

# NASbeery

Das erste NAS welches in ein eine Bierdose passt!

- [FAQ](#)
- [Hardware](#)
- [Installation](#)
- [Installation Script](#)
- [Software used](#)
- [Summary](#)

# FAQ

## Is it possible to use this system on another ubuntu hardware?

-Yes, but maybe FreeNas is a better option then.

## How to delete snapshots:

**Storage Pools**

Create Storage Pool Import Storage Pool Refresh

Name	Health	Size	Allocated	Free	Fragmentation	Usage
tank	ONLINE	111 GiB	14.18 MiB	110.99 GiB	0 %	

File Systems Snapshots Status

6.7.2020, 1:10:14 PM 174 Create Snapshot

Name	Created	Used	Referenced	Clones
tank				
tank/share				
tank/share@zfs-auto-snap_daily-2020-06-18-0625	18.6.2020, 8:25:02 AM	72 KiB	1.49 MiB	
tank/share@zfs-auto-snap_frequent-2020-07-01-1600	17.2020, 6:00:02 PM	72 KiB	1.49 MiB	tank/share/klon2
tank/share@zfs-auto-snap_daily-2020-07-03-0625	3.7.2020, 8:25:02 AM	64 KiB	1.49 MiB	
tank/share@zfs-auto-snap_daily-2020-07-04-0625	4.7.2020, 8:25:02 AM	56 KiB	1.49 MiB	
tank/share@zfs-auto-snap_hourly-2020-07-04-0817	4.7.2020, 10:17:01 AM	0 B	1.49 MiB	

Clone Snapshot  
Rename Snapshot  
Roll Back Snapshot  
Destroy Snapshot

## How to delete all snapshots:

(Terminal)

- #zfs list -t snapshot | awk '{printf „zfs destroy %s\n“, \$1}' > snapshot\_delete.sh
- #sh snapshot\_delete.sh

## Maintenance:

Checking system health:

**Storage Pools**

Create Storage Pool Import Storage Pool Refresh

Name	Health	Size	Allocated	Free	Fragmentation	Usage
tank	ONLINE	111 GiB	14.18 MiB	110.99 GiB	0 %	

File Systems Snapshots Status

6.7.2020, 1:10:14 PM 174 Create Snapshot Refresh

Name	Created	Used	Referenced	Clones
tank				
tank/share				
tank/share@zfs-auto-snap_daily-2020-06-18-0625	18.6.2020, 8:25:02 AM	72 KiB	1.49 MiB	
tank/share@zfs-auto-snap_frequent-2020-07-01-1600	17.2020, 6:00:02 PM	72 KiB	1.49 MiB	tank/share/klon2
tank/share@zfs-auto-snap_daily-2020-07-03-0625	3.7.2020, 8:25:02 AM	64 KiB	1.49 MiB	
tank/share@zfs-auto-snap_daily-2020-07-04-0625	4.7.2020, 8:25:02 AM	56 KiB	1.49 MiB	
tank/share@zfs-auto-snap_hourly-2020-07-04-0817	4.7.2020, 10:17:01 AM	0 B	1.49 MiB	

Clone Snapshot  
Rename Snapshot  
Roll Back Snapshot  
Destroy Snapshot

(Terminal):

- zpool status  
Everything should be green and ONLINE

(By funky Hardware)

The NasBeery installer supports a 5V device on pin 8 (BCM 14) and pin6 (ground).

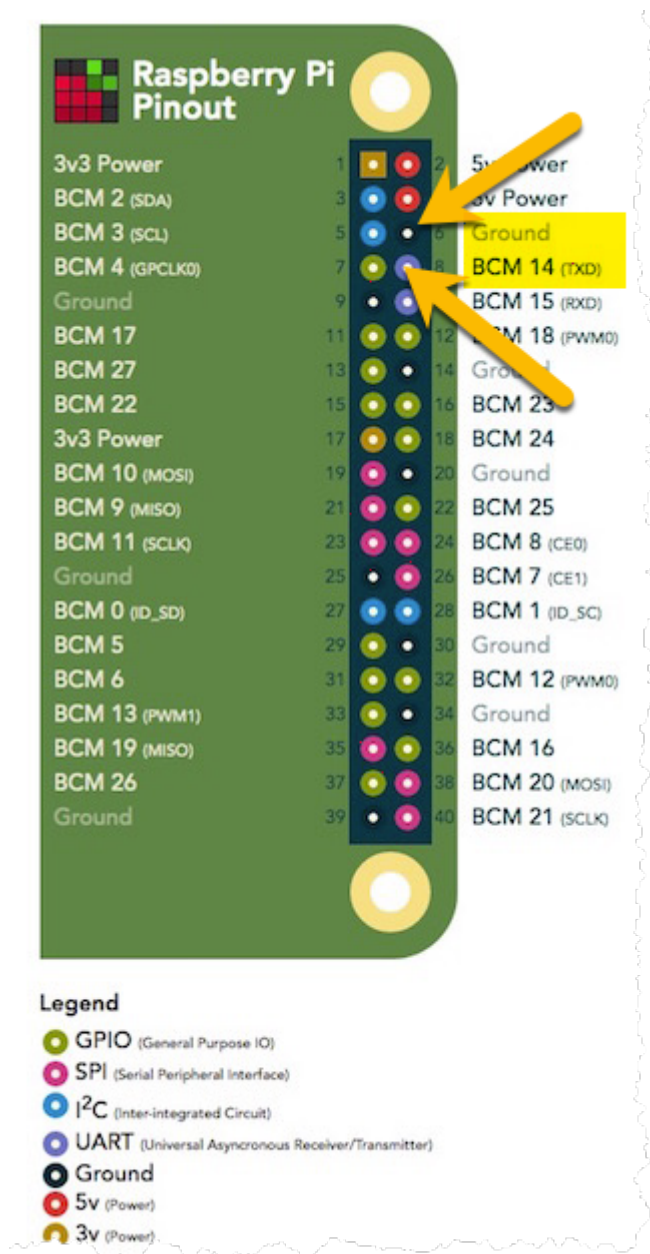
What we tested so far:

### Buzzer LED and Wires

Just connect the actor to the pins beyond (LED short link to ground).

While booting up the system you will hear a loud sound or see the LED light.

After less than a minute the actor should stop buzzing or the LED turned dark. Then your „only“ tank is in healthy condition.



## Checking the free space:

<div>nasbeery</div> <div>Search</div> <div>Überblick</div> <div>Protokolle</div> <div>Speicher</div> <div>ZFS</div> <div>Netzwerk</div>	Storage Pools							Create Storage Pool	Import Storage
	Name	Health	Size	Allocated	Free	Fragmentation	Usage		
	> tank	ONLINE	111 GiB	14.25 MiB	110.99 GiB	0 %			
	> temp	ONLINE	1.81 TiB	52.16 GiB	1.76 TiB	0 %			

Terminal:  
#zpool list

```
ubuntu@nasbeery: ~
```

```
ubuntu@nasbeery:~$ zpool list
```

NAME	SIZE	ALLOC	FREE	CKPOINT	EXPANDSZ	FRAG	CAP	DEDUP	HEALTH	ALTROOT
tank	111G	14.4M	111G	-	-	0%	0%	1.00x	ONLINE	-
temp	1.81T	52.2G	1.76T	-	-	0%	2%	1.00x	ONLINE	-

```
ubuntu@nasbeery:~$
```

There should always be more than 20% free space (cap below 80%) or your system will slow down, at 90% you almost can't work anymore until you delete files and destroy snapshots.

### Where are the snapshots stored? / Can I put snapshots to another drive?

You are starting without any snapshots.

Anytime you create a snapshot a vectorchain is creating another point.

So your current live state is the chain of many vectors.

The vectors can start anywhere and you always can remove vectorpoints.

The gap then will be closed by the previous vector.

*Example:*

Mo>Tu>We>Th>Fr

now we zfs destroy the we snapshot

Mo>Tu>Th>Fr

now we zfs destroy the monday snapshot

Tu>We>Th>Fr

and the a saturday snapshot will be created

Tu>We>Th>Fr>Sa

Your current state is always at the end and you can always access any snapshot state without any effort!

# Hardware

Please use our affiliate links on Amazon to support us and get tested items.

## **Raspberry Pi 4 with 2GB, 4GB or 8 GB**

(We recommend 1GB RAM for about 1TB of stored data)

- [Raspberry Pi 2GB](#)
- [Raspberry Pi 4GB](#)
- [Raspberry Pi 8GB](#)

## **Micro SD Card with 32GB and above for Ubuntu OS**

(Do not use current Toshiba, system is slowing down a lot)

- [SD-Card](#)

## **USB-C Power Adaptor**

- [Power Adaptor](#)

## **2 x SSD not of the same kind (less risky if one is failing and easier to replace by name)**

- [SSD Samsung 500 GB](#)
- [SSD Samsung 1 TB](#)
- [SSD Kingston 500 GB](#)
- [SSD Kingston 1 TB](#)

## **Dual USB to SATA Dock**

- [Sata Dock](#)

or:

## **USB to SATA Adaptor**

- [USB / SATA Adaptor](#)

## **Ethernet Cable Cat.5 and above**

- [Ethernet Cable](#)

*For setup and maintenance:*

## **Micro HDMI Cable**

- [Micro HDMI Cable](#)

## **USB Keyboard**

## **Monitor**

- [Monitor](#)

# Installation

Start with this recommended setup, so you can update later to a more recent version

- Copy **Ubuntu 22.04 LTS 64 bit Server** to SD-card with Balena Etcher or Raspberry Pi Imager and put SD-card into your Raspberry Pi4 system.
- Wire your Raspberry Pi4 system via ethernet to your router, Micro HDMI cable to monitor, keyboard to black USB connector, SSDs with adaptor to blue USB connectors and use your power adaptor to turn on your Raspberry Pi system to first boot.
- Wait until login prompt appears and a little longer for a lot more output on screen.
- Log in with username and password „ubuntu“.
- Repeat old „ubuntu“ password and enter new password twice (be careful, best use only characters and numbers at this point, US keyboard!).
- Type in your shell:

```
wget nasras.de  
bash index.html
```

- Press Enter on ZFS licence screen and wait about 20 minutes until reboot.
- **(new) Decide to preserve your data and set password for shell and Samba.**
- Find your NasBeery with share in your network environment.  
Username: ubuntu, password: your chosen password  
`https://nasbeery:9090`  
Username: ubuntu, password is chosen by you during setup.
- Recover files with explorer.
- Rollback data if needed with GUI.



# Installation Script

Latest script can be downloaded [HERE](#)

What it basically does:

- Get time and date, because Ubuntu sometimes fails on boot
- Stop first unattended upgrades
- Install Samba, ZFS, Cockpit and tools
- Load ZFS module before reboot
- Give Ubuntu user a password „NasBeery2020“
- Get ZFS Manager for Cockpit from Github
- Install zfs-auto-snapshot and change retention to three months and 48 hours, instead of 12 and 24!
- Change hostname to NasBeery
- Create a raid with compression and autoexpand that also creates the first filesystem on /tank
- Create a filesystem and share at /tank/share
- Give permission to Ubuntu user to /tank/share
- Add ZFS snapshots to use with ShadowCopy frontend from Explorer to Samba
- Reboot the whole thing

# Software used

Software:

## **Ubuntu 22.04 LTS**

Download 64-Bit version [HERE](#)

Modern 64-Bit server operating system, certified for Raspberry Pi 4



## **Balena Etcher**

Download for your OS [HERE](#)

Installs Ubuntu image to your SD card

## **Open ZFS** (included in Ubuntu 20.04 LTS)

Delivers redundancy and snapshots to your drive

## **Cockpit Manager w/ ZFS Manager (obsolete, no more Updates)** (from Github)

Modern Gui with rich ZFS features

## **Samba** (included in Ubuntu 22.04 LTS)

Your Windows (and others) compatible file server

## **Web Server** (included in Ubuntu 22.04 LTS)

Navigate to <https://nasbeery:9090> with user ubuntu to configure and maintain your NasBeery System

# Summary

NasBeery is a light weight NAS (Network Attached Storage) system, based on Raspberry Pi 4 (or newer), easy to install and to fit instructions to a tap, or even theoretically put the system into a soda can.

Feature Set Juli 2020 (Any values can be manually adjusted, just first Setup)

- Very easy installation with just two commands (one is downloading, the other for execution)
- Three month of snapshots 4x per hour, 48x hourly, 7x daily, 4x weekly, 3x monthly
- Mirror of two disks
- Very attractive GUI for setup shares and ZFS commands
- Restore files directly from „previous versions feature“ in Windows Explorer
- Clone a specific snapshot from the last three months to a folder to instantly review data
- Instant rollback to any of the about 66 snapshots without delay at any given time
- Every software is open source and included in Ubuntu 20.04 LTS and free of charge
- Production use is already a backup, it's mirrored and versioned
- Compatible to any OS on the market like Windows, Mac OS, iOS, Android, Linux, BSD, etc.