**Hinweis**: Falls ein Command nicht funktioniert → sudo davor schreiben

## Lizenzmodelle

|  |  |
| --- | --- |
| Modell | Beschreibung |
| Proprietäre Lizenz | Closed Source Software, kein Zugriff auf den Source-Code, Distribution und Modifikation in der Regel verboten |
| Free Software | Gewährt die vier Freiheiten (Benutzung, Untersuchung, Weitergabe, Veränderung); Source-Code meist offen |
| Open Source Lizenz | Source-Code offen, darf genutzt, verändert und verteilt werden; Offenlegung bleibt bei Weitergabe Pflicht |
| Shareware | Software darf kostenlos getestet werden, volle Nutzung meist nur nach Bezahlung |
| Freeware | Kostenlos nutzbar, jedoch ohne Garantie auf Offenlegung des Source-Codes |
| Public Domain | Ohne Urheberrechtsschutz, frei nutzbar, veränderbar und weitergebbar |
| Kommerzielle Lizenz | Gegen Bezahlung lizenziert; Nutzung nur im Rahmen der Lizenzbedingungen erlaubt |

### Gegenüberstellung

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lizenzart | Quellcode | Kosten | Nutzungsfreiheit | Modifikation | Zielgruppe |
| Proprietär | Geschlossen | Kostenpflichtig | Stark eingeschränkt | Meist verboten | Endnutzer, Firmen |
| Free Software | Offen | Oft kostenlos | Sehr hoch (viel Freiheit) | Erlaubt & erwünscht | Entwickler |
| Open Source | Offen | Kostenlos / Kosten | Hoch | Je nach Lizenz | Entwickler, Firmen |
| Shareware | Geschlossen | Erst frei, dann Kosten | Eingeschränkt (Testversion) | Nicht erlaubt | Privatnutzer |
| Freeware | Geschlossen | Kostenlos | Nutzung erlaubt | Meist nicht erlaubt | Privatnutzer |
| Public Domain | Variabel | Kostenlos | Komplett frei | Erlaubt ohne Einschränkung | Alle |
| Kommerziell | Geschlossen | Kostenpflichtig | Eingeschränkt | Nicht erlaubt | Firmen, Profis |

## Hypervisor

* **Typ 1** (Bare Metal): direkt auf Hardware installiert
* **Typ 2**: läuft auf bestehendem Betriebssystem

|  |  |
| --- | --- |
| Begriff | Beschreibung |
| Gast-Betriebssystem | OS innerhalb der VM, nutzt virtuelle Hardware (CPU, RAM, Disk) |
| Hypervisor | Virtualisierungssoftware, vermittelt zwischen Host-HW und Gast-OS, Beispiele: VirtualBox, VMware, Hyper-V |
| Host-Betriebssystem | Direkt auf Hardware installiert (bei Typ 2: Basis für Hypervisor), Beispiele: Windows, Linux, macOS |
| Hardware | Physische Grundlage (CPU, RAM, Disk, Netzwerkkarten), Ressourcen werden vom Hypervisor an die VMs verteilt |

## Paketmanager

|  |  |
| --- | --- |
| Befehl | Bedeutung |
| apt update | Aktualisiert Paketlisten (weiß, welche Versionen verfügbar sind) |
| apt upgrade | Aktualisiert alle installierten Pakete auf neueste Version |
| apt install <paket> | Installiert ein Paket |
| apt remove <paket> | Entfernt ein Paket, Konfigurationsdateien bleiben |
| apt purge <paket> | Entfernt ein Paket inkl. Konfigurationsdateien |
| apt search <suchbegriff> | Sucht nach Paketen in den Quellen |
| apt show <paket> | Zeigt Details zu einem Paket |
| apt list <suchbegriff> --installed | Zeigt installierte Pakete, die dem Suchbegriff entsprechen |

**Update vs. Upgrade**

* update: Aktualisiert nur Listen
* upgrade: Installiert wirklich neue Versionen

**Vorteile von Paketmanagern**

| Vorteil | Beschreibung |
| --- | --- |
| Automatische Abhängigkeiten | Fehlende Bibliotheken werden automatisch mit installiert |
| Zentrale Updates | Alle Programme können mit einem Befehl aktualisiert werden |
| Sicherheit | Pakete stammen aus geprüften Quellen |
| Systemintegration | Software wird korrekt eingebunden (Pfad, Menüs, Dienste) |
| Einfachheit | Installation, Update und Entfernung sind schneller und konsistenter |

## Dateisysteme

### Grundlagen

* Struktur zur Organisation von Daten
* Regelt Dateinamen, Speicherort, Rechte, Attribute
* Ohne Dateisystem → nur Rohdaten

### Überblick

* **FAT32**: sehr kompatibel, max. 4 GB Datei, bis 2 TB Partition → USB/SD
* **NTFS**: Windows-Standard, große Dateien, Rechteverwaltung → Windows/Externe Platten
* **EXT4**: Linux-Standard, Journaling, stabil → Linux-Systeme
* **Btrfs**: modern, Snapshots, Subvolumes → flexible Systeme
* **ZFS**: robust, Datenintegrität, große Datenmengen → Server/NAS

### Vergleich

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Dateisystem | Kompatibilität | Max. Datei | Besonderheiten | Typische Nutzung |
| FAT32 | Sehr hoch | 4 GB | Kein Rechtemanagement | USB, SD-Karten |
| NTFS | Windows, Linux teilw. | > 16 TB | Rechte, Journaling | Windows, Extern |
| EXT4 | Linux | 16 TB | Journaling, stabil | Linux-Systeme |
| Btrfs | Linux | 16 EB | Snapshots, Subvolumes | Flexible Systeme |
| ZFS | Linux, BSD, Solaris | 16 EB+ | Datenintegrität, Snapshots | Server, NAS |

### Partitionierung

* **Primär**: max. 4, OS oder FS
* **Erweitert**: Container für logische Partitionen
* **Logisch**: innerhalb erweitert, flexibel

**MBR vs. GPT**

* **MBR**: älter, max. 2 TB, max. 4 primäre Partitionen
* **GPT**: modern, bis 128 Partitionen, > 2 TB, für UEFI

## Partitionen erstellen

|  |  |  |
| --- | --- | --- |
| Schritt | Befehl | Hinweis |
| Partitionen anzeigen | lsblk | Alle Festplatten und Partitionen |
| Details prüfen | fdisk -l | Infos zu Partitionen und Größen |
| Neue Partition | fdisk /dev/sdX | n für neu, w speichern |
| Formatieren | mkfs.ext4 /dev/sdX1 | Erstellt EXT4 (andere FS mit mkfs.ntfs, etc.) |
| Mountpoint erstellen | mkdir /mnt/testpartition | Verzeichnis anlegen |
| Partition einbinden | mount /dev/sdX1 /mnt/testpartition | Partition mounten |
| Testen | df -Th | Eingebundene Partitionen anzeigen |
| Automount | /etc/fstab bearbeiten | z. B. /dev/sdX1 /mnt/testpartition ext4 defaults 0 2 |

## Wichtige Linux-Verzeichnisse

|  |  |
| --- | --- |
| Verzeichnis | Beschreibung |
| /bin | Systemprogramme für alle Benutzer |
| /sbin | Systemprogramme für Root |
| /dev | Gerätedateien |
| /etc | Systemweite Konfigurationen |
| /home | Benutzerverzeichnisse |
| /lib | Bibliotheken |
| /media | Automount für Wechselmedien |
| /mnt | Temporäre Mountpoints |
| /opt | Zusatzsoftware |
| /proc | Virtuelles Dateisystem für Prozesse/Kerneldaten |
| /run | Temporäre Laufzeitdaten |
| /srv | Daten für Dienste |
| /sys | Kernel-/Hardwareinfos |
| /tmp | Temporäre Dateien |
| /usr | Nutzerprogramme, Bibliotheken |
| /var | Logs, Datenbanken, Spool |

## Gruppen & Nutzer

|  |  |  |
| --- | --- | --- |
| Schritt | Befehl | Beschreibung |
| 1 | groupadd testgruppe | Erstellt Gruppe **testgruppe** |
| 2 | useradd -m -s /bin/bash testuser | Erstellt Benutzer **testuser** |
| 3 | echo "testuser:test" \| chpasswd | Setzt Passwort von **testuser** auf test |
| 4 | usermod -aG testgruppe testuser | Fügt **testuser** zur Gruppe hinzu |
| 5 | groups testuser | Gruppenzugehörigkeit prüfen |