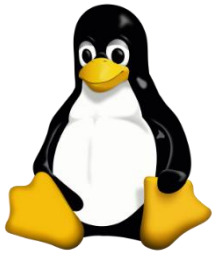
Benutzerverwaltung
Rechte – Teil 1



Benutzerverwaltung

- Typische Aufgaben im Zusammenhang mit Benutzerverwaltung
  - das Anlegen und Löschen von Benutzerkonten,
  - die Prüfung der Qualität von Passwörtern (Password Policy festlegen),
  - Änderungen von Passwörtern, welche die Benutzer vergessen haben und
  - die Überwachung des von Anwendern belegten Speicherplatzes

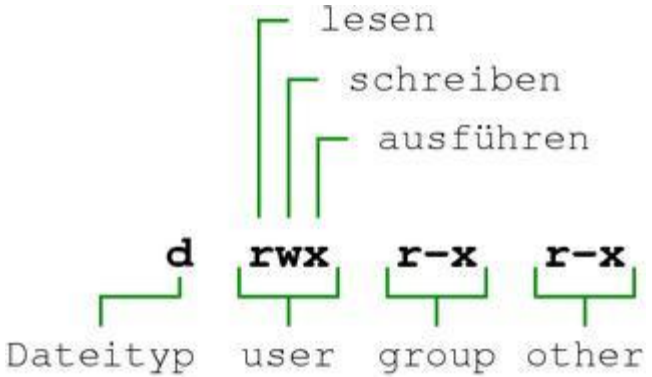
Standardrechte im Dateisystem

- jede Datei besitzt einen Eigentürmer (u) und ist einer Gruppe (g) zu geordnet
- für diese beiden sowie für alle anderen (o) werden nun jeweils drei Rechte vergeben oder verweigert: lesen (r), schreiben (w) und ausführen (x)
- nur der Eingentürmer oder root kann diese Rechte verändern
- mit “ls -l” kann man die Rechte anzeigen
- ls -l

```
drwxr-xr-x  1 proell    users     4096     2008-01-04 13:15 backup
drwxr-xr-x  1 mayer     users     4096     2008-03-22 23:47 ue1
-rwxr-xr-x  1 root      root      60       2008-01-07 00:00 cc2
------------  - ------------  ------------ ------ ---------------------- ---------
  Rights      Owner    Group    Size      Timestamp          Name
```

Zugriffskontrolle I

- 3 Klassen von Benutzerarten
  - Besitzer (Erzeuger) (***user)***
  - Gruppe (***group)***
  - ***Rest der Welt*** (***other)***
- 3 Arten von Zugriffen
  - lesen (**r**ead)
  - schreiben (**w**rite)
  - ausführen (e**x**ecute)
- Diese können unabhängig voneinander vergeben werden (nur vom Besitzer bzw. root)



Zugriffskontrolle II

Recht	Datei	Verzeichnis
r	Datei kann gelesen werden	Verzeichnisinhalt kann angezeigt werden (ls)
w	Datei kann geschrieben werden	Verzeichniseinträge können erstellt und verändert werden (mkdir, mv,…)
x	Datei kann ausgeführt werden	In Verzeichnis kann gewechselt werden (cd)

Rechte numerisch I

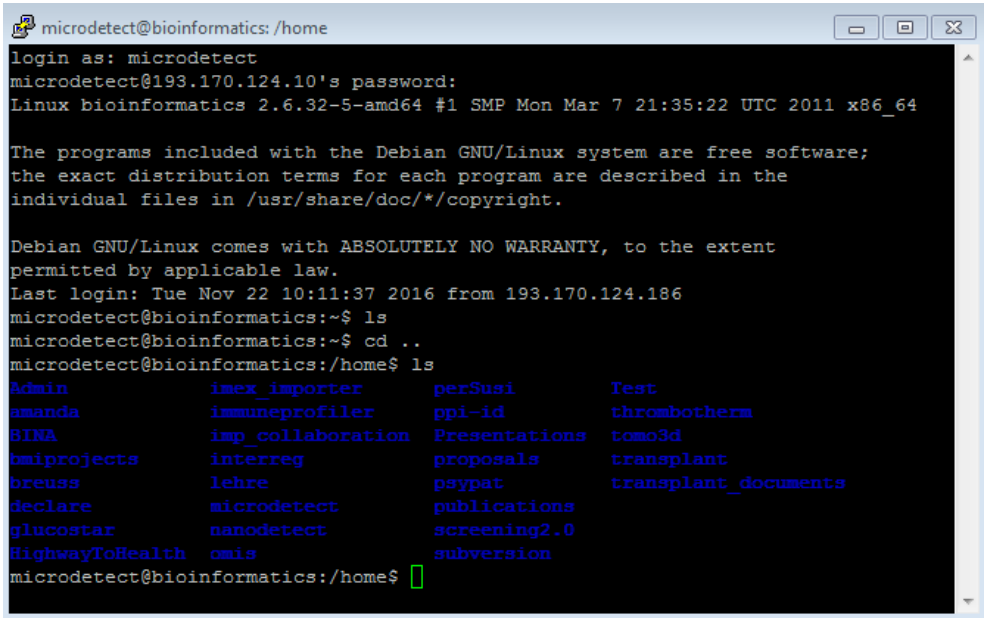
Recht	Oktalzahl
Read	4
Write	2
Execute	1

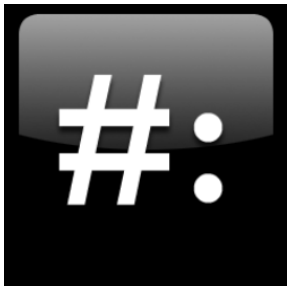
Pro Berechtigungsklasse wird eine Oktalzahl angegeben.
Zugeteiltes Recht ergibt sich aus der Summe der Einzelrechte:

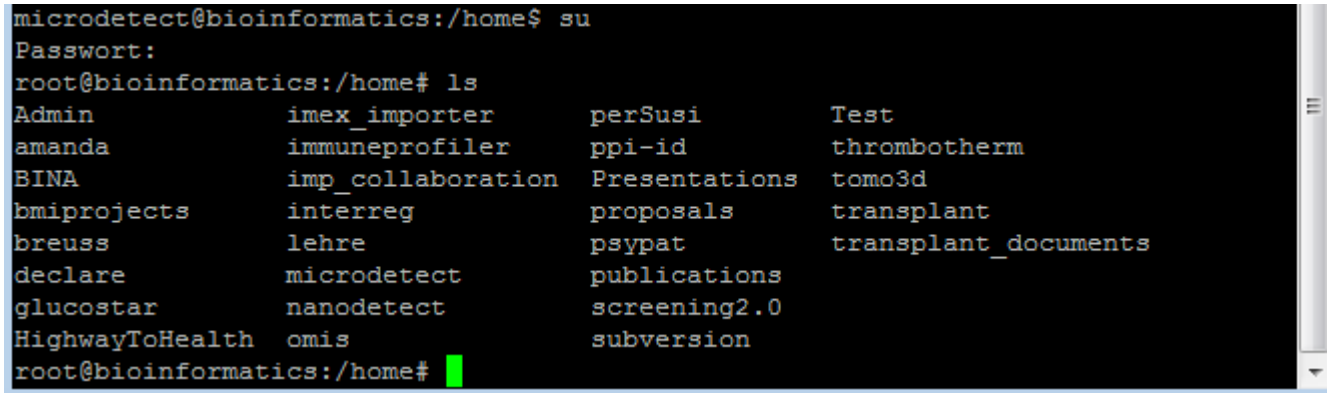
rwx =
read + write + execute ergibt
4+2+1 = 7

777 ergibt rwx rwx rwx (ugo)
754 ergibt rwx r-x r-- (ugo)

Kommando sudo







sudo

- sudo - **s**ubstitute **u**ser **do**
- standardmäßige Vorgehensweise in Ubuntu-Systemen
- Mit sudo lassen sich Programme, die sonst nur mit dem **root Account** ausgeführt werden dürfen, auch von Standardbenutzern starten.
- Damit ist es möglich, Programme mit root-Rechten auszuführen
- Verlangt ein Programm zur Ausführung die Rechte des Systemverwalters, so muss der Benutzer es mit vorangestelltem **sudo** ausführen.
- Beispiel: `sudo cat /etc/shadow`

Voraussetzung - sudo

- **sudo** kann nur von Benutzern aufgerufen werden, die auch dazu berechtigt wurden
- Berechtigungen für die Verwendung von **sudo** werden in der Datei `/etc/sudoers` festgelegt.
- `/etc/sudoers` ist ein normaler Textfile und sollte immer mit dem Befehl `visudo` bearbeitet und nicht mit einem normalen Texteditor. (sudo visudo)
  - gewährt Syntaxüberprüfung
  - erlaubt Zugriff nur durch eine Person
  - Der kleinste Tippfehler kann dazu führen, dass man sich aus dem System aussperrt.

sudo -Passwort

- Verwendet ein Standardbenutzer `sudo`, muss er zunächst sein Passwort eingeben, außer die Eingabe des Passworts ist durch eine Eintragung in `/etc/sudoers` explizit abgeschaltet.
- Normalerweise merkt `sudo` sich das Passwort für 5 Minuten
- Die Frist kann verlängert werden mit: `sudo -v`
  - Zeitstempel wird zurückgesetzt, so dass `sudo` erst nach weiteren 5 Minuten wieder nach dem Passwort fragt.
- Details und Konfiguration `/etc/sudoers`: https://wiki.ubuntuusers.de/sudo/Konfiguration

root versus sudo

- Benutzer **root** ist der oberste Systemadministrator eines Linux-Systems.
  - Er verfügt über die höchstmöglichen Rechte für das installierte System
  - Der **root** Account ist **nicht** für die alltägliche Verwendung des Systems gedacht, sondern nur für besondere Verwaltungsaufgaben, weil umfassende Risiken mit seiner Verwendung verbunden sind.
  - **root** Account darf **„ALLES“!**

- Standardinstallation eines Ubuntu-Systems
  - **root** Account ist **deaktiviert,** da das Passwort auf ungültig gesetzt ist.
  - Anmelden mit root daher unmöglich
  - Um den root Account zu aktivieren, muss man
    für den Benutzer root ein neues Passwort
    setzen
  - Anleitung im Dokument BasicsBenutzerverwaltung

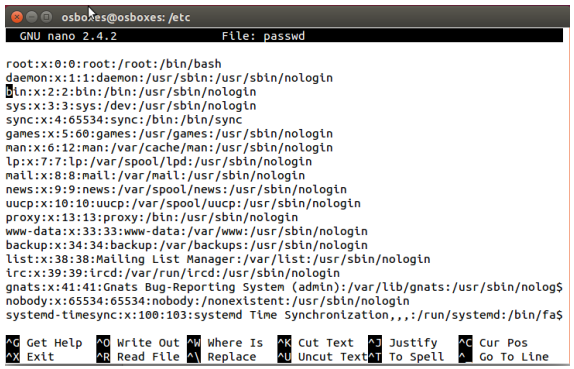
Klassische Benutzerverwaltung Unix

- Zur Verwaltung von **lokalen Benutzerkonten**
- Durch folgende Files realisiert
  - `/etc/passwd`

    Enthält eine Zeile pro Benutzer
  - `/etc/shadow`

    Enthält alle Passworte in verschlüsselter Form
  - `/etc/group`

    Enthält Information über die Zugehörigkeit einzelner Benutzer zu Benutzergruppen



- Graphisches Frontend zur Manipulation von Benutzern vorhanden, aber nicht unbedingt notwendig

Anatomie /etc/passwd

Besteht aus Zeilen in der Form:

`Loginname:Pwd:UID:GID:Kommentar:HomeVZ:LoginShell`

- UID – eindeutige Benutzernummer
- GID – Nummer der Primary Benutzergruppe

Einteilung in Systembenutzer und „menschliche Benutzer“

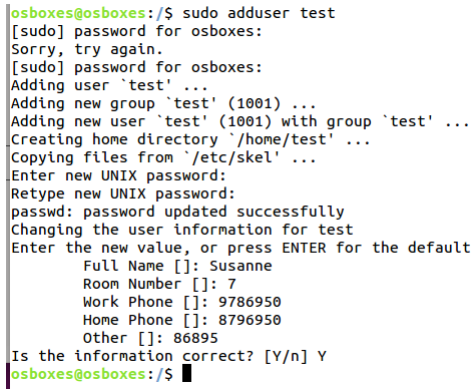
Beispiele:

```
bin:x:1:1:bin:/bin:
daemon:x:2:2:daemon:/sbin:
lp:x:4:7:lp:/var/spool/lpd:
     ……
```

Systembenutzer, dürfen sich nicht anmelden

```
root:x:0:0:root:/root:/bin/bash
doris:x:234:100:Doris Mueller:/home/doris:/bin/bash
vivek:x:235:100:Vinzenz Vektor:/home/vivek:/bin/bash
```

Menschliche Benutzer, dürfen sich anmelden



Normalen Benutzer anlegen Kommando sudo adduser

So liegen die UID und GID von Systembenutzern in diesem Fall zwischen 100 und 999 und die der normalen Benutzer zwischen 1000 und 29999.

/etc/passwd nach Anlegen eines neuen User

osboxes und test sind normale (menschliche) Benutzer

usbmux und lightdm sind Systembenutzer

```
osboxes@osboxes:/$ sudo adduser test
[sudo] password for osboxes:
Sorry, try again.
[sudo] password for osboxes:
Adding user `test' ...
Adding new group `test' (1001) ...
Adding new user `test' (1001) with group `test' ...
Creating home directory `/home/test' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for test
Enter the new value, or press ENTER for the default
        Full Name []: Susanne
        Room Number []: 7
        Work Phone []: 9786950
        Home Phone []: 8796950
        Other []: 86895
Is the information correct? [Y/n] Y
osboxes@osboxes:/$
```

```
usbmux:x:118:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
lightdm:x:119:126:Light Display Manager:/var/lib/lightdm:/bin/false
osboxes:x:1000:1000:osboxes.org,,,:/home/osboxes:/bin/bash
test:x:1001:1001:Susanne,7,9786950,8796950,86895:/home/test:/bin/bash
```

Systembenutzer anlegen
Kommando sudo adduser - -system test2

So liegen die UID und GID von Systembenutzern in diesem Fall zwischen 100 und 999 und die der normalen Benutzer zwischen 1000 und 29999.

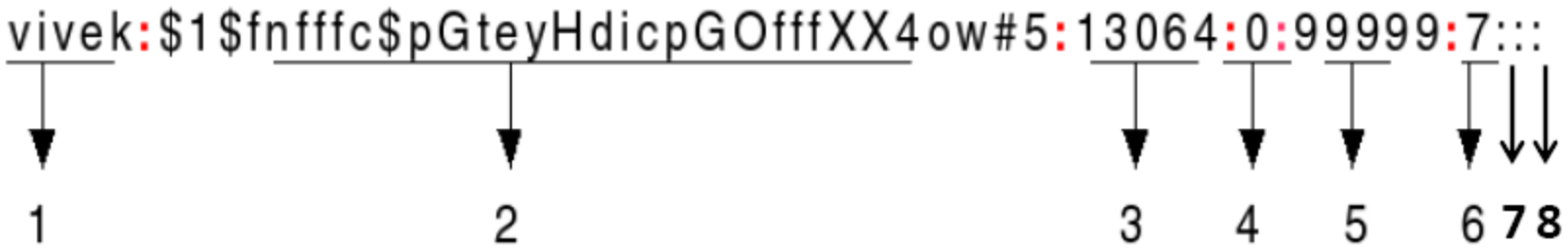
```
osboxes@osboxes:/etc$ sudo adduser --system test2
Adding system user `test2' (UID 120) ...
Adding new user `test2' (UID 120) with group `nogroup' ...
Creating home directory `/home/test2' ...
```

/etc/passwd nach Anlegen eines neuen User
osboxes und test sind normale (menschliche) Benutzer
test2 ist ein Systembenutzer

```
osboxes:x:1000:1000:osboxes.org,,,:/home/osboxes:/bin/bash
test:x:1001:1001:Susanne,7,9786950,8796950,86895:/home/test:/bin/bash
test2:x:120:65534::/home/test2:/bin/false
```

Anatomie /etc/shadow

- Nur Leserechte für Superuser
- Enthält verschlüsselte Passwörter



1 - Der Login Name
2 - Das verschlüsselte Kennwort
3 - Anzahl von Tagen zwischen dem 01.01.1970 und der letzten Kennwortänderung
4 - Zeit in Tagen, die zwischen zwei Kennwortänderungen liegen muss
5 - Zeit in Tagen, wie lange ein Kennwort gültig ist
6 - Zeit in Tagen, wie lange der Benutzer vor dem Auslaufen des Kennworts gewarnt wird
7 - Zeit in Tagen bis das Konto nach dem Auslaufen des Kennworts gesperrt wird
8 - Auslaufen des Kontos in Tagen seit dem 01.01.1970

Anatomie /etc/group

- Jeder Benutzer muss Mitglied in einer Gruppe sein
- Neben der in der passwd-Datei eingetragenen „primary group“ können hier pro Gruppe User angegeben werden, die der Gruppe zusätzlich (via secondary groups) angehören.
- Besteht aus Zeilen in der Form:
  - Gruppename:*:GID:Benutzerliste

Beispiel:

```
dialout:*:16:root,tatiana,steuer,vivek
```

```
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,osboxes
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:osboxes
floppy:x:25:
tape:x:26:
sudo:x:27:osboxes
audio:x:29:pulse
dip:x:30:osboxes
www-data:x:33:
backup:x:34:
```

Passwörter

- Linux speichert Passwörter verschlüsselt
  - Können auch vom Superuser nicht ermittelt werden
- Aus dem Klartext des Passwortes wird mit einer Einweg-Hashfunktion (3DES, MD5, Blowfish) das verschlüsselte Passwort ermittelt
  - wird in /etc/shadow gespeichert
- Einfaches Verfahren zum Knacken der Passwörter
  - alle Möglichkeiten durchprobieren
  - Passwortlänge entscheidend
  - Zeichensatz entscheidend

Zeit zur Ermittlung der Passwörter

Passwortlänge	*Zahl der möglichen Passwörter*	*Zeitbedarf zum Knacken*
1	62	keiner
2	3844	keiner
3	238.328	keiner
4	14.776.336	1,4 Sekunden
5	916.132.832	1,5 Minuten
6	56.800.235.584	1,5 Stunden
7	3.521.614.606.208	4 Tage
8	218.340.105.584.896	8 Monate
9	13.537.086.546.263.552	43 Jahre
10	839.299.365.868.340.224	2660 Jahre

Annahme:

1) 62 verschiedene Zeichen, die 26 lateinischen Groß- und Kleinbuchstaben sowie 10 Ziffern stehen zur Verfügung
2) 10 Millionen Kennwörter pro Sekunde können geprüft werden.

Zeit zur Ermittelung der Passwörter

- Zeichensatz spielt entscheidende Rolle
- Zeitbedarf zum Knacken, bei einer Passwortlänge von 8 Zeichen

Zeichensatz	*Zeichenzahl*	*Zahl der möglichen Pass-wörter*	*Zeitbedarf zum Knacken*
8-Bit ASCII	256	18.446.744.073.709.551.616	58.500 Jahre
7-Bit ASCII	128	72.057.594.037.927.936	228 Jahre
Buchstaben und Ziffern	62	218.340.105.584.896	8 Monate
nur Buchstaben	52	53.459.728.531.456	62 Tage
nur Kleinbuchstaben	26	208.827.064.576	6 Stunden
Wörter aus Wörterbuch	-	ca. 250.000	nahezu keiner