Linux Allgemein

- Drucken - CUPS
  - cups von allen Benutzern verwaltbar machen
  - PDF Cups für linux Terminalserver / nomaschine Drucker auf Client umleiten
- SSH Client
  - SSH-Sitzung teilen, ohne Kennwort oder Schlüssel weiterzugeben

Drucken - CUPS

cups von allen Benutzern verwaltbar machen

Beschreibung:

Außer Automatisch gefunde Drucker, kann man als normaler Benutzer keine Drucker installieren oder löschen etc.
Dafür kann man aber eine policy file anlegen.

Durchführung:

```
sudo nano /etc/polkit-1/localauthority/50-local.d/cups-allow-everything.pkla
```

Inhalt

```
[Allow users to do all printing actions without authentication]
Identity=unix-user:*
Action=org.opensuse.cupspkhelper.mechanism.*
ResultAny=yes
ResultInactive=yes
ResultActive=yes
```

Cups neustarten

```
sudo systemctl restart cups
```

PDF Cups für linux Terminalserver / nomaschine Drucker auf Client umleiten

Beschreibung:

Druckumleitung auf Terminalserver / nomaschine funktioniert nicht.
Wir können aber vom Client auf ein Verzeichnis vom Terminalserver zugreifen zum beispiel per sftp oder nomachine Verzeichnisumleitung.

Mittels CUPS PDF werden alle Dateien im Homeverzeichnis im Verzeichnis PDF gespeichert.
Auf dem Client läuft ein Script was diesen ordner beobachtet und wenn PDFs drin sind diese dann an den angebenen Druckewr sendet.

Durchführung:

Auf dem Server:

Cups printer installieren

```
sudo apt-get install printer-driver-cups-pdf
```

Alle Dokumente werden unter

```
~/pdf
```

abgespeichert. Vom Client aus dieses Verzeichnis mounten. Via sftp oder nomachines Laufwerkdurchreiche selbst, nextcloud
oder oder oder....

Auf dem Client:

Dem Autoprint script Namen den Benutzer dran hängen

```
sudo mkdir -p /usr/local/bin
sudo nano /usr/local/bin/auto-print-<benutzername>.sh
```

Inhalt:

Die Variablen Watch Dir und Printer Name anpassen

```
#!/bin/bash

# --- Konfiguration ---
WATCH_DIR="/home/<benutzername>/Schreibtisch/Stamm/home/<benutzername-terminalserver>/PDF"        #
Oder /var/spool/cups-pdf/duffy je nach Setup
PRINTER_NAME="TOSHIBA"        # Dein CUPS-Druckername (mit 'lpstat -p' prüfen)

# --- Endlosschleife ---
while true; do
    for file in "$WATCH_DIR"/*.pdf; do
        [ -e "$file" ] || continue  # Kein PDF vorhanden? Dann weiter

        echo "Drucke: $file"
        lp -d "$PRINTER_NAME" "$file" && rm -f "$file"
    done
    sleep 5  # alle 5 Sekunden prüfen
done
```

Ausführbar machen

```
sudo chmod +x /usr/local/bin/auto-print.sh
```

Systemd erstellen für den jeweiligen benutzer

```
sudo nano /etc/systemd/system/auto-print-<benutzername>.service
```

Inhalt:

Auch hier im Dateinamen den Benutzer anpassen

```
[Unit]
Description=Automatischer PDF-Drucker
After=network.target
```

```
[Service]
ExecStart=/usr/local/bin/auto-print-<benutzername>.sh
Restart=always
User=duffy
Environment=HOME=/home/<benutzername>
WorkingDirectory=/home/<benutzername>

[Install]
WantedBy=multi-user.target
```

Service aktibieren und starten, auch hier den benutzernamen wieder anpassen

```
sudo systemctl daemon-reexec
sudo systemctl daemon-reload
```

```
sudo systemctl enable --now auto-print-<benutzername>.service
```

SSH Client

SSH Client

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.