

# Benutzer und Rechte Teil 3

## Linux-Kurs der Unix-AG

Zinching Dang

11. Juli 2017



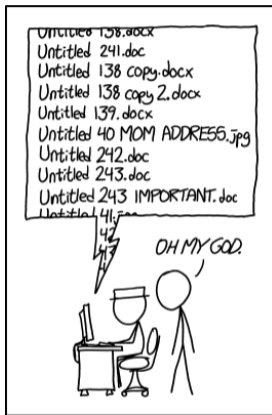
**Wiederholung: Benutzer & Gruppen**

**Zugriffsrechte & Besitzverhältnisse**

**Befehle & Optionen**

**Zusammenfassung & Ausblick**

# Übersicht



PROTIP: NEVER LOOK IN SOMEONE ELSE'S DOCUMENTS FOLDER.

<https://xkcd.com/1459>

# Wiederholung: Benutzer & Gruppen

## Benutzer

- ▶ haben eine eindeutige UID
  - ▶ Systembenutzer:  $UID < 1000$
  - ▶ „echte“ Benutzer:  $UID \geq 1000$
- ▶ haben eine primäre Gruppe (GID) und beliebig viele sekundäre Gruppen

## Gruppen

- ▶ haben eine eindeutige GID
- ▶ können beliebig viele Benutzer enthalten

# Wiederholung: Benutzer & Gruppen

## Systembenutzer `root`

- ▶ hat die UID 0
- ▶ ist zur Systemverwaltung gedacht

## Befehle als `root` ausführen

- ▶ mit `sudo`
- ▶ mit `su`

# Wiederholung: Benutzer & Gruppen

## Benutzer- und Gruppendatenbanken

- ▶ `/etc/passwd` – Benutzerdatenbank
  - ▶ `useradd` , `usermod` , `userdel`
- ▶ `/etc/group` – Gruppendatenbank
  - ▶ `groupadd` , `groupdel`
- ▶ `/etc/shadow` – Passwortdatenbank
  - ▶ `passwd`

# Zugriffsrechte und Besitzverhältnisse

## Allgemeines

- ▶ verschiedene Zugriffsrechte für Dateien und Verzeichnisse
  - ▶ Leserecht (**r**ead): Datei lesen/Verzeichnis öffnen
  - ▶ Schreibrecht (**w**rite): Datei verändern/Dateien anlegen oder löschen
  - ▶ Ausführungsrecht (**e**xecute): Datei als Programm ausführen/Verzeichnisse betreten und auf Inhalt zugreifen
- ▶ Dateien und Verzeichnisse haben Zugriffsrechte für
  - ▶ Besitzer: **u**ser
  - ▶ Gruppe: **g**roup
  - ▶ alle anderen: **o**thers
- ▶ das spezifischste Zugriffsrecht wird angewandt

# Zugriffsrechte und Besitzverhältnisse

## Zugriffsrechte und Besitzverhältnisse bestimmen

```
linuxkurs@ubuntu:~$ls -hl Dokumente/  
insgesamt 90K  
-rw-rw-r-- 1 linuxkurs proj1 6K Nov 6 Linux  
-rwxr-xr-x 1 linuxkurs users 80K Nov 6 ist  
drwxr-x--- 1 linuxkurs proj2 4K Nov 6 toll
```

Zugriffsrechte für  
Besitzer    andere  
          Gruppe

Besitzer    Gruppe



# Besitzverhältnisse im Detail

## Datei-Besitzer

- ▶ können Zugriffsrechte und Gruppen-Besitzer festlegen
- ▶ können in Verzeichnissen Dateien fremder Besitzer löschen
- ▶ können den Datei-Besitzer **nicht** ändern

## Gruppen-Besitzer

- ▶ wird vom Datei-Besitzer festgelegt
- ▶ hat die von ihm festgelegten Zugriffsrechte

# Besitzverhältnisse ändern – chown

## Allgemeines

- ▶ nur der Datei-Besitzer darf Gruppen-Besitzer ändern
- ▶ Dateien und Verzeichnisse können nicht „verschenkt“ werden
- ▶ nur `root` darf Datei-Besitzer ändern
- ▶ Argument: Besitzer und/oder Gruppe und Datei/Verzeichnis

## Wichtige Optionen

- ▶ `-R` : rekursiv auf ein Verzeichnis ausführen

# Besitzverhältnisse ändern – chown

## Beispiel

- ▶ `chown user1:group1 datei`
  - ▶ Besitzer und Gruppe festlegen
- ▶ `chown :group2 datei`
  - ▶ nur Gruppe festlegen
- ▶ `chown user1 datei`
  - ▶ nur Besitzer festlegen
- ▶ `chown user1: datei`
  - ▶ Besitzer festlegen, Gruppe auf die GID des Besitzers setzen

# Zugriffsrechte im Detail

## Zugriffsrechte

- ▶ sind für Dateien und Verzeichnisse definiert
  - ▶ Leserecht: **r** bzw. Wertigkeit 4
  - ▶ Schreibrecht **w** bzw. Wertigkeit 2
  - ▶ Ausführungsrecht **x** bzw. Wertigkeit 1
- ▶ können symbolisch oder oktal dargestellt werden
  - ▶ `rwxr-x---` – Dreier-Gruppen für Besitzer, Gruppe und alle anderen
  - ▶ `7 5 0` – Summe der Wertigkeiten anstatt einer Dreier-Gruppe
- ▶ werden mit `chmod` verändert

# Zugriffsrechte ändern – chmod

## Allgemeines

- ▶ nur der Besitzer darf Zugriffsrechte ändern
- ▶ Argumente: Zugriffsrechte und Datei/Verzeichnis
  - ▶ Zugriffsrechte in symbolischer oder oktalen Notation

## Wichtige Optionen

- ▶ `-R` : rekursiv auf ein Verzeichnis ausführen

# Zugriffsrechte – Notationen

## Symbolisch Notation

- ▶ für die Gruppen Besitzer (**u**ser), Gruppe (**g**roup) und alle anderen (**o**thers) werden die entsprechenden Rechte gesetzt
- ▶ können festgelegt (=), hinzugefügt (+) oder entzogen (-) werden

## Beispiel

- ▶ `chmod u=rwx,g=rx,o= datei`

# Zugriffsrechte – Notationen

## Symbolisch Notation

- ▶ für die Gruppen Besitzer (**u**ser), Gruppe (**g**roup) und alle anderen (**o**thers) werden die entsprechenden Rechte gesetzt
- ▶ können festgelegt (=), hinzugefügt (+) oder entzogen (-) werden

## Beispiel

- ▶ `chmod u+rwx,g+rx,o-rwx datei`

# Zugriffsrechte – Notationen

## Oktale Notation

- ▶ je eine Ziffer  $\in [0 - 7]$  für die Gruppen Besitzer (**u**ser), Gruppe (**g**roup) und alle anderen (**o**thers)
- ▶ übliche Nutzung mit direkter Festlegung der Rechte für die jeweiligen Gruppen

## Beispiel

- ▶ lesen, schreibe und ausführen:  $4 + 2 + 1 = 7$



# Zugriffsrechte – Notationen

## Oktale Notation

- ▶ je eine Ziffer  $\in [0 - 7]$  für die Gruppen Besitzer (**u**ser), Gruppe (**g**roup) und alle anderen (**o**thers)
- ▶ übliche Nutzung mit direkter Festlegung der Rechte für die jeweiligen Gruppen

## Beispiel

- ▶ `chmod 750 datei`

# Wichtige Befehle & Optionen

Befehl	Optionen	Funktion
<code>chown</code>	<code>-R</code>	Besitzer und Gruppe ändern
<code>chmod</code>	<code>-R</code>	Zugriffsrechte ändern

Zugriffsrecht	Wertigkeit
<b>r</b> ead	4
<b>w</b> rite	2
<b>e</b> xecute	1

# Zusammenfassung & Ausblick

## Zusammenfassung

- ▶ Besitzverhältnisse und Zugriffsrechte von Dateien und Verzeichnissen feststellen und ändern

## Nächstes Mal

- ▶ Prozesse und Logs