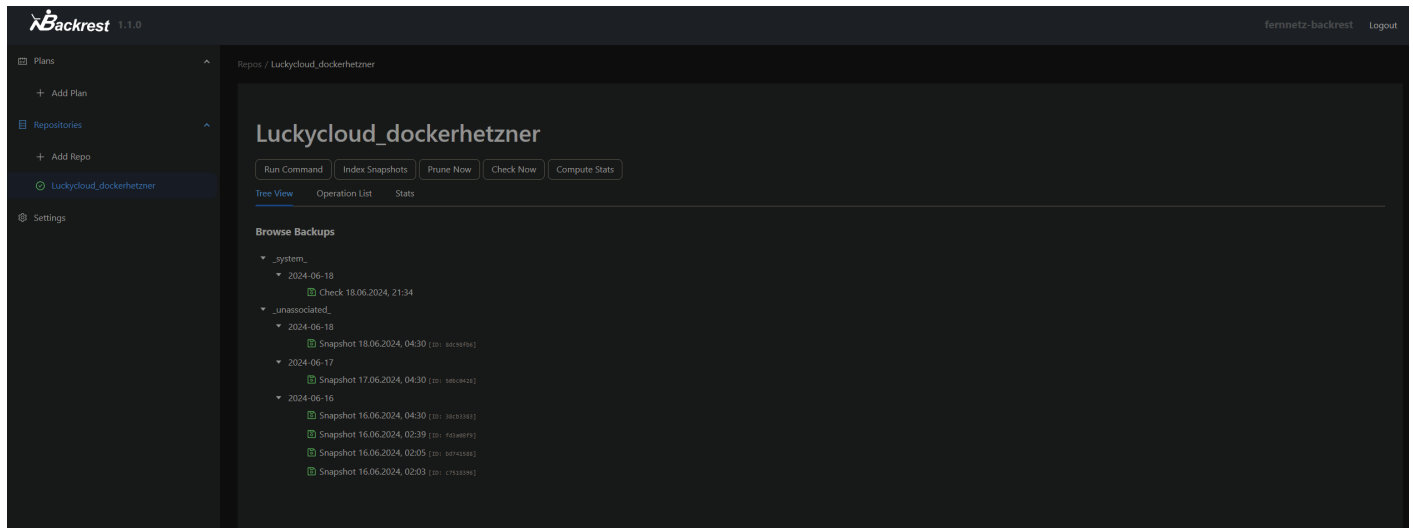


backrest (+ restic Befehle)

restic repo GUI mit Autobackups + Luckycloud S3-Storage oder Hetzner StorageBox



<https://github.com/garethgeorge/backrest>

```
version: "3.2"
```

```
services:
```

```
  backrest:
```

```
    image: garethgeorge/backrest
```

```
    container_name: backrest
```

```
    hostname: backrest
```

```
    volumes:
```

- data:/data
- config:/config
- cache:/cache
- /var/lib/docker/volumes:/docker-var-lib
- /data/compose:/docker-data-compose

```
  #- /MY-BACKUP-DATA:/userdata # [optional] mount local paths to backup here.
```

```
  #- /MY-REPOS:/repos # [optional] mount repos if using local storage, not necessary for
```

```
remotes e.g. B2, S3, etc.
```

```
  environment:
```

- BACKREST_DATA=/data # path for backrest data. restic binary and the database are

placed here.

- BACKREST_CONFIG=/config/config.json # path for the backrest config file.
- XDG_CACHE_HOME=/cache # path for the restic cache which greatly improves performance.
- TZ=Europe/Berlin # set the timezone for the container, used as the timezone for cron jobs.

restart: unless-stopped

ports:

- 9898:9898

volumes:

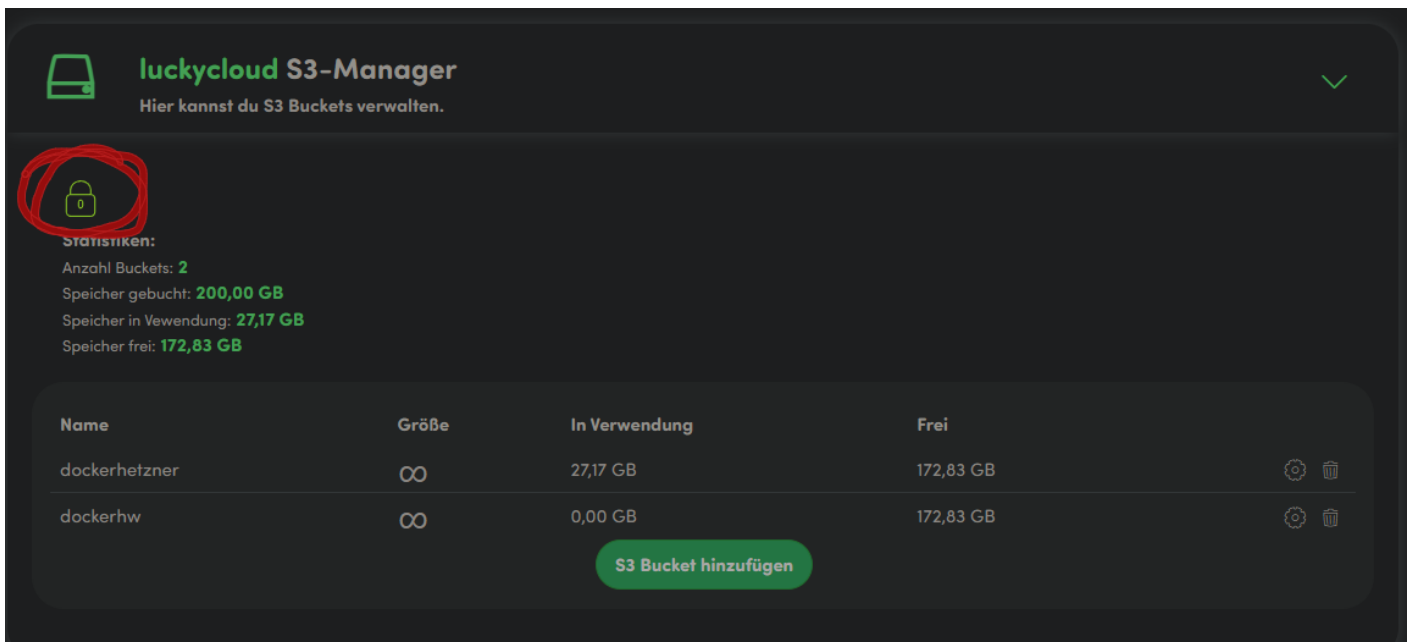
data:

config:

cache:

Konfiguration mit Luckycloud

In Luckycloud einen S3 Speicher anlegen:



luckycloud S3-Manager
Hier kannst du S3 Buckets verwalten.

Statistiken:
Anzahl Buckets: 2
Speicher gebucht: 200,00 GB
Speicher in Verwendung: 27,17 GB
Speicher frei: 172,83 GB

Name	Größe	In Verwendung	Frei
dockerherztner	∞	27,17 GB	172,83 GB
dockerhw	∞	0,00 GB	172,83 GB

S3 Bucket hinzufügen

auf das grüne Schloss drücken, um die Zugangsdaten AWS_ACCESS_KEY_ID und AWS_SECRET_ACCESS_KEY zu erhalten:

Lesen Sie ein Geheimnis

Auf dieser Seite können Sie ein Geheimnis über einen Link zur gemeinsamen Nutzung eines Geheimnisses weitergeben. Das Geheimnis wird im Link zur gemeinsamen Nutzung von Geheimnissen gespeichert und nicht auf dem Server.
Ein geheimer Link zum Teilen kann nur einmal verwendet werden.

Deine luckycloud-S3-Zugangsdaten

Server: s3-berlin1.luckycloud.de

access_key: GPd

secret_key: wQh

Passwort-geschützt:

Entschlüsseln!

die Zugangsdaten bei backrest als neues repo anlegen:

Backrest 1.1.0

Plans

- + Add Plan
- hetzner_to_luckycloud

Repositories

- + Add Repo
- Luckycloud_dockerhetzner

Settings

Edit Restic Repository

Repo Name: Luckycloud_dockerhetzner

Repository URL: s3https://s3-berlin1.luckycloud.de/dockerhetzner

Password: Unb

Env Vars: AWS_ACCESS_KEY_ID=GPd X3

AWS_SECRET_ACCESS_KEY=wQh

Prune Policy: Max Unused % After Prune 25

die repository url ist dann sowas wie

s3:https://<luckycloudserver>.luckycloud.de/<bucketname>

und ist zu finden im luckycloud portal:

Dein Konto

Zeitguthaben:	365 Tage
Cloud-Speicher:	0 GB
S3-Speicher:	200 GB
EURO/Monat:	5.60 €
Zahlart:	PayPal

Verwalten & Bezahlen

Deine Zugänge

luckycloud S3

Server:
https://s3-berlin1.luckycloud.de

Benutzername:
l

luckycloud S3

noch
172.83 GB
verfügbar

als Plan dann zb folgendes:

Update Plan

* Plan Name:

hetzner_to_luckycloud

* Repository:

Luckycloud_dockerhetzner

* Paths:

/backupdata

+ Add Path

Excludes:

+ Add Exclusion Glob

Excludes (Case Insensitive):

+ Add Case Insensitive Exclusion Glob

Backup Schedule:

Disabled

Max Frequency Hours

Max Frequency Days

Cron

Every

day

 at

4

 :

30

Backup Flags:

+ Set Flag

Retention Policy:

By Count

By Time Period

None

Yearly

2

Daily

7

Monthly

3

Hourly

24

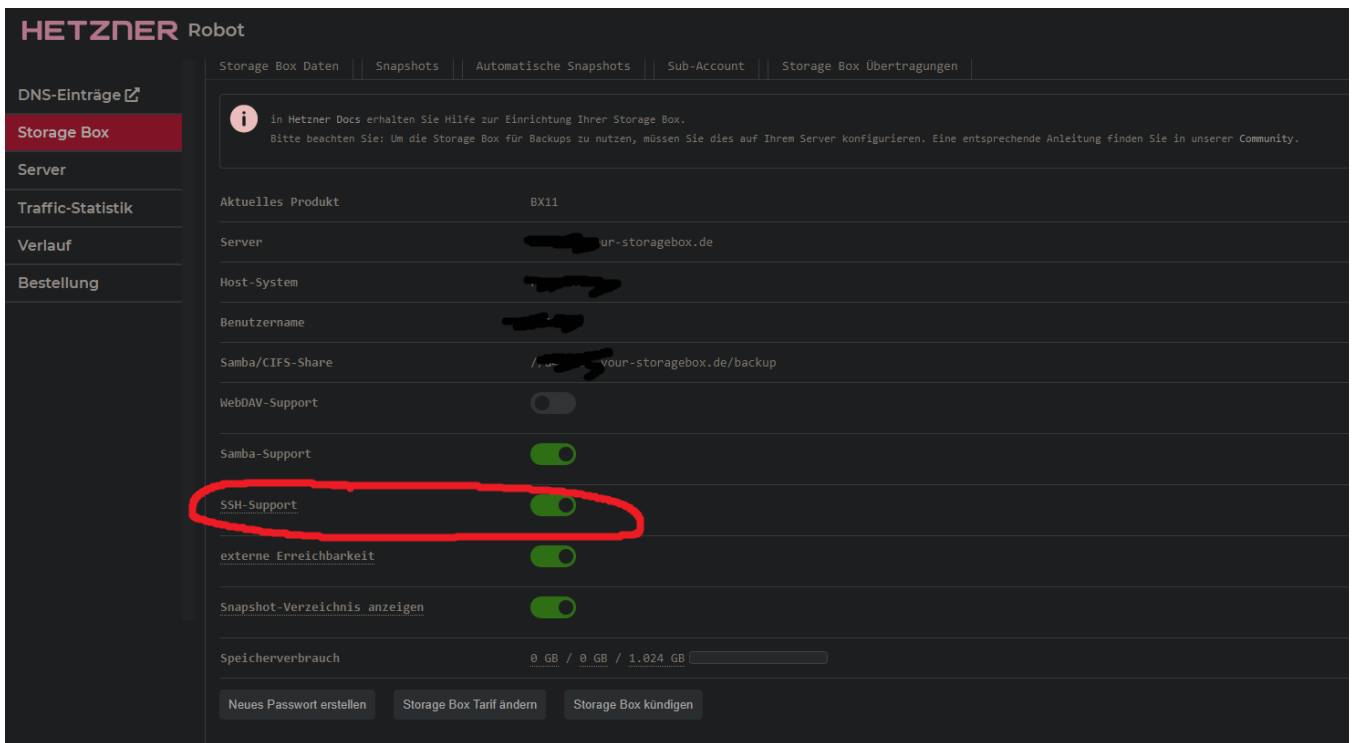
Weekly

5

oder /docker-var-lib als Path wenn man das Beispiel aus der docker-compose.yaml von oben nimmt

Konfiguration mit Hetzner Storage Box (SFTP)

1. Unter Hetzner Robot SSH aktivieren:



2. Auf dem Server, auf dem Docker mit backrest als Container läuft, einen SSH-Key erzeugen, wenn noch nicht geschehen:

```
ssh-keygen
```

3. SSH **Public** Key an Storage Box SSH Port 23 übertragen (kann auch id_ed25519.pub heißen oä, wenn es kein rsa ist):

```
cat ~/.ssh/id_rsa.pub | ssh -p23 uXXXXX@uXXXXX.your-storagebox.de install-ssh-key
```

4. testweise nochmal mit der Storage Box über ssh verbinden, sollte nun ohne Bestätigung klappen:

```
ssh -p 23 uXXXXX@uXXXXX.your-storagebox.de
```

oder mit Angabe zum Private-Key

```
ssh -i /root/.ssh/id_mein-admin_ed25519 -p 23 uXXXXX@uXXXXX.your-storagebox.de
```

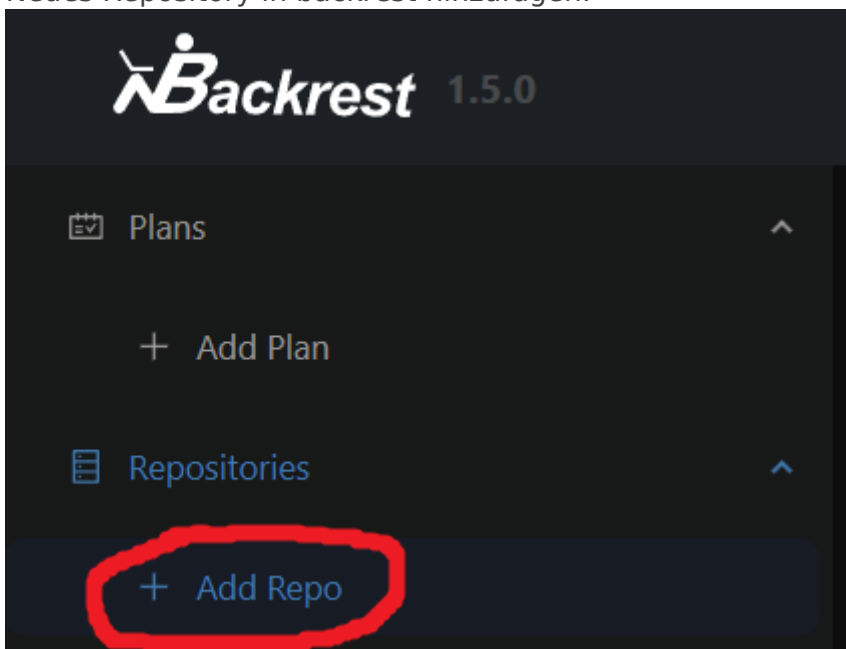
5. `/home/<user>/.ssh/known_hosts` und `/home/<user>/.ssh/id_ed25519` (SSH **Private** Key) vom Docker Server in das Config Volume vom Backrest Container kopieren, sodass backrest Zugriff auf den PrivateKey und die known_hosts Datei hat:

```
root@ubuntu-praxistool-arm64 /mnt/praxis-volume-01/docker-data/volumes/backrest_config/_data/ssh_keys
_data  ssh_keys  3  id_ed25519
config.json  822 B  id_rsa_restic-storagebox_pl
config.js~.2024-07-27-19-01-44  244 B  known_hosts
config.js~.2024-09-12-11-10-13  261 B
config.js~.2024-09-12-21-35-21  254 B
```

oder als Bsp auf hz-02:

```
root@hetzner-02-ubnt-amd /v/l/doc/volumes/backrest_config/_data/ssh_keys/id_pl-admin_ed25519
ssh_keys  id_pl-admin_ed25519  419 B
con~.json  known_hosts  898 B
```

6. Neues Repository in backrest hinzufügen:



Add Restic Repository

See [backrest getting started guide](#) for repository configuration instructions or check the [restic documentation](#) for more details about repositories.

* Repo Name: ✓

* Repository URI: ✓

Password: ✓ [Generate]

Env Vars:

Flags: ⊖

Repository URL bei SFTP hinzufügen:

```
sftp:user@host:/path/to/repo  
also zb  
sftp:u12345@u12345.your-storagebox.de:ResticBackup/DockerServer
```

Das Passwort für das Restic Backup kann man einfach mit klick auf Generate generieren lassen *udontsay.gif*

Flags (sftp.args) hinzufügen:

```
-o sftp.args="-i /config/ssh_keys/id_ed25519 -P 23 -o  
UserKnownHostsFile=/config/ssh_keys/known_hosts"
```

Falls es zu Problemen kommt, kann die SSH config angepasst werden, sodass man die SSH Flags weglassen kann:

Quelle: <https://forum.restic.net/t/usability-issue-with-sftp-and-sftp-command/1170/2>

OBACHT

Dieser "Trick" ist vollkommen sinnlos, weil man einen ssh-key und eine ssh-config in den flüchtigen ordner /root/.ssh im backrest-container legt. Nach jedem Neustart des Containers sind die Dateien also weg. Besser bei oben bleiben, wo man Flags hinzufügt und den SSH-Key sowie die known_hosts in das backrest_config Docker-Volume legt

```
docker exec -it backrest /bin/bash
```

```
nano /root/.ssh/config
```

.ssh/config:

```
Host storagebox  
  Hostname u12345.your-storagebox.de  
  Port 23  
  User u12345  
  IdentityFile /root/.ssh/id_admin_ed25519  
  ODER  
  IdentityFile /config/ssh_keys/id_admin_ed25519  
  JE NACHDEM WO DER SSH-KEY LIEGT!
```

wenn dannn noch WARNING UNPROTECTED PRIVATE KEY FILE Meldung kommt, einfach

```
chown 600 /config/ssh_keys/id_admin_ed25519
```

danach sollte

```
ssh storagebox
```

funktionieren und damit auch bei Repository URI:

```
sftp:storagebox:MEIN-PROJEKT/restic-backup
```

✓ Connected successfully to sftp:storagebox:MEIN-PROJEKT/restic-backup. No existing repo found at this location, a new one will be initialized

Getting Started

Add Restic Repository

See [backrest getting started guide](#) for repository configuration instructions or check the [restic documentation](#) for more details about repositories.

* Repo Name: ✓

* Repository URI: ✓

Password: ✓ [Generate]

Env Vars:

Flags:

Prune Policy: %

Clock for schedule:

Every on and at

restic unlock

<https://github.com/restic/restic/issues/1450>

Falls es zu folgender Fehlermeldung bei einem Backup kommt, kann man das Repository mit `restic unlock` wieder entsperren:

```
unable to create lock in backend: repository is already locked by PID 3604 on HOSTNAME by
HOSTNAME\User (UID 0, GID 0)
lock was created at 2025-08-17 22:58:46 (725h49m33.96593014s ago)
storage ID 42...bce
the `unlock` command can be used to remove stale locks
```

Lösung:

In die Shell des Backrest Docker-Container gehen und folgende Befehle eingeben:

```
restic -r /path/to/repository/external/drive unlock
# release previous lock
restic -r /path/to/repository/external/drive check
# Return some "not referenced in any index"
restic -r /path/to/repository/external/drive rebuild-index
# rebuild index
restic -r /path/to/repository/external/drive check
# return a few "not referenced in any index"
restic -r /path/to/repository/external/drive prune
# remove those few incomplete packs
restic -r /path/to/repository/external/drive check
# no errors were found
```

Wobei aus dem Hetzner-Storagebox SFTP Beispiel oben als Repository mit angegebenem SSH-Key folgende Zeile resultiert:

```
restic -r sftp:u12345@u12345.your-storagebox.de:ResticBackup/DockerServer -o sftp.args="-i
/config/ssh_keys/i
d_ed25519 -p 23 -o UserKnownHostsFile=/config/ssh_keys/known_hosts" check
```

Danach das repository-Passwort eingeben:

```
/ # restic -r sftp:u4 @u .your-storagebox.de: /ResticBackup -o sftp.args="-i /config/ssh_keys/i
d_ed25519 -p 23 -o UserKnownHostsFile=/config/ssh_keys/known_hosts" check
using temporary cache in /tmp/restic-check-cache-2037751025
create exclusive lock for repository
enter password for repository:
repository 9699ec26 opened (version 2, compression level auto)
created new cache in /tmp/restic-check-cache-2037751025
repo already locked, waiting up to 0s for the lock
unable to create lock in backend: repository is already locked by PID 21696 on [REDACTED] by [REDACTED] (UID 0, GID
0)
lock was created at 2025-08-17 01:25:22 (752h30m57.13612605s ago)
storage ID 41174d09
the `unlock` command can be used to remove stale locks
```

Backup wiederherstellen (Restic CLI)

In die Shell des Containers backrest gehen

```
# mit hz-03 verbinden
```

```
ssh hz-03
```

```
# mit lazydocker in den container backrest gehen:
```

```
lazydocker
```

```
# mit J/K hoch und runter zum Container "backrest" navigieren und Shift+E drücken für exec in shell:
```

```
root@hetzner-03-ubnt-amd: ~ X + v

Project
root

Containers
running backrest
running beszel-agen
running botpress
running botpress_db
running (healthy) cadvisor

Images
bitnami/mongodb
botpress/server
garethgeorge/backrest
garethgeorge/backrest
gcr.io/cadvisor/cadvisor

Volumes
local 72bfa4ab954aac99822c6ad
local 85cc8c574675ff5f13387d1
local 8f05c329912f40a1131af64
local aed7524c2353d76f9434130

Logs - Stats - Env - Config - Top
task queue reset.
2026-03-23T12:54:09.165+0100 INFO
set {"timezone": "Europe/Berlin"}
2026-03-23T12:54:09.165+0100 INFO

Menu
d remove
e hide/show stopped containers
p pause
s stop
r restart
a attach
m view logs
E exec shell
c run predefined custom command
b view bulk commands
w open in browser (first port is http)
enter focus main panel
[ previous tab
] next tab
/ filter list
```

```
lazydocker > exec shell in Container "backrest"
```

config.json des repositories anzeigen

```
# config des restic repositories anzeigen:  
cat config/config.json
```

```
/ # cat config/config.json  
{  
  "modno": 8,  
  "version": 4,  
  "instance": "restic-hz-03",  
  "repos": [  
    {  
      "id": "HetznerStorageBox",  
      "uri": "sftp:u123456@u123456.your-storagebox.de:/ResticBackup",  
      "guid": "72c3673e-1234-5678-9012-345678901234",  
      "password": "t1Lp0iF.....KnTqEQag",  
      "flags": [  
        "-o sftp.args=\"-i /config/ssh_keys/id_pl-admin_ed25519 -p 23 -o UserKnownHostsFile=/config/ssh_keys/known_hosts\"",  
      ],  
      "prunePolicy": {  
        "schedule": {  
          "cron": "0 0 1 * *",  
          "clock": "CLOCK_LAST_RUN_TIME"  
        },  
        "maxUnusedPercent": 10  
      },  
      "checkPolicy": {  
        "schedule": {  
          "cron": "0 0 1 * *",  
          "clock": "CLOCK_LAST_RUN_TIME"  
        }  
      }  
    }  
  ]  
}
```

wichtig sind uri, pw und flags für sftp des repos

restic cmd vorbereiten

```
# dann folgende ENVs im Container festlegen (aus config.json kopiert) mit  
export RESTIC_REPOSITORY="sftp:u123456@.....de:MEINEPRAXIS/ResticBackup"  
export RESTIC_PASSWORD="t1Lp0iF.....KnTqEQag"  
  
# und zuletzt einen alias für restic setzen mit  
alias restic='restic -o sftp.args="-i /config/ssh_keys/id_pl-admin_ed25519 -p 23 -o  
UserKnownHostsFile=/config/ssh_keys/known_hosts"'
```

restic Befehle anwenden und Dateien / Pfade suchen

```
# snapshots anzeigen mit repo-name und plan-name
restic snapshots

# oder kompakt
restic snapshots --compact

# oder nur bestimmter host
restic snapshots --host meinserver


# nach PDF-Dateien in bestimmtem Ordner suchen:
restic find --tag "plan:daily-docker-nextcloud" "*/PatientInnen MAX MUSTERMANN/MUSTERFRAU,
MAXI Gruppe ACT W1263456/Dokumentation Therapie/*.pdf"

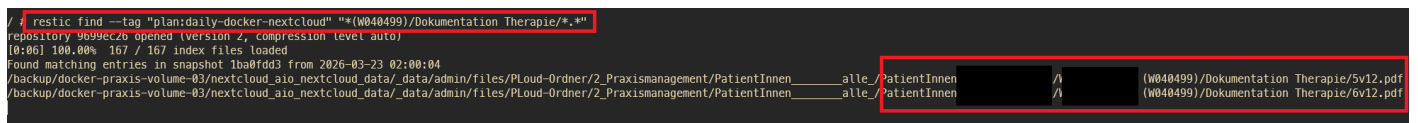
# alle dateien in bestimmtem Ordner:
restic find --tag "plan:daily-docker-nextcloud" "/backup/docker-praxis-volume-
03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-
Ordner/2_Praxismgmt/PatientInnen__alle_/PatientInnen MAX MUSTERMANN/MUSTERFRAU, MAXI Gruppe
ACT W1263456/Dokumentation Therapie/*.*"
```

Je genauer der Pfad, desto schneller die Suche

Beispiel für langsame Suche

```
restic find --tag "plan:daily-docker-nextcloud" "**(W040499)/Dokumentation Therapie/*.pdf"
```

= dauert ca 1min pro Snapshot!



```
/ # restic find --tag "plan:daily-docker-nextcloud" "**(W040499)/Dokumentation Therapie/*.pdf"
repository 9b99ec2b opened (version 2, compression level auto)
[0:06] 100.00% 167 / 167 index files loaded
Found matching entries in snapshot 1ba0fdd3 from 2026-03-23 02:00:04
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen__alle_/PatientInnen (W040499)/Dokumentation Therapie/5v12.pdf
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen__alle_/PatientInnen (W040499)/Dokumentation Therapie/6v12.pdf
```

Beispiel für schnelle Suche

```
restic find --tag "plan:daily-docker-nextcloud" "/backup/docker-praxis-volume-
03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-
Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen Behandler-Vorname\ Behandler-
Nachname/Pat-Nachname,\ Pat-Vorname\ \ (W
040499)/Dokumentation\ Therapie/*.*"
```

= innerhalb von 20sek alle Snapshots durchsucht

```
# restic find --tag "plan-daily-docker-nextcloud" "/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
(W040499)/Dokumentation Therapie/*.*"
repository 9099ec20 opened (version 2, compression level auto)
[0:05] 100.00% 167 / 167 index files loaded
Found matching entries in snapshot 1ba8fdd3 from 2026-03-23 02:00:04
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
Found matching entries in snapshot b676c5c4 from 2026-03-22 02:00:06
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
Found matching entries in snapshot d8c82731 from 2026-03-21 02:00:02
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
Found matching entries in snapshot d1452e2a from 2026-03-20 02:00:04
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
Found matching entries in snapshot 6627dfb2 from 2026-03-19 02:00:03
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
Found matching entries in snapshot 47661e75 from 2026-03-18 02:00:02
/backup/docker-praxis-volume-03/nextcloud_aio_nextcloud_data/_data/admin/files/PLoud-Ordner/2_Praxismanagement/PatientInnen____alle_/PatientInnen
```

Dateien wiederherstellen

Wiederherstellen entweder über GUI direkt zum richtigen Snapshot navigieren und herunterladen (siehe <https://wiki.praxisluebberding.de/doc/backup-wiederherstellen-backrest-gui-w04w1wkGAJ>) oder im CLI:

```
restic restore b676c5c4 --target "/backrest-restored/mein-wiederhergestellter-ordner" --include
"/backup/docker-praxis-volume-
03/nextcloud_aio_nextcloud_data/_data/admin/files/.../Dokumentation Therapie"
# danach im ordner /mnt/backrest-restored nachsehen und auf Dateien zugreifen (evtl über
mountain duck > sftp hz-03)
```

Achtung, immer wenn über die Web-UI (GUI) wiederhergestellt wird, landen die restores auf hz-01 im Ordner /mnt/backrest-restored, weil dort der backrest container für <https://backrest.praxisluebberding.de/> läuft.

Wenn man jedoch wie hier über die CLI wiederherstellt, gibt man einen lokalen Ordner an, zb auf hz-03 /mnt/backrest-restored

Backup wiederherstellen (Backrest GUI)

- 1) <https://backrest.praxisluebberding.de> öffnen
- 2) Backrest User & Passwort eingeben (in Bitwarden hinterlegt: <https://bw.praxisluebberding.de>)
- 3) Den Backup-Plan 'daily_docker_backup' auswählen

Backrest 1.6.2

Plans

+ Add Plan

daily_docker_backup

Repositories

+ Add Repo

HetznerStorageBox_PL

Settings

Summary

Repos

HetznerStorageBox_PL

Backups (30d):	Bytes Scanned (30d)	Bytes Added (30d)	
11	1.27 TiB	47.12 GiB	
ok	warning		
Next Scheduled Backup	Bytes Scanned Avg	Bytes Added Avg	
12.12.2024, 04:45	108.45 GiB	3.93 GiB	

4) letztes erfolgreichen Snapshot auswählen (grünes Icon)

5) Rechts den Snapshot Browser ausklappen

6) Den gewünschten Ordner oder Datei suchen

Backup NowRun CommandUnlock RepoClear Error History

Tree ViewList View

2024-12-12

Backup 12.12.2024, 04:45 [scheduled, waiting]

2024-12-11

Backup 11.12.2024, 04:45 [156.05 GiB in 12m10s, ID: 5f8e505f]

2024-12-10

Backup 10.12.2024, 04:45 [155.99 GiB in 14m50s, ID: c6d43b10]

2024-12-09

Backup 09.12.2024, 04:45 [99.1 GiB in 18s, ID: c56adda8]

2024-12-08

Backup 08.12.2024, 04:45 [99.18 GiB in 19s, ID: 6b140442]

2024-12-07

2024-12-06

2024-12-05

2024-12-01

2024-11-30

2024-11-24

2024-11-17

2024-11-10

2024-11-03

2024-10-31

2024-10-27

2024-09-30

2024-09-12

11.12.2024, 04:45Forget (Destructive)

11.12.2024, 04:57 - Forget in 2s [View Logs]

> Removed 1 Snapshots

11.12.2024, 04:45 - Snapshot in 12m9s

Details

Snapshot ID: 5f8e505f

User and Hostroot@backrest

Tagsplan:daily_docker_backup, created-by:pl-hetzner-01

Files Added121339

Files Changed759

Files Unmodified50401

Bytes Added356.93 MiB

Bytes Processed156.05 GiB

Files Processed172499

Snapshot Browser

/

docker-data-compose

docker-hetzner-volume

home-dir

onedrive-pl

Desktop

Erste Schritte mit OneDrive.pdf (1.12 MiB)

Praxis NEU

00 Notfall & Krise

01 PatientInnenmanagement

7) Restore to Path

09 IT

1 How to's

How to Drucker installieren FBS.mp4 (24.66 MiB)

Jitzi-Demo.mp4 (95.99 MiB) ⬇

Nuki-Türschlösser Akku we

Info

Restore to path

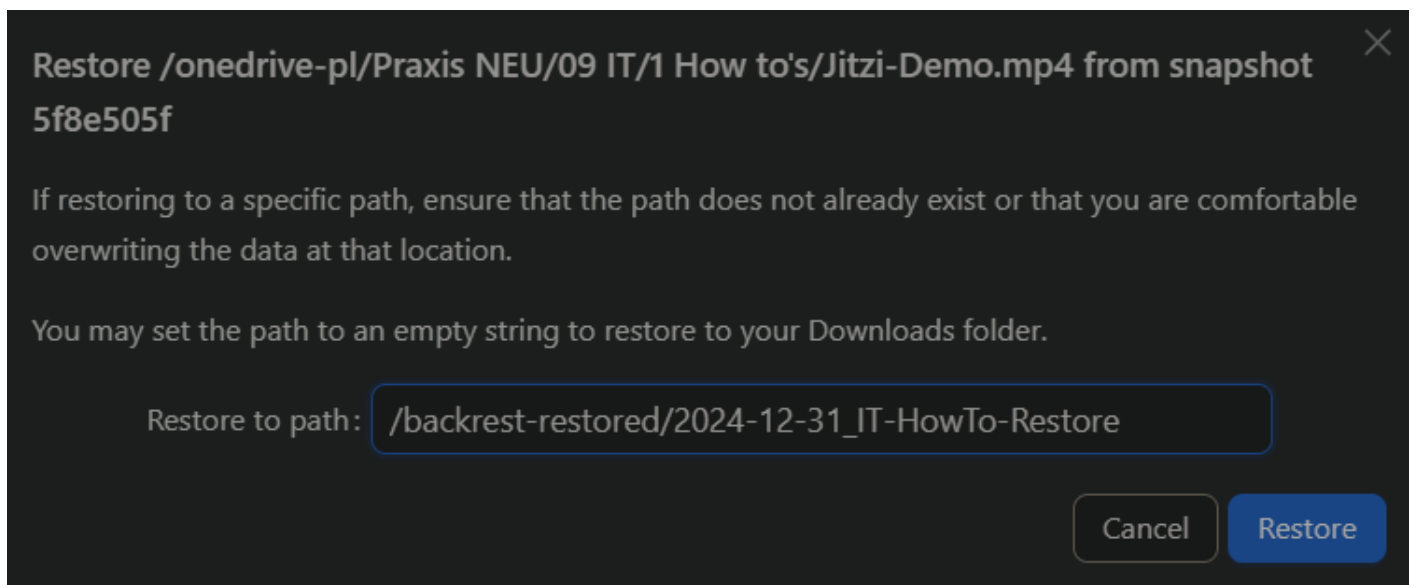
2 Drucker

4 Software

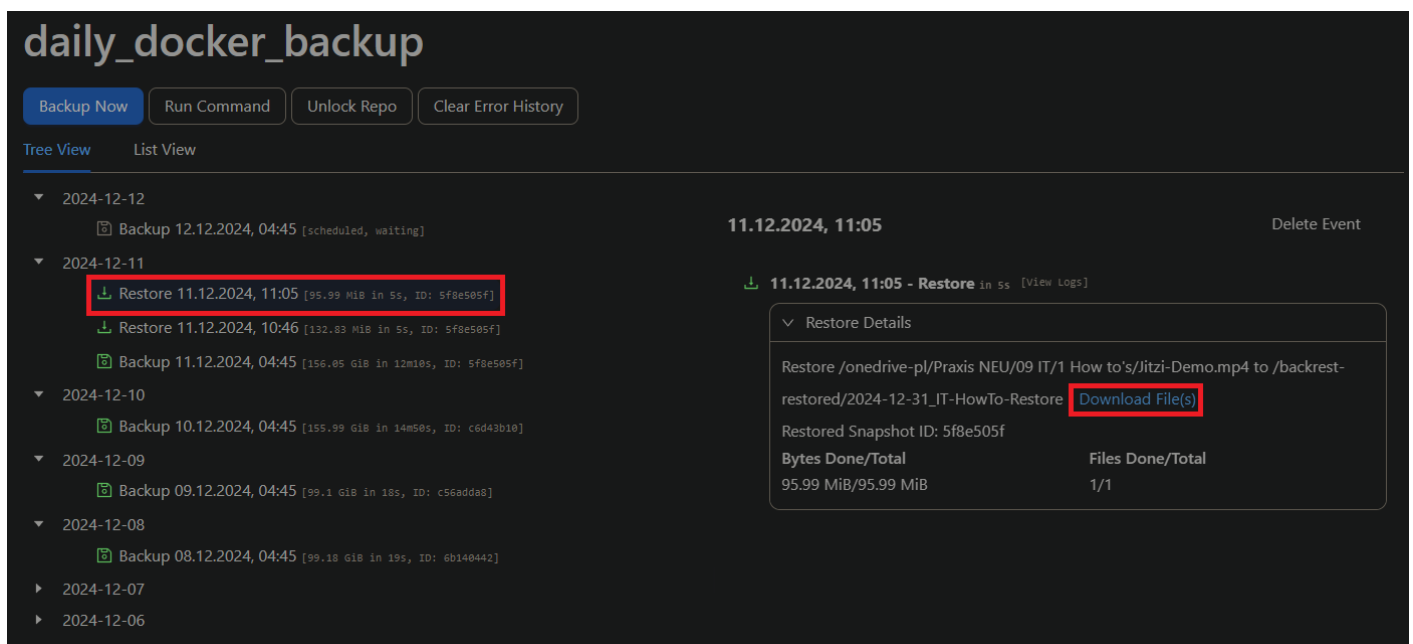
8) restore-pfad angeben:

/backrest-restored/NAME

zb /backrest-restored/2024-12-31_IT-HowTo-Restore



9) Zurück zur Übersicht, den Restore auswählen und auch Download Files drücken:



10) .tar.gz Datei herunterladen und entpacken

Revision #21

Created 2024-06-18 21:15:19 UTC

Updated 2026-03-25 11:08:42 UTC