

Backup auf ein NAS

Ziel

Der komplette relevante MagicMirror-Ordner eines Raspberry Pi wird manuell auf einem NAS gesichert.

Jedes Backup wird als komprimiertes TAR.GZ-Archiv gespeichert.

Das Backup:

- wird ausschließlich manuell gestartet
- läuft nach dem Start im Hintergrund weiter
- erhält einen Ordner mit Datum und Uhrzeit
- wird komprimiert gespeichert
- wird nach Erstellung vollständig geprüft
- erhält zusätzlich eine SHA-256-Prüfsumme
- wird nur gestartet, wenn das richtige NAS tatsächlich gemountet ist
- bleibt bei einem Fehler als `_INCOMPLETE` gekennzeichnet
- verhindert zwei gleichzeitig laufende Backups

Nicht gesichert werden:

```
node_modules
```

`node_modules` enthalten installierte Node.js-Abhängigkeiten.

Diese sollten insbesondere nach einem Wechsel auf einen neuen Raspberry Pi, eine andere Node.js-Version oder ein neues Betriebssystem neu installiert werden.

Backup-Struktur

Beispiel:

```
NAS
├── Backup-Freigabe
│   └── Manuelle_Backups
│       ├── MagicMirror_2026-10-07_12-30-00
│       │   ├── MagicMirror.tar.gz
│       │   └── MagicMirror.tar.gz.sha256
│       └── MagicMirror_2026-11-01_18-42-10
│           ├── MagicMirror.tar.gz
│           └── MagicMirror.tar.gz.sha256
```

Platzhalter dieser Anleitung

Platzhalter	Bedeutung
<PI_USER>	Benutzer auf dem Raspberry Pi
<NAS_HOST>	IP-Adresse oder DNS-Name des NAS
<NAS_SHARE>	SMB-Freigabe auf dem NAS
<NAS_USER>	Benutzername auf dem NAS
<NAS_PASSWORT>	Passwort des NAS-Benutzers
<UID>	numerische Benutzer-ID des Pi-Benutzers
<GID>	numerische Gruppen-ID der Gruppe nasusers

Vorbereitung auf einem neuen Raspberry Pi

1. Benutzer einrichten

Bei Raspberry Pi OS wird normalerweise bereits bei der Installation ein Benutzer angelegt.

Falls ein zusätzlicher Benutzer benötigt wird:

```
sudo adduser <PI_USER>
```

Optional sudo-Rechte vergeben:

```
sudo usermod -aG sudo <PI_USER>
```

Kontrolle:

```
id <PI_USER>
```

2. NAS-Gruppe anlegen

```
sudo groupadd nasusers
```

Benutzer hinzufügen:

```
sudo usermod -aG nasusers <PI_USER>
```

Anschließend neu anmelden oder den Raspberry Pi neu starten.

Kontrolle:

```
getent group nasusers
```

```
id <PI_USER>
```

3. UID und GID ermitteln

UID:

```
id -u <PI_USER>
```

GID:

```
getent group nasusers
```

Beispiel:

```
nasusers:x:1003:<PI_USER>
```

In diesem Beispiel wäre:

```
GID = 1003
```

4. CIFS-Unterstützung installieren

```
sudo apt update  
sudo apt install cifs-utils
```

Ein Samba-Server wird nicht benötigt.

Der Raspberry Pi greift lediglich als SMB/CIFS-Client auf das NAS zu.

5. Mountpunkt erstellen

```
sudo mkdir -p /mnt/nasbackup
```

6. Zugangsdaten zum NAS speichern

Verzeichnis anlegen:

```
sudo mkdir -p /etc/samba
```

Datei erstellen:

```
sudo nano /etc/samba/credentials-nasbackup
```

Inhalt:

```
username=<NAS_USER>  
password=<NAS_PASSWORT>
```

Optional:

```
domain=WORKGROUP
```

Datei absichern:

```
sudo chown root:root /etc/samba/credentials-nasbackup  
sudo chmod 600 /etc/samba/credentials-nasbackup
```

Kontrolle:

```
ls -l /etc/samba/credentials-nasbackup
```

Erwartet:

```
-rw----- 1 root root ...
```

7. NAS in /etc/fstab eintragen

Vorher Sicherung:

```
sudo cp /etc/fstab /etc/fstab.backup
```

Bearbeiten:

```
sudo nano /etc/fstab
```

Eintrag:

```
//<NAS_HOST>/<NAS_SHARE> /mnt/nasbackup cifs  
credentials=/etc/samba/credentials-  
nasbackup,vers=3.0,iocharset=utf8,uid=<UID>,gid=<GID>,file_mode=0660,dir_mod  
e=0770,_netdev,nofail,x-systemd.automount,x-systemd.mount-timeout=10s 0 0
```

Bedeutung der Mount-Optionen

Option	Bedeutung
credentials=...	NAS-Zugangsdaten liegen in einer geschützten Datei
vers=3.0	SMB 3 verwenden
uid=<UID>	lokaler Eigentümer der Dateien

Option	Bedeutung
gid=<GID>	lokale Gruppe der Dateien
file_mode=0660	Lese-/Schreibzugriff für Benutzer und Gruppe
dir_mode=0770	Zugriff auf Verzeichnisse für Benutzer und Gruppe
_netdev	Netzwerk-Dateisystem
nofail	Boot funktioniert auch bei ausgeschaltetem NAS
x-systemd.automount	NAS wird erst bei tatsächlichem Zugriff gemountet
x-systemd.mount-timeout=10s	begrenzt die Mount-Wartezeit

8. Automount aktivieren

```
sudo systemctl daemon-reload
sudo systemctl start mnt-nasbackup.automount
```

Status:

```
systemctl status mnt-nasbackup.automount --no-pager -l
```

Korrekt:

```
Active: active (waiting)
```

9. NAS-Verbindung testen

Mount durch Zugriff auslösen:

```
ls -la /mnt/nasbackup
```

Prüfen:

```
findmnt /mnt/nasbackup
```

Es muss ein CIFS-Eintrag für den erwarteten Share vorhanden sein.

10. Speicherplatz prüfen

```
df -h /mnt/nasbackup
```

11. Schreibzugriff prüfen

```
touch /mnt/nasbackup/test_backup.txt
ls -l /mnt/nasbackup/test_backup.txt
```

```
rm /mnt/nasbackup/test_backup.txt
```

Erst wenn dieser Test funktioniert, das Backup-Script einrichten.

Backup-Script einrichten

1. Script-Verzeichnis anlegen

```
mkdir -p ~/bin
```

2. Script erstellen

```
nano ~/bin/backup_magicmirror.sh
```

Inhalt:

```
#!/usr/bin/env bash

SOURCE="$HOME/MagicMirror"
NAS="/mnt/nasbackup"
BACKUP_BASE="$NAS/Manuelle_Backups"
LOGDIR="$HOME/magicmirror-backup-logs"

# An eigenes NAS anpassen:
EXPECTED_SOURCE="//<NAS_HOST>/<NAS_SHARE>"

check_nas() {

    if ! timeout 15s stat "$NAS" >/dev/null 2>&1; then
        echo "FEHLER: NAS ist nicht erreichbar."
        return 1
    fi

    MOUNT_INFO="$(findmnt -T "$NAS" -n -o FSTYPE,SOURCE | tail -n 1)"

    FSTYPE="$(awk '{print $1}' <<< "$MOUNT_INFO")"
    MOUNT_SOURCE="$(awk '{print $2}' <<< "$MOUNT_INFO")"

    if [[ "$FSTYPE" != "cifs" || "$MOUNT_SOURCE" != "$EXPECTED_SOURCE" ]];
then
        echo "FEHLER: NAS-Share ist nicht korrekt eingebunden."
        echo "Gefunden: $MOUNT_INFO"
        return 1
    fi
}
```

```
fi

return 0
}

if [[ "${1:-}" == "--worker" ]]; then

    FINAL_DEST="$2"
    TEMP_DEST="${FINAL_DEST}_INCOMPLETE"

    ARCHIVE="$TEMP_DEST/MagicMirror.tar.gz"
    CHECKSUM="$TEMP_DEST/MagicMirror.tar.gz.sha256"

    # Nur ein Backup gleichzeitig.
    exec 9>/tmp/magicmirror-backup.lock

    if ! flock -n 9; then
        echo "FEHLER: Es läuft bereits ein MagicMirror-Backup."
        exit 1
    fi

    echo
    echo "=====
    echo "MagicMirror Backup"
    echo "Start:  $(date)"
    echo "Quelle:  $SOURCE"
    echo "Ziel:    $FINAL_DEST"
    echo "=====
    echo

    if ! check_nas; then
        echo "BACKUP ABGEBROCHEN: NAS nicht verfügbar."
        exit 2
    fi

    mkdir -p "$TEMP_DEST" || exit 3

    echo "Erstelle MagicMirror-Archiv ..."
    echo "node_modules werden bewusst NICHT gesichert."
    echo

    tar \
        --create \
        --gzip \
```

```
--dereference \  
--exclude='MagicMirror/node_modules' \  
--exclude='MagicMirror/node_modules/*' \  
--exclude='*/node_modules' \  
--exclude='*/node_modules/*' \  
--file="$ARCHIVE" \  
--directory="$HOME" \  
MagicMirror  
  
RESULT=$?  
  
if [[ $RESULT -ne 0 ]]; then  
  
    echo  
    echo "BACKUP FEHLGESCHLAGEN"  
    echo "tar Fehlercode: $RESULT"  
    echo "Unvollständiges Backup:"  
    echo "$TEMP_DEST"  
  
    exit "$RESULT"  
fi  
  
echo  
echo "Prüfe gzip ..."  
  
if ! gzip -t "$ARCHIVE"; then  
    echo "FEHLER: gzip-Prüfung fehlgeschlagen."  
    exit 4  
fi  
  
echo "gzip: OK"  
  
echo  
echo "Prüfe TAR ..."  
  
if ! tar -tzf "$ARCHIVE" >/dev/null; then  
    echo "FEHLER: TAR-Prüfung fehlgeschlagen."  
    exit 5  
fi  
  
echo "tar: OK"  
  
echo  
echo "Erzeuge SHA-256-Prüfsumme ..."  
  
(
```

```
    cd "$TEMP_DEST" || exit 1
    sha256sum MagicMirror.tar.gz > MagicMirror.tar.gz.sha256
) || exit 6

(
    cd "$TEMP_DEST" || exit 1
    sha256sum -c MagicMirror.tar.gz.sha256
) || exit 7

echo "SHA-256: OK"

mv "$TEMP_DEST" "$FINAL_DEST" || exit 8

echo
echo "===== "
echo "BACKUP ERFOLGREICH"
echo "Ende: $(date)"
echo
echo "Archiv:"
echo "$FINAL_DEST/MagicMirror.tar.gz"
echo
echo "Prüfsumme:"
echo "$FINAL_DEST/MagicMirror.tar.gz.sha256"
echo
echo "Größe:"
du -h "$FINAL_DEST/MagicMirror.tar.gz"
echo "===== "

exit 0
fi

if [[ ! -d "$SOURCE" ]]; then

    echo "FEHLER: MagicMirror-Ordner wurde nicht gefunden:"
    echo "$SOURCE"

    exit 1
fi

echo "Prüfe NAS ..."

if ! check_nas; then

    echo
```

```
echo "BACKUP NICHT GESTARTET"
echo "Das NAS ist offline oder der Share nicht gemountet."
echo "Es wurden KEINE Daten lokal nach /mnt/nasbackup geschrieben."

exit 1
fi

mkdir -p "$BACKUP_BASE"
mkdir -p "$LOGDIR"

STAMP="$(date '+%Y-%m-%d_%H-%M-%S' )"

DEST="$BACKUP_BASE/MagicMirror_$STAMP"

LOG="$LOGDIR/MagicMirror_$STAMP.log"

echo "NAS ist erreichbar."
echo
echo "Backup wird im Hintergrund gestartet."
echo "Quelle: $SOURCE"
echo "Ziel:   $DEST"
echo "Log:    $LOG"
echo

nohup "$0" --worker "$DEST" >"$LOG" 2>&1 < /dev/null &

PID=$!

echo "Backup-Prozess gestartet."
echo "PID: $PID"
echo
echo "Status ansehen mit:"
echo "tail -f \"$LOG\""
```

3. Script ausführbar machen

```
chmod +x ~/bin/backup_magicmirror.sh
```

Backup verwenden

Backup starten

```
~/bin/backup_magicmirror.sh
```

Danach läuft die Sicherung im Hintergrund.

Das Terminal bzw. die SSH-Verbindung kann geschlossen werden.

Status ansehen

Neuestes Log:

```
tail -f "$(ls -t ~/magicmirror-backup-logs/*.log | head -1)"
```

Ctrl+C beendet nur die Loganzeige.

Das Backup läuft unabhängig davon weiter.

Prüfen, ob ein Backup läuft

```
pgrep -af 'backup_magicmirror|tar|gzip'
```

Keine Ausgabe bedeutet, dass aktuell kein Backup-Prozess läuft.

Erfolgreiches Backup

Im Log muss am Ende erscheinen:

```
gzip: OK  
tar: OK  
MagicMirror.tar.gz: OK  
SHA-256: OK  
  
BACKUP ERFOLGREICH
```

Ein fertiges Backup trägt keinen Zusatz `_INCOMPLETE`.

Backup überprüfen

SHA-256 prüfen

In das Backup-Verzeichnis wechseln:

```
cd /mnt/nasbackup/Manuelle_Backups/MagicMirror_YYYY-MM-DD_HH-MM-SS
```

Prüfen:

```
sha256sum -c MagicMirror.tar.gz.sha256
```

Erwartet:

```
MagicMirror.tar.gz: OK
```

gzip prüfen

```
gzip -t MagicMirror.tar.gz
```

Keine Ausgabe bedeutet Erfolg.

Inhalt des Archivs ansehen

```
tar -tzf MagicMirror.tar.gz | less
```

Archiv auf dem NAS entpacken

Falls das Archiv lediglich angesehen werden soll, kann es in einen separaten Ordner entpackt werden.

Beispiel:

```
mkdir MagicMirror_ENTPACKT
```

```
tar -xzf MagicMirror.tar.gz -C MagicMirror_ENTPACKT
```

Da Symlinks beim Backup dereferenziert werden und `node_modules` nicht enthalten sind, besteht das Archiv überwiegend aus normalen Dateien und Verzeichnissen.

Wiederherstellung auf einem neuen Raspberry Pi

1. Grundsystem installieren

Auf dem neuen Raspberry Pi zuerst:

- Raspberry Pi OS installieren
- Benutzer einrichten
- Netzwerk konfigurieren
- NAS-Mount nach dieser Anleitung einrichten
- passende Node.js-Version installieren
- MagicMirror-Grundvoraussetzungen installieren

2. Backup prüfen

```
cd /mnt/nasbackup/Manuelle_Backups/MagicMirror_YYYY-MM-DD_HH-MM-SS  
  
sha256sum -c MagicMirror.tar.gz.sha256
```

Nur bei:

```
MagicMirror.tar.gz: OK
```

weiterarbeiten.

3. Vorhandenen MagicMirror-Ordner sichern

Falls bereits ein Ordner existiert:

```
mv ~/MagicMirror ~/MagicMirror_alt
```

4. Backup wiederherstellen

```
tar -xzf /mnt/nasbackup/Manuelle_Backups/MagicMirror_YYYY-MM-DD_HH-MM-SS/MagicMirror.tar.gz -C ~
```

Danach existiert wieder:

```
~/MagicMirror
```

5. Node.js-Abhängigkeiten neu installieren

node_modules befinden sich absichtlich nicht im Backup.

Deshalb müssen die Abhängigkeiten passend zur neuen Umgebung neu installiert werden.

Für den MagicMirror-Hauptordner und installierte Drittmodule ist die jeweils aktuelle Installationsanleitung zu verwenden.

Das ist insbesondere bei einem Wechsel auf einen neuen Pi oder eine neue Node.js-/MagicMirror-Version wichtig.

Verhalten bei nicht erreichbarbarem NAS

Das Script prüft vor dem Start:

- ob der Mountpunkt erreichbar ist
- ob dort tatsächlich CIFS verwendet wird
- ob genau der konfigurierte NAS-Share gemountet ist

Ist das NAS offline, wird das Backup nicht gestartet.

Dadurch kann nicht versehentlich auf die lokale SSD unter

```
/mnt/nasbackup
```

geschrieben werden.

Wichtige Dateien

Pfad	Funktion
~/MagicMirror	zu sichernder MagicMirror-Ordner
~/bin/backup_magicmirror.sh	Backup-Script
~/magicmirror-backup-logs	Backup-Logs
/mnt/nasbackup	NAS-Mountpunkt
/mnt/nasbackup/Manuelle_Backups	Backup-Ziel
/etc/samba/credentials-nasbackup	NAS-Zugangsdaten
/etc/fstab	NAS-Automount-Konfiguration

Kurzfassung

Backup starten:

```
~/bin/backup_magicmirror.sh
```

Status ansehen:

```
tail -f "$(ls -t ~/magicmirror-backup-logs/*.log | head -1)"
```

Backup prüfen:

```
cd /mnt/nasbackup/Manuelle_Backups/MagicMirror_YYYY-MM-DD_HH-MM-SS  
sha256sum -c MagicMirror.tar.gz.sha256
```

Ein Backup gilt als gültig, wenn:

- im Log BACKUP ERFOLGREICH steht
- der Ordner keinen Zusatz _INCOMPLETE besitzt
- die SHA-256-Prüfung OK meldet

From:

<https://wiki.mahlen.eu/> - **Smart-Home Wiki**

Permanent link:

https://wiki.mahlen.eu/doku.php?id=magicmirror:magicmirror_tipps:backup&rev=1791367156

Last update: **07.10.2026**

