

[Startseite](#) » [Nextcloud Installationsanleitungen](#) » Nextcloud Sicherung und Wiederherstellung

[NEXTCLOUD INSTALLATIONSANLEITUNGEN](#)

Nextcloud Sicherung und Wiederherstellung

von [Carsten Rieger](#) | Aktualisiert [11. Oktober 2020](#)



Nextcloud Sicherung und Wiederherstellung

. . .

Ein neuerer Guide ist verfügbar: [Nextcloud Backup \(Borg\) mit De-Duplizierung](#)

. . .

M

it dem skizzierten Vorgehen wird der Nextcloud Server regelmäßig gesichert und rotiert. Die Sicherungen enthalten u.a.

- das Web – und Datenverzeichnis: `/var/www/nextcloud` und `/var/nc_data`
- sowie die Nextcloud Datenbank.

Ersetzen Sie im weiteren Verlauf bitte den `osusername` und die Datenbank bzw . das Datenbankpasswort `nextcloud` mit Ihren Werten.

Beginnen wir und wechseln dazu in den `sudo` Mode:

```
sudo -s
```

1. Nextcloud Wartungsmodus aktivieren:

```
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:mode --on
```

2. Sicherung des Webverzeichnisses:

```
tar -cpzf /home/osusername/ncserver_`date +%w``.tar.gz -C /var/www/nextcloud .
```

3. Sicherung des Datenverzeichnisses:

```
tar -cpzf /home/osusername/ncdata_`date +%w``.tar.gz -C /var/nc_data .
```

4. Sicherung der Datenbank:

```
mysqldump --single-transaction -h localhost -unextcloud -pnextcloud nextcloud >  
/home/osusername/ncdb_`date +%w``.sql
```

5. Nextcloud Wartungsmodus deaktivieren:

```
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:mode --off
```

Sicherung erstellt.

Wiederherstellung der zuvor erstellten Nextcloud Sicherung

Die Sicherung besteht u.a. aus

- dem Webverzeichnis: /var/www/nextcloud
- dem Datenverzeichnis: /var/nc_data/appdata_<id>
- der Datenbank: nextcloud.

Wir demonstrieren die exemplarische Herstellung einer Montags-Sicherung (_1). Wechseln Sie dafür zuerst in den sudo Mode:

```
sudo -s
```

1. Nextcloud Wartungsmodus aktivieren:

```
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:mode --on
```

2. Stoppen aller relevanten Services:

```
/usr/sbin/service nginx stop  
/usr/sbin/service php7.4-fpm stop
```

3. Löschen der alten Verzeichnisse:

```
rm -r /var/www/nextcloud/  
rm -r /var/nc_data/
```

4. Anlegen der Verzeichnisse und Entpacken der Sicherung:

```
mkdir -p /var/www/nextcloud/  
mkdir -p /var/nc_data/  
tar -xpf /home/osusername/ncserver_1.tar.gz -C /var/www/nextcloud/  
tar -xpf /home/osusername/ncdata_1.tar.gz -C /var/nc_data/
```

5. Berechtigungen setzen:

```
chown -R www-data:www-data /var/nc_data /var/www
```

6. Nextcloud Datenbank wiederherstellen:

– Löschen der alten Datenbank

```
mysql -h localhost -uroot -pnextcloud -e "DROP DATABASE nextcloud"
```

– Anlegen einer leeren Datenbank

```
mysql -h localhost -uroot -pnextcloud -e "CREATE DATABASE nextcloud CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci"
```

– Berechtigungen setzen

```
mysql -h localhost -uroot -pnextcloud -e "GRANT ALL PRIVILEGES on nextcloud.* to nextcloud@localhost"
```

– Wiederherstellen der Datenbank

```
mysql -h localhost -unextcloud -pnextcloud nextcloud < /home/osusername/ncdb_1.sql
```

7. Neustart der relevanten Services:

```
/usr/sbin/service php7.4-fpm start  
/usr/sbin/service nginx start
```

8. Nextcloud fingerprint setzen:

Nach einem Daten- oder Datenbankrestore sollte immer `occ maintenance:data-fingerprint` ausgeführt werden. Dieser Befehl passt den ETag aller Dateien an und teilt den Sync-Clients mit, dass Dateien verändert wurden.

```
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:data-fingerprint
```

9. Nextcloud Wartungsmodus deaktivieren:

```
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:mode --off
```

Done.

Nachfolgen stelle ich Ihnen ein exemplarisches Backupsript zur Verfügung, das nur noch hinsichtlich der Verzeichnisse und Datenbankparameter angepasst werden muss:

```
#!/bin/bash
#enable Maintenance Mode to prevent users from working with Nextcloud
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:mode --on
#create directory first: mkdir /backup -p
BACKUP_STORE=/backup
#create directory first: mkdir /backup-repository -p
ARCHIVE_STORE=/backup-repository
#declare dateformat and numbering of backups
CURRENT_TIME_FORMAT="%w"
#print start date/time
echo "START: $(date)"
#list of folders to be backed up, feel free to add/remove directories
FOLDERS_TO_BACKUP=(
"/root/"
"/etc/fail2ban/"
"/etc/letsencrypt/"
"/etc/mysql/"
"/etc/nginx/"
"/etc/php/"
"/etc/ssh/"
"/etc/pam.d/"
"/etc/ssl/"
"/var/www/"
"/var/nc_data/"
"/nc_data/rainloop-storage/"
)
#declare the backup filename
ARCHIVE_FILE="$ARCHIVE_STORE/nc_backup_$(date +%CURRENT_TIME_FORMAT).tar.gz"
#change directory
cd $BACKUP_STORE
#start rsync to back up the folders
for FOLDER in ${FOLDERS_TO_BACKUP[@]}
do
    if [ -d "$FOLDER" ];
    then
        echo "Copying $FOLDER..."
        rsync -AarX --delete $FOLDER $BACKUP_STORE
    else
        echo "Skipping $FOLDER (does not exist!)"
    fi
done
#copy the fstab
[ -f /etc/fstab ] && cp /etc/fstab $BACKUP_STORE/etc/
#copy the mail configuration
[ -f /etc/msmtprc ] && cp /etc/msmtprc $BACKUP_STORE/etc/
#create a database back up
mysqldump --single-transaction -h localhost -unextcloud -pnextcloud nextcloud >
$BACKUP_STORE/ncdb_date +%w".sql
#print the database backup size
```

```
mysql -hlocalhost -unextcloud -pnextcloud -e "SELECT table_schema
'DB',round(sum(data_length+index_length)/1024/1024,1) 'Size (MB)' from
information_schema.tables WHERE table_schema = 'nextcloud';"
#create the directories
mkdir -p $(dirname $ARCHIVE_FILE)
#compress all data
tar -cpzf $ARCHIVE_FILE .
#print back up size
echo "nc_backup size: $(stat --printf='%s' $ARCHIVE_FILE | numfmt --to=iec)"
#stop all services
/usr/sbin/service nginx stop
/usr/sbin/service mysql stop
/usr/sbin/service redis-server stop
/usr/sbin/service php7.4-fpm stop
#remove copied files
[ -f $BACKUP_STORE/ncdb_date +"%w".sql ] && rm -f $BACKUP_STORE/ncdb_date +"%w".sql
[ -f /etc/msmtprc ] && rm -f $BACKUP_STORE/etc/msmtprc
[ -f /etc/fstab ] && rm -f $BACKUP_STORE/etc/fstab
#restart all services
/usr/sbin/service nginx stop
/usr/sbin/service mysql restart
/usr/sbin/service redis-server restart
#enable if Collabora and/or OnlyOffice are used
/usr/bin/docker restart COLLABORAOFFICE
/usr/sbin/service php7.4-fpm restart
/usr/sbin/service nginx restart
#disable maintenance mode
sudo -u www-data php7.4 /var/www/nextcloud/occ maintenance:mode --off
#Nextcloud optimizations
#Ubuntu:
/usr/bin/redis-cli -s /var/run/redis/redis-server.sock <<EOF
#Debian: /usr/bin/redis-cli -s /var/run/redis/redis.sock <<EOF
FLUSHALL
quit
EOF
sudo -u www-data php7.4 /var/www/nextcloud/occ files:scan --all
sudo -u www-data php7.4 /var/www/nextcloud/occ files:scan-app-data
#check for Nextcloud updates
echo "Nextcloud apps are checked for updates..."
sudo -u www-data php7.4 /var/www/nextcloud/occ app:update --all
#print end date/time
echo "END: $(date)"
exit 0
```

Über eine Spende würden sich meine Frau, meine Zwillinge und ich sehr freuen!

© Carsten Rieger IT-Services

Carsten Rieger

Carsten Rieger ist ein angestellter Senior IT-Systemengineer und zudem auch als Kleinunternehmer (Freelancer) aktiv. Er arbeitet seit 2005 im Linux- und Microsoftumfeld, ist ein Open Source Enthusiast und hoch motiviert, Linux Installationen und Troubleshooting durchzuführen. Dabei arbeitet er vorrangig mit Debian und Ubuntu Linux, Nginx und Apache Webservern, MariaDB, PHP, Cloud Infrastrukturen (bspw. Nextcloud) und auch vielen anderen Open Source Projekten (bspw. HAProxy, Jitsi, BigBlueButton, CheckMK etc.). Zudem engagiert er sich ehrenamtlich für die [Dr. Michael & Angela Jacobi Stiftung](#) - und das schon seit 2012.

32 BEITRÄGE

AUTOR

