

# Windows-Freigaben mit smbnetfs einbinden

Smbnetfs ist ein Userspace Dateisystem mit dem transparent und User-bezogen Windows-Freigaben eingebunden werden können.

Zwar gibt es verschiedene Möglichkeiten, Windows-Freigaben in Linux-Desktops einzubinden, aber die meisten haben ihre Nachteile:



- Ein Einbinden über die Dateimanager von Gnome, Mate oder KDE hat z.B. den Nachteil, dass die Shares dann in virtuellen Filesystemen eingebunden werden (z.B. gvfs bei Gnome und Mate), die einigen Programmen nicht zur Verfügung stehen.
- Das Einbinden über einen Eintrag in der fstab mit cifs mag für ein Ein-Benutzersystem in Frage kommen, stößt aber bei Multi-User Umgebungen an seine Grenzen.

Eine Lösung stellt smbnetfs dar, das Windows-Freigaben im User-Space mountet - und auch von den Nutzer\_innen persönlich verwaltet werden kann. Der Nachteil: es kommt nicht mit einer grafischen Konfigurationsmöglichkeit daher.

Nach Installation mit

```
apt-get install smbnetfs
```

Und der Zugehörigkeit in der fuse-Gruppe

```
adduser USERNAME fuse
```

lassen sich mit dem einfachen Befehl

```
smbnetfs ./mountpoint
```

Windows-Freigaben mounten. Das Tool erwartet eine Konfigurationsdatei unter `./smb/smbnetfs.conf`. Im einfachsten Fall sieht diese so aus:

[~/smb/smbnetfs.conf](#)

```
auth SERVER "USERNAME" "PASSWORT"
```

In diesem Fall versucht smbnetfs alle Server über netbios zu suchen/finden - und für den Server „SERVER“ sich mit den angegebenen Credentials zu authentifizieren. Für unterschiedliche Server sind mehrere Zeilen möglich.

Soll lediglich ein Server eingebunden werden (und/oder funktioniert aufgrund von Firewall-Regeln (oder Routing) netbios nicht richtig) kann folgendes konfiguriert werden:

~/smb/smbnetfs.conf

```
# Passwort den SERVER
auth SERVER "user" "password"

# wir wollen explizit Shares von folgenden Hosts einbinden
host SERVER visible=true

# Wir wollen keine anderen Hosts browsen:
smb_query_browsers      "false"

# wenn das Share versteckt ist, trotzdem zeigen:
show_hidden_hosts      "true"
```



Die smbnetfs.conf muss die Rechte 600 haben, wenn dort Passwörter referenziert werden, sonst verweigert sich smbnetfs. Die Konfigurationsdateien können auch mit include-Direktiven gesplittet werden - bspw. in eine smbnetfs.aut und smbnetfs.host.

Unter /etc/smbnetfs.conf liegt eine Default-Datei. Diese scheint aber bei Abwesenheit einer User spezifischen Konfiguration nicht genutzt zu werden, sondern dient nur als Beispiel für die Konfiguration. Außerdem soll smbnetfs auch auf eine vorhandene /etc/samba/smb.conf oder eine smb.conf im ~/smb Verzeichnis zurückgreifen. Welche Parameter daraus gelesen werden, wird aus der Dokumentation aber nicht klar.

## Automatisches Einbinden

Am einfachsten ist das Mounten über die .profile im User-Verzeichnis.

~/profile

```
# smbnetfs
/usr/local/bin/sambausermount.sh start
```

z.B. mit folgendem Skript beim Login zu mounten:

/usr/local/bin/sambausermount.sh

```
#!/bin/bash

SMBMOUNTDIR=~/samba
test -d $SMBMOUNTDIR || mkdir $SMBMOUNTDIR

case "$1" in
```

```
start)
  # nur mounten, wenn's noch nicht passiert ist sonst beenden
  mount | grep $USER | grep -q smbnetfs && exit 0

  # Teste auf Vorhandensein von Konfiguration
  if ! test -e ~/.smb/smbnetfs.conf; then
    zenity --info --text "Keine Konfigurationsdatei für das Einbinden
von Samba-Verzeichnissen gefunden. Bitte prüfe, ob ~/.smb/smbnetfs.conf
existiert oder wende dich vertrauensvoll an deinen Administrator"
    exit 0
  fi

  if ! grep -q ^auth .smb/smbnetfs.conf; then

    #zenity --info --text "Du musst noch dein Passwort in deiner
Konfigurationsdatei für die Windows-Freigaben konfigurieren. Die Datei
wird jetzt geöffnet. Bitte editieren und speichern."

    pluma ~/.smb/smbnetfs.conf

    /usr/bin/smbnetfs $SMBMOUNTDIR

  else
    /usr/bin/smbnetfs $SMBMOUNTDIR
  fi
;;
stop)
  # nur was tun, wenn's gemountet ist.
  mount | grep $USER | grep smbnetfs || exit 0
  /bin/fusermount -u $SMBMOUNTDIR
;;
esac
```

In der `.bash_logout` könnte dann entsprechend stehen:

`.bash_logout`

```
# smbnetfs
/usr/local/bin/sambausermount.sh stop
```

Das funktioniert so lange nur eine Login-Shell geöffnet, bzw. in diesem Fall: geschlossen wird. Das ist bei grafischen Oberflächen der Fall, einzelne grafische Terminals öffnen keine Login-Shells - und rufen auch die `.bash_logout` beim Schließen entsprechend nicht auf.

Alternativ kann auch in der Desktop-Umgebung die entsprechende Befehlszeile als Autostart konfiguriert werden. Z.B. einfach:

## smbnetfs samba

Allerdings funktioniert das automatische Aushängen beim Logout nicht - und das Share bleibt bis zum nächsten Reboot gemountet.

## Authentifizierung über gnome-keyring Manager

smbnetfs kann auch auf die Verbindungsinformationen, die im gnome-keyring gespeichert sind, zurück greifen. In diesem Fall kann auf die auth-Zeile in der smbnetfs.conf verzichtet werden - bzw. das Passwort kann leer gelassen werden.

Voraussetzung ist, dass die Samba-Freigabe schon einmal über den Gnome- oder Mate-Filemanager eingebunden wurde (als gvfs) und im Gnome-Keyring entsprechende Credentials existieren.

## Weitere Informationen:

Die Konfigurationsmöglichkeiten finden sich hier:

- <http://sourceforge.net/p/smbnetfs/git/ci/803699c6044aed0c2b25b23d22e5c0641be715fe/tree/conf/smbnetfs.conf>

Und eine Anleitung im Ubuntu-Wiki:

- [https://wiki.ubuntuusers.de/Samba\\_Client\\_SMBNetFS](https://wiki.ubuntuusers.de/Samba_Client_SMBNetFS)

From:  
<https://wiki.datenkollektiv.net/> - **datenkollektiv.net**

Permanent link:  
<https://wiki.datenkollektiv.net/public/smbnetfs>

Last update: **2015/08/13 11:19**

