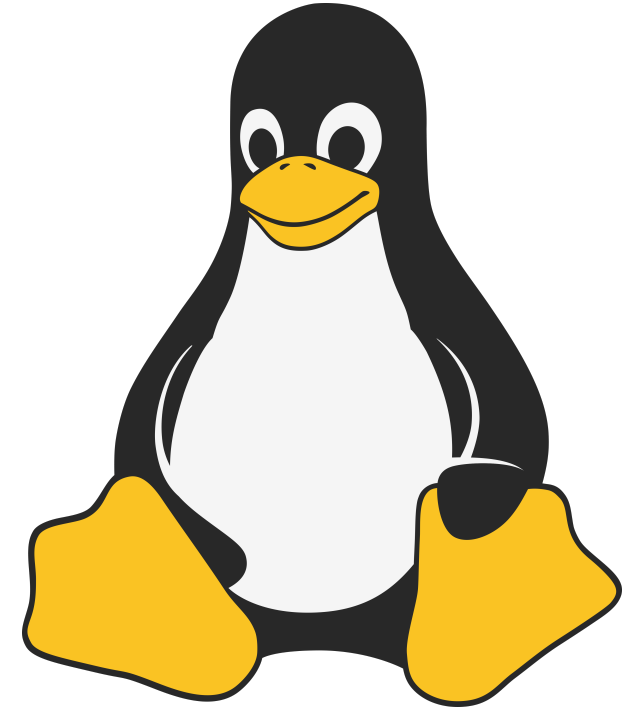


Shell-Praxis (Teil 4)



Dateiattribute, Zugriffsrechte, Identität,
Dateiinhalte ausgeben

Inhaltsverzeichnis

- Verzeichnisse anlegen/löschen: `mkdir`, `rmdir`
- Kommando: `ls`
- Benutzerverwaltungsdateien
- Dateiattribute
- Zugriffsrechte
- Bestimmung der eigenen Identität: `id`

- Inhalt von Dateien ausgeben: `cat`
- Inhalt der Benutzerverwaltungsdateien
- Kopf- und Fußzeilen ausgeben: `head` und `tail`
- Zeilen filtern mit `grep`

Verzeichnisse anlegen/löschen: `mkdir`, `rmdir`

Zum Start befinden wir uns im Home-Verzeichnis des angemeldeten Benutzers.

- `pwd` (print working directory) zeigt das aktuelle Verzeichnis an.
- `mkdir` (make directory) erstellt ein neues Verzeichnis.
- `rmdir` (remove directory) löscht ein Verzeichnis. Das Verzeichnis muss leer sein.
- `cd` (change directory) wechselt in das angegebene Verzeichnis.

```
hermann@debian:~$ pwd
/home/hermann
hermann@debian:~$ mkdir my-tests # create directory
hermann@debian:~$ rmdir my-tests # remove directory
hermann@debian:~$ mkdir my-tests # create directory again
hermann@debian:~$ cd my-tests # change to directory
hermann@debian:~/my-tests$ pwd # show current directory
/home/hermann/my-tests
hermann@debian:~/my-tests$ ls
hermann@debian:~/my-tests$ ls -a
.  ..
hermann@debian:~/my-tests$ ls -l
insgesamt 0
hermann@debian:~/my-tests$ ls -al
insgesamt 8
drwxrwxr-x   2 hermann hermann 4096 Nov  9 15:13 .
drwxr-x---+ 63 hermann hermann 4096 Nov  9 15:13 ..
hermann@debian:~/my-tests$
```

Kommando: `ls`

- `ls` (list) zeigt die Einträge des angegebenen Verzeichnisses an. Ist kein Verzeichnis angegeben, werden die Einträge des aktuellen Verzeichnisses angezeigt. In diesem ist das Verzeichnis `my-tests` leer. Es gibt also keine Einträge.
- `ls -a` zeigt alle Einträge an, auch die versteckten. Bei einem leeren Verzeichnis sind das die Einträge `.` und `..`. `.` repräsentiert das aktuelle Verzeichnis, `..` das übergeordnete.

- `ls -l` zeigt die Einträge ausführlich (mit den wichtigsten Attributen) an. In diesem Fall gibt es keine Einträge.
- `ls -al` zeigt alle Einträge ausführlich an. In diesem Fall gibt es zwei Einträge: `.` und `..`.

Benutzerverwaltungsdateien

Das Verzeichnis `/etc` enthält viele Konfigurationsdateien und -verzeichnisse. Drei wichtige Dateien für die Benutzerverwaltung sind:

- `/etc/passwd`: enthält Infos über die Benutzer des Systems.
- `/etc/shadow`: enthält die verschlüsselten Passwörter der Benutzer.
- `/etc/group`: enthält Infos über die Gruppen des Systems.

Dateiattribute

Dateiattribute am Beispiel von `/etc/passwd`, `/etc/shadow`,
`/etc/group`

```
hermann@debian:~/my-tests$ ls -l /etc/passwd /etc/shadow /etc/group
-rw-r--r-- 1 root root 1316 Sep 20 13:13 /etc/group
-rw-r--r-- 1 root root 3175 Sep 20 13:13 /etc/passwd
-rw-r----- 1 root shadow 1586 Sep 20 13:13 /etc/shadow
```

`ls -l` zeigt die Attribute (Meta-Daten) von Dateien und Verzeichnissen an.

Dateiattribute: Beispiel: `/etc/passwd`

- `-` (Minus-Zeichen in der ersten Spalte) zeigt an, dass es sich um eine Datei handelt. (Ein `d` würde auf ein Verzeichnis hinweisen, ein `l` auf einen symbolischen Link.)
- `rw-r--r--` zeigt die Zugriffsrechte an.
 - `rw-` (1. Tripel) bedeutet, dass der Besitzer der Datei Lese- und Schreibrechte hat, aber keine Ausführungsrechte.
 - `r--` (2. Tripel) bedeutet, dass die Gruppe, zu der die Datei gehört, nur Leserechte hat.
 - `r--` (3. Tripel) bedeutet, dass alle anderen Benutzer nur Leserechte haben.

- `1` zeigt an, dass es nur einen Hardlink auf die Datei gibt.
- `root` ist der Besitzer der Datei.
- `root` ist die Gruppe, zu der die Datei gehört.
- `3175` ist die Größe der Datei in Bytes.
- `Sep 20 13:13` ist das Datum der letzten Modifikation.
- `/etc/passwd` ist der Dateiname.

Zugriffsrechte

Welches Tripel zeigt meine Zugriffsrechte an?

Dazu muss zunächst die eigene Identität bestimmt werden (`id`).

1. Bin ich der Eigentümer (owner) der Datei? Wenn ja, dann gelten die Rechte des 1. Tripels.
2. Bin ich Mitglied einer Gruppe (group), zu der die Datei gehört? Wenn ja, dann gelten die Rechte des 2. Tripels.
3. Bin ich ein anderer (others), also weder Eigentümer noch Mitglied einer Gruppe, zu der die Datei gehört? Es gelten die Rechte des 3. Tripels.

Bestimmung der eigenen Identität: `id`

Das Kommando `id` zeigt die Identität des angemeldeten Benutzers an.

- Benutzernummer (UID) und Benutzername
- Gruppennummer (GID) und Gruppenname
- Gruppenzugehörigkeit zu weiteren Gruppen

```
hermann@debian:~/my-tests$ id
uid=1000(hermann) gid=1000(hermann) Gruppen=1000(hermann),24(cdrom),...,27(sudo),...
```

Inhalt von Dateien ausgeben: **cat**

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
...
hermann:x:1000:1000:Hermann Hueck:/home/hermann:/bin/bash
...
```

Das Kommando **cat** (concatenate) gibt den Inhalt der als Argument angegebenen Datei auf die Standardausgabe aus.

Inhalt von Dateien blättern: `less`

Das Kommando `less` ist ein Pager, der den Inhalt von Dateien seitenweise anzeigt. Es ist interaktiv und erlaubt das Blättern vorwärts und rückwärts. Das ist oft bei langen Dateien hilfreich.

`less` liest auch von der Standardeingabe, sodass man damit auch die Ausgabe von anderen Kommandos seitenweise betrachten kann.

Ist `less` gestartet, dann blättert ...

- das Kommando `.` eine Zeile vorwärts,
- das Kommando `SPACE` eine Seite vorwärts,
- das Kommando `b` eine Seite zurück und
- das Kommando `q` beendet `less`.

Inhalt der Benutzerverwaltungsdateien

Struktur der Zeilen in `/etc/passwd`

Jede Zeile in der Datei `/etc/passwd` enthält Angaben zu einem Benutzer. Eine Zeile hat sieben Felder, die durch Doppelpunkte getrennt sind.

```
username:password:uid:gid:comment:homedir:shell
```

1. Benutzername
2. Passwort (in der Regel `x`, da Passwörter nicht mehr hier sondern in `/etc/shadow` gespeichert werden)
3. Benutzer-ID (UID)
4. Gruppen-ID (GID)
5. Kommentar (optional, enthält in der Regel den vollen Namen des Benutzers)
6. Home-Verzeichnis
7. Login-Shell

/etc/shadow

`/etc/shadow` enthält die verschlüsselten Passwörter der Benutzer und kann von normalen Benutzern nicht gelesen werden.

```
hermann@debian:~/my-tests$ ls -l /etc/shadow
-rw-r----- 1 root shadow 1586 Sep 20 13:13 /etc/shadow
hermann@debian:~/my-tests$ cat /etc/shadow
cat: /etc/shadow: Keine Berechtigung
```

Mit vorangestelltem `sudo` würde das Kommando `cat /etc/shadow` mit root-Rechten ausgeführt werden und nicht fehlschlagen.

Kopf- und Fußzeilen ausgeben: `head` und `tail`

- Das Kommando `head -n <file>` gibt die ersten n Zeilen einer Datei aus (Standard: 10).
- Das Kommando `tail -n <file>` gibt die letzten n Zeilen einer Datei aus (Standard: 10).

```
hermann@debian:~/my-tests$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
```

```
hermann@debian:~/my-tests$ head -3 /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
```

```
hermann@debian:~/my-tests$ tail /etc/passwd
saned:x:123:135:./var/lib/saned:/usr/sbin/nologin
hplip:x:124:7:HPLIP system user,,,:/run/hplip:/bin/false
nm-openconnect:x:126:137:NetworkManager OpenConnect plugin,,,:/var/lib/NetworkManager:/usr/sbin/nologin
fwupd-refresh:x:997:997:Firmware update daemon:/var/lib/fwupd:/usr/sbin/nologin
davfs2:x:127:138:./var/cache/davfs2:/usr/sbin/nologin
swtpm:x:128:140:virtual TPM software stack,,,:/var/lib/swtpm:/bin/false
libvirt-qemu:x:64055:109:Libvirt Qemu,,,:/var/lib/libvirt:/usr/sbin/nologin
libvirt-dnsmasq:x:129:142:Libvirt Dnsmasq,,,:/var/lib/libvirt/dnsmasq:/usr/sbin/nologin
colord:x:125:136:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
hermann:x:1000:1000:Hermann Hueck:/home/hermann:/bin/bash
```

```
hermann@debian:~/my-tests$ tail -2 /etc/passwd
colord:x:125:136:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
hermann:x:1000:1000:Hermann Hueck:/home/hermann:/bin/bash
```

Zeilen filtern mit **grep**

- Das Kommando **grep <pattern> <file>** (global regular expression print) filtert Zeilen aus einer Datei heraus, die das angegebene Muster enthalten.
- Häufige Optionen:
 - **-v** (invert match): Zeilen ausgeben, die das Muster nicht enthalten.
 - **-i** (ignore case): Groß- und Kleinschreibung ignorieren.
 - **-n** (line number): Zeilennummer der Trefferzeile mit ausgeben.

```
hermann@debian:~/my-tests$ grep hermann /etc/passwd  
hermann:x:1000:1000:Hermann Hueck:/home/hermann:/bin/bash
```

Das folgende Kommando zeigt die Gruppe **hermann** (Gruppenname in der 1. Spalte) sowie alle Gruppen, in denen der Benutzer **hermann** Mitglied ist (Liste der Benutzernamen einer Gruppe in der 2. Spalte).

- Vergleichen Sie die Ausgabe mit der Ausgabe des **id**-Kommandos.


```
hermann@debian:~/my-tests$ grep hermann /etc/group
cdrom:x:24:hermann
floppy:x:25:hermann
sudo:x:27:hermann
audio:x:29:pulse,hermann
dip:x:30:hermann
video:x:44:hermann
plugdev:x:46:hermann
users:x:100:hermann
netdev:x:106:hermann
bluetooth:x:111:hermann
lpadmin:x:113:hermann
scanner:x:116:saned,hermann
hermann:x:1000:
```

```
hermann@debian:~/my-tests$ id
uid=1000(hermann) gid=1000(hermann) Gruppen=1000(hermann),24(cdrom),...,27(sudo),...
```