt`** Sortiert eine Datei zeilenweise.
- **date** Aktuelles Datum und Uhrzeit.
- **who** Gibt eine Liste der zur Zeit angemeldeten Benutzer aus.

Übungsbeispiele

- Erstellen Sie eine Liste, die alle Dateien aus Ihrem Homedir enthält, und speichern Sie das Ergebnis unter dem Namen `liste.txt`.
- Hängen Sie an diese Datei das aktuelle Datum an.
- Geben Sie eine nach Usernamen sortierte Liste aller zur Zeit angemeldeten User am Bildschirm aus.

Übungsbeispiele

- Wieso kommt "1" raus wenn man folgenden Befehl eingibt:
`who | grep $USER | wc -l`
- Kann das Ergebnis auch größer als eins sein?
- **Anm:** `$USER` ist eine Umgebungsvariable, die den Usernamen (Loginname) beinhaltet.