

Um Hilfe zu einzelnen Befehlen zu erhalten

... koennen entweder

1. die man-pages aufgerufen weden, z.B. man ls gibt ausführliche Hilfe zum ls-Befehle
2. oder eine Kurzhilfe erhalten durch z.B. ls --help

Dateien

Typen

Typ	Beschreibung
Datei	<p>Im Englischen <i>regular, plain</i> oder auch <i>ordinary file</i> genannt. Technisch eine zeichen-adressierbare Zeichenfolge. Man unterscheidet traditionell:</p> <ul style="list-style-type: none">• Textdateien, deren Zeilenstruktur durch das Newline-Zeichen festgelegt wird. Mit <i>unbewaffneten Auge</i> lesbar.• Binärdateien (Programme, komprimierte Dateien, Bilder, ...), deren Inhalt erst in einem bestimmtem Kontext (Programmaufruf, Entpacker, Bildbetrachter) sichtbar wird.
Verzeichnis	<p>Im Englischen <i>directory</i> genannt. Verzeichnisse sind Dateien mit einem durch das System festgelegten und garantierten Inhalt, nämlich Informationen über die Dateien und Verzeichnisse, die in diesem Verzeichnis abgelegt sind. Jedes Verzeichnis hat mindestens zwei Einträge: Einmal einen Verweis auf sich selbst ».« und einmal einen Verweis auf das übergeordnete Verzeichnis »..«, womit die Baumstruktur zur Wurzel zurückverfolgt werden kann.</p>
Spezielle Dateien	<p><i>special files</i> sind für Otto Normaluser normalerweise ohne Bedeutung. Mit ihnen regelt das System den Zugriff auf Geräte und ermöglicht die Kommunikation von Programmen untereinander.</p> <ul style="list-style-type: none">• Gerätedateien in /dev.• <i>named pipes</i>: Benannte Datenkanäle zwischen Programmen.• <i>sockets</i>: Datenkanal zwischen Programmen auch über das Netzwerk.• symbolische Links: Verweise auf andere Dateien.

Namen

Dateinamen

- sind maximal 255 Zeichen lang,
- unterscheiden Groß- und Kleinbuchstaben,
- müssen innerhalb eines Verzeichnisses eindeutig sein,
- sollten keine Sonderzeichen enthalten. Schon Umlaute sind problematisch, aber die Verwendung von

/ " ' * ; - & ? () [] ~ ! \$ { } > < # @

nennt man *asking for trouble*, da diese Zeichen in der Shell eine reservierte Bedeutung haben und damit zu Problemen beim Kommandieren von Linux-Befehlen führen.

Verzeichnisse

Begriffe

Begriff	Erklärung
Arbeitsverzeichnis	Das Verzeichnis, in dem Sie sich gerade befinden. Mit <i>cd</i> (<i>change directory</i>) können Sie das Arbeitsverzeichnis ändern und mit <i>pwd</i> (<i>print working directory</i>) anzeigen lassen, in welchem Verzeichnis Sie sich gerade befinden.
Home-Verzeichnis	Das Verzeichnis, in dem Sie sich nach dem Anmelden befinden.
.	Das Arbeitsverzeichnis
..	Das übergeordnete Verzeichnis (<i>parent directory</i>)
~	Das eigene Home-Verzeichnis
~user	Das Home-Verzeichnis des Nutzers <i>user</i>
--	Nach einem Verzeichniswechsel das vorherige Arbeitsverzeichnis

Pfad

Die Verzeichnisfolge von einem Verzeichnis im Dateisystem zu einer bestimmten Datei nennt man Pfad. Man unterscheidet:

relative Pfadnamen	Der Startpunkt der Verzeichnisfolge ist das aktuelle Verzeichnis, z. B. ../../g/grote/.profile. Syntaktisch beginnt ein relativer Pfadname ohne Schrägstrich »/«.
absolute Pfadnamen	Der Startpunkt der Verzeichnisfolge ist das root-Verzeichnis, z. B. /dfs/u/g/grote/.profile

Ein Pfadname darf nicht länger als 1023 Zeichen sein.

Datei-Information anzeigen: »ls«

ls [option ...] [datei ...]

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus
-l	Ausführlich (long)
-d, --directory	Gibt bei Verzeichnissen den Eintrag, nicht den Inhalt aus
-a, --all	Gibt auch versteckte (.-) Dateien aus
-t	Sortiert nach der Änderungszeit
-R, --recursive	Zeigt auch Unterverzeichnisse an

Zum Beispiel ergibt *ls -l* / auf meiner Workstation:

```
$ ls -l /
total 37
drwxr-xr-x  2 root  svstem   2944 Feb 21 15:06 bin
drwxr-xr-x  3 root  svstem   1024 Feb 20 16:34 boot
lrwxrwxrwx  1 root  svstem    12 Feb 28 12:47 cdrom -> /media/cdrom
drwxr-xr-x 18 root  svstem   2016 Feb 28 11:05 dev
drwxr-xr-x  2 root  svstem    256 Feb 20 14:52 dfs
drwxr-xr-x 43 root  system   6752 Feb 28 11:05 etc
```

```

lrwxrwxrwx    1 root    system      13 Feb 28 12:47 floppy -> /media/floppy
drwxr-xr-x    3 root    system      288 Feb 20 15:21 home
drwxr-xr-x    6 root    system     2720 Feb 21 15:11 lib
drwxr-xr-x    2 root    system      256 Feb 20 15:05 lost+found
drwxr-xr-x    5 root    system      352 Feb 20 15:21 media
drwxr-xr-x    2 root    system      256 Sep 22 15:08 mnt
drwxr-xr-x   13 root    system      608 Feb 20 15:17 opt
drwxr-xr-x    2 root    system      256 Feb 20 15:05 proc
drwx-----  10 root    system     1088 Feb 28 11:05 root
drwxr-xr-x    4 root    system     7456 Feb 21 15:06 sbin
-rw-r--r--    1 root    system        0 Feb 28 11:05 success
drwxrwxrwt    4 root    system     1376 Feb 28 16:40 tmp
drwxr-xr-x   17 root    system     1024 Feb 25 11:29 usr
drwxr-xr-x   16 root    system      736 Feb 20 15:15 var
lrwxrwxrwx    1 root    system      10 Feb 28 13:05 zip -> /media/zip

```

Arbeitsverzeichnis wechseln: »cd«

cd [*verzeichnis*]

Wie schon erwähnt, ändert cd das Arbeitsverzeichnis. Ohne Argument aufgerufen, wird das Home-Verzeichnis zum Arbeitsverzeichnis. cd ist genau wie pwd ein Kommando, das in der Shell eingebaut ist.

Verzeichnis anlegen: »mkdir«

mkdir (*make directory*) legt ein Verzeichnis an.

mkdir [*option*] *verzeichnis* ..

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus
-p, --parents	Legt falls nötig auch Verzeichnisse auf dem Pfad an

Verzeichnis löschen: »rmdir«

rmdir (*remove directory*) löscht ein Verzeichnis, wenn es leer ist.

rmdir [*option*] *verzeichnis* ..

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus

Dateien löschen: »rm«

rm (*remove*) löscht Dateien und Verzeichnisse. Letztere auch dann, wenn sie nicht leer sind.

rm [*option*] *datei* ..

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus
-i, --interactive	Frägt vor dem Löschen
-f, --force	Löscht ohne Nachfrage
-r, -R, --recursive	Löscht auch Unterverzeichnisse

Ohne Optionen löscht rm keine schreibgeschützten Dateien.

Dateien kopieren: »cp«

cp (*copy*) kopiert Dateien. Je nach Befehls-Schreibweise oder verwendeter Befehls-Option kann man eine Datei, viele Dateien oder Datei-Teilbäume, also rekursiv, kopieren. Besondere Sorgfalt ist nötig für ungewöhnliche Dateien wie Gerätedateien, symbolische Links, usw.

Ist das letzte Befehlsargument ein Verzeichnis, so wird die Datei (*oder*: werden die Dateien) in das Verzeichnis kopiert. Ist das letzte Befehlsargument dagegen kein Verzeichnis, so darf der Befehl nur zwei Datei-Namen als Argumente haben. Soll ein Verzeichnis kopiert werden, so ist die Option »-R« anzugeben.

cp *[option]* *alt neu*

cp *[option]* *datei .. verzeichnis*

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus
-R, --recursive	Rekursiv
-p, --preserve	Attribute wenn möglich erhalten
-d, --no-dereference	Verweise erhalten
-a, --archive	-dpR

Dateien bewegen/umbenennen: »mv«

mv (*move*) bewegt Dateien oder benennt sie um. Das "oder" des letzten Satzes ist nicht-exklusiv.

Prinzipiell werden Dateien im Dateibaum von einer Stelle zur anderen bewegt, indem die Daten unter gleichem Namen kopiert und die Quelle anschließend gelöscht wird. Aus der Besprechung der I-Nodes wissen wir, dass dieses Bild manchmal nicht stimmt. Dateien können beim Bewegen auch ihren Namen ändern, insbesondere auch dann, wenn sich der Ort im Dateibaum nicht ändert. Und damit ist klar, warum die Aktion "Datei umbenennen" eine Teilmenge der Aktion "Datei bewegen" ist, und warum es keinen eigenen Befehl zum Umbenennen einer Datei gibt.

Wie beim cp-Befehl wird durch die Anzahl und den Typ der Argumente der Verarbeitungs-Modus bestimmt. Bei zwei Argumenten wird die erste Datei mit dem zweiten Namen umbenannt. Bei mehr als zwei Argumenten muss das letzte Argument ein Verzeichnis sein. Die vorher auf der Befehlszeile angegebenen Dateien werden in dieses Verzeichnis bewegt.

mv *[option]* *alt neu*

mv *[option]* *datei .. verzeichnis*

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus

Die "magische" Tab-Taste

In Microsoft-Produkten hieße sie wahrscheinlich "Do what I mean"-Taste und würde als jüngstes Beispiel künstlicher Intelligenz gefeiert. 8-)

In Wirklichkeit hilft sie beim Schreiben von Befehlen, indem sie, wenn möglich, den Befehl zu Ende schreibt. Sie weiß, wo syntaktisch ein Befehl oder ein Dateiname stehen könnte und gibt passende Vorschläge zur Vervollständigung.

Pager: »less«

Um sich Dateien anzusehen, die länger als eine Bildschirmseite sind, verwendet man less. Dieser Pager gibt jeweils eine Bildschirmseite aus und wartet dann auf einen Befehl. less wird auch zur Anzeige der Hilfeseiten bei man benutzt. Ein nicht ganz so leistungsfähiger Pager ist more.

less [option] [datei] ..

Befehl	Bedeutung
h	Gibt eine Onlinehilfe aus
<space>, f	Blättert eine Seite weiter
b	Blättert eine Seite zurück
<enter>, <Pfeil nach unten>	Blättert eine Zeile weiter
<Pfeil nach oben>	Blättert eine Zeile zurück
/muster , ?muster	Sucht das Muster vorwärts bzw rückwärts
n, N	Sucht das nächste bzw. vorige Vorkommen des Suchmusters
q	Beendet das Programm

Dateien (ent)komprimieren: »gzip«

gzip komprimiert eine Datei, die nach der Komprimierung die Namensweiterung »gz« bekommt. Im Unterschied zu entsprechenden Windows-Programmen werden hier nicht mehrere Dateien in einem Archiv komprimiert gesammelt.

gzip [option] [datei]

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus
-v, --verbose	Ausführliche Ausgaben
-d, --decompress	Entkomprimiert (auch gunzip)
-c, --stdout	Schreibt auf die Standardausgabe (auch gzcata)

Dateien archivieren: »tar«

Der Name tar (*tape archive*) kommt noch aus einer Zeit, als man ausschließlich auf Magnetbänder archivierte. Auch heute archiviert man noch auf Magnetbänder, aber nicht mehr mit diesem Kommando. Heute dient tar hauptsächlich dazu, aus einem Teilbaum eine sequentielle, häufig auch komprimierte Datei zu erzeugen. Genutzt wird dies zur Software-Verteilung im Internet. Der Empfänger eines solchen tar-Archivs entpackt den Teilbaum wieder und installiert aus diesem die Software. Die Kombination "tar + Komprimierer" entspricht funktional den Windows-"Zippern".

Obwohl tar sehr viele Optionen hat, braucht man im täglichen Umgang nur sehr wenige. Häufig will man ein komprimiertes tar-Archiv einfach nur auspacken und anschließend den Inhalt, wenn es sich etwa um Software handelt, installieren. Das »Auspacken« erfolgt in zwei Schritten:

Schritt 1: Man lässt sich das Inhaltsverzeichnis (Option: »t«) des Archivs anzeigen um festzustellen, ob das Archiv ein Top-Level-Verzeichnis bereits enthält. Das ist häufig, aber nicht immer der Fall. Dann wird nämlich das gesamte Archiv in dieses Verzeichnis entpackt und in das aktuelle Verzeichnis kopiert. Fehlt dieses Verzeichnis, so legt man es an und wechsle mit dem Arbeitsverzeichnis dorthin. Anhand des Archiv-Namens kann man die Komprimierungsmethode erkennen: Ein »tgz« steht für gzip-te und ein »tbz2« steht für bzip2-te Dateien. Das Kommando etwa für das Archiv »xawtv.tgz« lautet:

```
tar -ztf xawtv.tgz
```

Schritt 2: Nun wird das Archiv ausgepackt (Option: »x«):

```
tar -zxvf xawtv.tgz
```

Die Option »v« sorgt für Fortschrittsmeldungen auf dem Bildschirm.

tar [option] [datei ..]

Option	Bedeutung
--help	Gibt eine Liste aller Optionen aus
-v, --verbose	Ausführliche Ausgaben
-c, --create	Archiv erzeugen
-t, --list	Inhalt anzeigen
-x, --extract	Archiv auspacken
-f file, --file=file	<i>file</i> ist der Name der Archivdatei (- bedeutet Standardausgabe)
-z, --gzip, --gunzip	Archiv mit gzip (ent)komprimieren
-j, --bzip2, --bunzip2	Archiv mit bzip2 (ent)komprimieren