Linux Rechte

Anwenderarten

Linux kennt 3 Anwender Arten:

- Eigentümer einer Resource: **user**
- Mitglied einer Gruppe: **group**
- Alle anderen: **other**

Berechtigungen

- r– Lesen von Dateien oder Verzeichnissen.
- -w- Dateien oder Verzeichnisse Ändern:
  - Anlegen, überschreiben, löschen.
- --x Datei ausführen, im Verzeichnis wechseln.

Für jeder Datei oder Verzeichnis sind die Zugriffsrechte für den Anwender, eine Gruppen und für alle anderen festgelegt.

User mit Home Verzeichnis anlegen

```
# useradd -D test
# ls -ld /home/test
drwx------ 3 test test 4096 28.02.2023
```

User mit gegebene uid und gid anlegen:

```
# groupadd -g 6000 anton
# useradd -u 6000 -g 6000 -m anton
# ls -ld /home/anton
drwx------ 3 anton anton 4096 28.02.2023
```

Mit der Option -D wurde das Home Verzeichnis angelegt.
Bei der zweite Variante wird es die option -m gegeben.

- **d** besagt, dass es sich um ein Verzeichnis handelt.
- **rwx** ist nur für USER gesetzt, anderen haben hier nicht zu suchen.

Gruppen Zugehörigkeit und default shell, Passwort vergeben / ändern

```
# usermod -G wheel -s /bin/ksh  # Unter Linux ist die Shell meistens die bash
# passwd test
Geben Sie ein neues Passwort ein:
Unsicheres Passwort: Das Passwort ist kürzer als 8 Zeichen
Geben Sie das neue Passwort erneut ein:
passwd: alle Authentifizierungsmerkmale erfolgreich aktualisiert.
```

Der User darf Verwaltungsaufgaben vornehmen, deswegen wurde ihn die Mitgliedschaft in der Gruppe **wheel** eingeräumt.

Als User Test einlogen

```
$ su - test
$ ls -la
insgesamt 40
drwx------  4 test test 4096 27. Feb 16:08 .
drwxr-xr-x. 4 root root 4096 27. Feb 15:44 ..
-rw-------  1 test test   43 27. Feb 16:01 .bash_history
-rw-r--r--  1 test test   18  2. Jan 12:43 .bash_logout
-rw-r--r--  1 test test  141  2. Jan 12:43 .bash_profile
-rw-r--r--  1 test test  492  2. Jan 12:43 .bashrc
drwx------  2 test test 4096 27. Feb 16:08 .cache
-rw-r--r--  1 test test  172 19. Sep 12:42 .kshrc
drwxr-xr-x  4 test test 4096 22. Jul 2022  .mozilla
-rw-------  1 test test   48 27. Feb 16:08 .xauthPPajCu
$ id
uid=1001(test) gid=1001(test) Gruppen=1001(test),10(wheel)
```

Sonder Rechte

- `--s------` Ausführen mit der Eingentümerrechte.
- `-----s---` Ausführen mit der Gruppenrechte. Bei Verzeichnisse Vererbung der Gruppe.
- `--------t` Wenn eine Verzeichnis diesen Flag hat können Dateien nur vom Eigentümer der Datei gelöscht werden.
- `--------t` Früher: ausführbare Datei verbleibt im RAM, schnellere Ladezeit.

Was bedeutet ein großes S oder T bei der Ausgabe von ls -l

```
$ chmod 7666 u0
$ ls -ln u0
-rwSrwSrwT  1 1000 1000 0 28. Feb 17:17 u0
```

Die S-Bits für Owner (SUI) und Group (SGID) sind gesetzt. jedoch nicht das x-bit. das gleiche gilt für das t-bit (Sticky bit).

Beispiel: /tmp Verzeichnis

```
$ ls -ld /tmp
drwxrwxrwt 27 root root 820 28. Feb 16:20 /tmp
```

Beispiel: /usr/bin Verzeichnis

```
$ cd /usr/bin/; ls -l | egrep '^-..s|^-.....s'
-rwsr-xr-x  1 root root          62336 18. Jan 01:00 at
-rwsr-xr-x  1 root root          74640  6. Mär 01:00 chage
-rws--x--x  1 root root          28696 21. Jan 01:00 chfn
-rws--x--x  1 root root          24552 21. Jan 01:00 chsh
-rwsr-xr-x  1 root root          53768 19. Jan 01:00 crontab
-rwxr-sr-x  1 root mail          24744 19. Jan 01:00 dotlockfile
-rwsr-xr-x  1 root root          49248 21. Jan 01:00 mount
-rwsr-xr-x  1 root root          38192  6. Mär 01:00 newgrp
-rwsr-xr-x  1 root root          32760 19. Jan 01:00 passwd
-rwsr-xr-x  1 root root          32704 30. Mär 02:00 pkexec
-rwxr-sr-x  1 root plocate      323360 20. Jan 01:00 plocate
-rwxr-sr-x  1 root screen       517824 21. Jan 01:00 screen
-rwsr-xr-x  1 root root          58144 21. Jan 01:00 su
---s--x--x  1 root root         202336  1. Mär 01:00 sudo
-rwsr-xr-x  1 root root          36896 21. Jan 01:00 umount
-rwxr-sr-x  1 root tty           24568 21. Jan 01:00 write
```

Capabilities, besondere Zugriffrechte für Programme

```
$ cd /usr/bin/; getcap *
arping cap_net_raw=p
clockdiff cap_net_raw=p
newgidmap cap_setgid=ep
newuidmap cap_setuid=ep
```

Die Capabilities erlauben es feingranulare Berechtigungen bestimmte Programme zu geben.

Gemeinsamen Ordner mit Guppenrechte

```
# mkdir /video
# chmod 3770 /video
# chgrp video /video
# ls -lnd /video
drwxrws--T 0 39 2 10. feb 08:00 /video
```

Anwender, die Mitglied der Gruppe video sind dürfen Dateien und Ordner anlegen, die erzeugte Kinder erben die Gruppe vom übergeordnete Verzeichnis. Das SGID Bit wir auch an Verzeichnisse vererbt.

Das Sticky Bit wird nicht vererbt!

User und Gruppen Administration

- useradd
- usermod
- userdel
- groupadd
- groupmod
- groupdel

Siehe Man Pages!

Rechte beim Anlegen einer Datei

```
$ umask
0022
$ > u
$ ls -ln u
-rw-r--r-- 1 1000 1000 0 28. Feb 17:02 u
$ umask 0000
$ > u0
$ ls -ln u0
-rw-rw-rw- 1 1000 1000 0 28. Feb 17:07 u0
```

Die “2” verhindern, dass die w bits für group und other gesetzt werden. Dies kann aber geändert werden.

Dateien und Ordner Rechte ändern

```
$ chmod 777 /tmp/u0
$ ls -ln u0 /tmp/u0
-rwxrwxrwx 1 1000 1000 0 28. Feb 17:07 u0
$ chmod ugo=rw  /tmp/u0
$ ls -ln u0 /tmp/u0
-rw-rw-rw- 1 1000 1000 0 28. Feb 17:07 u0
$ chmod u+x  /tmp/u0
$ chmod g-w  /tmp/u0
ls -ln u0 /tmp/u0
-rwxr--r- 1 1000 1000 0 28. Feb 17:07 u0
$ chmod -R a=rwX /tmp/dir
$ls -ldn
drwxrwxrwx 2 1000 1000 0 28. Feb 18:07 dir
$ ls -ln dir
-rw-rw-rw- 2 1000 1000 0 28. Feb 18:03 file
```

Das “X” im letzte Befehl besagt setze das “x” bit für Directories, lasse das “x” bit, bei Dateien unverändert.

Eigentum

- **chown** *kann auch Eigentümer und Gruppe gleichzeitig* ändern (own user:group).
- **chgrp**

UID - GID

```
$ id
uid=1001(test) gid=1001(test) Gruppen=1001(test),10(wheel)
$ newgrp wheel
$ id
uid=1001(test) gid=10(wheel) Gruppen=1001(test),10(wheel)
$ > w
$ ls -ln w
-rw-r--r-- 1 1001 10  0 12. Feb 07:17 w
```

Die Gruppen zugehörigkeit neu angelegte Verzeichnissen oder Dateien wird entsprechend der primäre Gruppe gesetzt, hier die Gruppe wheel.

Rechte bei neue Dateien und Verzeichnissen

- **Dateien uid** und **gid** des Anwenders werden entspechend das **umask** gesetzt, das **x** Recht wird nicht gesetzt.
- **Verzeichnisse** wie Dateien, das **x** Recht wird jedoch gesetzt.

Posix ACL

- verschienene Anwender können auf bestimmte Daten gleichberechtigt zurückgreifen oder unterschieldieliche Rechte erhalten.
- Basieren auf das UNIX Rechtemodel (User, Group, Other mit rwx Rechte).
- Die Berechtigungen der Elternverzeichnis werden vererbt.
- Viel flexibel als das Standard Rechte Model.

ACL Lesen und Setzen

- **getfacl**
- **setfacl**

Die Rechte werden entsprechend der Posix ACL vererbt. Dateien und Ordner können mehrere User und / oder Gruppen angehören.

ACL Syntax

- d:user|group|other:Recht
  - rechte die vererbt werden.
  - Beispiel **d:u:anna:rw-,d:u:bernd:r-- <Verzeichnis>**
    Anna darf Lesen und Schreiben, Bernd nur Lesen.
- user|group|other:Recht
  - Rechte für eine bestimmte Datei oder Verzechnis festlegen.
  - Beispiel **u:bernd:rw- <Datei>**

ls und ACL

```
$ mkdir dir
$ chmod o=rwxt dir
$ ls -ln dir
drwxrwxrwt 2 1000 1000 40 28. Feb 17:20 dir
$ setfacl  -m d:u::rwx dir
$ getfacl dir
# file: dir
# owner: me
# group: me
# flags: --t
user::rwx
group::rwx
other::rwx
default:user::rwx
default:group::rwx
default:other::rwx
$ ls -lnd dir
drwxrwxrwt+ 2 1000 1000 40 10. Feb 17:20 dir
```

ACL Mask

- Mit der Maske wird für Anwender, Gruppe und Andere die maximale Rechte auf der Wert der Mask begrenzt.
    - Formel: Recht & Mask. Recht = rwx, Mask = r-- ⇒ r--
- Die Maske ist nicht wirklich sinnvoll, sie sollte nicht gesetzt werden

Die Rechte werden entsprechend der Maske eingeschränkt. Hier wird schreiben untersagt. Dies kann zu unschöne Effekte führen.

Baum stuktur mit ACL nachträglich versehen

```
# ls -ld video
drwxrwx--- 3 root root 4096 Feb 12 12:43 video
# chgrp -R video video # ev. video/* video
# setfacl -R -n -m d:u::rwx,d:g:video:rwx,g:video:rwx video
# # Dateien und Verzeichnis rechte korrigieren mit:
# chmod -R g-x video
# chmod -R g+X video
```

Die Maske wird nur aud Verzeichnisse angewand, ohne **d:m::rwx** wären die Dateien mit das Ausführungsrecht für der Gruppe versehen.

ACL Einträge entfernen

Alles:

```
$ setfacl -R -b <Verzeichnis>
```

Ein bestimmten User:

```
$ setfacl -R -x d:u:anton <Verzeichnis>
$ setfacl -x u:anton <Verzeichnis oder Datei>
```

Beispiel Familie Unternehmen ACL: Besitzer und Mitgliedschaften

Anwender	Gruppen Mitgliedschaft	Rechte $HOME
vater	vater, eltern	rwx------
mutter	mutter, eltern	rwx------
tochter	tochter, kinder	rwx------
sohn	sohn, kinder	rwx------

Ordner	Gruppe	Rechte
grusel	eltern	rwx
Zeichentrick	eltern kinder	rwx rx

Beispiel Familie Unternehmen: ACL der Verzeichnissen setzen

```
# setfacl -m d:u::rwx,d:g:eltern:rwx,d:o:- grusel
# setfacl -m d:u::rwx,d:g:eltern:rwx,d:o:-,d:g:kinder:rx zeichentrick
```

```
# getfacl zeichentrick
# file: zeichentrick
# owner: root
# group: root
user::rwx
group::r-x
group:eltern:rwx
group:kinder:r-x
mask::rwx
other::---
default:user::rwx
default:group::r-x
default:group:eltern:rwx
default:group:kinder:r-x
default:mask::rwx
default:other::---
```

Beispiel reiche Familie

Die Kinder haben ein Betreuer oder betreuerin, diese Person darf die Gruselfilme sehen und Zeichentrickfile einstellen.

```
$ setfacl -R -m d:g:betreuer:r-x grusel
$ setfacl -R -m d:g:betreuer:rwx zeichentrick
$ getfacl zeichentrick
# file: zeichentrick
...
group:eltern:rwx
group:kinder:r-x
group:betreuer:rwx
...
default:group:eltern:rwx
default:group:kinder:r-x
default:group:betreuer:rwx
...
```

Diese Rechte werden den bereits vorhandenen hinzugefügt.

Sohnemann den Zugriff verweigern/erlauben

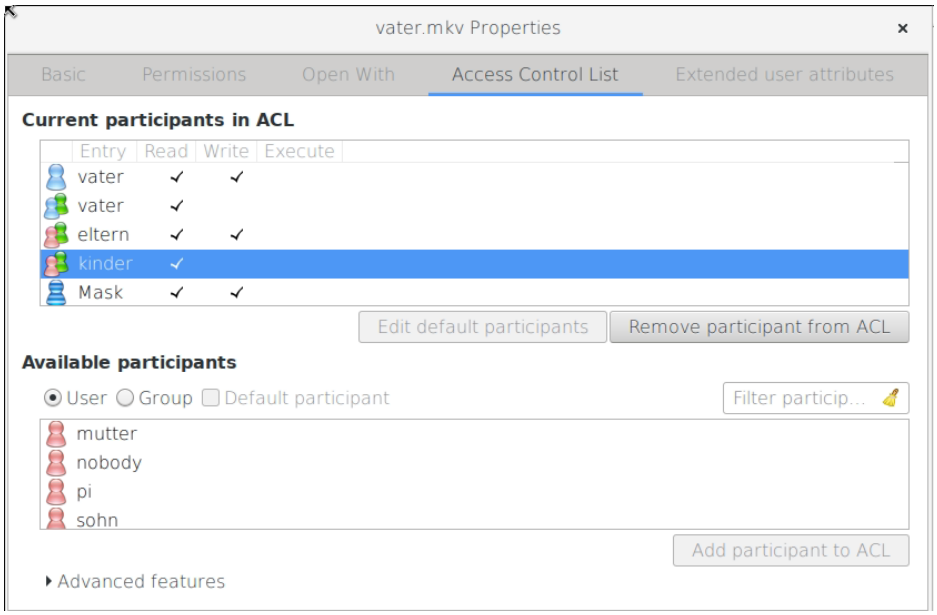
Login verweigern

```
# usermod -L sohn
```

Login wieder erlauben

```
# usermod -U sohn
```

ACL und GUI



eiciel

Dolphin der KDE Datei-Manager kann auch verwendet werden. Für einige weitere Desktop (MATE, ...) kann Eiciel als Plugin installiert werden.

Anwender und Gruppen im Netzwerk

- User und Gruppen sind mit ein 32 Bit Nummer versehen. Bei den heutigen Linux Distributionen ist das Bereich 0 bis 999 für System Accounts und Gruppen reserviert.
- Der erste Anwender erhält üblicherweise die UID 1000.
- Sollen Dienste von verschiedene Anwendern im Netzwerk mittels NFS v3 in Anspruch genommen werden, sollte beim Anlegen der Anwendern eine Netzwerkweite passende UID vorgegeben werden.

Die Sicherheit von NFS v3 ist nicht so gut, der Transfer erfolgt unverschlüsselt. Die Autorisierung erfolgt anhand der UID und GID.

UID und GID einem Anwender nachträglich ändern

```
# usermod -u 6001 anton
# groupmod -g 6001 anton
# chgrp -R anton /home/anton
```

Für Dateien außerhalb von $HOME müssen die UID und GID manuell angepasst werden.

NFS v3 - Server

Datei **/etc/exports**

```
/share  *(rw,sync,no_subtree_check)
```

NFS v3 - Client

Von der Kommando Zeile einbinden:

```
# mount -t nfs -o nfsvers=3,acl raspi:/share /mnt
```

oder in der Datei **/etc/fstab**, es wird beim hochfahren automatisch eingebunden

```
raspi:/share  mnt  nfs  nfsvers=3,acl  0  0
```

Samba geht auch

Untreuen können auch mit ein Windows System auf die Freigaben zurückgreifen.
Es wäre aber gut die ACL auf den Anwender zu beziehen:

```
setfacl -R -b *
setfacl -R -m d:u:vater:rwx,d:u:mutter:rwx,u:vater:rwx,u:mutter:rwx,d:o::---,o::---
setfacl -R -m d:u:sohn:r-x,d:u:tochter:r-x,u:sohn:r-x,u:tochter:r-x zeichentrick
chmod -R a-x *
chmod -R ug+X *
```

Samba Vorbereitung

Für sämtliche Anwender nachstehendes ausführen:

```
# smbpasswd -a <USER>
```

smb.Conf anpassen

```
[global]
    ...
    vfs objects = acl_xattr
    map acl inherit = yes
    store dos attributes = yes
[Familie]
    path = /share
    writable = yes
    browseable = yes
    valid users = vater, mutter, @eltern, sohn, tochter, @kinder
    write list = vater, mutter, @eltern
```

Samba Server für Windows sichtbar machen

- **wsdd** Paket installieren.
- **wsdd** enablen und starten.

Vor- und Nachteile von SMB und NFSv3

- NFSv3 ist nicht verschlüsselt.
    - In eine sichere Umgebung ist es kein gravierenden Nachteil.
    - Mittels ein VPN wie Wireguard kann NSFv3 verschlüsselt werden.
- NFSv4 bietet eigene ACL, ist verschlüsselt jedoch komplizierten zu verwalten als NFSv3
- NFS ist für Windows HOME nicht vorhanden.
- Windows Systeme sind Case insensitive, **O**rdner und **o**rdner sind identisch!
- Windows Share werden nicht im Linux-Dateimanager angezeigt.
    - Sie können jedoch eingebunden werden, Eintrag in fstab.
    - Im Datei Manager die Adresse smb://<windows-hostname|IP> eintragen, die mögliche getreilten Ordner werden angezeigt.
- Windows kennt zwar die POSIX-ACL, die Gruppen ACL werden jedoch nicht beachtet.
- SMB wird von alle gängigen Betriebssysteme unterstüzt.