

SCI Linux Einführung

Christian Bernhardt

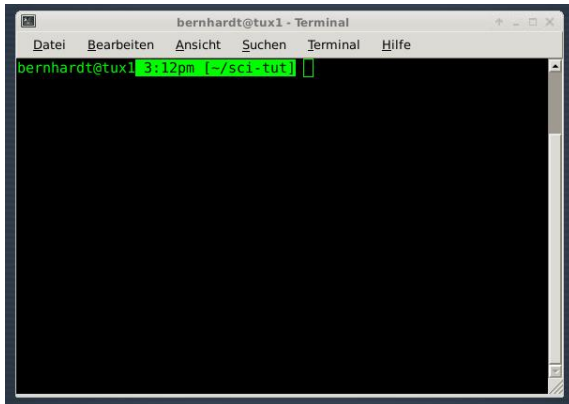
Service Center Informatik

bernhardt@cs.uni-kl.de
sci@cs.uni-kl.de

04.04.2017

Die Textkonsole (Shell) öffnen

- über den Menüeintrag *Terminal*
- mit dem Shortcut **strg + alt + t**



... und nutzen (1)

Befehle

- Befehlsname (z.B. **ls**) + Optionen (z.B. **-a**) + Argumente (z.B. **/home/p_muster**)
- Bsp.: `ls -a /home/p_muster`: Zeigt alle Dateien und Verzeichnisse im Verzeichnis `/home/p_muster` an

Tab-Completion

- mit *TAB* lassen sich Eingaben (z.B. Befehle, Pfade) vervollständigen, sofern die bisherige Eingabe eindeutig ist.
- bei mehrdeutiger Eingabe zeigt zweimaliges „Tabben“ alle möglichen Vervollständigungen.

... und nutzen (2)

History

- speichert alle Eingaben
- mit den Pfeiltasten \uparrow , \downarrow kann man vorherige Eingaben anzeigen und erneut ausführen.

Kopieren und Einfügen (im Terminal)

- *strg + shift + c* : Kopieren
- *strg + shift + v* : Einfügen
- Makieren mit der Maus und mittlere Maustaste : Kopieren und Einfügen

man

man <*befehl* | *datei*>

- Dokumentation zu Befehlen, speziellen Dateien, . . .
- Stichwortsuche in der Man-Page mit /<Stichwort>, nächster Treffer mit *n*
- scrollen mit ↑, ↓ *Bild* ↑, *Bild* ↓
- Beenden mit *q*

Linux Dateisystem-Hierarchie

- Baumstruktur
- Wurzel-Verzeichnis /
- Unterverzeichnisse unter /
 Bsp. /bin, /usr, /usr/local, /verz1/verz2/verz3
- **/home/<username>**: Heimatverzeichnis des Nutzers *username*, enthält die Nutzerdaten des Nutzers *username*
 Bsp.: **/home/p_muster**
- . = aktuelles Verzeichnis
- .. = übergeordnetes Verzeichnis

Achtung!

- *absolute* und *relative* Pfade

`cd /home/p_muster/uebung` (absolut)
≠
`cd home/p_muster/uebung` (relativ)

Navigation im Dateisystem

pwd

- gibt das aktuelle Verzeichnis aus

cd *<pfad>*

- wechselt das Verzeichnis nach *<pfad>*
- **cd** ohne Pfadangabe wechselt in das eigene Heimatverzeichnis
- **cd ..** wechselt in das übergeordnete Verzeichnis

Beispiele:

- *cd /v1/v2/v3* wechselt nach */v1/v2/v3*
- *cd v1* wechselt in das **Unterverzeichnis** *v1*
- *cd ../v1* gehe eins höher (..) und dann in das Verzeichnis *v1*

Anzeigen von Dateien

ls

- zeigt Inhalt des aktuellen Verzeichnisses an
- **ls** *<pfad>* : zeigt den Inhalt von *<pfad>*
- **-a** : zeigt auch versteckte Dateien/Verzeichnisse (*.<name>*)
- **-l** : mehr Informationen (Datei-/ Verzeichnisrechte, Größe, Zeitstempel)

Ausgeben von Dateien

less <datei>

- zeigt Inhalt der Datei <datei> in einem Pager an.
- suchen mit /<Stichwort>, weitersuchen mit *n*
- scrollen mit ↑, ↓ *Bild* ↑, *Bild* ↓. Beenden mit *q*

cat <datei>

- gibt den Inhalt der Datei <datei> auf die Textkonsole aus

mkdir, touch, Editoren

mkdir <verz>

- legt das Verzeichnis <verz> an.
- **mkdir -p** <pfad/verz> : legt die Verzeichnisstruktur an

touch <datei>

- legt eine leere Datei <datei> an.

Editoren

- gedit (graphisch, einfach zu bedienen)
- nano (Konsole, schnell, viele Funktionen, gewöhnungsbedürftig)
- vi / vim / gvim (Konsole (außer gvim), schnell, viele Funktionen, gewöhnungsbedürftig)

cp, mv, rm

cp *<quelle>* *<ziel>*

- kopiert Dateien.
- **-r** kopiert Verzeichnisse rekursiv
- **-p** erhält Dateiattribute (Besitzer, Zeistempel, . . .)

mv *<quelle>* *<ziel>*

- verschiebt Verzeichnisse und Dateien

rm *<datei | verz>*

- löscht leere! Verzeichnisse und Dateien.
- **-r** löscht Verzeichnisse rekursiv
- **-f** keine Sicherheitsabfrage
- löschen \neq in den Papierkorb verschieben

tar

tar

- **cvf** *<archiv>.tar <verz>*
archiviert das Verzeichnis *<verz>* in die Datei *<archiv>.tar*
- **cvfz** *<archiv>.tar.gz <verz>*
zusätzliche Kompression mit gzip
- **cvfj** *<archiv>.tar.bz2 <verz>*
zusätzliche Kompression mit bzip2
- **tvf** *<archiv>.tar*
zeigt den Inhalt von *<archiv>.tar*
- **xvf** *<archiv>.tar*
entpackt das Archiv *<archiv>.tar* in das aktuelle Verzeichnis

tar Beispiele

- **tar cvf** *backup-daten01.tar daten*
archiviert das Verzeichnis *daten* in die Datei *backup-daten01.tar*
- **tar cvfz** */Backup/homes/user01.tar.gz /home/user01*
archiviert und komprimiert das Verzeichnis */home/user01* in die Datei *user01.tar.gz* im Verzeichnis */Backup/homes*
- **tar xvf** *backup-daten01.tar*
entpackt das Archiv *backup-daten01.tar* in das aktuelle Verzeichnis
- **tar xvfz** */Backup/backup-daten02.tar.gz -C /home/user01*
Entpackt das Archiv */Backup/backup-daten02.tar.gz* nach */home/user01*

zip, unzip

zip, unzip

- **zip -r** *<archiv>.zip <verz>*
packt das Verzeichnis *<verz>* in die Datei *<archiv>.zip*
- **zip -l** *<archiv>.zip*
zeigt den Inhalt von *<archiv>.zip*
- **unzip** *<archiv>.zip*
entpackt die Datei *<archiv>.zip* in das aktuelle Verzeichnis

weitere Komprimierer

- bzip2
- gzip
- 7z

Besitzer, Gruppe und Rechte (1)

Dateien und Verzeichnisse

- gehören zu einem *Besitzer* (**u**) und einer *Gruppe* (**u**)
- besitzen Rechte zum *Lesen* (**r**), *Schreiben* (**w**) und *Ausführen* (**x**)
für *Besitzer* (**u**), *Gruppe* (**g**) und *Andere* (**o**)

Rechte:

- *Lesen* (**r**): Inhalt anzeigen
- *Schreiben* (**w**): Inhalt ändern, Dateien/Verzeichnisse anlegen
- *Ausführen* (**x**): Datei ausführen, in Verzeichnis wechseln

Besitzer und Gruppe und Rechte (2)

`ls -l <datei |verz>` Zeigt den Besitzer, die Gruppe und die Rechte

- `drwxrwxr-x` 4 p_muster users 4096 Apr 13 12:50 uebungen
- 1. Spalte: *Dateityp* (1. Zeichen) und *Rechte* (Zeichen 2-10)
Rechte **Besitzer** (**u**) (Zeichen 2-4)
Rechte **Gruppe** (**g**) (Zeichen 5-7)
Rechte **Andere** (**o**) (Zeichen 8-10)
- 3. Spalte: *Besitzer* (**u**) → p_muster
- 4. Spalte *Gruppe* (**g**) → users

Rechte Beispiel 1

`drwxrwxr-x 4 p_muster users 4096 Apr 13 12:50 uebungen`

- **Zeichen 1:** d: Verzeichnis
- **Zeichen 2-4:** Rechte Besitzer (u)
rwx : lesen, schreiben, ausführen
- **Zeichen 5-7:** Rechte Gruppe (g)
rwx : lesen, schreiben, ausführen (ins Verzeichnis wechseln):
- **Zeichen 8-10:** Rechte Andere(o)
r-x : lesen, ausführen (ins Verzeichnis wechseln):

Rechte Beispiel 1

```
-rwxr-x--- 4 p_muster users 4096 Apr 13 12:50 HelloWorld.py
```

- **Zeichen 1:** - Datei
- **Zeichen 2-4:** Rechte Besitzer (u)
rwx : lesen, schreiben, ausführen
- **Zeichen 5-7:** Rechte Gruppe (g)
r-x : lesen, ausführen
- **Zeichen 8-10:** Rechte Andere(o)
--- : kein Zugriff

Rechte ändern

chmod *<wer> +|-|= <rechte> <datei |verz>*

- ändert die Rechte von Dateien und Verzeichnissen
- -R : rekursives ändern
- **u** (user): Besitzer, **g** (group): Gruppe, **o** (others): Andere, **a** (all): Alle
- **+** : Recht hinzufügen, **-** : Recht entfernen, **=** : Rechte auf die angegebenen setzen, alle anderen entfernen
- **r** (read) : lesen, **w** (write) : schreiben, **x** (execute) : ausführen
- **,** : Trennzeichen

Beispiele: Rechte ändern

- **chmod o-x prog01.bin**
Andere (**o**) : Ausführrecht (**x**) für die Datei *prog01.bin* entfernen (-)
- **chmod -R a+r Bilder**
Alle (**a**) : Leserecht (**r**) für den Ordner Bilder, dessen Unterordner und Dateien erteilen (+)
- **chmod ug=rwx,o=r skripte**
Besitzer (**u**) und Gruppe (**g**): Vollzugriff, Andere (**o**): Leserecht (**r**) auf das Verzeichnis / die Datei *skripte* setzen (=)
- **chmod u+x,g-x,o-x prog02.bin**
Besitzer (**u**): Ausführrecht (**x**) erteilen, Gruppe (**g**) und Andere (**o**): Ausführrecht (**x**) entziehen (-)

Besitzer und Gruppe ändern

chgrp <group> <verz |datei>

- ändert die Gruppe von <verz |datei> auf <group>
- **-R** rekursiv

chown <user> <verz |datei>

- ändert den Besitzer eines Verzeichnisse oder einer Datei .
- **-R** rekursiv

Achtung!

Bei Gruppenänderung muss der Besitzer Mitglied der neuen Gruppe sein. Nur *root* kann den Besitzer ändern.

ssh, scp

ssh *<user>@<host>*

- startet Shell auf entferntem Rechner
- verschlüsselt

scp

- kopiert Dateien zwischen Rechnern mittels ssh
- **-r** Rekursives Kopieren für Verzeichnisse
- vom **lokalen** zum **entfernten** Rechner:
scp </pfad/zur/quelle> <user>@<host>:<pfad/zum/ziel>
- vom **entfernten** zum **lokalen** Rechner:
scp <user>@<host>:<pfad/zur/quelle> <pfad/zum/ziel>

scp Beispiel

```
scp uebung01.txt p_muster@tux1.cs.uni-kl.de:/home/tutor/  
abgaben/uebung01.txt
```

Kopiert die **Quelldatei** *uebung01.txt* als **Nutzer** *p_muster* in das **Verzeichnis** */home/tutor/abgaben* auf dem **entfernten Rechner** *tux1.cs.uni-kl.de*

Java

Java

- Version: **javac** *-version*, **java** *-version*
- Compilieren (OpenJDK): **javac** *<name>.java*
- Ausführen (OpenJDK): **java** *<name>*
- Oracle Java 8: **javac8**, **java8**

C

C

- Version: **gcc --version**
- Compilieren: **gcc <name>.c** (erzeugt ausführbare Binärdatei *a.out*)
- Ausführen: *./a.out*
- Besser: Kompilieren: **gcc -o <outputname> <name>.c**
(erzeugt ausführbare Binärdatei *<outputname>*) und
Ausführen: *./<outputname>*
- **Achtung!** Programm muss ausführbar sein (Rechte).

Python

Python

- Version: **python** *--version*
- Ausführen: **python** *<name>.py* für Systemdefault oder explizit **python2.7**
oder: *./<name>.py*, wenn *<name>.py* ausführbar ist.