

Back In Time

Umzug auf einen anderen PC



Allgemeines

In der Regel soll eine Datensicherung für eine lange Zeit die neu erstellten Daten in einem Home-System aufnehmen. Auch bei diesem Datensicherungssystem gilt es die Funktionalität im Auge zu behalten. Auch in einem NAS mit 2 oder mehr Festplatten kann es mal zum Ausfall einzelner Platten kommen. Dann sollte man in der Lage sein, auch einen Umzug der Datensicherung auf einen anderen Rechner zu bewerkstelligen.

Wenn die Daten einmal umziehen müssen

Auch hier gibt es Möglichkeiten. Zunächst verweise ich auf den Artikel bei ubuntuusers: https://wiki.ubuntuusers.de/Back_In_Time/#Backup-auf-neues-Laufwerk-umziehen

Um das, was dort steht, auch umzusetzen, ist es jedoch erforderlich, dass der NAS-Nutzer Kenntnisse im Umgang mit Konsolen - Befehlen und Programmen hat.

Welche Hardware wird benötigt

Für das Speichern der zu sichernden Daten wird ein externer Datenträger, der mindestens die Größe der gespeicherten Daten hat, benötigt. Ein USB-Stick wird nicht ausreichen. Eine externe Festplatte muss es schon sein. Die externe Festplatte muss über einen USB-Port an das NAS anschließbar sein.

Vorgehensweise

Da das NAS in der Regel keinen Video-Ausgang hat wird die Verbindung über das Netzwerk hergestellt. Im NAS muss die SSH-Funktionalität aktiv sein. Anschließend kann eine entsprechende Verbindung am heimische PC hergestellt werden.

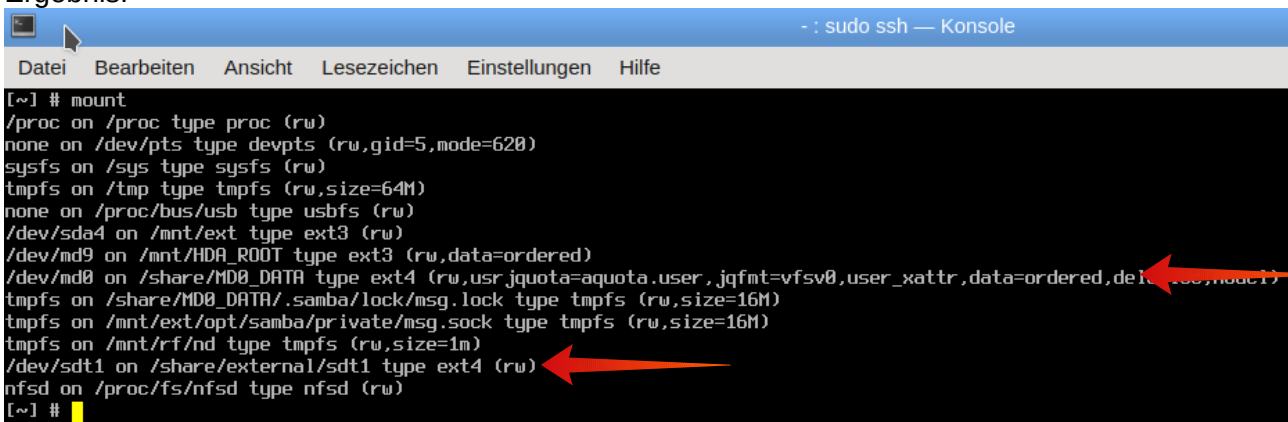
Für eine entsprechende Verbindung eignet sich auch das Programm „putty“. Ich beschreibe hier aber die Variante über die Konsole.

In der Konsole wird die Verbindung hergestellt:

```
ssh [benutzername]@[rechnername].[domain]
(um Problemen vorzubeugen sollte der Benutzer root oder admin Rechte haben)
```

A screenshot of a terminal window. The window has a blue title bar and a menu bar with the following items: Datei, Bearbeiten, Ansicht, Lesezeichen, Einstellungen, Hilfe. The terminal content shows the prompt 'ulrich@asus ~ \$' followed by the command 'sudo ssh root@qnap219p.lan' which is being typed, with a yellow cursor at the end of the command.

Ist die Verbindung hergestellt, gilt es Quell- und Zielverzeichnis zu finden. Wenn man Glück hat, wird die angeschlossene USB-Platte automatisch gemountet. Ein Aufruf von „mount“ zeigt das Ergebnis.



```
[~] # mount
/proc on /proc type proc (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
sysfs on /sys type sysfs (rw)
tmpfs on /tmp type tmpfs (rw,size=64M)
none on /proc/bus/usb type usbfs (rw)
/dev/sda4 on /mnt/ext type ext3 (rw)
/dev/md9 on /mnt/HDA_ROOT type ext3 (rw,data=ordered)
/dev/md0 on /share/MD0_DATA type ext4 (rw,usr_jquota=aquota.user,jqfmt=vfsv0,user_xattr,data=ordered,delimiters=,mode=1)
tmpfs on /share/MD0_DATA/.samba/lock/msg.lock type tmpfs (rw,size=16M)
tmpfs on /mnt/ext/opt/samba/private/msg.sock type tmpfs (rw,size=16M)
tmpfs on /mnt/rf/nd type tmpfs (rw,size=1m)
/dev/sdt1 on /share/external/sdt1 type ext4 (rw)
nfsd on /proc/fs/nfsd type nfsd (rw)
[~] #
```

Hat man nicht soviel Glück, muss zumindest das Zielverzeichnis gesucht und anschließend gemountet werden.

Danach kann mit der Herstellung einer Sicherungskopie begonnen werden. Das Programm „rsync“ eignet sich dazu am besten.

```
rsync -avH --progress [quelle] [ziel]
```

Beispiel:

```
rsync -avH --progress /share/MD0_DATA/Public/home/ulrich* /share/external/sdt1/
```

Die so angestoßene Datenübertragung kann mehrere Stunden dauern.

Hinweise

Back In Time erstellt inkrementelle Datensicherungen. Das heißt, dass nur veränderte Daten neu geschrieben werden. Für alles andere werden Hardlinks gesetzt.

Rsync berücksichtigt dies und erstellt eine genaue Kopie der Ursprungsdaten.

Sofern das alte und neue Datensicherungssystem im gleichen Netz vorhanden sind, braucht der Umweg über die externe Festplatte nicht zu erfolgen. Über „rsync“ können dann Quelle und Ziel direkt angesprochen werden.