# SSH Client

- SSH-Sitzung teilen, ohne Kennwort oder Schlüssel weiterzugeben

# SSH-Sitzung teilen, ohne Kennwort oder Schlüssel weiterzugeben

**Wofür:** Ein zweites Programm, ein Kollege am selben Rechner oder ein KI-Assistent soll auf einem Server arbeiten. Auf dem Server lässt sich aber kein öffentlicher Schlüssel hinterlegen (z. B. weil die Platte voll ist), und das Kennwort soll niemand sehen.

**Prinzip:** Du baust die Verbindung selbst auf und tippst das Kennwort nur in deinem eigenen Terminal. OpenSSH legt dabei eine Steuer-Datei (Socket) an. Weitere `ssh`-Aufrufe auf *demselben Rechner* benutzen diese offene Verbindung mit. Ein Kennwort oder ein Schlüssel ist dafür nicht nötig, und das Kennwort wird nirgends gespeichert.

# 1. Verbindung aufbauen (du selbst)

```
ssh -o ControlMaster=yes -o ControlPath=~/.ssh/cm-ZIEL -o ControlPersist=3h root@SERVER-IP
```

| Option | Bedeutung |
|---|---|
| `ControlMaster=yes` | Diese Verbindung wird zur „Haupt“-Verbindung, an die sich andere anhängen dürfen. |
| `ControlPath=~/.ssh/cm-ZIEL` | Ort der Steuer-Datei. Wähle einen sprechenden Namen pro Ziel, z. B. `cm-pbs`. |
| `ControlPersist=3h` | Die Verbindung bleibt nach deinem Abmelden noch 3 Stunden offen. |

Das Kennwort tippst du in dieses Terminal. Danach kannst du normal weiterarbeiten oder die Sitzung schließen. Die Verbindung im Hintergrund bleibt bis zum Ablauf von `ControlPersist` bestehen.

**Wichtig:** Wenn du dich vorher schon mit normalem `ssh` angemeldet hast, kann sich niemand an diese Sitzung hängen. Sie hat keine Steuer-Datei. Öffne deshalb ein zweites Terminal und baue die Verbindung mit dem Befehl oben neu auf.

# 2. Die geteilte Verbindung benutzen (der Helfer)

```
ssh -o ControlPath=~/.ssh/cm-ZIEL root@SERVER-IP 'hostname; df -h /'
```

Damit nichts hängen bleibt, wenn die Verbindung schon zu ist, empfiehlt sich zusätzlich `-o BatchMode=yes`. Dann fragt SSH nicht nach einem Kennwort, sondern meldet sofort einen Fehler.

# 3. Kontrolle und Beenden

```
# Läuft die Verbindung noch?
ssh -O check -o ControlPath=~/.ssh/cm-ZIEL root@SERVER-IP

# Sofort schließen (für alle, die sie mitbenutzen)
ssh -O exit -o ControlPath=~/.ssh/cm-ZIEL root@SERVER-IP
```

# Sicherheit

- Wer die Steuer-Datei benutzen kann, arbeitet mit **deinen Rechten** auf dem Server. Sie liegt in `~/.ssh/` und gehört nur deinem Benutzer. Andere lokale Benutzer kommen also nicht ran, root auf deinem Rechner aber schon.
- Teile die Verbindung nur so lange wie nötig. Halte `ControlPersist` kurz und beende die Verbindung danach mit `-O exit`.
- Auf Kundensystemen: Kläre vorher, was der Helfer tun darf. Nur lesen ist der sichere Standard, Änderungen nur nach Rückfrage.
- Die Steuer-Datei wirkt nur auf dem Rechner, auf dem sie liegt. Ein entfernter Zugriff ist darüber nicht möglich.

# Fehlerbilder

| Meldung | Ursache |
|---|---|
| `Permission denied (publickey,password)` | Es existiert keine Steuer-Datei. Die Verbindung wurde ohne `ControlMaster=yes` aufgebaut, ist abgelaufen oder läuft auf einem anderen Rechner. |
| `Control socket connect(...): No such file or directory` | Der Pfad stimmt nicht oder die Verbindung ist schon geschlossen. |