

Nextcloud Backup mit BorgBackup

[Link zum Video](#)

Installationsanleitung

NAS mounten und Ordner erstellen

Hier werden die Ordner Daten, temp und restore in einem gemounteten Laufwerk erstellt.

```
mkdir -p /mnt/smb-ds/daten /mnt/smb-ds/temp /mnt/smb-ds/restore
```

Borg Backup installieren

```
apt install -y borgbackup
```

Repository erstellen

```
borg init -e repokey-blake2 /mnt/smb-ds/daten/
```

Das Backup-Skript erstellen

Mit Hilfe von Nano wird das Skript im Root-Verzeichnis erstellt.

```
nano /root/backup.sh
```

Inhalt des Skripts (**die Ordner dementsprechend anpassen**):

```
#!/bin/bash
#####

### Beispieldaten:
### logDirectory="/backup_logs/"
### backupDiscMount="/backup/"
### localBackupDir="/backup/temp"
```

```
#####
export BORG_PASSPHRASE='<borg-password>'
export BORG_UNKNOWN_UNENCRYPTED_REPO_ACCESS_IS_OK=yes
export BORG_RELOCATED_REPO_ACCESS_IS_OK=yes
startTime=$(date +%s)
currentDate=$(date --date @"$startTime" +"%Y%m%d_%H%M%S")
currentDateReadable=$(date --date @"$startTime" +"%d.%m.%Y - %H:%M:%S")
logDirectory="/var/log/"
logFile="${logDirectory}/${currentDate}.log"
backupDiscMount="/backup/daten/"
localBackupDir="/backup/temp"
borgRepository="${backupDiscMount}/"
borgBackupDirs="/home/data/ /var/www/nextcloud/ $localBackupDir/"
nextcloudFileDir='/var/www/nextcloud'
webserverServiceName='apache2'
webserverUser='www-data'
nextcloudDatabase='nextcloud'
dbUser='root'
dbPassword='<db-password>'
fileNameBackupDb='nextcloud-db.sql'
if [ ! -d "${logDirectory}" ]
then
    mkdir -p "${logDirectory}"
fi
errecho() { cat <<< "$@" 1>&2; }
exec > >(tee -i "${logFile}")
exec 2>&1
if [ "${id -u}" != "0" ]
then
    errecho "ERROR: This script has to be run as root!"
    exit 1
fi
if [ ! -d "${localBackupDir}" ]
then
    errecho "ERROR: The local backup directory ${localBackupDir} does not exist!"
    exit 1
fi
echo -e "\n##### Start des Backups: ${currentDateReadable} #####\n"
echo -e "Daten werden zusammengestellt"
dpkg --get-selections > "${localBackupDir}/software.list"
```

```

sudo -u "${webserverUser}" php ${nextcloudFileDir}/occ maintenance:mode --on
echo "apache2 wird gestoppt"
systemctl stop "${webserverServiceName}"
echo "Datenbanksicherung wird erstellt"
mysqldump --single-transaction --routines -h localhost -u "${dbUser}" -p"${dbPassword}"
"${nextcloudDatabase}" > "${localBackupDir}/${fileNameBackupDb}"
echo -e "\nBackup mit borgbackup"
borg create --stats \
    $borgRepository::"${currentDate}" \
    $localBackupDir \
    $borgBackupDirs
echo
echo "webserver wird gestartet"
systemctl start "${webserverServiceName}"
sudo -u "${webserverUser}" php ${nextcloudFileDir}/occ maintenance:mode --off
rm "${localBackupDir}"/software.list
rm -r "${localBackupDir}/${fileNameBackupDb}"
borg prune --progress --stats $borgRepository --keep-within=7d --keep-weekly=4 --keep-monthly=6
endTime=$(date +%s)
endDateReadable=$(date --date @"$endTime" +"%d.%m.%Y - %H:%M:%S")
duration=$((endTime-startTime))
durationSec=$((duration % 60))
durationMin=$((duration / 60 % 60))
durationHour=$((duration / 3600))
durationReadable=$(printf "%02d Stunden %02d Minuten %02d Sekunden" $durationHour $durationMin
$durationSec)
echo -e "\n##### Ende des Backups: ${endDateReadable} (${durationReadable}) #####\n"
echo -e "Plattenbelegung:\n"
df -h ${backupDiscMount}

```

Das Skript ausführbar machen

```
chmod +x /root/backup.sh
```

Automatisch um 03:00 Uhr ausführen lassen

```
# Cronjobs öffnen
```

```
crontab -e
```

```
# Ans Ende der Datei folgenden Befehl hinzufügen
```

```
0 3 * * * /root/backup.sh > /dev/null 2>&1
```

Erstellte Backups anzeigen lassen

```
borg list /backup/daten
```

Backup in Restore mounten um Daten wiederherzustellen

```
borg mount /mnt/backup/daten::<date> /mnt/backup/restore/
```

```
# Beispiel
```

```
borg mount /mnt/backup/daten::20200602_084033 /mnt/backu/restore/
```

Unmount Backup

```
borg umount /restore/
```

Die Bash für das Backup befindet sich unter `root/backup.sh`

Die Datei wird über Cronjob jede Nacht um 03:00 Uhr gestartet

Restore

Als root anmelden

```
# Backups anzeigen:
```

```
borg list /mnt/smb-ds/daten
```

```
# mount backup to browse
```

```
borg mount /mnt/smb-ds/daten::<date> /restore/
```

und weiter

Die Daten liegen im Borg Backup:

Daten: mnt/smb-ds/restore/mnt/Daten

Nextcloudfiles: Ordner Var

Datenbank: in smb-ds / *.sql

Backup zurückspielen

Ein Backup anzufertigen ist natürlich nur eine Seite der Medaille. Im Falle des Falles muss das Backup auch zurückgespielt werden können.

Vor der Wiederherstellung versetzen wir Nextcloud wie gewohnt in den Maintenance-Mode:

```
cd /var/www/nextcloud  
sudo -u www-data php occ maintenance:mode --on
```

Da wir hier ein komplettes Backup wieder einspielen, sollte der Webserver zunächst ein mal gestoppt werden:

```
# Hier kommt es darauf an welcher Server läuft (Nginx/Apache2)  
service apache2 stop
```

Vor der Wiederherstellung der Dateien der Cloud werden als erstes die entsprechenden Verzeichnisse gelöscht und neu angelegt. Dies sorgt dafür, dass keine Dateien aus dem Zustand vor dem Backup übrigbleiben:

```
rm -r /var/www/nextcloud/  
rm -r /var/nextcloud_data/ # Bei mir /mnt/Daten  
  
mkdir -p /var/www/nextcloud/  
mkdir -p /var/nextcloud_data/ # Bei mir /mnt/Daten
```

Nun gilt es zunächst das Nextcloud-Verzeichnis aus dem Backup wiederherzustellen. Da das Datum als Zeitstempel für das Backup diente, muss bei den folgenden Befehlen immer das korrekte Datum verwendet werden:

```
# gemountetes Backup
cd /mnt/smb-ds/restore
ls
```

Die Dateien wiederherstellen

```
rsync -Aax /mnt/smb-ds/restore/mnt/Daten/ /mnt/data/ --progress
```

```
rsync -Aax /mnt/smb-ds/restore/var/www/nextcloud/ /var/www/nextcloud/ --progress
```

Anschließend werden die Verzeichnis-Berechtigungen noch einmal explizit gesetzt:

```
chown -R www-data:www-data /var/www/nextcloud
chown -R www-data:www-data /var/nextcloud_data
```

Zum Schluss wird noch die Datenbank wiederhergestellt. Dazu wird zunächst die Datenbank entfernt und neu angelegt (für die Ausführung der Befehle muss das Passwort des Nextcloud-Datenbankusers angegeben werden):

```
mysql -h localhost -u nextclouduser -p -e "DROP DATABASE nextcloud"
mysql -h localhost -u nextclouduser -p -e "CREATE DATABASE nextcloud"
```

Das Einspielen des Backups geschieht durch folgenden Befehl:

```
mysql -h localhost -u nextclouduser -p nextcloud <
/mnt/Share/Backups/Nextcloud/NextcloudBackup_DB_20170912.sql
```

Nun wird der Webserver wieder gestartet:

```
service apache2 start
```

Redis installieren

Damit Nextcloud bei mir richtig läuft muss noch redis installiert werden:

Nextcloud Advanced Configuration

Nun wird Nextcloud wieder aus dem Maintenance-Mode geholt.

```
cd /var/www/nextcloud
sudo -u www-data php occ maintenance:mode --off
```

Zum Schluss muss noch ein Befehl abgesetzt werden, damit Nextcloud-Clients mitbekommen können, dass sich hier etwas durch das Wiederherstellen eines Backups verändert hat:

```
cd /var/www/nextcloud
sudo -u www-data php occ maintenance:data-fingerprint
```

Nach diesen Schritten wurde ein Backup von Nextcloud wiederhergestellt und es kann wieder mit der Cloud gearbeitet werden.

Backup unmount

```
# Unmount the Backup
borg umount /restore/
```

Backups Bereinigen

Beispiele

```
# delete a single backup archive:
$ borg delete /path/to/repo::Monday
# actually free disk space:
$ borg compact /path/to/repo

# delete all archives whose names begin with the machine's hostname followed by "-"
$ borg delete --prefix '{hostname}-' /path/to/repo

# delete all archives whose names contain "-2012-"
$ borg delete --glob-archives '*-2012-*' /path/to/repo

# see what would be deleted if delete was run without --dry-run
$ borg delete --list --dry-run -a '*-May-*' /path/to/repo

# delete the whole repository and the related local cache:
$ borg delete /path/to/repo
You requested to completely DELETE the repository *including* all archives it contains:
repo                Mon, 2016-02-15 19:26:54
root-2016-02-15     Mon, 2016-02-15 19:36:29
newname             Mon, 2016-02-15 19:50:19
Type 'YES' if you understand this and want to continue: YES
```

Fehlerbehebung

Failed to create/acquire the lock

```
borg break-lock /mnt/smb-ds/daten/
```

Revision #8

Created 21 March 2023 10:09:52 by Hermann

Updated 7 June 2024 06:34:05 by Hermann