

Vorkurs - Tag 2

FSI Informatik

Uni Erlangen-Nürnberg

11. Oktober 2006

- 1 Help! I need somebody...
- 2 More rights for the world
- 3 Secure Login in the Wild Wild West

Typische Situation

Frage: "Mist, wie konnte ich nur uvw mit Befehl xyz machen?"

Antwort: "man pages!"

man-pages ...

- sind die erste Anlaufstelle für Fragen zu Kommandozeilenbefehlen.
- sind die erste Anlaufstelle für Fragen zur Programmierschnittstelle.
- gibt es für (fast) jeden Befehl.

Und wie benutze ich diese man-pages?!?

Typische Verwendung

```
man <Befehl>
```

Und dann? Lesen!

```
CP(1)
```

```
User Commands
```

```
NAME
```

```
cp - copy files and directories
```

```
SYNOPSIS
```

```
cp [OPTION]... SOURCE DEST
```

```
cp [OPTION]... SOURCE... DIRECTORY
```

```
cp [OPTION]... --target-directory=DIRECTORY SOURCE
```

Wie ist eine man-page aufgebaut? (2)

Typischer Aufbau

NAME hier findet ihr den Namen und eine Kurzbeschreibung des Befehls.

SYNOPSIS die Aufrufsyntax mit möglichen Optionen. Dabei stehen Optionen die nicht erforderlich sind in eckigen Klammern.

DESCRIPTION eine ausführliche Beschreibung des/der Befehle und Optionen.

OPTIONS manchmal sind die Optionen auch in einem eigenen Unterpunkt aufgeführt.

Wie ist eine man-page aufgebaut? (2)

Typischer Aufbau

AUTHOR ab und zu findet man auch die ursprünglichen Autoren in den man pages aufgelistet.

BUGS welche bekannten Fehler das Programm hat, ist hier zu finden.

SEE ALSO verwandte man pages oder zusätzliche Informationsquellen.

Wie ist eine man-page aufgebaut? (2)

Typischer Aufbau

AUTHOR ab und zu findet man auch die ursprünglichen Autoren in den man pages aufgelistet.

BUGS welche bekannten Fehler das Programm hat, ist hier zu finden.

SEE ALSO verwandte man pages oder zusätzliche Informationsquellen.

Aber es gibt auch Ausnahmen!

Viele man pages halten sich an den oben beschriebenen Aufbau. Aber auch hier gilt: *Ausnahmen bestätigen die Regel!* Also nicht überrascht sein, wenn der Aufbau mal anders ist.

Die wichtigsten Tasten

- **Scrollen (zeilenweise):** Cursortasten hoch/runter
- **Scrollen (seitenweise):** PageUp/PageDown
- **Suchen:** /suchbegriff<ENTER>
- **nächster Treffer:** n
- **vorheriger Treffer:** N
- **beenden:** q

Gruppenkonzept

- jeder Benutzer ist in min. einer Gruppe (im CIP: *immdstud*)
- man kann in mehr als einer Gruppe sein
- über Gruppen können Berechtigungen auf Verzeichnissen und Dateien geregelt werden

Ein paar Grundlagen!

Gruppenkonzept

- jeder Benutzer ist in min. einer Gruppe (im CIP: *immdstud*)
- man kann in mehr als einer Gruppe sein
- über Gruppen können Berechtigungen auf Verzeichnissen und Dateien geregelt werden

Einteilung der Benutzer

Owner Diesem Benutzer "gehört" die Datei/das Verzeichnis

Group Dieser Gruppe "gehört" die Datei/das Verzeichnis

Other Alle anderen

Und welche Berechtigungen hat eine Datei/ein Verzeichnis?

- `ls -l` zeigt eine ausführliche Ausgabe
- dabei zeigt die erste Zeichengruppe die Rechte an

Ein Beispiel:

```
bash:> ls -l test
-rwxr-xr-x  1 sichkoll  immdstud  256 Oct  7 test
```

Ein Beispiel:

```
bash:> ls -l test
-rwxr-xr-x  1 sichkoll  imdstud  256 Oct  7 test
```

Und was heisst das jetzt?

- das erste Zeichen zeigt den Typ an (z.B. *d* für ein Verzeichnis oder *-* für normale Dateien)
- die nächsten drei Zeichen zeigen die Rechte für den **Owner**
- das zweite Zeichentripel zeigt die Rechte für die **Group**
- und (logischerweise) die verbleibenden drei Zeichen, die Rechte für den Rest der Welt

Ein Beispiel:

```
bash:> ls -l test
-rwxr-xr-x  1 sichkoll  immdstud  256 Oct  7 test
```

r? w? x?

- *x* = ausführbar
- *w* = schreibbar
- *r* = lesbar

Ein Beispiel:

```
bash:> ls -l test
-rwxr-xr-x  1 sichkoll  immdstud  256 Oct  7 test
```

Aber ich kann besser mit Zahlen statt mit Zeichen

- kein Problem:
 - *ausführbar* = 1
 - *schreibbar* = 2
 - *lesbar* = 4
- für jedes Tripel können diese Zahlen aufaddiert werden zu einer Zahl zwischen 0 und 7.

Ein Beispiel:

```
bash:> ls -l test
-rwxr-xr-x  1 sichkoll  immdstud  256 Oct  7 test
```

Das Beispiel erklärt

- ausführbar, schreibbar und lesbar für den Besitzer ($1 + 2 + 4 = 7$)
- ausführbar und lesbar für die Gruppe ($1 + 4 = 5$)
- ausführbar und lesbar für alle Anderen ($1 + 4 = 5$)

Ein anderes Beispiel:

```
bash:> ls -l
drwxr-xr-x  1 sichkoll  immdstud  256 Oct  7 test
```

Aber Moment! Wie können Verzeichnisse ausführbar sein?

Für Verzeichnisse gilt:

- Ausführbar bedeutet, dass man das Verzeichnis betreten kann (*cd <Verzeichnis>*)
- Schreibbar bedeutet, dass man weitere Verzeichnisse und Dateien darin anlegen und auch löschen kann
- Lesbar bedeutet, dass man den Inhalt des Verzeichnisses auflisten kann (z.B. mit *ls*)

Änderungen im Rechtesystem

Benutzer ändern

- `chown <login> <Datei|Verzeichnis>`
- `chown -R <login> <Datei|Verzeichnis>` (rekursiv: auch alle Unterverzeichnisse und enthaltene Dateien)

Gruppe ändern

- `chgrp <group> <Datei|Verzeichnis>`
- `chgrp -R <group> <Datei|Verzeichnis>` (rekursiv)

Rechte ändern

- `chmod <mod> <Datei|Verzeichnis>`
- `chmod -R <mod> <Datei|Verzeichnis>` (rekursiv)

Versteckte Dateien

Normales `ls`

```
bash:> ls  
a.txt    mein_pic.jpg
```

`ls -a` zeigt "alles"

```
bash:> ls -a  
.      ..      .bash_history  a.txt  mein_pic.jpg
```

- `ls -a` zeigt wirklich alle Einträge des Verzeichnisses an
- Einträge, die mit einem "." beginnen, werden als "müssen nicht immer sichtbar sein" betrachtet und versteckt, z.B.:
 - "." ist immer das aktuelle Verzeichnis
 - ".." ist immer das übergeordnete Verzeichnis
 - ".bash_history" enthält z.B. Befehle, die früher eingegeben wurden
 - ".vimrc" enthält die Konfiguration für den vim-Editor
- Viele Befehle wie z.B. `cp *` arbeiten in der Grundform nur auf nicht-versteckten Dateien

Typische Situation

- es ist Sonntag Abend
- Montag, 10 Uhr ist Abgabe der Algo-Übungen
- man hat kein Auto
- es fährt kein Bus mehr
- das Fahrrad ist platt
- es regnet
- und man hat natürlich keinen Regenschirm
- ...

SSH? Was ist das?

von Wikipedia: ... ist sowohl ein Programm als auch ein Netzwerkprotokoll, mit dessen Hilfe man sich über eine verschlüsselte Netzwerkverbindung auf einem entfernten Computer einloggen und dort Programme ausführen kann.



Wie benutzt man SSH?

Unter Linux

- sollte bereits auf deinem Rechner installiert sein
- danach gibst du einfach in einem Terminal ein:

```
ssh login@CIP-Rechnername.informatik.uni-erlangen.de
```
- *login* ist dein CIP-Login, z.B. sifoobar
- *CIP-Rechnername* ist beispielsweise faui00a (findet man auf den Monitoren)
- benutze dein CIP-Passwort um dich einzuloggen

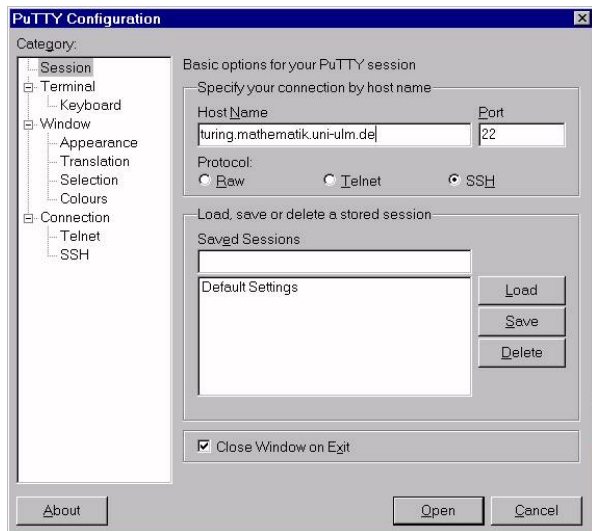
Tipp

im CIP kannst du `login@` und `.informatik.uni-erlangen.de` auch weglassen, es reicht also z.B.: `ssh faui00a`

Unter Windows

- für Windows existieren verschiedene SSH-Programme
- wir empfehlen putty, welches kostenlos unter
`http://www.putty.nl/download.html`
zu bekommen ist
- auch hier `CIP-Rechnername.informatik.uni-erlangen.de` als Host nehmen
- einloggen mit CIP-Account

putty-Screenshot



Kopieren zwischen Rechnern

scp

scp kann Dateien von einem Rechner auf einen anderen kopieren und verwendet ssh für die Authentifizierung

Unter Linux

```
scp login@rechnername:<pfad+name> <ziel>
```

Beispiel:

```
scp sicakauf@fau05.informatik.uni-erlangen.de:datei .
```

kopiert die Datei 'datei' aus dem *home* von sicakauf in das aktuelle Verzeichnis (durch den '.' angegeben)

Unter Windows

um Dateien zwischen Linux und Windows auszutauschen, kann man WinSCP verwenden (<http://winscp.net>)

winscp-Screenshot

The screenshot shows the WinSCP interface with the following details:

- Title Bar:** wiki - test@localhost - WinSCP
- Menu Bar:** Local, Mark, Files, Commands, Session, Options, Remote, Help
- Address Bar:** test@localhost
- Local Panel (Left):** D:\Documents\wiki
- Remote Panel (Right):** /home/martinp/public_html/wiki/wiki
- Local File List:**

Name	Size	Changed	A.
integration.txt	2 942	26.4.2005 10:19:18	a
interfaces.txt	1 924	6.4.2005 16:38:39	a
introduction.txt	1 308	4.5.2005 17:49:10	a
languages.txt	2 238	5.5.2005 12:25:06	a
operation_mask.txt	2 035	25.4.2005 10:35:48	a
protocols.txt	6 470	4.5.2005 21:02:04	a
remote_comman...	4 574	4.5.2005 10:46:50	a
requirements.txt	5 259	4.5.2005 9:29:20	a
resume.txt	2 923	4.5.2005 20:10:34	a
screenshots.txt	857	3.5.2005 10:26:40	a
scripting.txt	2 931	4.5.2005 10:03:26	a
security.txt	2 080	5.4.2005 17:00:05	a
setup.txt	52	15.12.2004 13:5...	a
shell_session.txt	966	18.2.2005 16:43:12	a

25 945 B of 405 KB in 9 of 157

- Remote File List:**

Name	Size	Changed	Rights
..		4.5.2005 21:05:09	rw-rw-rwx
.htaccess	31	10.11.2004 22:4...	rw-rw-rw-
commandline.txt	2 752	4.5.2005 11:13:40	rw-rw-rw-
config.txt	1 562	29.4.2005 11:03:36	rw-rw-rw-
contributions.txt	1 064	5.5.2005 13:43:50	rw-rw-rw-
directory_cache...	1 861	25.3.2005 9:56:49	rw-rw-rw-
dragext.txt	3 977	1.5.2005 9:53:44	rw-rw-rw-
faq.txt	3 962	4.5.2005 14:10:16	rw-rw-rw-
faq_commandlin...	102	17.12.2004 12:4...	rw-rw-rw-
faq_dir_default.txt	961	8.2.2005 13:04:36	rw-rw-rw-
faq_download_t...	745	29.4.2005 10:57:04	rw-rw-rw-
faq_drag_move...	1 590	8.2.2005 12:54:32	rw-rw-rw-
faq_error_code.txt	696	17.3.2005 14:00:24	rw-rw-rw-
faq_filemanager...	503	14.4.2005 17:17:14	rw-rw-rw-

6 690 B of 102 MB in 4 of 22

Toolbar: F2 Rename, F4 Edit, F5 Copy, F6 Move, F7 Create Directory, F8 Delete, F9 Properties, F10 Quit

Status Bar: 12 914 B, 751 KB, aes, SFTP (v3), 0:15:22