

Debian Datei-Systeme

- [Samba Freigabe in Debian anbinden](#)
- [Partition von Debian im laufenden Betrieb vergrößern](#)

Samba Freigabe in Debian anbinden

Einleitung

In diesem Artikel geht es darum, wie wir in *Debian* eine *Samba-Freigabe* anbinden bzw. mounten können. Dieses nutzen wir, um z.B. Daten auf einem anderen Server speichern zu können. In meinem Fall nutze ich dies, um meine *Proxmox Backups*, die vom **Proxmox Backup Server** geschrieben werden, auf einer *Hetzner Storagebox* zu speichern.

Dabei binden wir die Freigabe einmal in der aktuellen Sitzung an, und erstellen dann einen Eintrag in der `/etc/fstab` Datei, damit die Freigabe auch nach einem Neustart verfügbar ist. Dadurch soll sich das Laufwerk automatisch nach einem Neustart verbinden.

Durchführung

Im ersten Schritt müssen wir überprüfen, ob das Paket `cifs-utils` auf unserem Server installiert ist. Falls nicht, wird das Paket mit dem nachstehenden Befehl auf unserem Server installiert.

```
sudo apt install cifs-utils -y
```

Im nächsten Schritt müssen wir ein *Verzeichnis* erstellen. Mit diesem Verzeichnis wird dann die *Samba-Freigabe* verbunden. Die Dateien aus der *Samba Freigabe* finden wir dann in diesem Verzeichnis. Ich lege dieses Verzeichnis im `/mnt` Verzeichnis an. Der Name und der Ort des Verzeichnisses kann dabei frei gewählt werden.

```
sudo mkdir /mnt/ds_smbshare
```

Jetzt erstellen wir eine Datei, in dem wir die Anmeldedaten für die Samba-Freigabe ablegen. Um die Sicherheit zu erhöhen, beginnt der Datei-Name mit einem Punkt. Dadurch ist diese auf den ersten Blick unsichtbar.

```
sudo nano /root/.smbcredentials
```

In diese Datei fügen wir folgenden Inhalt ein:

```
username=<Benutzername>
password=<Passwort>
```

Diese Datei machen wir aus Sicherheitsgründen nur für den *root Benutzer* lesbar. Alle anderen Benutzer können diese Datei nicht öffnen und damit nicht den Inhalt ausfindig machen.

```
chmod 400 /root/.smbcredentials
```

Jetzt im nächsten Schritt verbinden wir die *Samba-Freigabe* mit unserer aktuellen Sitzung. Dadurch sind wir in der Lage, Dateien auf den Server hochzuladen.

```
# Befehl
sudo mount -t cifs -o rw,vers=3.0,credentials=/root/.smbcredentials //<ip-adresse oder host-
name>/<freigabename> /<lokales Verzeichnis>

# Beispiel
sudo mount -t cifs -o rw,vers=3.0,credentials=/root/.smbcredentials //192.168.50.6/media
/mnt/ds_smbshare
```

Um die Freigabe nach einem Neustart automatisch zu verbinden, müssen wir die `/etc/fstab` Datei editieren. Dazu öffnen wir diese im ersten Schritt.

```
sudo nano /etc/fstab
```

Und dort fügen wir in die letzte Zeile den folgenden Code ein, den wir noch auf unsere Gegebenheiten anpassen müssen. Sobald wir diese Datei gespeichert haben, sollte unser Netzlaufwerk nach einem Neustart zur Verfügung stehen.

```
//<ip-adresse oder hostname>/<freigabe> /<lokales verzeichnis> cifs
vers=3.0,credentials=/root/.smbcredentials
```

Partition von Debian im laufenden Betrieb vergrößern

Einleitung

In dieser Anleitung vergrößern wir die vorhandene Debian Partition auf das Maximum. So können wir die Daten-Partition unseres Servers vergrößern, um mehr Daten auf der Festplatte abzulegen. Dies ist besonders vorteilhaft in Virtualisierungsumgebungen.

WICHTIG! Es sollte vorher in jedem Fall ein Backup vom Server gemacht werden! Unter Umständen kann die Veränderung der Partition zu Datenverlust führen!

Partition vergrößern

Im ersten Schritt müssen wir mit **fdisk** die Festplatte "öffnen":

```
sudo fdisk /dev/sda
```

Im nächsten Schritt arbeiten wir die Abfragen ab. Bei mir sieht das wie folgt aus:

```
WARNING: DOS-compatible mode is deprecated. It's strongly recommended to
switch off the mode (command 'c') and change display units to
sectors (command 'u').

Command (m for help): d
Selected partition 1

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-83220, default 1): 64
Last cylinder, +cylinders or +size{K,M,G} (63-83220, default 83220): 83220

Command (m for help): w
```

```
The partition table has been altered!
```

```
Calling ioctl() to re-read partition table.
```

```
Syncing disks.
```

Sobald das abgeschlossen ist, können wir mit dem Befehl `lsblk` überprüfen, ob die Erweiterung erfolgreich durchgeführt wurde.