

# Server & Desktop Backup-Lösungen

Die ersten Sicherungen werden eine Weile dauern. Die folgenden sind viel schneller, aber das hängt davon ab, wie viel geändert wurde. Es werden nur die Änderungen gespeichert.

## rsync

Folge zunächst unserer [rsync](#) Anleitung.

Die Snapshots werden lokal und über die rsync-Daemon remote gespeichert.

Diese Backup-Lösung ist nur für ein internes Netzwerk zu empfehlen. Verwende stattdessen eine verschlüsselte Sicherungsmethode mit [borg](#).

## Abhängigkeiten

Das Skript benötigt `inetutils` für den `hostname` Befehl.

```
pacman -S inetutils
```

## Anmeldeinformationen

```
echo "$password" > /etc/rsyncd.password  
chmod 400 /etc/rsyncd.password
```

## Skript

Füge deine Daten zu `DAEMONUSER=""` und `DAEMONHOST=""` hinzu.

```
nano /root/rsnapbackup.sh
```

```
#!/bin/sh  
  
## Based on:  
## my own rsync-based snapshot-style backup procedure  
## (cc) marcio rps AT gmail.com  
  
# config vars  
  
SRC="/"   
SNAP="/root/backup/"  
OPTS="--rltogiPhv --stats --delay-updates --delete --chmod=a-w"  
EXCL="--exclude-from=/root/backup-filter.rule"  
DAEMONUSER=""
```

```
DAEMONHOST=""
HOSTNAME=$(hostname)
MINCHANGES=1

# run this process with real low priority

ionice -c 3 -p $$
renice +12 -p $$

# List and save installed packages
pacman -Qn | awk '{ print $1 }' > /root/pkglist

# sync

rsync $OPTS $EXCL $SRC $SNAP/latest >> $SNAP/rsync.log

# check if enough has changed and if so
# make a hardlinked copy named as the date

COUNT=$( wc -l $SNAP/rsync.log | cut -d" " -f1 )
if [ $COUNT -gt $MINCHANGES ] ; then
    DATETAG=$(date +%Y-%m-%d-%H:%M)
    if [ ! -e $SNAP/$DATETAG ] ; then
        cp -al $SNAP/latest $SNAP/$DATETAG
        chmod u+w $SNAP/$DATETAG
        mv $SNAP/rsync.log $SNAP/$DATETAG
        chmod u-w $SNAP/$DATETAG
    fi
fi

rsync -avAXHP --delete --password-file=/etc/rsyncd.password $SNAP
rsync://$DAEMONUSER@$DAEMONHOST/archive/backup/$HOSTNAME

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

## Ordner und Dateien ausschließen

Dies ist ein Beispiel. Füge alles hinzu, was du nicht sichern möchtest. Und ändere den home \$USER.

```
nano /root/backup-filter.rule
```

```
/dev/*
/proc/*
/sys/*
/tmp/*
/run/*
/mnt/*
/media/*
/lost+found
```

```
# root user
/root/backup/*
/root/.cache/*
# Home user
/home/$USER/.cache/*
```

## borg

Folge zunächst unserem [borg](#) Tutorial.

Die Snapshots werden über SSH gespeichert.

## Skript

Vergiss nicht, zuerst das Borg Repo zu erstellen und dem Skript die Anmeldedaten hinzuzufügen.

```
borg init --encryption=keyfile-blake2 --make-parent-dirs
ssh://username@remote.host.address:$port>/~/backups/borg/{hostname}
```

Füge deine ausgeschlossenen Ordner/Dateien hinzu --exclude '/home/\*/ .cache/\*' \ und füge unter :: '{hostname}-{now}' \ die Ordner/Dateien hinzu, die du sichern willst.

```
#!/bin/sh

# Setting this, so the repo does not need to be given on the commandline:
export BORG_REPO=ssh://username@example.com:2022/~/backups/borg/{hostname}

# See the section "Passphrase notes" for more infos.
export BORG_PASSPHRASE='XYZl0ngandsecurepa_55_phrasea&&123'

# some helpers and error handling:
info() { printf "\n%s %s\n\n" "$( date )" "$*" >&2; }
trap 'echo $( date ) Backup interrupted >&2; exit 2' INT TERM

info "Starting backup"

# Backup the most important directories into an archive named after
# the machine this script is currently running on:

borg create                                \
  --verbose                               \
  --filter AMEhsx                         \
  --list                                  \
  --stats                                 \
  --progress                             \
  --verbose                               \
  --show-version                          \
  --show-rc                               \
```

```
--compression zstd,11      \
--exclude-caches            \
--exclude '/home/*/.cache/*' \
--exclude '/var/tmp/*'      \
                             \
::'{hostname}-{now}'        \
/etc                       \
/home                     \
/root                    \
/var                     \
```

```
backup_exit=$?
```

```
info "Pruning repository"
```

```
# Use the `prune` subcommand to maintain 7 daily, 4 weekly and 6 monthly
# archives of THIS machine. The '{hostname}-' prefix is very important to
# limit prune's operation to this machine's archives and not apply to
# other machines' archives also:
```

```
borg prune                \
--list                    \
--prefix '{hostname}-'    \
--show-rc                 \
--keep-daily 7            \
--keep-weekly 4           \
--keep-monthly 6          \
--keep-yearly 1           \
```

```
prune_exit=$?
```

```
# use highest exit code as global exit code
global_exit=$(( backup_exit > prune_exit ? backup_exit : prune_exit ))
```

```
if [ ${global_exit} -eq 0 ]; then
    info "Backup and Prune finished successfully"
elif [ ${global_exit} -eq 1 ]; then
    info "Backup and/or Prune finished with warnings"
else
    info "Backup and/or Prune finished with errors"
fi
```

```
exit ${global_exit}
```

## Crontab - rsync und borg

Folge zunächst unserem [crontab](#) Tutorial und füge folgendes für den Root- Benutzer hinzu:

```
@daily /root/rsnapbackup.sh
```

```
@daily /root/bsnapbackup.sh
```

- @yearly
- @annually
- @monthly
- @weekly
- @daily
- @hourly
- @reboot

## Syncthing

Folge zunächst unserem [Syncthing](#) Tutorial für beide Geräte (Backupserver und Datengerät).

### Gerät hinzufügen

Füge den Backupserver zu deinem Client unter Remote Devices hinzu.

### Ordner hinzufügen

- Füge einen Ordner unter Folder hinzu und wähle unter General den Ordner aus der gesichert werden soll.
- Wähle unter Sharing den Backupserver aus.
- Unter File Versioning kannst du die Staggered File Versioning hinzufügen, die dir mehr Sicherheit gibt. Aber schau unter <https://docs.syncthing.net/users/versioning.html> nach und wähle aus was dir am Besten gefällt.
- Check auch Advanced und Folder type und wähle wieder, was am besten zu dir passt. Keepass kann zum Beispiel mit Send & Receive verwendet werden, wenn du deine Datenbank auf beiden Geräten synchronisieren möchtest.

From:

<http://wiki.techsaviours.org/> - **Your Digital Privacy DIY Solutions | TECH SAVIOURS .ORG**

Permanent link:

<http://wiki.techsaviours.org/de/backup/server?rev=1662169233>

Last update: **2022/10/24 08:24**

