

# Linux Basiswissen

## Agenda

### 1. Distributionen

- [Überblick](#)

### 2. Verzeichnisse und Dateitypen

- [Verzeichnisaufbau](#)
- [Dateitypen](#)

### 3. Basisbefehle

- [In den Root-Benutzer wechseln](#)
- [Wo bin ich ?](#)
- [Praktische Ausgabe von langen Seiten - less](#)
- [Datei anlegen - touch](#)
- [Autovervollständigen \\* und tab](#)
- [Welches Programm wird verwendet](#)

### 4. Erweiterte Befehle (Nice to have)

- [Alias Befehle anzeigen](#)
- [Welche Bibliotheken verwendet ein ausführbares Programm](#)

### 5. Dateien und Verzeichnisse

- [Mit cd im System navigieren](#)
- [Verzeichnisse in Listenansicht mit versteckten Dateien anzeigen -> ls -la](#)
- [Inhalt in Datei schreiben und anhängen](#)
- [Verzeichnisse und Dateien löschen](#)
- [Kopieren/Verschieben/Umbenennen von Dateien und Files](#)

### 6. Prozesse

- [Prozesse anzeigen - ps/pstree -p](#)

### 7. Benutzer, Gruppen und Rechte

- [Rechte](#)
- [Dateien für Benutzer und Gruppen](#)
- [Benutzer anlegen](#)
- [sudo Benutzer erstellen](#)

### 8. Dateimanipulation/Unix Tools

- [cat/head/tail-Beginn/Ende einer Datei anzeigen](#)
- [zcat - Inhalte einer mit gzip komprimierten Datei anzeigen](#)
- [wc - Zeilen zählen](#)
- [Bestimmte Zeilen aus Datei anzeigen - grep](#)
- [Erweiterte Suche mit Grep](#)

### 9. Logs/Loganalyse

- [Logfile beobachten](#)
- [Dienste debuggen](#)

### 10. Variablen

- [Setzen und verwenden von Variablen](#)

## 11. Dienste/Runlevel(Targets verwalten)

- [Die wichtigsten systemctl/service](#)

## 12. Partitionierung und Filesystem

- [parted and mkfs.ext4](#)

## 13. Boot-Prozess und Kernel

- [Grub konfigurieren](#)
- [Kernel-Version anzeigen](#)
- [Kernel-Module laden/entladen/zeigen](#)

## 14. Hilfe

- [Hilfe zu Befehlen](#)

## 15. Grafische Oberfläche und Installation

- [Gnome unter Ubuntu installieren](#)
- [X-Server - Ausgabe auf Windows umleiten](#)
- [Installations-Images-Server](#)

## 16. Wartung und Aktualisierung

- [Aktualisierung des Systems](#)
- [Paketmanager apt/dpkg](#)
- [Archive runterladen und entpacken](#)

## 17. Firewall und ports

- [ufw \(uncomplicated firewall\)](#)
- [firewalld](#)
- [Scannen und Überprüfen mit telnet/nmap](#)

## 18. Netzwerk/Dienste

- [IP-Adresse von DHCP-Server holen \(quick-and-dirty\)](#)
- [Auf welchen Ports lauscht mein Server](#)

## 19. Literatur

- [Literatur](#)

# Distributionen

## Überblick

### Multi-Purpose - Distributionen (Ideal zum Starten)

#### Redhat-Familie

```
Centos
Redhat. - rpm / (yum / dnf)
Fedora
```

#### Debian Familie

```
Debian
Ubuntu. - dpkg / apt
Mint
```

#### SuSE - Familie

```
SLES (SuSE Linux Enterprise)
OpenSuSE
```

### Distris zur Sicherheitsüberprüfung / Hacken

```
Kali Linux
Parrot. - Distributionen zum Hacken
```

### Live-DVD (Linux ohne Installation)

- Knoppix - Live DVD - brauche nicht installieren

### Spezial-Linuxe, z.B. für Router

```
OpenWRT
DDWRT
```

### Seite mit Übersicht aller Linux-Distros

- <https://distrowatch.com/>

# Verzeichnisse und Dateitypen

## Verzeichnisaufbau

### /etc

- Verzeichnis für Konfigurationsdateien

### /dev

- Devices (Alle Gerätedateien - Ein- und Ausgabegeräte, wie bspw. Festplatten, Mouse)

### /mnt

- früher viel verwendet:
- für händisches Einhängen gedacht (per Hand mounten)

### /media

- das neue / moderne (wird heutzutage meistens verwendet)
- Verzeichnis für automatisch eingehängte Devices (z.B. usb-stick)

### /opt

- Große Softwarepaket (z.B. LibreOffice, OpenOffice, Dritt-Anbieter)

### /boot

- Files for booting (e.g. kernel, grub.cfg, initial ramdisk)

### /proc

- Schnittstelle zwischen Kernel und User-Space (für Programme, Benutzer)
- Kommunikation erfolgt über Dateien

### /root

- Heimatverzeichnis des root-Benutzers

### /run

- Dateien mit Prozess-ID für laufenden Services
- um diese gut beenden zu können

### /tmp

- Temporäre Dateien
- Löschen von Dateien kann unter /etc/tmpfiles.d verwaltet werden (erfolgt von systemd auf Tagesbasis)

### /sys

- wie proc
- Schnittstelle zwischen Kernel und User-space

### /var (=variable daten)

- Hier liegen Daten, die sich häufig ändern
- Log-Dateien, Datenbanken, Spool-Dateien, Cache-Dateien

### /lib

- Bibliotheken (.so, .ko) wie unter Windows \*dll's

### /sbin

- Programme zur Systemadministration

## **/bin**

- Normale Programme für alle (executables)

## Dateitypen

### Wo ?

- Erste Spalte bei ls -la

### Welche ?

```
- file
d directory
l symbolischer Link
c Character-Device (Eingabegerät: Zeichenorientiert z.B. Tastatur)
b Block-Device (Ausgabegerät): Blockorientiert, z.B. Festplatte)
```

## Basisbefehle

### In den Root-Benutzer wechseln

```
## einloggen als normaler Benutzer z.B. benutzer: kurs  
sudo su -  
## eingeben des Passworts des Benutzers
```

## Wo bin ich ?

```
## 1. Ich erkenne es am prompt (Beginn der Zeile )
```

```
## pwd - Print working directory
```

```
pwd
```



## Praktische Ausgabe von langen Seiten - less

### Open a file with less

```
##  
less /etc/services  
  
## Why ?  
## Leichtere Navigation
```

### Pipen mit less (ausgabe an less schicken)

```
ls -la | less  
cat /etc/services | less
```

### Suchen in less

```
##Innerhalb von less  
/suchbegriff + RETURN  
## nächstes Suchergebnis  
n
```

### Springen ans Ende/an den Anfang

```
## Innerhalb von less  
## ans Ende  
G  
## an den Anfang  
1g  
## zu einer bestimmten Zeile (Zeile 5)  
5g
```

### In die Hilfe rein

```
h  
## wieder raus  
q
```

**Datei anlegen - touch**

```
touch dateiname
```

## Autovervollständigen \* und tab

### Autovervollständigen \*

```
## show all entries in directory starting with tod
## * = zero or more characters
echo tod*
## tod todo todotext
```

### Autovervollständigen tab

```
echo tod <TAB><TAB> # bei mehreren Einträgen
echo todol<TAB> # bei einem weiteren Eintrag
```

### Welches Programm wird verwendet

```
## Sucht in der Pfad-Variablen $PATH nach dem programm  
## und zeigt ersten Fund --> d.h. dieses Programm würde ausgeführt  
which false
```

## Erweiterte Befehle (Nice to have)

### Alias Befehle anzeigen

```
## keine wirkliche Befehle, sondern nur andere Schreibweise/Abkürzungen
## kann u.U. so auf anderen Distris nicht vorhanden sein
alias
```

### Welche Bibliotheken verwendet ein ausführbares Programm

```
ldd /usr/bin/ls
```

## Dateien und Verzeichnisse

### Mit cd im System navigieren

#### Ins Heimatverzeichnis und Wurzelverzeichnis (C: unter Windows) wechseln

```
## Ins Heimatverzeichnis wechseln
## cd ohne alles
cd

## Ins Wurzelverzeichnis des Filesystems wechseln // Windows -> C:\
cd /
```

#### Wie in ein Verzeichnis wechseln (relativ und absolut)

```
## relativ - nur in ein Unterverzeichnis meines bestehenden Verzeichnisses
cd etc

## absolut - wechselt dort rein, egal wo ich bin
cd /etc
```

**Verzeichnisse in Listenansicht mit versteckten Dateien anzeigen -> ls -la**

```
ls -la
```



## Inhalt in Datei schreiben und anhängen

### Inhalte in Datei schreiben / anhängen

```
cd /home/kurs
## eingefügt am anfang, überschreibt alte Inhalte
ls -la > todo
## angehängt
echo "hans hat durst" >> todo
```

## Verzeichnisse und Dateien löschen

### Dateien und Verzeichnisse löschen

```
## bei symbolischen Links wird nur der symbolische Link und nicht die Datei gelöscht  
rm symlink  
## Datei löschen  
rm dateiname  
## Verzeichnis löschen  
rm -r verzeichnis
```

## Kopieren/Verschieben/Umbenennen von Dateien und Files

### Dateien umbenennen, verschieben, kopieren

```
## wenn Zeilverzeichnis nicht existiert -> Fehler !
cp -a todo.txt /dokumente/
## wenn zielverzeichnis nicht existiert, wird dokumente2 erstellt als file - > Achtung !!
cp -a todo.txt /dokumente2

## umbenennen
mv datei1 neuernamedatei1

## verschieben
mv datei1 /dokumente
```

### Rechte behalten bei kopieren

```
## -a macht das
cp -a todo.txt todoneu.txt

## ohne -a werden symbolische links aufgelöst und die Rechte des ausführenden gesetzt
cp ab cd
```

## Prozesse

### Prozesse anzeigen - ps/pstree -p

#### Prozesse anzeigen

```
ps -ef  
ps aux # x alle Prozesse anzeigen, die nicht an ein Terminal gebunden sind
```

#### systemctl (läuft Dienst)

```
systemctl status sshd
```

#### Prozeßbaum anzeigen (meist nicht für die Praxis notwendig)

```
pstree -p
```

## Benutzer, Gruppen und Rechte

### Rechte

#### Aufbau triple

```
kurs@ubuntu2004-101:~$ # rwx | rw- | r--  
kurs@ubuntu2004-101:~$ # u   g   o  
kurs@ubuntu2004-101:~$ # 421 | 42- | 4--  
kurs@ubuntu2004-101:~$ # 7   | 6   | 4
```

### Berechtigungen mit Symbolen setzen

```
chmod g+w,o+r testfile
```

## Dateien für Benutzer und Gruppen

```
cd /etc  
cat passwd  
cat shadow  
cat group
```

## Benutzer anlegen

### Benutzer anlegen (auf Ubuntu)

```
## for shell script
useradd

## for admins interactive
adduser
```

## **sudo Benutzer erstellen**

### **Benutzer zum Sudo benutzer machen**

```
adduser newuser
usermod -aG sudo newuser
### testing
su - newuser
groups # see if we are in groups sudo
id # shows the same but more info
## need to enter password here
sudo su -
```



## Dateimanipulation/Unix Tools

### cat/head/tail-Beginn/Ende einer Datei anzeigen

#### cat mit Zeilennummer

```
cat -n /etc/services
```

#### Die ersten -x Zeilen anzeigen

```
## ersten 10 Zeilen anzeigen  
head /etc/services
```

```
## Ersten 20 Zeilen  
head -n 20 /etc/services
```

#### Die letzten -x Zeilen anzeigen

```
## die letzten 10 Zeilen  
tail /etc/services
```

```
## die letzten 40 Zeilen  
tail -n 40 /etc/services
```

### Ausgabe der letzten Zeilen und ausgabe in Datei

```
cd /var/log  
tail -n 100 syslog.1 >> fehlerlog  
cat fehlerlog
```

**zcat** - Inhalte einer mit gzip komprimierten Datei anzeigen

**wc** - Zeilen zählen

**Datei**

```
wc -l /etc/services
```

**Zeilen aus Befehl**

```
ls -la | wc -l
```

## Bestimmte Zeilen aus Datei anzeigen - grep

### Beispiele

```
## alle Zeilen in den tcp vorkommt
cat /etc/services | grep tcp
## alle Zeilen in denen tcp nicht vorkommt
cat /etc/services | grep -v tcp
## alle Zeilen in denen tcp nicht vorkommt
## egal ob gross oder klein geschrieben.
cat /etc/services | grep -iv TCP

cat /etc/services | grep '#'
cat /etc/services | grep "#"
cat /etc/services | grep "^#"
## alle Zeilen, die am Anfang der Zeile kein # haben
cat /etc/services | grep -v "^#"
cat /etc/services | grep -v "^#" > /root/services
cat /etc/services | grep -v "^#" | head -n 20

cat /etc/services | grep -v "s$"
## alle Zeilen die als letztes Zeichen ein s haben
cat /etc/services | grep "s$"
```

### Recursive Suchen (grep -r)

```
grep -r "PermitRootLogin" /etc
```

## Erweiterte Suche mit Grep

### Nach einzelnen Wort suchen (Wort muss so vorkommen)

```
cat /etc/services | grep -i -w 'protocol'
```

### Eines der Begriffe soll vorkommen

```
## Achtung, unbedingt -E für extended regex verwendet
cat /etc/services | grep -E 'protocol|mysql'
```

### Eines der Wort soll am Anfang der Zeile vorkommen

```
## egrep ist das gleiche wie grep -E
egrep -i '^(mysql|Maira)' /etc/services
```

### x-Zeilen vor bzw. nach "Finde-(Grep-)" - Ergebnis anzeigen

```
## -A x-Zeilen danach, z.B. -A 4 --> 4 Zeilen danach
## -B x-Zeilen davor
egrep -A 4 -B 4 -i '^(mysql|Maira)' /etc/services '^(mysql|Maira)' /etc/services
```

### Einzelne Zeichen als Suchmuster nehmen

```
## 0, dann zwei beliebige Zeichen, dann tcp
grep '0..tcp' /etc/services
## 0, dann ein beliebiges Zeichen, dann tcp
grep '0.tcp' /etc/services
```

### Tatsächlich eine Punkt suchen

```
## /root/dateinamen
hans.txt
hans1txt
peter.txt

grep 'hans\.txt' /root/dateinamen

root@ubuntu2004-101:/etc# grep 'hans\.txt' /root/dateinamen
hans.txt
root@ubuntu2004-101:/etc# grep 'hans.txt' /root/dateinamen
hans.txt
hans1txt
```

### Einzelne Zeichen sollen vorkommen

```
root@ubuntu2004-101:~# echo "Klaus" >> /root/namen
root@ubuntu2004-101:~# echo "klaus" >> /root/namen
root@ubuntu2004-101:~# grep '[kK]l' /root/namen
Klaus
```

```
klaus
root@ubuntu2004-101:~# grep '[kK][la]' /root/namen
Klaus
klaus
root@ubuntu2004-101:~# echo "karin" >> /root/namen
root@ubuntu2004-101:~# grep '[kK][la]' /root/namen
Klaus
klaus
karin
```

```
echo "Klaus1" >> /root/namen
root@ubuntu2004-101:~# echo "Klaus2" >> /root/namen
root@ubuntu2004-101:~# grep '[kK][la]aus[0-9]' /root/namen
```

## Mengeangabe

```
## Achtung unbedingt egrep oder grep -E verwenden
cat /root/namen
AxB nix
AxB nix
abc nix
a nix

egrep '^[a-zA-Z]{1,3} nix' /root/namen
```

```
echo "ab nix" >> /root/namen
## Mindestens 2 Zeichen
root@ubuntu2004-101:~# egrep '^[a-zA-Z]{2,} nix' /root/namen
AxB nix
AxB nix
abc nix
ab nix
```

## Nach Zahlen Suchen

```
echo "12345 namen" >> /root/namen
grep "[[:digit:]]\{5\}" /root/namen
```

## Cheatsheets

- <https://cheatography.com/tme520/cheat-sheets/grep-english/>

## Ref:

- <https://www.cyberciti.biz/faq/grep-regular-expressions/>

## Logs/Loganalyse

### Logfile beobachten

```
## Terminal 1
tail -f /var/log/syslog

## Terminal 2 - write to logfile e.g.
logger meine_nachricht
```

## Dienste debuggen

### Walkthrough

```
## Dienst startet nicht

## Schritt 1 : status -> was sagen die logs (letzte 10 Zeilen)
systemctl status mariadb.service

## Nicht fündig-> Schritt 2:
journalctl -xe

## Nicht fündig -> Schritt 3:
journalctl -u mariadb.service

## Nicht fündig -> Schritt 4:
## Spezifisches Log von Dienst suchen
## und evtl. LogLevel von Dienst hochsetzen
## z.B. bei mariadb (durch Internetrecherche herausfinden)
less /var/log/mysql/error.log

## Nicht fündig -> Schritt 5
## Allgemeines Log
## Debian/Ubuntu
/var/log/syslog
## REdhat/Centos
/var/log/messages
```

### Find error in logs quickly

```
cd /var/log/mysql
cat error.log | grep -i error
```

# Variablen

## Setzen und verwenden von Variablen

```
DATEiname=/etc/services
echo $DATEiname

# Werte hochzählen
ZAHL=4
let ZAHL=ZAHL+1
echo $ZAHL

cat $DATEiname
# wird nicht der Inhalt verwendet sondern der Name $DATEiname
cat '$DATEiname'
cat "$DATEiname"

# Befehl ausführen und Rückgabewert anzeigen
date
echo $?

# Wert aus ausgeführtem Befehl in Variable schreiben
DATUM=$(date)
echo $DATUM
echo $DATUM >> /var/log/datumslog
```



## Dienste/Runlevel(Targets verwalten)

### Die wichtigsten systemctl/service

#### systemctl Beispiele

```
## Status eines Dienstes überprüfen
service sshd status
systemctl status sshd

## Wie heisst der Dienst / welche Dienste gibt es ?
systemctl list-units -t service
## für apache
systemctl list-units -t service | grep ^apache
## die Abkürzung
systemctl -t service | grep ^apache

## Dienst aktivieren
systemctl enable apache2
## Ist Dienst aktiviert
systemctl is-enabled apache2
enabled
echo $?
0 # Wenn der Dienst aktiviert ist

## Dienst deaktivieren (nach Booten nicht starten)
systemctl disable apache2
systemctl is-enabled
disabled
echo $?
1 # 1 wenn nicht aktiviert

## Rebooten des Servers
## verweist auf systemctl
reboot
systemctl reboot
shutdown -r now

## Halt (ohne Strom ausschalten)
halt
systemctl halt
shutdown -h now

## Poweroff
poweroff
systemctl poweroff
```

### Wie sehe ich, wie ein Service konfiguriert ist / Dienstekonfiguration anzeigen ?

```
## z.B. für Apache2
systemctl cat apache2.service
```

## Wie kann ich rausfinden, wie die runlevel als targets heissen ?

```
cd /lib/systemd/system
root@ubuntu2004-104:/lib/systemd/system# ls -la run*target
lrwxrwxrwx 1 root root 15 Jan  6 20:47 runlevel0.target -> poweroff.target
lrwxrwxrwx 1 root root 13 Jan  6 20:47 runlevel1.target -> rescue.target
lrwxrwxrwx 1 root root 17 Jan  6 20:47 runlevel2.target -> multi-user.target
lrwxrwxrwx 1 root root 17 Jan  6 20:47 runlevel3.target -> multi-user.target
lrwxrwxrwx 1 root root 17 Jan  6 20:47 runlevel4.target -> multi-user.target
lrwxrwxrwx 1 root root 16 Jan  6 20:47 runlevel5.target -> graphical.target
lrwxrwxrwx 1 root root 13 Jan  6 20:47 runlevel6.target -> reboot.target
```

## Welche Dienste sind aktiviert/deaktiviert

```
systemctl list-unit-files -t service
```

## Dienste bearbeiten

```
systemctl edit sshd.service
## Dann eintragen
[Unit]
Description=Jochen's ssh-server
## Dann speichern und schliessen (Editor)

systemctl daemon-reload
systemctl status
```

## Targets (wechseln und default)

```
## Default runlevel/target auslesen
systemctl get-default
## in target wechseln
systemctl isolate multi-user
## Default target setzen (nach start/reboot)
systemctl set-default multi-user
```

## Alle Target anzeigen in die ich reinwechseln kann (isolate)

```
## Ubuntu
grep -r "AllowIsolate" /lib/systemd/system
/lib/systemd/system/reboot.target
...
...
...
systemctl isolate reboot.target
```

## Dienste maskieren, so dass sie nicht gestartet werden können

```
systemctl mask apache2
## kann jetzt gestartet werden
systemctl start apache2
```

```
## de-maskieren
systemctl unmask apache2
## kann wieder gestaret werden
systemctl start apache2
```

## **systemctl Cheatsheet**

- [https://access.redhat.com/sites/default/files/attachments/12052018\\_systemd\\_6.pdf](https://access.redhat.com/sites/default/files/attachments/12052018_systemd_6.pdf)

# Partitionierung und Filesystem

## parted and mkfs.ext4

### Walkthrough

```
## Schritt 1: Platte in virtualbox oder gui-interface anlegen

## Schritt 2: Platte identifizieren
lsblk

## Schritt 3: Platte partitionieren
mkpart /dev/sdb1
mklabel gpt
mkpart data2 ext4 2048s 500M # data2 ist name der Partition bei gpt
quit

## Schritt 4: Partition formatiert
lsblk # Partition identifiziert
mkfs.ext4 /dev/sdb1

## Schritt 5: Mount-Punkt erstellen
mkdir /mnt/platte

## Schritt 6: einhängen und aushängen
mount /dev/sdb1 /mnt/platte
umount /mnt/platte

## Schritt 7: Persistent konfigurieren
## Eintragen in /etc/fstab
/dev/sdb1 /mnt/platte ext4 defaults 0 0

## Schritt 8: Test, ob fstab gut ist (keine Fehler)
mount -av # v steht für geschwätzig.

## Wenn das klappt: Schritt 9
reboot
```

# Boot-Prozess und Kernel

## Grub konfigurieren

### Walkthrough

```
## Step 1
## z.B. timeout hochsetzen, wie lange er mit Booten im Bootmenu wartet
cd /etc/default
vi grub
### make wanted changes
##GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=5

## Step 2
update-grub

## Step 3 - reboot

## When grub menu appears enter arrow-down arrow-up ONCE
## Dann zählt er nicht weiter runter und bootmenu bleibt stehen.

## Mit e kann man einen boot-eintrag für den nächsten Boot ändern

## Ändern und dann CTRL bzw. STRG + x für das Booten nach Änderung

## Step 4 - be happy
```

## Kernel-Version anzeigen

```
uname -a
```

## Kernel-Module laden/entladen/zeigen

### Walkthrough

```
## show kernel modules
lsmod
## kernel - module entladen
modprobe -r psmouse
lsmod | grep psmouse # now not present
## damit wieder laden
modprobe psmouse
lsmod | grep psmouse # now present
```

### Wo leben die Kernel - Module

```
### kernel version is used, find out kernel version with
uname -a

cd /lib/modules/5.4.0-66-generic

## e.g. psmouse
find /lib/modules -name psmouse*
/lib/modules/5.4.0-66-generic/kernel/drivers/input/mouse/psmouse.ko
```

# Hilfe

## Hilfe zu Befehlen

### Möglichkeiten der Hilfe

```
## anhand von ps

vi -h
ps --help
man ps
info ps
```

### -h oder --help --> eines geht immer

```
## Beispiel ls
ls -h # geht nicht für Hilfe
ls --help # geht !
```

### Navigation in den man-pages

```
q - verlassen von man
Pfeil oben/unten
PageUp/PageDown
G # für ans Ende der Datei springe
lg # in die erste Zeile
```

### Suche mit in man-pages

```
/Suchwort [Enter]
n # nächster Treffer (kleines n)
N # letzter Treffer
```



## Grafische Oberfläche und Installation

### Gnome unter Ubuntu installieren

```
sudo apt install tasksel  
sudo tasksel install ubuntu-desktop
```

## **X-Server - Ausgabe auf Windows umleiten**

- [https://www.thomas-krenn.com/de/wiki/Grafische\\_Linux\\_Programme\\_remote\\_von\\_einem\\_Windows\\_PC\\_mit\\_Xming\\_nutzen](https://www.thomas-krenn.com/de/wiki/Grafische_Linux_Programme_remote_von_einem_Windows_PC_mit_Xming_nutzen)

## Installations-Images-Server

- <https://ubuntu.com/download/server#download>

## Wartung und Aktualisierung

### Aktualisierung des Systems

```
apt update
apt upgrade
apt dist-upgrade

## oder geht auch auf älteren Systemen
apt-get update
apt-get upgrade
apt-get dist-upgrade
```

## Paketmanager apt/dpkg

### Alle Pakete anzeigen, die installiert sind auf dem System

```
dpkg -l
```

### Alle Paket die zur Verfügung stehen

```
apt list
```

### Wo sind die Repos konfiguriert

```
cat /etc/apt/sources.list  
cd /etc/apt/sources.list.d
```

### Paket deinstallieren und aufräumen

```
## mit konfigurationsdateien deinstallieren  
apt purge mariadb-server  
## konfigurationsdateien stehen lassen  
apt remove mariadb-server  
  
## Aufräumen / alle Pakete die nicht mehr benötigt werden  
apt autoremove
```

### Pakete händisch mit dpkg installieren

```
## Schritt 1: Im Browser  
## Paket online finden und Link kopieren (Browser - Rechte Maustaste Link kopieren)  
  
## Schritt 2: auf dem Linux Server  
sudo apt install wget  
cd /usr/src  
wget http://archive.ubuntu.com/ubuntu/pool/main/a/acl/acl_2.2.53-10build1_amd64.deb  
sudo dpkg -i acl_2.2.53-10build1_amd64.deb
```

### Pakete mit apt search suchen

```
## Vorbereitung  
apt update  
  
## suche nache apache  
apt search apache  
## mit pager  
apt search apache | less  
  
## Alle Paket in denen apache am Anfang der Zeile  
apt search ^apache | less
```

### Installieren mit apt install

```
## mit genauem Namen  
apt install apache2
```

## Archive runterladen und entpacken

```
## Walkthrough
## Schritt 1: Download-Link in Browser kopieren (rechte Maustaste)

## Schritt 2:
cd /usr/src
# falsche Dateiname -> umbenannt.
wget https://github.com/phayes/geoPHP/tarball/master
mv master master.tar.gz
## Schritt 3: Sicherheitsverzeichnis anlegen und entpacken
mkdir foo
mv master.tar.gz foo
cd foo
tar xvf master.tar.gz
```

## Firewall und ports

**ufw (uncomplicated firewall)**

**Läuft der Dienst für die Firewall**

```
systemctl status ufw
```

**Ist die Firewall scharfgeschaltet ?**

```
ufw status
```

**Firewall aktivieren (Achtung ssh)**

```
## Neue ssh - Verbindungen werden nicht angenommen
## Bestehende Bedingungen (ESTABLISHED) bleiben erhalten
ufw enable
ufw status
```

**Port hinzufügen**

```
ufw allow 22 # for tcp and udp
## or
ufw allow ssh # uses /etc/services for detection of port - number
ufw status
```

## firewalld

### Install firewalld and restrict ufw

```
## Schritt 1: ufw deaktivieren
systemctl stop ufw
systemctl disable ufw
ufw disable # zur Sicherheit
ufw status
## -> disabled # this has to be the case

## Schritt 2: firewalld
apt install firewalld
systemctl status firewalld
systemctl status ufw
```

### Is firewalld running ?

```
## is it set to enabled ?
systemctl status firewalld
firewall-cmd --state
```

### Command to control firewalld

- firewall-cmd

### Best way to add a new rule

```
## Step1: do it persistent -> written to disk
firewall-cmd --add-port=82/tcp --permanent

## Step 2: + reload firewall
firewall-cmd --reload
```

### Zones documentation

man firewalld.zones

### Zones available

```
firewall-cmd --get-zones
block dmz drop external home internal public trusted work
```

### Active Zones

```
firewall-cmd --get-active-zones
## in our case empty
```

### Show information about all zones that are used

```
firewall-cmd --list-all
firewall-cmd --list-all-zones
```



## Add Interface to Zone ~ Active Zone

```
firewall-cmd --zone=public --add-interface=enp0s3 --permanent
firewall-cmd --reload
firewall-cmd --get-active-zones
public
    interfaces: enp0s3
```

## Default Zone

```
## if not specifically mentioned when using firewall-cmd
## .. add things to this zone
firewall-cmd --get-default-zone
public
```

## Show services

```
firewall-cmd --get-services
```

## Adding/Removing a service

```
firewall-cmd --permanent --zone=public --add-service=ssh
firewall-cmd --reload
firewall-cmd --permanent --zone=public --remove-service=ssh
firewall-cmd --reload
```

## Add/Remove ports

```
## add port
firewall-cmd --add-port=82/tcp --zone=public --permanent
firewall-cmd --reload

## remove port
firewall-cmd --remove-port=82/tcp --zone=public --permanent
firewall-cmd --reload
```

## Enable / Disabled icmp

```
firewall-cmd --get-icmptypes
## none present yet
firewall-cmd --zone=public --add-icmp-block-inversion --permanent
firewall-cmd --reload
```

## Working with rich rules

```
## Documentation
## man firewalld.richlanguage

## throttle connectons
firewall-cmd --permanent --zone=public --add-rich-rule='rule family=ipv4 source
address=10.0.50.10/32 service name=http log level=notice prefix="firewalld rich rule INFO:
" limit value="100/h" accept'
```

```
firewall-cmd --reload #
firewall-cmd --zone=public --list-all

## port forwarding
firewall-cmd --get-active-zones
firewall-cmd --zone=public --list-all
firewall-cmd --permanent --zone=public --add-rich-rule='rule family=ipv4 source
address=10.0.50.10 forward-port port=42343 protocol=tcp to-port=22'
firewall-cmd --reload
firewall-cmd --zone=public --list-all
firewall-cmd --remove-service=ssh --zone=public

##

## list only the rich rules
firewall-cmd --zone=public --list-rich-rules

## persist all runtime rules
firewall-cmd --runtime-to-permanent
```

## References

- <https://www.linuxjournal.com/content/understanding-firewalld-multi-zone-configurations#:~:text=Going%20line%20by%20line%20through,or%20source%20associated%20with%20it.>
- <https://www.answertopia.com/ubuntu/basic-ubuntu-firewall-configuration-with-firewalld/>

## Scannen und Überprüfen mit telnet/nmap

## Netzwerk/Dienste

### IP-Adresse von DHCP-Server holen (quick-and-dirty)

#### Walkthrough

```
## Ip nicht gesetzt - kurzfristig eine IP holen
ip a # zeigt die Netzwerkschnittstellen an.
dhclient enp0s8 # ip - Adresse für Schnittstelle enp0s8 holen
ip a
```

## Auf welchen Ports lauscht mein Server

```
## Zeigt alle ports an auf die gelauscht wird (ipv4)
lsof -i
## alternative
netstat -tupel
```

## Literatur

### Literatur

### Literatur

- [Linux Grundlagen für Anwender und Administratoren](#)
- [Linux Systemadministration I für Anwender und Administratoren](#)
- [Alle Unterlagen](#)

### Cheatsheet

- [Cheatsheet bash](#)