13. Nützliche Hilfen

Ändern der Zugriffsrechten von Files

Ändern des Eigentümers von Files

```
chown [-R] <Nutzername> <Filename> {<Filename>}
```

Ändern des Eigentümers der Files in den spezifizierten neuen Nutzernamen.

-R – Eigentümer rekursiv absteigend ändern

Ändern der Gruppe von Files

```
chgrp [-R] <Gruppenname> <Filename> {<Filename>}
```

Ändern der Gruppenzugehörigkeit der Files in den spezifizierten neuen Gruppennamen.

-R – Gruppe rekursiv absteigend ändern

Ändern der Zugriffsmodies von Files

```
chmod [-R] <Zugriffsrechte> <Filename> {<Filename>}

<Zugriffsrechte>:=<numerische Zugriffsrechte> |
                  <symbolische Zugriffsrechte>
  numerische Zugriffsrechte:
    400  -  Leserechte Eigentümer
    200  -  Schreibrechte Eigentümer
    100  -  Ausführungsrechte Eigentümer
    040  -  Leserechte Gruppe
    020  -  Schreibrechte Gruppe
    010  -  Ausführungsrechte Gruppe
    004  -  Leserechte Welt
    002  -  Schreibrechte Welt
    001  -  Ausführungsrechte Welt

   Kombinationen sind zulässig z.B.
      755 - alles für den Eigentümer, der Rest
            darf alles ausser verändern.
 Beispiel:
   chmod 755 xyz abc
   chmod -R 700 xyz abc Dab
```

symbolische Zugriffsrechte:

```
[<Ziel>]<Operator>[<Recht>]

    Rechte:
      r - lesen
      w - schreiben
      x - ausführen
    Ziel
      u - Nutzer
      g - Gruppe
      o - Welt
    Operatioren
      +  - Recht hinzufügen
      -  - Recht streichen
      =  - Recht absolute setzen
```

Beispiele:

```
chmod u=rx xxx
chmod og= xxx
```


Verwalten der File-Creation-Maske

```
umask  [<Maske>]
```

Setzen und Anzeigen der Bits der File-Creation-Maske eines Prozesses. Wird kein Parameter angegeben werden die gesetzten Bits der File-Creation-Maske angezeigt. In der Maske werden die Bits oktal kodiert, für die beim Erzeugen eines neuen Files keine Zugriffsrechte gegeben werden sollen.

Beispiele:

```
umask      - Anzeigen der File-Creation-Maske des Prozesses
umask 022  - Schreibrechte für Gruppe und Other bei neuen Files
             prinzipiell gelöscht.
umask 077  - alle Zugriffsrechte für Gruppe und Other bei neuen
             Files prinzipiell gelöscht.
```

Benennung von Files

Umbenennen von Files

```
mv [-fi] <Quellfile> <Ziel-Direktory>
mv [-fi] <Quellfile> <Ziel-File>
```

Umbenennen des Quellfiles(alt) in das Zielfile(neu). Ist das Zielfile ein Direktory, wird der Basis-Name des Quellfiles der Filename im Zieldirektory.

```
-f  - überschreiben von bestehenden Files ohne nachfragen
-i  - interaktiv
```

Beispiele:

```
mv xxx yyy     - umbenennen des Files xxx in yyy
mv xxx Yyy     - Yyy ist eine Direktory. Das File
                 wird in Yyy/xxx umbenannt.
```


Erzeugen eines neuen Namens für ein File

```
ln [-fns] <Quellfile> <Zielfile>
```

Erzeugen eines neuen Namens <Zielfile> für ein bestehendes File <Quellfile>.

```
-s   - erzeugen eines symbolischen Namens
-n   - bestehende Files nicht ueberschreiben
-f   - Erzeugen des neuen Files ohne Nachfrage
```

Beispiele:

```
ln xxx yyy
ln -x /vol/fob-vol3/mi99/xxx xxx
```

Kopieren von Files

```
  cp [-i|-p] <Quellfile>  <Zielfile>

  cp [-i|-p] <Quellfile1> {<Quellfile2>} <Zieldirektory>

  cp [-i|-p] [-r|-R] <Quelldirektory> <Zieldirektory>
```

Kopieren eines oder mehrer Files in ein anderes File bzw. in eine andere Direktory. Direktories können rekursiv kopiert werden (Option -r).

```
  -i  - interaktiv (beim Überschreiben von Files)
  -p  - Kopieren der Zugriffsrechte und Zeiten
  -r  - rekursives Kopieren
  -R  - rekursives Kopieren ohne Pipes
```


Operationen über Files

Berechnen der Anzahl von Zeilen, Wörtern und Zeichen in einem File

```
  wc [-c][-m][-l][-w] <Filename> {<Filename>}

     -c    Anzahl der Bytes ausgeben

     -m    Anzahl der Zeichen ausgeben

     -l    Anzahl der Zeilen

     -w    Anzahl der Worte

   Beispiele:

    > wc mkthml
      11     50   409   mkhtml
    >
```

```
Inhalt eines Files nach verschiedenen Gesichtspunkten sortieren

   sort  [Optionen]  [<Filename>]

   Inhalt des Files (oder Standardeingabe) nach verschiedenen
   Gesichtspunkten (Optionen) sortieren und auf Standard-
   ausgabe ausgeben.

      -b   ignorieren führender Leerzeichen
      -f   ignorieren Groß-Klein-Schreibung
      -i   ignorieren nicht-druckbare Zeichen
      -n   numerische Sortierung
      -r   reverse Sortierung
      u.s.w.

   Beispiele:

      sort abc
      sort -r abc
```



```
Files vergleichen
-----------------

Vergleichen von Files Zeile bei Zeile

  diff  [Optionen]  <Filename1> <Filename2>
  diff  [Optionen]  <Directoryname1> <Directoryname2>

  Optionen:
    für Files
     -b        - führende Leerzeichen ignorieren
     -i        - ignoriere Groß-/Kleinbuchstaben
     -t        - Tabulatoren expandieren
     -w        - ignoriere Leerzeilen (einschließlich Blanks und Tabs)
     -c        - Differenzprotokoll mit 3 Zeilen Umgebung der Differenz
     -e        - erzeugen eines Scriptes fuer ed
     -n        - ähnlich -e
     -C number - Differenzprotokoll mit <number> Zeilen Umgebung
    für Directories
     -s        - identische Files anzeigen
     -l        - langes Format
     -r        - rekursiv


Vergleich von Files Zeiche bei Zeichen

  cmp  [-l][-s]  <Filename1> <Filename2>

      -l   - Ausgabe der Zeichen oktal und Adresse dezimal
      -s   - nur Rückkehrwert erzeugen, keine Ausgabe
```

```
Suchen
------

Suchen von Zeichenketten in Files

  grep    [-chn] [-l] [-q]  "Zeichenkette" <File1> {<File>}
  egrep   [Optionen]  <regulärer Ausdruck> <File1> {<File>}

    -c  - nur zählen
    -h  - Filename unterdrücken
    -l  - nur Filenamen anzeigen
    -q
    -n  - Zeilennummern ausgeben
    -s  - keine Fehlerausgabe
    -v  - inverse Ausgabe (Zeilen, die Zeichenkette nicht enthalten)

  Beispiele:

    grep mail *
    grep -cl mail *
```



```
Suchen von Files im Filesystem

  find <Direktory> [Optionen] [Aktionen]

     Optionen
      -name <name>      - suche File mit Namen <name>
      -ctime <n>        - suche Files, deren Status vor <n> Tagen
                             +n   älter als n Tage
                              n   genau n Tage
                             -n   weniger als n Tage
                           geändert wurde
      -group <name>     - suche Files der Gruppe <name>
      -user  <name>     - suche Files des Nutzers <name>
      -perm  <mode>     - suche Files mit Zugriffsrechten <mode>
                                - genau diese Bits gesetzt
                              - - diese Bits sind auch gesetzt
      -type c           - Filetype (b,c,d,f,l,p,s)

     Aktionen
        -exec <Kommando>  - ausführen des Kommandos <Kommando> für
                             das gefundene File
        -print            - gefundenen Filename ausgeben

   Beispiele:
      find / -name core -print
      find / -name core -exec rm {} \;
      find /usr/bin -perm 4111 -user root -print
      find /usr/bin -perm -4111 -user root -print
      find /usr/bin -perm -4000 -user root -print
```

```
Informationen abzweigen
-----------------------
Lesen von Stadardeingabe und Schreiben auf Standardausgabe und
in ein File

  tee [-ai]  <File>

  Optionen

      -a    - Anfügen der Ausgabe an das File
      -i    - ignoriere Interrupts

  Beispiel:

     ls -lisa | tee xxx
```



```
Kompremieren von Files
----------------------

GNU-Zip - Packen und Entpacken von Files mittels Lempel-Ziv-Kodierung

Gepackte Files haben die Endung:  .gz

  gzip [ -acdfhlLnNrtvV19 ] [-S <suffix>] [ <name> ... ]
  gunzip [ -acfhlLnNrtvV ] [-S <suffix>] [ <name> ... ]
  gzcat [ -fhLV ] [ name ... ]

    gzip   - packen
    gunzip - entpacken
    gzcat  - anzeigen auf Standardausgabe

  Beispiele:

      gzip abc          # es entsteht   abc.gz
      gunzip abc.gz     #  es entsteht abc
      gcat   abc.gz     #  abc wird nach Standardausgabe ausgegeben
                        #  abc.gz bleibt unverändert
```

```
GNU-BZip - Packen und Entpacken von Files mittels
           Block-Sorting-Text-Compremierung (Burrows-Wheeler)

Gepackte Files haben die Endung: bz2

  bzip2 [ -cdfkstvzVL123456789 ] [ <filenames> ... ]
  bunzip2  [ -fkvsVL ] [ <filenames> ... ]
  bzcat [ -s ] [ <filenames> ... ]
  bzip2recover  <filename>

    bzip2         - packen
    bunzip2       - entpacken
    bzcat         - anzeigen auf Standardausgabe
    bzip2recover  - entpacken mit fehlerhafter Prüfsumme

  Beispiele:

      bzip2 abc             # packen, es entsteht   abc.bz2
      bunzip2 abc.bz2       # entpacken, es entsteht abc
      bzcat   abc.bz2       # abc wird nach Standardausgabe ausgegeben
                            # abc.bz2 bleibt unverändert
      bzip2recover abc.bz2 # entpacken mit fehlerhafter Prüfsumme
```



```
Einpacken von Files und Directories
-----------------------------------

Erzeugen und entpacken von Bandarchiven
tar - tape archive

Erzeugen eines Tapearchives:

  tar  c[bBeEfFhiklnopPqvwX[0-7]] [block] [tar-file] [exclude-file]
       {-I include-file | -C directory | file  | file }  ...

  Beispiel:
     tar -cvf xyz.tar  ./xyz

Erzeugen eines Inhaltsverzeichnisses eines Tapearchives

  tar  t[BefFhiklnqvX[0-7]] [tarfile] [exclude-file]
       {-I include-file | file}  ...

  Beispiel:
     tar -tvf xyz.tar

Auspacken eines Tapearchives

  tar x[BefFhiklmnopqvwX[0-7]] [tarfile] [exclude-file] [file ...]

  Beispiel:
     tar -xvf xyz.tar
```

```
Informationskommandos
---------------------

Grafische Oberfläche zum Lesen von Manuals

    xman


Suchen von Files in Locate-Datenbasis

  locate [-d path | --database=path] [-e | --existing]
         [-i | --ignore-case ] [--version] [--help] pattern...


Bestimmen von Binaries, Sources und Manual-Pages von Kommando-Files

   whereis [-bmsu]  [-B|M|S directory  ... -f] filename ...



Bestimmen des benutzten Filenamen oder Aliases für ein Kommando

   which [ filename ....]
```



```
Bestimmen der Plattenbelegung

  df [-F FSType] [-abegklntV] [-oFSType-specific_options]
       [block_device | directory | file | resource ... ]

  Beispiele:
     df
     df -k
     df -k /vol/fob-vol3


Bestimmen der Größe von Direktories und Files

  du [-adkLr] [-o|-s] [file ...]

  Beispiele:
    du
    du .
    du -sk .
    du -sk abc
```

```
Drucken von Files
-----------------

Druckauftrag absenden

    lp [-c]  [-m]  [-p]  [-s]  [-w]  [-d destination]
       [-f form-name]  [-H special-handling]  [-n number]
       [-o option]  [-P page-list]  [-q priority-level]
       [-S character-set | print-wheel]   [-t title]
       [ -T content-type  [-r]] [-y mode-list]  [ file ... ]

     Beispiel:
         lp -d lpbeat   postscritptfile
         lp -d lpemu    postscritptfile

Druckauftrag absenden

     lpr [-P destination]  [-# number]  [-C class]
         [-J job]  [-T title]  [-i  [ indent ]]
         [-1 | -2 | -3 | -4 font] [-w cols] [-m]
         [-h] [-s] [-filter_option]  [ file ... ]

    Beispiele
       lpr -P lpbeat  postscritptfile
       lpr -P lpemu    postscritptfile
```



```
Druckerwarteschlange anschauen

     lpq [-P destination]  [-l]  [+[ interval]] [request-ID ... ]
          [user ...]

    Beispiel:
       /usr/ucb/lpq -l


Druckzustand anschauen

    lpstat  [-d] [-r] [-R] [-s] [-t] [-a [list]]
            [-c [list]] [-f [list] [-l]] [-o [list]]
            [-p [list] [-D] [-l]]  [-S [list] [-l]]
            [-u [login-ID-list]] [-v [list]]

    Beispiel:
       lpstat -t
```

Druckerauftrag löschen

für Aufträge, die mit lpr abgeschickt wurden:

```
lprm [-P destination]  [-]  [request-ID  ...] [user ...]
```

Beispiele:

```
lprm -P lpemu  607
lprm    301
```

für Aufträge, die mit lp abgeschickt wurden:

```
cancel [request-ID ... ]  [destination ...]
cancel -u user ...  [destination ...]
```

Beispiele:

```
cancel lpbeat-607
```


Postsriptfile erzeugen

```
a2ps   [Optionen]  --output=<Ausgabefile>  [<Eingabefile>]
```

a2ps konvertiert ASCII-Files in Postscriptfiles. Standardmäßig wird von Standardeingabe gelesen und auf den Standarddrucker gedruckt.

Beispiele:

```
a2ps --help                 - anzeigen aller Optionen (Hilfe)
a2ps  xxx                   - Die Datei xxx wird auf dem
                              Standarddrucker gedruckt
a2ps --output=xxx.ps  xxx   - Die Datei xxx wird in das
                              File xxx.ps umgewandelt
lp xxx.ps
```

```
SVN - Versionsverwaltungssysteme
================================

           Wie verwaltet man komplexe Quelltexte ohne daß
           man mal vor dem Nichts steht

Versionsverwaltungssystem:

   Was wird verwaltet?
      Gewöhnliche Dateien - Text-Files, in der Regel keine Binärdaten
            z.B.:  Quelltexte, Texte, Makefiles, Konfigurationsfiles,...
      Mehrere Versionen werden aufbegehoben  und bei Bedarf
             wieder erzeugt.

   Ziele von Versionsverwaltungssystemen:
     Kontrollierten Zugriff ermöglichen
     Frühere Versionen wiederherstellen
     Kontinuierliche Generationsfolge sichern
     Dokumentation der Änderungen
     Parallelen Zugriff organisieren - Synchronisation
     Speicherplatz sparen
     Kosten reduzieren
     Zeit sparen
```



```
SVN - subversion (Next Generation Open Source Version Control)

Open Source Versions Control System
Erste Ideen 2000. Abgeleitet von CVS. Beseitigt die
Unzulänglichkeiten von alten Systemen:
               Direktory Versionierung
               bessere Fileverwaltung (History)
               Metadatenverwaltung
               Verbesserter Netzwerkzugriffe
                     (Apache, SSH, separater Server)
               gut strukturiert
               gut verwaltbar
               sicherer als CVS.
Besteht aus einem Nutzerkommando und mehreren Administrations-
kommandos, Anbindung an Eclipse

Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato
Version Control with Subversion
O'Reilly 2004, ISBN 0-596-00448-6

Institut betreibt einen zentralen SVN-Server

       https://svn.informatik.hu-berlin.de/
```

Arbeitsweise von Versionsverwaltungssystemen:

Versionsverwaltungssysteme benutzen in der Regel ein Repository als Speicher für die Daten. Die Nutzer agieren grundsätzlich in Sandboxes, die Kopien des Repositories darstellen. Die Verwaltungssysteme organisieren den Datentransport zwischen Repository und Sandbox.

Initialisierung einer Datenbasis
    init

Füllen der Datenbasis mit einem Anfangszustand
    import

Ausgeben von Daten der Datenbasis an die Nutzer (Programmierer)
    checkout

Einfügen von Daten der Nutzer in die Datenbasis
    commit

Bilden einer neuen Version
    import

Aktualisieren der Datenbasis eines Nutzers
    update

Subversion Kommandos – Übersicht

```
svn
        Kommandozeilen Client-Programm
svnversion
        Programm zum Erzeugen eines Statusreports für eine Kopie
svnlook
        Tool zur Inspektion eines Subversion-Repository
svnadmin
        Tool zur Administration des Subversion-Repository
svndumpfilter
        Tool zur Erzeugung eines Dumpfiles
svnserve
        Standalone Server, auch mit SSH nutzbar

Hilfe:
    man svn
    svn help
    svn help <subcommando>
```

```
Arbeiten mit svn help
---------------------

svn help

Aufruf: svn UNTERBEFEHL [Optionen] [Parameter]
Geben Sie 'svn help UNTERBEFEHL' ein, um Hilfe zu einem Unterbefehl
zu erhalten.

Die meisten Unterbefehle akzeptieren Datei und/oder Verzeichnisparameter,
wobei die Verzeichnisse rekursiv durchlaufen werden. Wenn keine Parameter
angegeben werden, durchläuft der Befehl das aktuelle Verzeichnis rekursiv.

Verfügbare Unterbefehle:
   add
   blame (praise, annotate, ann)
   cat
   checkout (co)
   cleanup
   commit (ci)
   copy (cp)
   delete (del, remove, rm)
   diff (di)
   export
   help (?, h)
   import
   info
   list (ls)
   log
   merge
```



```
   mkdir
   move (mv, rename, ren)
   propdel (pdel, pd)
   propedit (pedit, pe)
   propget (pget, pg)
   proplist (plist, pl)
   propset (pset, ps)
   resolved
   revert
   status (stat, st)
   switch (sw)
   update (up)

Subversion ist ein Programm zur Versionskontrolle.
Für weitere Informationen siehe: http://subversion.tigris.org/
```

```
svn help update

update (up): Aktualisiert die Arbeitskopie mit Änderungen aus dem Projektarch
Aufruf: update [PFAD...]

  Ist keine Revision angegeben, wird die Arbeitskopie auf den aktuellen
  Stand der HEAD-Revision gebracht. Ansonsten wird die Arbeitskopie mit der
  durch -r angegebenen Revision synchronisiert.
  Für jedes aktualisierte Objekt wird eine Zeile mit einem Buchstaben für die
  Aktion ausgegeben. Diese haben die folgenden Bedeutungen

    A  Added    - Hinzugefügt
    D  Deleted  - Gelöscht
    U  Updated  - Aktualisiert
    C  Conflict - Konflikt
    G  merGed   - Zusammengeführt

  Ein Buchstabe in der ersten Spalte symbolisiert eine Aktualisierung der
  Datei, während Aktualisierungen der Dateieigenschaften in der zweiten Spalt
  angezeigt werden.

Gültige Optionen:
  -r [--revision] arg      : ARG (manche Befehle akzeptieren auch Wertebereic
                             Ein Revisions Parameter kann sein:
                                NUMBER       Revisionsnummer
                                "{" DATE "}" Revision zum Startdatum
                                "HEAD"       Neueste im Projektarchiv
                                "BASE"       Basisrevision der Arbeitskopie
                                "COMMITTED"  Letzte übertragene Revision bei
```



```
                                               oder vor BASE
                                "PREV"       Letzte Revision vor COMMITTED
  -N [--non-recursive]     : Nicht rekursiv hinabsteigen
  -q [--quiet]             : So wenig wie möglich ausgeben
  --diff3-cmd arg          : Verwende ARG als Merge Programm
  --username arg           : Benutzername ARG angeben
  --password arg           : Passwort ARG angeben
  --no-auth-cache          : Anmeldeinformation nicht zwischenspeichern
  --non-interactive        : Keine interaktiven Rückfragen ausgeben
  --config-dir arg         : Benutzerkonfigurationsdateien aus dem Verzeichni
```

```
Kurzfassung Subversion-Nutzung
------------------------------

Möglichkeiten des Zugriffs auf ein Repository für Nutzer:

    file:///pfad        - lokales Filesystem

    http://host/pfad    - Zugriff über http und WebDAV            *
                          über Apache-Server                      *
                                                                  *
    https://host/pfad   - Zugriff über http und WebDAV            *
                          über Apache-Server aber mit SSL         *

    svn://host/pfad     - Zugriff über svnserve-Server

    svn+ssh://host/pfad - Zugriff über ssh und svn,
                          wie svn aber über ssh-Tunnel
```



```
Arbeiten mit den Daten eines Repositories

mkdir Sandbox      # Sandbox erzeugen
cd Sandbox
# Füllen der Sandbox
svn checkout https://svn.informatik.hu-berlin.de/svn/unix-09/Einleitung  mein
ls mein
Makefile  README sysconf2.c  sysconf.c
cd mein
vi sysconf.c      # ändern
# Senden der Änderung
svn commit


Anzeigen:

svn ls https://svn.informatik.hu-berlin.de/svn/unix-09/Einleitung
svn log https://svn.informatik.hu-berlin.de/svn/unix-09/Einleitung


Im Browser unter

       https://svn.informatik.hu-berlin.de/svn/unix-09/Einleitung

anzeigbar.
```