

Übung: finde die minimal nötigen Berechtigungen

```
mkdir -m 000 /tmp/dir1 /tmp/dir2
echo geheim > file1
chmod 000 file1
```

Welche Berechtigungen müssen jeweils bei dir1, dir2 und file1 hinzugefügt werden, damit file1 mit

```
cp file1 dir1/
```

ins Verzeichnis dir1 kopiert werden kann?

Welche Berechtigungen müssen jeweils bei dir1, dir2 und file1 hinzugefügt werden, damit file1 mit

```
mv /tmp/dir1/file1 /tmp/dir2/
```

ins Verzeichnis dir2 verschoben werden kann?

Übung: mit umask Berechtigung setzen

Welche umask muss man setzen, damit neu angelegte

- Dateien die Berechtigung rw- r- -w- und
- Verzeichnisse die Berechtigung rwx r-x -w-

bekommen.

Befehle Dateirechte

Dateizugriffsrechte betrachten	ls -l datei
Verzeichnisrechte betrachten	ls -ld verzeichnis
Besitzer der Datei ändern	chown benutzer datei
Gruppe der Datei ändern	chown :users datei chgrp users datei

SUID Recht für die Datei /bin/cat setzen	Programm starten und real bzw. effective UID ansehen: <code>chmod u+s /bin/cat</code> Als Nutzer: <code>cat &</code> <code>ps -C cat -o cmd,ruser,euser</code>
Wo darf ich schreiben?	<code>find / /dev -xdev -writable -ls</code>
Was darf eine Gruppe?	<code>find / /dev -xdev -group users -ls</code>
Alle SUID-root Dateien finden	<code>find / -xdev -type f -user root -perm /4000 -ls 2>/dev/null</code> <code>find / -xdev -type f -perm -u=s -ls</code>
Allen Programmen das Ausführungsrecht nehmen	<code>find verzeichnis/ -type f -perm /0111 -exec chmod a-x {} +</code>

Beispiel: Schreibrechte im Verzeichnis - mehr als man denkt

```
mkdir /test
id nutzer17
```

```
uid=1001(nutzer17) gid=100(users) groups=100(users),16(dialout),33(video)
```

```
id iw
```

```
uid=1000(iw) gid=100(users)
groups=100(users),16(dialout),17(audio),33(video)
```

```
chown iw:users /test/
chmod g+w /test/
ls -ld /test/
```

```
drwxrwxr-x  2 iw users 4096 Oct 10 17:30 /test/
```

```
su - iw
cat <<EOF > /test/unveraenderbar.txt
```

Dies ist ein unveraenderlicher Text

```
EOF
chmod u=rw,g=r,o=r /test/unveraenderbar.txt
logout
su - nutzer17
ls -l /test/unveraenderbar.txt
```

```
-rw-r--r--  1 iw users 36 2005-10-10 17:32 /test/unveraenderbar.txt
```

```
vi /test/unveraenderbar.txt
```

```
ls -la /test/
```

```
insgesamt 12
drwxrwxr-x  2 iw      users 4096 2005-10-10 17:35 .
drwxr-xr-x 22 root    root  4096 2005-10-10 17:30 ..
-rw-r--r--  1 nutzer17 users  34 2005-10-10 17:35 unveraenderbar.txt
```

Übung: Weniger Rechte für Eigentümer und Gruppe

```
/tmp/datei356 :
```

```
#!/bin/bash
echo executable
```

```
chmod 356 /tmp/datei356
```

Wer darf was?

rwx	Mitglied der Gruppe	nicht Mitglied der Gruppe
Besitzer	---	---
nicht Besitzer	---	---

Beispiel: Gruppenzugehörigkeiten kleben am Prozess länger als man denkt

```
groupadd projekt
useradd -m -G projekt iw
touch /tmp/datei
chown root:projekt
chmod g+w /tmp/datei
su - iw
id iw
echo test1 >> /tmp/datei
su -
usermod -G users iw
id iw
logout
id iw
echo test1 >> /tmp/datei
```

Beispiel: Ungewollter Eigentümerwechsel nach dem Löschen eines Nutzers

```
useradd -m chef
userdel chef
useradd -m raumpfleger
ls -l /home
```

Beispiel: Auswirkung von mount-Optionen

```
mkdir /mnt/sda2
mount -o ro /dev/sda2 /mnt/sda2
touch /mnt/sda2/test
```

Weitere Mount-Optionen mit Auswirkungen auf Dateirechte:

Option	Bedeutung
nodev	Geräte Dateien sind nicht erlaubt
noexec	Ausführbare Dateien sind nicht erlaubt
nosuid	S-BIT wird ignoriert
ro	Dateien sind nicht veränderbar

Beispiel: Ungewollter Eigentümerwechsel bei Backup und Restore

Server A

```
useradd -u 2000 nutzer_a
useradd -u 2001 nutzer_b
useradd -u 2002 nutzer_c
useradd -u 2003 nutzer_d
```

Server B

```
useradd -u 2000 nutzer_a
useradd -u 2003 nutzer_b
useradd -u 2001 nutzer_e
```

```
mkdir /tmp/backup
```

Server A

```
mkdir /tmp/workdir
```

```
touch /tmp/workdir/file_{a,b,c,d}  
chown nutzer_a /tmp/workdir/file_a  
chown nutzer_b /tmp/workdir/file_b  
chown nutzer_c /tmp/workdir/file_c  
chown nutzer_d /tmp/workdir/file_d
```

```
rsync -a /tmp/workdir/file_* server_b:/tmp/backup
```

Server B

```
ls -l /tmp/backup
```

Server A

```
mkdir /tmp/restore  
rsync -a server_b:/tmp/backup/file_* /tmp/restore  
ls -l /tmp/restore
```

Links

- [Sind ACLs unter Windows "besser"?](#)
- [Erste Schritte mit posix-acls](#)
- [Detaillierte Beschreibung Linux ACLs](#)

From:

<https://wiki.lab.linuxhotel.de/> - **Linuxhotel Wiki**

Permanent link:

<https://wiki.lab.linuxhotel.de/doku.php/lpi1:dateirechte?rev=1508223835>

Last update: **2017/11/22 11:20**

