

# Praktische Datenverarbeitung

- Erste Schritte mit Linux (Debian)
  - Anmelden und „root“ werden
  - Paketverwaltung „apt“
  - Informationen und Hilfe
  - Umgang mit Dateien und Verzeichnissen
  - Platzhalter
  - Meta- und Sonderzeichen
  - Umlenkung und Pipes
  - Aliase
  - Dateien in \$HOME
  - Benutzer und Gruppen anlegen und bearbeiten
    - Identität wechseln
    - „Multiuser“-System

---

Erste Schritte mit Linux  
Aufwand etwa 4-6 UE + Übung

Das ist ein erster Einstieg in die Benutzung von Linux an der Kommandozeile.

- Halten Sie Ihr System auf dem aktuellen Stand
- Finden Sie Hilfe und Anleitung
- Die ersten Befehle und Kommandos
- Legen Sie Benutzer und Gruppen an
- Üben, üben, üben :-)

# Debian - Anmeldung



14.11.23Horst Bräuner2

In der Regel melden Sie sich als „normaler“ Benutzer an und wechseln in die Rolle des Systemadministrators „root“ nur bei Bedarf. Damit ist die Gefahr, dass Sie unabsichtlich systemrelevante Dateien, Dienste und Eigenschaften ändern / verschieben / löschen geringer.

Das gilt nicht nur für die grafische Anmeldung. Auch an der Kommandozeile sollten Sie nicht von vornherein als privilegierter Benutzer arbeiten.

Wenn Sie sich in der grafischen Umgebung angemeldet haben, können Sie die Kommandozeile, die shell, über das Menü „Anwendungen“ -> „Terminal“ (oder „Anwendungen -> Zubehör -> Terminal“) aufrufen.



# Linux – privilegierter User „root“

```
hbraeuner@tardis:~$
hbraeuner@tardis:~$ su -
Passwort:
root@tardis:~#
```

Anmeldung als „root“ vermeiden  
mit su – zum Administrator machen

14.11.23
Horst Bräuner
3

Wenn Sie privilegiert arbeiten müssen, wechseln Sie Ihre Rolle. An der Kommandozeile können Sie sich mit dem „switch user“-Kommando „su“ zum „root“ (oder jedem anderen Benutzer) machen. Das Kommando kann mit Parametern aufgerufen werden. Hilfreich ist der Parameter „-“. Damit vermeiden Sie es, Einstellungen des normalen Benutzers mit in die privilegierte Umgebung zu nehmen.

Eine ausführliche Erklärung liefert die Hilfe zu dem Kommando:

**SU - ...**

... startet die Shell als Anmelde-Shell mit einer Umgebung ähnlich zu einer realen Anmeldung:

- löscht alle Umgebungsvariablen außer TERM und jene, die durch --whitelist-environment angegeben wurden.
- initialisiert die Umgebungsvariablen HOME, SHELL, USER, LOGNAME und PATH.
- wechselt in das Home-Verzeichnis des Zielbenutzers.
- setzt argv[0] der Shell auf »-<, um die Shell zur Anmelde-Shell zu machen.

Kurz gesagt, verhindern Sie durch „su -“ an Stelle des einfachen „su“, dass Ihnen Einstellungen unter geschoben werden, die möglicherweise schädlich sind. Mehr ist für die Praxis nicht relevant.



## Linux – Paketverwaltung (1)

- **Advanced Packaging Tool (APT)**
  - ~ Software – Pakete - Verwaltungstool
  - setzt auf dpkg (**D**ebian **P**ackage) auf
- Konfiguration in /etc/apt/sources.list
  - Paketquellen
  - CD / DVD / Disk / Verzeichnis / HTTP / FTP / ...
- Pakete im „Debian“ (.deb) Format
- Frontends
  - apt
  - apt-get, apt-cache, ...
  - aptitude (Text)
  - synaptic (Gnome, lxde, xfce ~ GUI)
  - adept (KDE ~ GUI)
- apt-proxy / apt-mirror
  - eigener mirror für Pakete
  - entlastet Internet-Netzwerkverkehr
  - ca. 200 - 300 GB Platzbedarf, je nach gespiegelten Quellen und Paketen
  - Proxy überwacht Paketstruktur und Aktualität

siehe auch Artikel: <https://wiki.ubuntuusers.de/apt/apt/>


Software wird in „.deb“-basierten Linuxen normalerweise über das Advanced Packaging Tools „APT“ verwaltet. Dieses setzt auf dpkg auf und ergänzt es. Das Tool wird über einfache Textdateien konfiguriert. Das sind Listen mit URLs von Software-Archiven, vergleichbar mit den Playstores von Mobilgeräten.

Es gibt für APT verschiedene Frontends. Grafische Varianten sind zum Beispiel „synaptic“ oder „adept“. Synaptic haben Sie bei der grafischen Installation automatisch mit installiert.

siehe auch Artikel: <https://wiki.ubuntuusers.de/apt-mirror/>

Sie haben bei der Installation einen Spiegelservers („Mirror“) im Internet angegeben. Mit „apt-proxy“ können Sie einen solchen Spiegel zu sich selbst holen. Wenn Sie dann weitere Linux(e) installieren, geben Sie als Spiegel Ihren eigenen Server an. Damit sind Sie nicht auf eine dauerhafte Internet-Verbindung angewiesen.

In der Praxis werden Sie mehrere solcher Spiegel verwenden. Der erste Spiegel zieht sich automatisch die aktuelle Software aus dem Internet. Einen Abzug davon nutzen Sie zum Testen von Installationen. Es könnte ja sein, dass sich irgendwo eine Fehlfunktion in ein Stück Software eingeschlichen hat. Einen fehlerfreien Abzug dieses Abzugs nutzen Sie dann, um Ihre Linux(e) sicher zu installieren.

## Linux – Paketverwaltung (2)

```
root@host# apt-get
root@host#
```

apt - Paketverwaltung

```
root@host# apt-cache search [Paket]
root@host#
```

apt search [Paket]

```
root@host# apt-get install man
root@host#
root@host# apt-get install makepasswd
root@host#
```

apt install [Paket]

```
root@host#
root@host# apt-get install man makepasswd
root@host#
```

Beispiele, sofern  
„man, makepasswd,  
aptitude, ...“ noch  
nicht installiert

```
root@host# apt-get install aptitude
```

aptitude - Paketverwaltung

```
root@host# aptitude
root@host# aptitude search [Paket]
root@host# aptitude install [Paket]
root@host# aptitude install man makepasswd
```

14.11.23

Horst Bräuner

5

Die wichtigsten Kommandos zum Umgang mit Paketen.

**install** ... installiert ein Paket, sofern es noch nicht installiert ist

**remove** ... löscht ein Paket, sofern es vorhanden ist

**purge** ... löscht ein Paket und entfernt alle relevanten Einstellungen dazu

**autoremove** ... löscht Pakete, die nicht mehr benötigt werden. Entweder weil sie veraltet sind oder mit anderen Paketen installiert wurden, die nicht mehr vorhanden sind, z.B. deinstalliert wurden.

Beim Installieren prüft APT, ob ein Paket von anderen Paketen abhängig ist, zum Beispiel von Bibliotheken („Libraries“), die von mehreren Programmen genutzt werden. Ist dies der Fall, versucht APT die abhängigen Pakete ebenfalls zu installieren.

Ausführliches Kapitel:

<https://www.debian.org/doc/manuals/debian-reference/ch02.de.html>

## Linux – Paketverwaltung (3)



```
root@deb18adt:~# apt update
Holen:1 http://security.debian.org/debian-security stretch/updates InRelease [94,3 kB]
Ign:2 http://ftp.de.debian.org/debian stretch InRelease
Holen:3 http://ftp.de.debian.org/debian stretch-updates InRelease [91,0 kB]
OK:4 http://ftp.de.debian.org/debian stretch Release
Holen:5 http://security.debian.org/debian-security stretch/updates/main Sources [199 kB]
Holen:6 http://security.debian.org/debian-security stretch/updates/main amd64 Packages [485 kB]
Holen:7 http://security.debian.org/debian-security stretch/updates/main Translation-en [215 kB]
Es wurden 1.085 kB in 3 s geholt (345 kB/s).
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen.... Fertig
Aktualisierung für 4 Pakete verfügbar. Führen Sie »apt list --upgradable« aus, um sie anzuzeigen.
root@deb18adt:~#
```

update = Listen  
neuer/aktualisierbarer Pakete  
herunterladen

14.11.23Horst Bräuner6

Weitere Funktion von APT ist es, Pakete auf dem aktuellen Stand zu halten. Als Referenz dienen die Spiegelservers, die in der Konfiguration von APT eingetragen sind.

**apt update** ... Liste verfügbarer Pakete aktualisieren

**apt upgrade** ... das System durch Installation/Aktualisierung der Pakete hochrüsten

**apt full-upgrade (apt dist-upgrade)** ... das System durch Entfernung / Installation / Aktualisierung der Pakete vollständig hochrüsten / aktualisieren




## Linux – Paketverwaltung (4)



```
root@deb18adt:~# apt dist-upgrade
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut.
Statusinformationen werden eingelesen... Fertig
Paketaktualisierung (Upgrade) wird berechnet... Fertig
Die folgenden Pakete werden aktualisiert (Upgrade):
  ghostscript libgs9 libgs9-common libpng16-16
4 aktualisiert, 0 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.
Es müssen 7.727 kB an Archiven heruntergeladen werden.
Nach dieser Operation werden 2.048 B Plattenplatz freigegeben.
Möchten Sie fortfahren? [J/n] J
Holen:1 http://security.debian.org/debian-security stretch/updates/main amd64 libpng16-16 amd64 1.
6.28-1+deb9u1 [280 kB]
Entpacken von ghostscript (9.26a-dfsg-0+deb9u2) über (9.26a-dfsg-0+deb9u1) ...
Vorbereitung zum Entpacken von .../libgs9_9.26a-dfsg-0+deb9u2_amd64.deb ...
Entpacken von libgs9:amd64 (9.26a-dfsg-0+deb9u2) über (9.26a-dfsg-0+deb9u1) ...
Vorbereitung zum Entpacken von .../libgs9-common_9.26a-dfsg-0+deb9u2_all.deb ...
Entpacken von libgs9-common (9.26a-dfsg-0+deb9u2) über (9.26a-dfsg-0+deb9u1) ...
libgs9-common (9.26a-dfsg-0+deb9u2) wird eingerichtet ...
libpng16-16:amd64 (1.6.28-1+deb9u1) wird eingerichtet ...
Trigger für libc-bin (2.24-11+deb9u4) werden verarbeitet ...
libgs9:amd64 (9.26a-dfsg-0+deb9u2) wird eingerichtet ...
ghostscript (9.26a-dfsg-0+deb9u2) wird eingerichtet ...
Trigger für libc-bin (2.24-11+deb9u4) werden verarbeitet ...
root@deb18adt:~#
```

dist-upgrade = Upgrade (Paketaktualisierung)  
für die komplette Distribution durchführen,  
siehe apt-get(8)

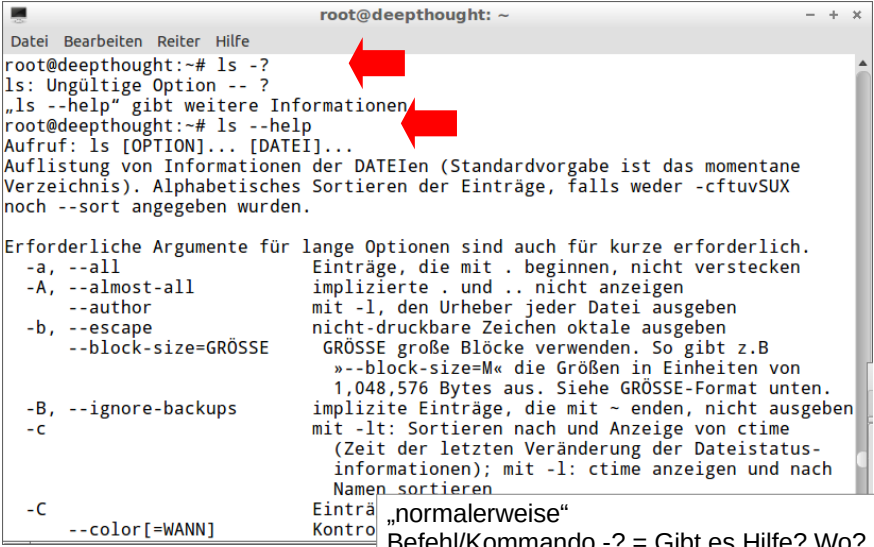
Hier das Beispiel für ein „dist-upgrade“ bzw. „full-upgrade“.



---

## Linux – Informationen und Hilfe



```

root@deeptthought: ~
Datei Bearbeiten Reiter Hilfe
root@deeptthought:~# ls -?
ls: Ungültige Option -- ?
„ls --help“ gibt weitere Informationen
root@deeptthought:~# ls --help
Aufruf: ls [OPTION]... [DATEI]...
Auflistung von Informationen der DATEIen (Standardvorgabe ist das momentane
Verzeichnis). Alphabetisches Sortieren der Einträge, falls weder -cftuvSUX
noch --sort angegeben wurden.

Erforderliche Argumente für lange Optionen sind auch für kurze erforderlich.
-a, --all                Einträge, die mit . beginnen, nicht verstecken
-A, --almost-all        implizierte . und .. nicht anzeigen
--author                mit -l, den Urheber jeder Datei ausgeben
-b, --escape            nicht-druckbare Zeichen oktale ausgeben
--block-size=GRÖSSE     GRÖSSE große Blöcke verwenden. So gibt z.B
                        »--block-size=M« die Größen in Einheiten von
                        1,048,576 Bytes aus. Siehe GRÖSSE-Format unten.
-B, --ignore-backups    implizite Einträge, die mit ~ enden, nicht ausgeben
-c                      mit -lt: Sortieren nach und Anzeige von ctime
                        (Zeit der letzten Veränderung der Dateistatus-
                        informationen); mit -l: ctime anzeigen und nach
                        Namen sortieren
-C                      „normalerweise“
--color[=WANN]          Einträ
                        Kontro

```

„normalerweise“  
 Befehl/Kommando -? = Gibt es Hilfe? Wo?  
 Befehl/Kommando --help = Anleitung und Optionen


14.11.23
Horst Bräuner
8

Hilfe und Informationen zu den einzelnen Kommandos gibt es meist durch die Parameter `-?` `--?` `--help` nach dem Kommando / Befehl. Da Linux gemeinschaftlich von vielen Entwicklern gestaltet wird, ist es nicht durchgehend gleich, welcher Parameter zur Hilfe führt. Das bedeutet, dass Sie manchmal mehrere Varianten ausprobieren müssen.


Falls Sie bei der Installation „deutsch“ als Systemsprache eingestellt und installiert haben, ist ein großer Teil der Hilfe ebenfalls auf deutsch übersetzt. Das erleichtert Ihnen den Einstieg.

Im obigen Beispiel sehen Sie das Kommando „ls“ (Liste der Dateien im aktuellen Verzeichnis) mit der Variante `-?`, die nicht zur Hilfe führt, und die Variante `--help`, die den Befehl „ls“ und seine Parameter erklärt.






---



# Linux – Information und Hilfe



root@deepthought: ~

Datei Bearbeiten Reiter Hilfe

LS(1) User Commands LS(1)

**NAME**

ls - list directory contents

**SYNOPSIS**

ls [OPTION]... [FILE]...

**DESCRIPTION**

List information about the FILES (the current directory by default). Sort entries alphabetically if none of **-cftuvSUX** nor **--sort** is specified.

Mandatory arguments to long options are mandatory for short options too.

**-a, --all**  
do not ignore hidden files

**-A, --almost-all**  
do not ignore hidden files, **./** and **..**

**--author**

Manual page ls(1) line 1

Manual Pages = Handbuch

man Befehl/Kommando = Handbuch zum Befehl/Kommando

man man = Wie funktionieren die Handbücher

Paket ggf. mit

user@host# apt install man

installieren

14.11.23
Horst Bräuner
9

Zusätzlich gibt es zu den meisten Kommandos / Befehlen Handbücher bzw. Anleitungen sogenannte „Manual Pages“ oder man-pages.

**man-pages** werden mit dem Kommando „man“ gefolgt von dem Befehl, zu dem man das Handbuch wünscht, aufgerufen. Der Befehl „man man“ liefert sogar das Handbuch zum Handbuch.

Innerhalb eines Handbuchs können Sie mit den Pfeiltasten oder den BildAuf/BildAb Tasten blättern. Zusätzliche Hilfe erhalten Sie durch drücken von „h“. Die Anzeige des Handbuchs beenden Sie durch drücken von „q“ (für „quit“). Lassen Sie sich zur Übung ein paar Handbuchseiten anzeigen. Beispielsweise

```
user@host:~$ man apt
user@host:~$ man ls
user@host:~$ man cp
user@host:~$ man more
user@host:~$ man less
user@host:~$ man grep
```



## Linux – Informationen und Hilfe

```
user@host# man [Befehl] [Abschnitt]
user@host# man hdparm 8
user@host# man bash 1
```

ohne Angabe des Abschnitts  
wird Abschnitt 1 angezeigt

### Manual Pages - Abschnitte

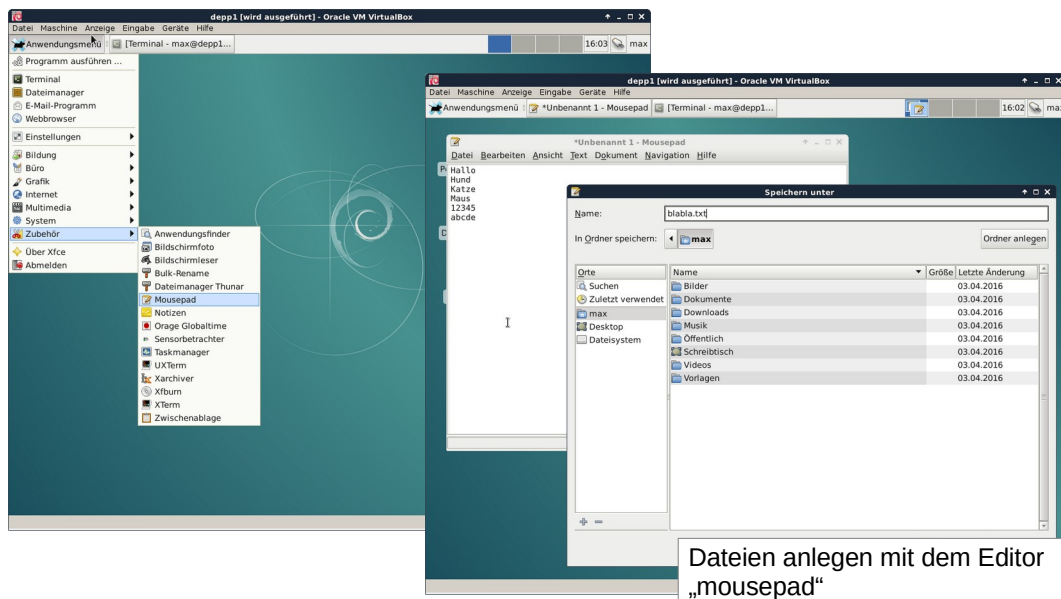
Abschnitt 1: Nutzerbefehle und Kommandos  
Abschnitt 2: Systemaufrufe  
Abschnitt 3: Subroutinen und Bibliotheksaufrufe  
Abschnitt 4: Dateiformate (BSD: Treiber- und Netzwerk-Schnittstellen)  
Abschnitt 5: Header, Tabellen und Makropakete (BSD: Dateiformate)  
Abschnitt 6: Spiele und Demosoftware  
Abschnitt 7: Treiber- und Netzwerk-Schnittstellen (BSD: Header, Tabellen und Makropakete)  
Abschnitt 8: BSD: Systemverwaltung  
Abschnitt 9: Kernel-interne Schnittstellen  
Abschnitt n: Neu (wird z. B. von Tcl/Tk verwendet)

Die Einteilung der Abschnitte ist  
NICHT verbindlich und variiert  
von Distribution zu Distribution

Die Handbücher sind in Abschnitte unterteilt. Grundsätzlich sollten die Handbuchseiten den oben genannten Abschnitten folgen. Da gemeinschaftlich entwickelt wird, ist das leider nicht konsequent durchgehalten. Zudem ist die Trennung oft schwer, da Kommandos sehr viele Funktionen in einem Befehl vereinen können.

Eine Liste nützlicher Befehle haben Sie erhalten. Rufen Sie die Handbücher dazu auf und experimentieren Sie mit einfachen „ls“ Befehlen.

# Dateien und Verzeichnisse - Editor



Dateien anlegen mit dem Editor „mousepad“

Wie erwähnt, werden Konfigurationen in einfachen Textdateien abgelegt. Eine Textverarbeitung ist eher ungeeignet, um diese Dateien zu bearbeiten. Besser geeignet ist beispielsweise „Mousepad“ als grafischer Editor. In Ihrer grafischen Installation finden Sie ihn unter „Zubehör“.


Legen Sie zur Übung ein paar Textdateien an.



---



## Dateien und Verzeichnisse - „nano“



GNU nano 2.7.4
Neuer Puffer
Verändert

Haupt-Hilfe für Nano

Nano wurde konzipiert, die Funktionalität und die Benutzerfreundlichkeit des UW-Pico-Texteditors zu imitieren. Es gibt vier Hauptbereiche: Die Titelzeile zeigt die Version des Programms, den Namen der momentan editierten Datei und ob die Datei verändert wurde oder nicht. Das Hauptfenster enthält die zu bearbeitende Datei. Die Statuszeile (die dritte Zeile von unten) zeigt wichtige Meldungen. Die untersten zwei Zeilen listen die meistgebrauchten Tastenkombinationen von Nano auf.

Tastenkombinationen werden wie folgt abgekürzt: Kombinationen mit der Strg-Taste werden mit einem ^ ausgedrückt und können auch eingegeben werden, indem Esc zwei Mal gedrückt wird. Escape-Sequenzen werden mit dem Meta-Symbol (M) angegeben und können mit Alt, Cmd oder Meta eingegeben werden, abhängig von Ihrer Tastatureinstellung. Zudem kann durch zweimaliges Drücken von Esc und anschließender Eingabe einer dreistelligen Zahl von 000 bis 255 das Zeichen mit dem entsprechenden Wert eingegeben werden. Die folgenden Tasten(-kombinationen) sind im Hauptfenster verfügbar. Alternative Tasten stehen in Klammern:

|   |   |
|---|---|
| ^G (F1)      Diese Hilfe anzeigen<br>^X (F2)      Aktuellen Dateipuffer schließen / Nano beenden<br>^O (F3)      Datei speichern<br>^R (F5)      Weitere Datei in die aktuelle Datei einfügen | Dateien anlegen mit dem Editor „nano“<br>root@host:~# nano<br>^ = <span style="border: 1px solid black; padding: 2px 5px;">Strg</span> / <span style="border: 1px solid black; padding: 2px 5px;">Ctrl</span> |
|---|---|

|                              |                                 |                                 |
|------------------------------|---------------------------------|---------------------------------|
| ^X Beenden<br>^L Auffrischen | ^P Zeile zurück<br>^N Zeile vor | ^Y Seite zurück<br>^V Seite vor |
|------------------------------|---------------------------------|---------------------------------|

14.11.23
Horst Bräuner
12

Ein weiterer Editor für die Kommandozeile ist „nano“. Rufen Sie den Editor einfach an der Kommandozeile mit „nano“ auf. Bei deutscher Systemsprache erhalten Sie zu Beginn eine kurze Anleitung. Steuern Sie den Editor indem Sie die „Steuerung“-Taste STRG (Engl. Tastatur CTRL) und den entsprechenden Buchstaben gleichzeitig drücken.

Heutzutage ist „nano“ auf fast allen Linux-Systemen installiert.

[https://de.wikipedia.org/wiki/Nano\\_\(Texteditor\)](https://de.wikipedia.org/wiki/Nano_(Texteditor))

Im weiteren Verlauf erhalten Sie noch eine Beschreibung des Editors „vi“.

# Umgang mit Dateien und Verzeichnissen

(1)

```
Terminal - horst@charon: ~/meinedaten
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe
horst@charon:~$ pwd
/home/horst
horst@charon:~$ mkdir meinedaten
horst@charon:~$ cd meinedaten/
horst@charon:~/meinedaten$ ls
horst@charon:~/meinedaten$ touch datei1.txt
horst@charon:~/meinedaten$ touch datei2.txt
horst@charon:~/meinedaten$ touch datei21.txt
horst@charon:~/meinedaten$ touch datei22.txt
horst@charon:~/meinedaten$ ls
datei1.txt datei21.txt datei22.txt datei2.txt
horst@charon:~/meinedaten$ cat datei2.txt
dieser
Text
ist
nicht
sortiert
horst@charon:~/meinedaten$ cp datei1.txt datei3.txt
horst@charon:~/meinedaten$ ls
datei1.txt datei21.txt datei22.txt datei2.txt datei3.txt
horst@charon:~/meinedaten$
```

| Kommando | Wirkung  |
|----------|--|
| pwd      | Zeige aktuelles Verzeichnis  |
| mkdir    | ein Verzeichnis anlegen  |
| cd       | in ein Verzeichnis wechseln  |
| ls       | Inhalt eines Verzeichnisses anzeigen                                   |
| touch    | eine leere Datei anlegen oder das Datum einer bestehenden Datei ändern |
| cat      | Inhalt einer Datei „am Stück“ anzeigen                                 |
| cp       | eine Datei kopieren  |

14.11.23

Horst Bräuner

13

Legen Sie für einfache Übungen ein paar Dateien mit nano oder mousepad an.

Beispielsweise, wie oben dargestellt, eine Datei datei2.txt mit dem Inhalt „dieser Text ist nicht sortiert“, jedes Wort in einer anderen Zeile.

Legen Sie zusätzlich eine sehr große Datei an, deren Inhalt über mehrere (Bildschirm-)Seiten läuft. Sie können diese Datei gerne durch „cut & paste“ eines größeren Textes erzeugen. Oder nutzen Sie das Kommando

```
user@host:~$ man man > Handbuch.txt
```

um ein Handbuch zu den Handbüchern zu erstellen.

Mit den oben genannten Befehlen haben Sie schon die ersten Werkzeuge parat.

Sie müssen diese Werkzeuge nicht auswendig oder „blind“ bedienen können. Sie sollten für die Praxis nur wissen, dass es diese Werkzeuge gibt und was „man“ damit anfangen kann.

## Umgang mit Dateien und Verzeichnissen (2)

```
Terminal - horst@charon: ~/meinedaten
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe
horst@charon:~/meinedaten$ ls
datei1.txt datei21.txt datei22.txt datei2.txt datei3.txt datei4.txt
horst@charon:~/meinedaten$ ln -s datei3.txt bla.bla
horst@charon:~/meinedaten$ ls
bla.bla datei21.txt datei2.txt datei4.txt
datei1.txt datei22.txt datei3.txt
horst@charon:~/meinedaten$ ls -la
insgesamt 24
drwxrwxr-x 2 horst horst 4096 Apr 12 20:11 .
drwxr-xr-x 38 horst horst 4096 Apr 12 20:09 ..
lrwxrwxrwx 1 horst horst 10 Apr 12 20:11 bla.bla -> datei3.txt
-rw-rw-r-- 1 horst horst 0 Apr 12 19:50 datei1.txt
-rw-rw-r-- 1 horst horst 0 Apr 12 19:50 datei21.txt
-rw-rw-r-- 1 horst horst 31 Apr 12 19:51 datei22.txt
-rw-rw-r-- 1 horst horst 648 Apr 12 19:52 datei2.txt
-rw-rw-r-- 1 horst horst 29 Apr 12 20:01 datei3.txt
-rw-rw-r-- 1 horst horst 23 Apr 12 20:09 datei4.txt
horst@charon:~/meinedaten$ wc datei3.txt
1 6 29 datei3.txt
horst@charon:~/meinedaten$ wc -l datei3.txt
1 datei3.txt
horst@charon:~/meinedaten$
```

| Kommando | Wirkung   |
|----------|---|
| more     | Inhalt einer Datei „scheibchenweise“ anzeigen     |
| less     | Inhalt einer Datei seitenweise anzeigen           |
| ln       | eine Verknüpfung erstellen                        |
| wc       | Zeilen, Wörter oder Zeichen in einer Datei zählen |

Weitere Werkzeuge zum anzeigen sind „more“ und „less“. Mit „more“ zeigen Sie den Inhalt einer Datei Seite für Seite nacheinander an. „Less“ geht einen Schritt weiter. Mit „less“ können Sie innerhalb der Anzeige blättern. Das Kommando „q“ beendet „less“.

„ln“ erstellt einen Link (~ Zeiger) auf ein Ziel / eine Datei. Es gibt symbolische und harte Links. Einen „harten“ Link können Sie nur setzen, wenn das Ziel vorhanden ist. Ein „symbolischer“ Link setzt kein Ziel voraus. Ist das Ziel nicht vorhanden ist ein „symbolischer“ Link einfach „tot“. Sie finden in Linux Systemen viele Konfigurationsdateien als symbolische Links. Symbolische Links erkennen Sie am „l“ an erster Stelle der Eigenschaft einer Datei, siehe oben den Link „bla.bla“.

Mit „wc“ (word count) können sie Zeilen, Wörter und Bytes innerhalb von Dateien zählen“.

Für nähere Informationen rufen Sie die man-pages zu dem jeweiligen Kommando auf.

## Umgang mit Dateien und Verzeichnissen (3)

```
Terminal - horst@charon: ~/meinedaten
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe
horst@charon:~/meinedaten$ wc -l datei22.txt
5 datei22.txt
horst@charon:~/meinedaten$ cat datei22.txt
dieser
Text
ist
nicht
sortiert
horst@charon:~/meinedaten$ sort datei22.txt
dieser
ist
nicht
sortiert
Text
horst@charon:~/meinedaten$
```

| Kommando | Wirkung   |
|----------|-----------|
| sort     | sortieren |

Mit dem „sort“-Kommando können Sie Text innerhalb eine Datei sortieren. Testen Sie dies mit Dateien:

### **datei2.txt**

dieser  
Text  
ist  
nicht sortiert

### **datei3.txt**

Pferd  
Katze  
Maus  
Ameise

### **datei4.txt**

10 Affe  
6 Huhn  
7 Tiger  
14 Zebra  
3 Maus

Verwenden Sie auch das Kommando „sort -n datei4.txt“, oder das Kommando „sort -r [Dateiname]“ um „rückwärts“ zu sortieren.

## Umgang mit Dateien und Verzeichnissen (4)

```
Terminal - horst@charon: ~  
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe  
horst@charon:~/meinedaten$ grep sortiert datei22.txt  
sortiert  
horst@charon:~/meinedaten$ grep i datei22.txt  
dieser  
ist  
nicht  
sortiert  
horst@charon:~/meinedaten$ find . -name datei3.txt  
./datei3.txt  
horst@charon:~/meinedaten$ find /home -name datei3.txt  
/home/horst/meinedaten/datei3.txt  
horst@charon:~/meinedaten$ rm datei3.txt  
horst@charon:~/meinedaten$ ls -la  
insgesamt 16  
drwxrwxr-x  2 horst horst 4096 Apr 12 20:18 .  
drwxr-xr-x 38 horst horst 4096 Apr 12 20:09 ..  
lrwxrwxrwx  1 horst horst  10 Apr 12 20:11 file01 -> datei3.txt  
-rw-rw-r--  1 horst horst   0 Apr 12 19:50 datei1.txt  
-rw-rw-r--  1 horst horst   0 Apr 12 19:50 datei21.txt  
-rw-rw-r--  1 horst horst  31 Apr 12 19:51 datei22.txt  
-rw-rw-r--  1 horst horst 648 Apr 12 19:52 datei2.txt  
horst@charon:~/meinedaten$ cd ..  
horst@charon:~$ rm -rf meinedaten  
horst@charon:~$
```

| Kommando | Wirkung   |
|----------|---|
| grep     | eine Datei nach Mustern durchsuchen                 |
| find     | eine Datei innerhalb eines Verzeichnisbaumes suchen |
| rm       | eine Datei / Verzeichnis löschen                    |

14.11.23

Horst Bräuner

16

„grep“ sucht im Inhalt von Dateien

Testen Sie „grep“ mit Ihren Dateien von voriger Seite.

```
user@host:~$ grep Maus datei3.txt  
user@host:~$ grep Maus *  
user@host:~$ grep er *  
user@host:~$ grep -i ti *
```


„find“ sucht Dateien innerhalb eines Verzeichnisbaumes. Sie finden auf einigen Systemen das Kommando „locate“ (locate [Dateiname]), das etwas komfortabler ist.

### VORSICHT

„rm“ **löscht** Dateien, **ohne Rückfrage und Papierkorb**

„rm -r“ **löscht** Verzeichnisse und Dateien rekursiv, **ohne Rückfrage und Papierkorb**



# Platzhalter

Terminal - horst@charon: ~/testdaten

```
horst@charon:~/testdaten$ ls
datei1 datei11 Datei2 datei2b Datei33
Datei1 datei2 datei22 datei3  Datei33a
horst@charon:~/testdaten$ ls *2
datei2 Datei2 datei22
horst@charon:~/testdaten$ ls datei?
datei1 datei2 datei3
horst@charon:~/testdaten$ ls D*
Datei1 Datei2 Datei33 Datei33a
horst@charon:~/testdaten$ ls *3?
Datei33 Datei33a
horst@charon:~/testdaten$ ls *a
Datei33a
horst@charon:~/testdaten$ ls ?at*
datei1 datei11 Datei2 datei2b Datei33
Datei1 datei2 datei22 datei3  Datei33a
horst@charon:~/testdaten$
```

\* = beliebige Menge von Zeichen  
? = genau 1 Zeichen


14.11.23Horst Bräuner17

Platzhalter oder „Wildcards“ können Aktionen erleichtern, da sie eine Vielzahl von Zeichen ersetzen können.


Sie haben auf voriger Seite bereits mit „\*“ für einen beliebigen Dateinamen gearbeitet.

- \* steht für eine beliebige Menge von Zeichen
- ? ist der Platzhalter für genau 1 Zeichen


Platzhalter können aber auch Aktionen gerade deswegen gefährlich machen. Zum Beispiel sollte das Löschen von Dateien mit Platzhaltern (`rm datei*`) überlegt werden. Es kann sein, dass Sie sonst plötzlich sehr viel Platz auf Ihrer Festplatte haben :-)



---



## Meta- und Sonderzeichen (1)



\

"..."

'...'

nachfolgendes Zeichen hat keine Sonderbedeutung  
user@host\$ rm \\*

die Zeichen zwischen den Hochkommata haben keine  
Sonderbedeutung (Ausnahme: \\$ ``)

die Zeichen zwischen den Hochkommata haben keine  
Sonderbedeutung

```

horst@charon:~/testdaten$ ls
*      *blafasel?.doc  dat\ei2007$.*
*blafasel?.2007$.doc.geheim  datei.*      **nocheinedatei.xls
horst@charon:~/testdaten$ ls \*
*
horst@charon:~/testdaten$ ls **
*      *blafasel?.2007$.doc.geheim  *blafasel?.doc  **nocheinedatei.xls
horst@charon:~/testdaten$ ls *
*      *blafasel?.doc  dat\ei2007$.*
*blafasel?.2007$.doc.geheim  datei.*      **nocheinedatei.xls
horst@charon:~/testdaten$ ls *.*
datei.*  dat\ei2007$.*
horst@charon:~/testdaten$ ls dat\ei2007$.*
ls: Zugriff auf datei2007$.* nicht möglich: Datei oder Verzeichnis nicht gefunde
n
horst@charon:~/testdaten$ ls "dat\ei2007$.*"
dat\ei2007$.*
horst@charon:~/testdaten$ ls 'dat\ei2007$.*'
dat\ei2007$.*
horst@charon:~/testdaten$ ls dat\\ei2007\$.*
dat\ei2007$.*
horst@charon:~/testdaten$

```

14.11.23
Horst Bräuner
18

Der Vollständigkeit halber noch ein paar Worte zu Meta- und Sonderzeichen:

Sie könnten eine Datei \*.\* nennen (oder noch kreativer sein ;-) siehe oben), indem Sie

```
user@host:~$ touch \*.\*
```

verwenden. Denken Sie bitte daran, dass es eine ganze Anzahl von Menschen gibt, die in der „heilen Welt proprietärer Betriebssysteme“ leben. Diese Menschen könnten Sie mit solchen Dateinamen zur Verzweiflung bringen :-) Zudem kann es sein, dass solche Dateien als Anhänge in E-Mail geblockt werden.

Zwischen Hochkommata ist normalerweise „Frieden“ und Text wird einfach als Zeichenfolge behandelt. Ausnahme davon der „\“ (wie oben erwähnt), „\$“ für Variablen und `` Backquotes für Befehle innerhalb von Texten.

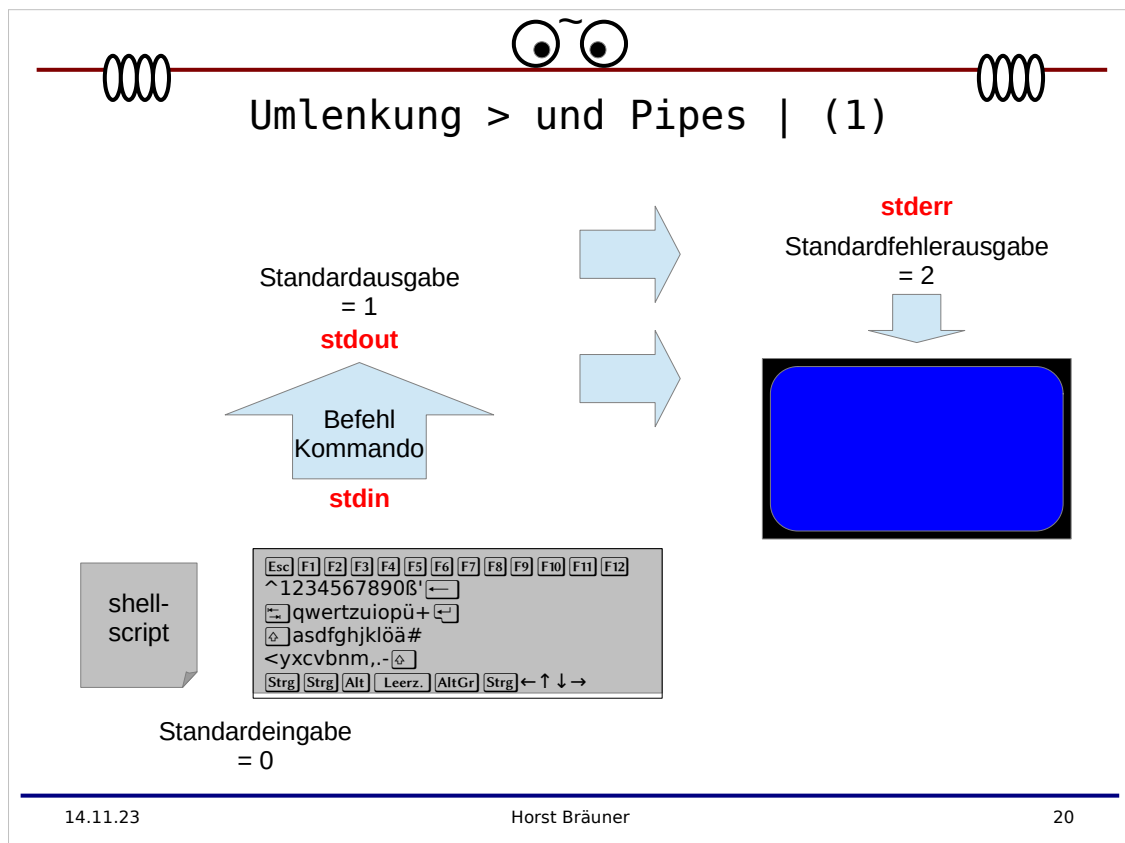


## Meta- und Sonderzeichen (2)

- ~ Tilde = HOME-Verzeichnis
- Bindestrich = zum vorigen Verzeichnis
- # Kommentarsymbol, z.B. in Skripten
- ; Semikolon = Trennung von 2 Befehlen
- & „Kaufmanns“-Und = im Hintergrund starten
- \$ Dollar = Inhalt von Variablen ausgeben
- () runde Klammer = Kommandos in einer Subshell ausführen
- `` Backquotes = Text als Kommando verstehen

```
Terminal - horst@charon:/opt/virt
Datei  Bearbeiten  Ansicht  Terminal  Gehe zu  Hilfe
horst@charon:/opt/virtual$ pwd
/opt/virtual
horst@charon:/opt/virtual$ cd ~
horst@charon:~$ pwd
/home/horst
horst@charon:~$ cd -
/opt/virtual
horst@charon:/opt/virtual$ #da passiert nix
horst@charon:/opt/virtual$ pwd ; ls
/opt/virtual
Adobe.vdi
horst@charon:/opt/virtual$ echo $HOME $UID $PWD
/home/horst 1000 /opt/virtual
horst@charon:/opt/virtual$ echo `date`
date
horst@charon:/opt/virtual$ echo `date`
So 14. Apr 15:28:38 CEST 2013
horst@charon:/opt/virtual$
```

- die ~ Tilde ist synonym für das Home-Verzeichnis eines Benutzers
- der - Bindestrich steht für das vorige Verzeichnis
- der # Hash-Tag oder „Gartenzaun“ oder „Raute“ wird als Zeichen vor Kommentaren verwendet
- das ; Semikolon trennt zwei Befehle
- das & führt Kommandos im Hintergrund aus
- das \$ Zeichen steht für eine Variable (Variable var wird über \$var angesprochen)
- mit einem Kommando in () Klammern, wird diese in einer Subshell ausgeführt
- die `` Backquotes (Akzent-Taste) führen innerhalb eines Textes ein Kommando aus



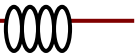


Linuxe können Geräte umleiten oder miteinander verknüpfen.

stdin oder 0 ist die Standarddeingabe. Das kann eine Tastatur sein. Das kann auch ein Script sein, das Befehle ausführt.

stdout oder 1 ist die Standardausgabe. Das kann ein Bildschirm sein. Das kann auch eine Datei sein, wenn ein script ausgeführt wird. Es macht zum Beispiel wenig Sinn an einem Bildschirm Informationen anzuzeigen, wenn niemand zusieht. Und, nicht alle Systeme haben Bildschirme.

stderr oder 2 ist die Standardfehlerausgabe. Das kann ein Bildschirm sein. Das kann auch eine Datei sein oder ein Drucker oder irgend etwas anderes, das Fehler dokumentieren soll.

Wenn Sie mit einem PC interaktiv arbeiten ist stdin die Tastatur und stdout und stderr Ihr Bildschirm.



## Umlenkung > und Pipes | (2)

- Umlenkung
  - > lenkt die Ausgabe von stdin um und legt die Ausgabe(datei) ggf. an oder überschreibt sie, falls schon vorhanden.
  - >> lenkt die Ausgabe von stdin um und fügt sie an eine bestehende Datei an.
    - Beispiele:
      - user@host\$ man man > manman.txt
      - user@host\$ echo „halli hallo“ > hallo.txt
      - user@host\$ echo „nochmal hallo“ >> hallo.txt
    - Beispiel Umlenkung Ausgabe...**
      - user@host\$ cp -v /etc/hosts bla.txt 1>klappt 2>problem
      - user@host\$ rm bla.txt
      - user@host\$ mv -v /etc/hosts bla.txt 1>klappt 2>problem
- Pipes
  - | verknüpft Befehle/Kommandos miteinander, so dass das Ergebnis des ersten Kommandos durch das 2te bearbeitet wird
    - Beispiele:
      - user@host\$ who | wc -l
      - user@host\$ ps ax | grep -i tty | sort -n

14.11.23

Horst Bräuner

21

stdin, stdout und stderr können auf Linux-Systemen umgeleitet werden.

> lenkt die Ausgabe von stdin (hier: Tastatur) in eine Datei um. Falls es die Datei schon gibt, wird diese überschrieben.

>> lenkt die Ausgabe ebenfalls um, fügt sie aber ans Ende einer Datei an. Falls es die Datei nicht gibt, wird sie erstellt.

Die Ausgabedateien `manman.txt`, `hallo.txt` usw. im obigen Beispiel können Sie sich mit `cat`, `more`, `less` anschauen.

Im obigen Beispiel wird in der Zeile ...

```
user@host:~$ cp -v /etc/hosts bla.txt 1>klappt 2>problem
```


... versucht die Datei `/etc/hosts` nach `/home/user/bla.txt` zu kopieren.

```
user@host:~$ mv -v /etc/hosts bla.txt 1>klappt 2>problem
```

... versucht die Datei `/etc/hosts` nach `/home/user/bla.txt` zu verschieben.

Falls das jeweils klappt, wird die Ausgabe (stdout (=1) zeigt sie an) in die Datei `/home/user/klappt` geschrieben. Falls nicht, erfolgt die Ausgabe in `/home/user/problem`.

„|“ Pipes verknüpfen Kommandos und können auch mehrfach hintereinander verwendet werden. Im obigen letzten Beispiel werden alle aktiven (ax) Prozesse (ps) im System angezeigt, mit „grep“ die Bildschirmprozesse (tty) gefiltert und mit „sort“ numerisch aufsteigend sortiert.



# Aliase

```
Terminal - horst@charon: ~
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe
horst@charon:~$ häh
häh: Befehl nicht gefunden.
horst@charon:~$ alias häh=id
horst@charon:~$ häh
uid=1000(horst) gid=1000(horst) Gruppen=1000(horst),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),110(lpadmin),123(sambashare)
horst@charon:~$
```

```
Terminal - horst@charon: ~/testdaten
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe
horst@charon:~/testdaten$ ll
insgesamt 16
drwxrwxr-x  2 horst horst 4096 Apr 13 17:20 ./
drwxr-xr-x 39 horst horst 4096 Apr 13 17:16 ../
-rw-rw-r--  1 horst horst   8 Apr 13 17:18 jkhl
-rw-rw-r--  1 horst horst   8 Apr 13 17:18 l
-rw-rw-r--  1 horst horst   0 Apr 13 17:17 lasdkfjl
horst@charon:~/testdaten$ alias ImachDichPlattAlta='rm -i *'
horst@charon:~/testdaten$ ImachDichPlattAlta
rm: Normale Datei »jkhl« entfernen? n
rm: Normale Datei »l« entfernen? n
rm: Normale leere Datei »lasdkfjl« entfernen? n
horst@charon:~/testdaten$
```

Aliase können als Abkürzungen von Befehlen oder Varianten von Kommandos verwendet werden. Z.B. um versehentliches Löschen mit „rm“ zu verhindern:

```
alias rm='rm -i'
```

14.11.23

Horst Bräuner

22


Aliase können dazu verwendet werden System-Kommandos durch eigene Kommandos zu ersetzen.

der Befehl „rm“ löscht bekanntlich Dateien ohne Rückfrage. „rm -i“ würde „sind Sie sicher?“ rückfragen. Wenn Sie nun den Alias `rm = „rm -i“` setzen, fragt Sie auch rm jedesmal beim Löschen einer Datei „sind Sie sicher?“. Das kann das Leben als Linux-Administrator erleichtern.

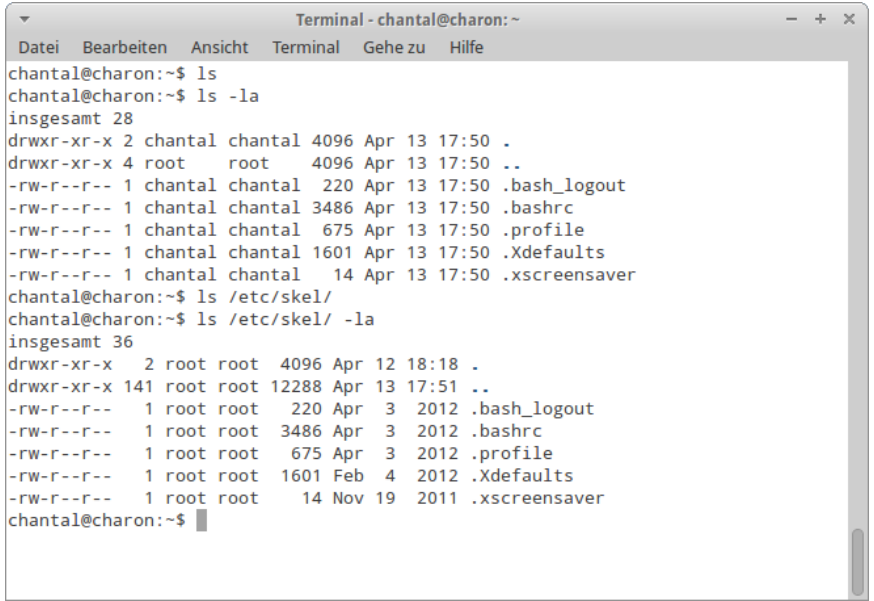
Wenn Sie den alias „von Hand“ an der Kommandozeile setzen, ist er nach dem nächsten Ab- und Anmelden wieder weg.

Sie finden Beispiele für Aliase in der Datei `.bashrc` (Punkt beachten) in Ihrem HOME-Verzeichnis. Dort könnten Sie auch den „rm“-Alias einbauen. Schauen Sie sich die Datei `.bashrc` mit `cat` oder `nano` an und ergänzen Sie die dortigen Aliase. Sichern Sie mit „cp“ die Datei vorher ;-)

```
user@host:~$ cp .bashrc .bashrc.sicherung
user@host:~$ nano .bashrc
```

## Dateien in \$HOME



```
Terminal - chantal@charon: ~
Datei Bearbeiten Ansicht Terminal Gehe zu Hilfe
chantal@charon:~$ ls
chantal@charon:~$ ls -la
insgesamt 28
drwxr-xr-x 2 chantal chantal 4096 Apr 13 17:50 .
drwxr-xr-x 4 root root 4096 Apr 13 17:50 ..
-rw-r--r-- 1 chantal chantal 220 Apr 13 17:50 .bash_logout
-rw-r--r-- 1 chantal chantal 3486 Apr 13 17:50 .bashrc
-rw-r--r-- 1 chantal chantal 675 Apr 13 17:50 .profile
-rw-r--r-- 1 chantal chantal 1601 Apr 13 17:50 .Xdefaults
-rw-r--r-- 1 chantal chantal 14 Apr 13 17:50 .xscreensaver
chantal@charon:~$ ls /etc/skel/
chantal@charon:~$ ls /etc/skel/ -la
insgesamt 36
drwxr-xr-x 2 root root 4096 Apr 12 18:18 .
drwxr-xr-x 141 root root 12288 Apr 13 17:51 ..
-rw-r--r-- 1 root root 220 Apr 3 2012 .bash_logout
-rw-r--r-- 1 root root 3486 Apr 3 2012 .bashrc
-rw-r--r-- 1 root root 675 Apr 3 2012 .profile
-rw-r--r-- 1 root root 1601 Feb 4 2012 .Xdefaults
-rw-r--r-- 1 root root 14 Nov 19 2011 .xscreensaver
chantal@charon:~$
```

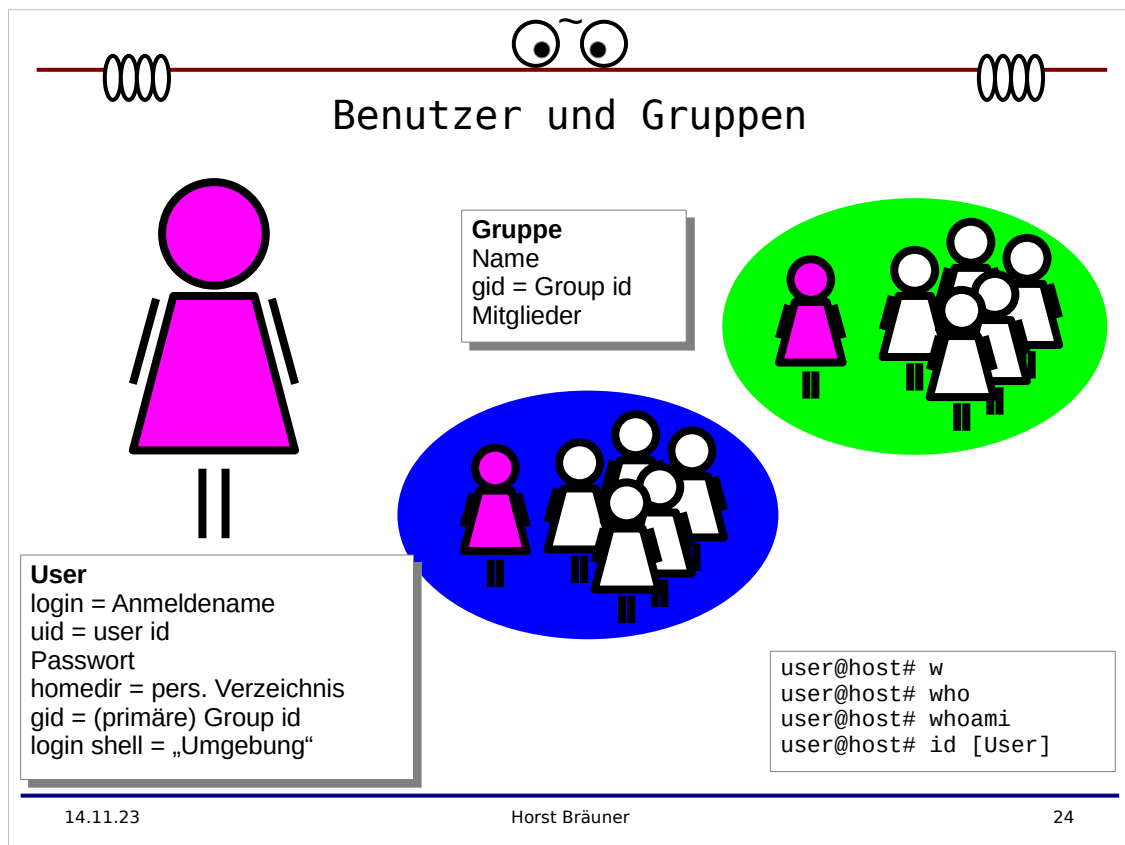
14.11.23Horst Bräuner23

Jeder Benutzer erhält normalerweise ein eigenes Verzeichnis, ganz für sich alleine. Es wird als Homedir, Home-Directory, Home-Verzeichnis oder \$HOME bezeichnet. \$HOME weil die Variable \$HOME den Pfad zu diesem Verzeichnis enthält. Sie können sich die Variable mit

```
user@host:~$ echo $HOME
```

anzeigen lassen. Beachten Sie die GROSS/klein-Schreibung.

Im obigen Beispiel sehen Sie eine Liste der Dateien, die bereits in Ihrem Home-Verzeichnis vorhanden sind. Dateien, die eine Punkt am Anfang haben, sind „unsichtbar“. Das normale „ls“-Kommando oder der Dateimanager zeigen sie nicht an. Das soll verhindern, dass „unbedarfte“ Anwender diese Dateien aus Versehen löschen oder an diesen Dateien manipulieren.




Ein Benutzer / User wird durch eine eindeutige ID (user identifier, UID) gekennzeichnet. User haben normalerweise ein Passwort und ein persönliches Verzeichnis. Sie sind in einer primären Gruppe und haben eine login-shell, das heißt eine eigene, persönliche Umgebung.


Gruppen haben eine ID und einen „sprechenden“ Namen.

Ein User kann in mehreren Gruppen Mitglied sein. **MERKE:** Gruppen können NICHT in Gruppen Mitglied sein.








---




## Benutzer - /Gruppen-Verwaltung



- /etc/passwd
  - Benutzer und ihre Eigenschaften
- /etc/group
  - Gruppen und zugehörige Benutzer
- /etc/shadow
  - Benutzer + verschlüsseltes Passwort
- /etc/skel
  - Inhalt wird nach \$HOME (pers. Verzeichnis des Users) kopiert




14.11.23
Horst Bräuner
25

Die Verwaltung von Usern und Gruppen erfolgt in Dateien. Schauen Sie sich die Dateien an. Sie benötigen für die shadow-Datei root-Rechte.

```
user@host:~$ cat /etc/passwd
user@host:~$ cat /etc/group
user@host:~$ su -
root@host:~# cat /etc/shadow
```

Im Verzeichnis „/etc/skel“ (= skeleton, Gerüst) liegen Dateien, die beim **Anlegen** eines Benutzers in dessen Home-Verzeichnis kopiert werden.



## Benutzer anlegen und bearbeiten

```

max@depp1:~$ su -
Passwort:
root@depp1:~# useradd -m jennifer
root@depp1:~# passwd jennifer
Geben Sie ein neues UNIX-Passwort ein:
Geben Sie das neue UNIX-Passwort erneut ein:
passwd: Passwort erfolgreich geändert
root@depp1:~#
root@depp1:~# adduser jaqueline
Lege Benutzer »jaqueline« an ...
Lege neue Gruppe »jaqueline« (1002) an ...
Lege neuen Benutzer »jaqueline« (1002) mit Gruppe »jaqueline« an ...
Erstelle Home-Verzeichnis »/home/jaqueline« ...
Kopiere Dateien aus »/etc/skel« ...
Geben Sie ein neues UNIX-Passwort ein:
Geben Sie das neue UNIX-Passwort erneut ein:
passwd: Passwort erfolgreich geändert
Benutzerinformationen für jaqueline werden geändert.
Geben Sie einen neuen Wert an oder drücken Sie ENTER für den Standardwert
Vollständiger Name []:
Zimmernummer []:
Telefon geschäftlich []:
Telefon privat []:
Sonstiges []:
Sind die Informationen korrekt? [J/n] J
root@depp1:~#

```

useradd = cmdline  
adduser = interaktiv, config über  
/etc/adduser.conf  
usermod = cmdline  
userdel / deluser

14.11.23
Horst Bräuner
26

Abgesehen von grafischen Tools, gibt es (mindestens ;-) ) zwei Kommandos, mit denen User angelegt werden können.

**useradd** legt einen User an; ohne Passwort, ohne Login-shell, ohne Home-Directory; all das müssen Sie nachträglich anlegen.

**adduser** fragt mehrere Informationen ab und legt dann einen User mit Passwort und Home-Directory an, siehe oben im Beispiel

**usermod** ändert / modifiziert einen User, siehe „usermod --help“

**userdel / deluser** löscht einen User aber (meist) nicht dessen Home-Directory; das Löschen des Home-Verzeichnisses muss extra konfiguriert werden oder per Parameter angegeben werden, s. „userdel --help“ bzw. „deluser –help“

**Grafische Tools:** die für diese Vorlesung installierte „xfce“-Desktop Umgebung hat kein eigenes grafisches Tool zum Benutzermanagement. Sie können jedoch beispielsweise den „gnome“ user-manager dafür verwenden. Spätestens nach einem Neustart ist er unter „Einstellungen“ und „System“ zu finden.

```
root@host:~# apt install gnome-system-tools
```



## Gruppen anlegen und bearbeiten

```
root@depl1:~#  
root@depl1:~# groupadd maedels  
root@depl1:~#  
root@depl1:~# addgroup jungs  
Lege Gruppe »jungs« (GID 1004) an ...  
Fertig.  
root@depl1:~#  
root@depl1:~# id jennifer  
uid=1001(jennifer) gid=1001(jennifer) Gruppen=1001(jennifer)  
root@depl1:~#  
root@depl1:~# usermod -G maedels jennifer  
root@depl1:~#  
root@depl1:~# id jennifer  
uid=1001(jennifer) gid=1001(jennifer) Gruppen=1001(jennifer),1003(maedels)  
root@depl1:~#  
root@depl1:~# adduser jaqueline maedels  
Füge Benutzer »jaqueline« der Gruppe »maedels« hinzu ...  
Benutzer jaqueline wird zur Gruppe maedels hinzugefügt.  
Fertig.  
root@depl1:~#  
root@depl1:~# id jaqueline  
uid=1002(jaqueline) gid=1002(jaqueline) Gruppen=1002(ja  
root@depl1:~#  
root@depl1:~#
```

groupadd = cmdline  
addgroup = „interaktiv“  
groupmod  
groupdel / delgroup  
  
Benutzer zu Gruppe hinzufügen:  
usermod -G [Gruppe] [User]  
  
adduser [User] [Gruppe]

Für die Verwaltung von Gruppen gibt es die Varianten:

**goupadd** = Anlegen ohne Rückmeldung

**addgroup** = Anlegen mit kurzer Rückmeldung

**groupmod** = Ändern der Eigenschaften einer Gruppe, z.B.  
Gruppenname

**goupdel / delgroup** = Löschen einer Gruppe

Benutzer werden mit ...

```
root@host:~# usermod -G [Gruppenname] [Username]
```

ODER

```
root@host:~# adduser [Username] [Gruppenname]
```

zu Gruppen hinzugefügt.



# Identität wechseln



← „didi“

im „Terminal“ ...  
su = Switch User  
su - = Switch User mit „neuer“ Umgebung

14.11.23Horst Bräuner28

wie bereits am Anfang erwähnt, können Sie mit dem „switch user“-Kommando in andere Identitäten wechseln. Das „-“ nach dem Kommando bewirkt, dass Sie eine eigene, neue Umgebung erhalten.

~

# „Multiuser“-System

hier 3 Terminal-Sitzungen:

```
didi@depp1:~$ su -          # user= root
didi@depp1:~$ su - max     # user=max
didi@depp1:~$ su - moritz  # user=moritz
alternativ: didi@depp1:~$ ssh -l [username] localhost
```

14.11.23 Horst Bräuner 29

Legen Sie zur Übung ein paar Benutzer an und wechseln Sie Ihre Identität.

```
root@host:~# addgroup girls
root@host:~# addgroup boys
root@host:~# adduser max
root@host:~# adduser moritz
root@host:~# adduser max boys
root@host:~# adduser moritz boys
root@host:~# adduser hanni
root@host:~# adduser nanni
root@host:~# adduser hanni girls
root@host:~# adduser nanni girls
```

Spielen Sie mit verschiedenen Identitäten :-)



## „Multiuser“-System - Übung

- als root / im root-Fenster
  - Benutzer / Gruppen anlegen
  - Verzeichnis(se) unter /opt anlegen
  - mit chown / chgrp / chmod Berechtigungen ändern
- mit **mehreren** „su -“ - Identitäten prüfen ...
  - Zugriff auf Verzeichnisse (ls (-la), mkdir, cd, ...)
  - Zugriff auf Dateien (touch, cp, rm, mv, ...)
  - Berechtigungen ändern (chown, chmod, ...)
  - \$HOME „absichern“ („700“)

Versuchen Sie die Kommandos

**chown = change owner** ... „ändere den Eigentümer“

`chown [USER] datei1.txt` = ändere den Eigentümer der Datei „datei1.txt“ und lasse die Gruppe der Datei „datei1.txt“ wie sie ist

`chown [USER:GROUP] datei1.txt` = ändere den Eigentümer der Datei „datei1.txt“ und die Gruppe der Datei „datei1.txt“.


**chmod = change modus** ... „ändere die Eigenschaften (Rechte)“

`chmod 754 datei1.txt` = ändere die Berechtigung der Datei datei1.txt darauf, dass der Eigentümer alle Rechte (7, rwx), die Gruppe Lese- und Ausführrechte (5, rx) und der Rest der Welt nur Leserechte (4, r) hat. Zu den Rechten, siehe Folien „Grundlagen“.

Beispiele zum Üben:

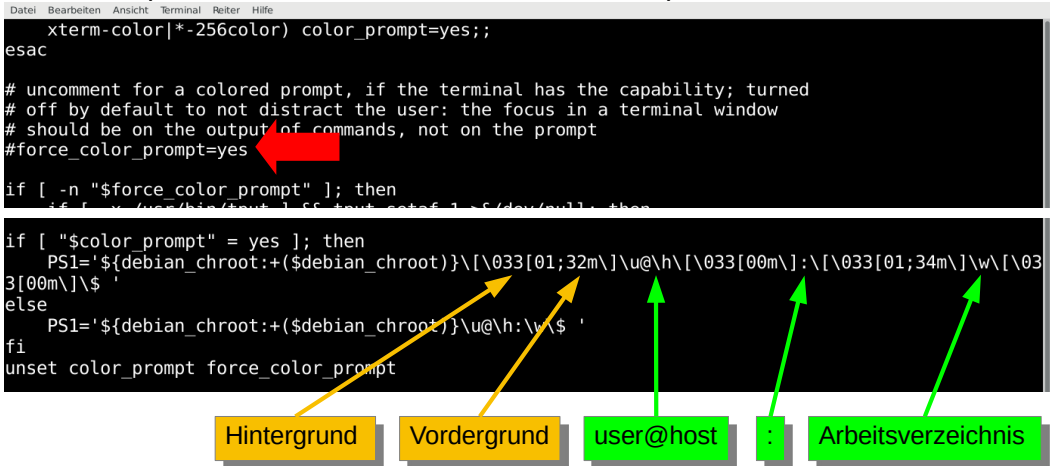
- Legen Sie als User „max“ Dateien an und versuchen Sie diese als „moritz“ zu Lesen. Ändern Sie als „max“ die Berechtigungen und schauen Sie sich als „moritz“ das Ergebnis an.
- Erstellen Sie eine Gruppe „boys“ mit den Mitgliedern „max“ und „moritz“ und eine Gruppe „girls“ mit den Mitgliedern „hanni“ und „nanni“. Legen Sie als eines der „girls“ Dateien an und schauen sich das Ergebnis als einer der „boys“ an. Ändern Sie dann die Berechtigungen der Gruppen an den Dateien.

**Hinweis:** im Verzeichnis /tmp haben in aller Regel alle Benutzer Schreibrechte



# „Multiuser“-System - Übung

- prompt/alias für jeden Benutzer individuell einstellen (/home/[USER]/.bashrc)
- s. <https://wiki.ubuntuusers.de/Bash/Prompt/>



```

xterm-color[*-256color) color_prompt=yes;;
esac

# uncomment for a colored prompt, if the terminal has the capability; turned
# off by default to not distract the user: the focus in a terminal window
# should be on the output of commands, not on the prompt
#force_color_prompt=yes

if [ -n "$force_color_prompt" ]; then
    if [ -x "$(command -v tput)" ]; then
        tput setaf 1 && tput setab 1 && echo -n
    fi
fi

if [ "$color_prompt" = yes ]; then
    PS1='${debian_chroot:+($debian_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\]:\[\033[01;34m\]\w\[\033[00m\]\$ '
else
    PS1='${debian_chroot:+($debian_chroot)}\u@\h:\w\$ '
fi
unset color_prompt force_color_prompt
  
```

Hintergrund
Vordergrund
user@host
:
Arbeitsverzeichnis

14.11.23
Horst Bräuner
31

Falls Sie die Umgebung des Benutzers, die „shell“, etwas aufpeppen wollen, können Sie sich an der Datei .bashrc versuchen, die Sie bei „Aliase“ schon kennen gelernt haben. Kopieren Sie sich zur Sicherheit die vorhandene Datei.

```
user@host:~$ cp .bashrc .bashrc.sicherung
```

und ändern Sie dann in der Datei .bashrc die Anzeige des „Prompt“, also der Eingabeaufforderung. Sie können dafür beispielweise den nano-Editor verwenden.

siehe Artikel: <https://wiki.ubuntuusers.de/Bash/Prompt/>

~

„Multiuser“-System - Beispiel

Datei Bearbeiten Ansicht Terminal Reiter Hilfe

moritz@deb18adt:~\$

moritz@deb18adt:~\$

Datei Bearbeiten Ansicht Terminal Reiter Hilfe

max@deb18adt:~\$

max@deb18adt:~\$

Datei Bearbeiten Ansicht Terminal Reiter Hilfe

didi@deb18adt:~\$

didi@deb18adt:~\$

Datei Bearbeiten Ansicht Terminal Reiter Hilfe

chantal@deb18adt:12:04:09--bash--\$

chantal@deb18adt:12:04:10--bash--\$

didi@depp1:~\$ setterm --inversescreen on / off

14.11.23Horst Bräuner32

Je nach Ihrer Kreativität erscheint der Prompt bei jedem User anders :-)