

Backupserver petit-backup

Hardware

- Asrock J4125-ITX
- 256GB NVME mit Adapter in PCIe Slot
- 4x3TB HDD (SATA)
- TODO: Gehäuse (maybe <https://www.ebay.de/itm/223102561977>)

Grundsetup

Ubuntu 22.04

User krumel angelegt

```
#Passwordless Sudo
sudo visudo
#Change line: %sudo    ALL=(ALL:ALL) NOPASSWD:ALL

#Hostname
echo "petit-backup" | sudo tee /etc/hostname

#Add SSH Key
mkdir ~/.ssh
echo "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCSLI18nShd47L6o4dL2sIbhJA1WdXXc7BBSqhsLTBMVziY60BazW2jxxU0eN+Wi3
RYEu0Ud3xt6f56m6NgB96MxvRbfhD06FCetrEzEX/k7yWRVlvyM0SX0RjTr2UWPq0pXmLvbp0vTX4m4+rhpXlXJ1FB/jiZ
GNYvQEXot8PFTkMBdP0rHsdXiHhJvJy8Y/jDoErrCrK+Yger9ziCeskr3t/KET1nD6e/g4lQwVr7YftMw9s/0RiSVU4VQn
UHjMiyXMpg8SD54YkmaQ8TJ14dQ3LVvMjXNGhg3fmmmxQMwot64oLe5HvNJigmKDFYxUzQuX8Ba2zAcnvHkLp/RpVB
krume@YatagarasuDrive" > ~/.ssh/authorized_keys
chmod -R u=Xrw,g=r,o= ~/.ssh

# Updates und purge Garbage
sudo apt purge snap cloud-init
sudo rm -rf /etc/cloud
sudo apt update && sudo apt upgrade
sudo apt autoremove
```

```
# ZFS
sudo apt install zfsutils-linux

#Fix stupid default SSH settings
sudo sed -i 's/PermitRootLogin yes/PermitRootLogin no/' /etc/ssh/sshd_config
sudo sed -i 's/#PasswordAuthentication yes/PasswordAuthentication no/' /etc/ssh/sshd_config
```

/etc/netplan/00-installer-config.yaml

```
# This is the network config written by 'subiquity'
network:
  ethernets:
    eno1:
      addresses:
        - 192.168.2.57/24
      gateway4: 192.168.2.1
      nameservers:
        addresses:
          - 192.168.2.20
        search:
          - grafschafft.loc
  version: 2
```

BackupPC

```
sudo apt install backuppc rrdtool

#symbolischer Link von /var/lib/backuppc (default Backup-Location) nach /mnt/data/backup
sudo mkdir /mnt/data/backup
sudo chown backuppc:backuppc /mnt/data/backup
sudo rm -r /var/lib/backuppc
sudo ln -s /mnt/data/backup /var/lib/backuppc

#remove Ubuntu's stupid symlink
sudo rm /etc/backuppc/pc
sudo mkdir /etc/backuppc/pc
sudo chown -R backuppc:www-data /etc/backuppc/pc
sudo mv /etc/backuppc/localhost.pl /etc/backuppc/pc
```

```
#import zpool
sudo zpool import -a -f

#fix old perms
sudo chown -R backuppc:backuppc /mnt/data/backup
```

/etc/backuppc/apache.conf

```
Alias /backuppc /usr/share/backuppc/cgi-bin/

<Directory /usr/share/backuppc/cgi-bin/>
    AllowOverride None

    # Uncomment the line below to ensure that nobody can sniff important
    # info from network traffic during editing of the BackupPC config or
    # when browsing/restoring backups.
    # Requires that you have your webserver set up for SSL (https) access.
    #SSLRequireSSL

    Options ExecCGI FollowSymlinks
    AddHandler cgi-script .cgi
    DirectoryIndex index.cgi

    AuthUserFile /etc/backuppc/htpasswd
    AuthType basic
    AuthName "BackupPC admin"

    <RequireAll>
        # Comment out this line once you have setup HTTPS and uncommented
        SSLRequireSSL

        #Require local

        # This line ensures that only authenticated users may access your backups
        Require valid-user
    </RequireAll>
</Directory>
```

/etc/backuppc/htpasswd

```
backuppc:$apr1$66V0rgF9$Lpoi3yC6ka9cZHTtWH28C1
```

/etc/backuppc/config.pl

Kopiert von altem System
(siehe unten)

Ursprüngliches Setup auf Odroid C4

Armbian von [hier](#).

Grundsetup

Armbian Setup

```
Create root password: *****
Repeat root password: *****

Choose default system command shell:

1) bash
2) zsh

1

Please provide a username (eg. your first name): admin

Create user (admin) password: *****
Repeat user (admin) password: *****

Please provide your real name: Admin

Set user language based on your location? [Y/n] n

At your location, more locales are possible:
[..]
Please enter your choice: 154) Skip generating locales
```

System Setup

```
#Passwordless Sudo
sudo visudo
#Edit line: %sudo    ALL=(ALL:ALL) NOPASSWD:ALL
```

```

#Image auf neuen Stand bringen
sudo apt update
sudo apt upgrade
sudo apt dist-upgrade

#Hostname
echo "petit-backup" | sudo tee /etc/hostname

#Add SSH Key
mkdir ~/.ssh
echo "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCsLI18nShd47L6o4dL2sIbhJAlWdXXc7BBSqhsLTBMVziY60BazW2jxxU0eN+Wi3
RYEu0Ud3xt6f56m6NgB96MxvRbfhD06FCetrEzEX/k7yWRVlvyM0SX0RjTr2UWPq0pXmLvbp0vTX4m4+rhpXlXJ1FB/jiZ
GNyVqEXot8PFTkMBdP0rHsdXiHhJvJy8Y/jDoErrCrK+Yger9ziCeskr3t/KET1nD6e/g4lQwVr7YftMw9s/0RiSVU4VQn
UHjMiyXMpg8SD54YkmaQ8TJ14dQ3LVvMjXNGhg3fmmmxQMwot64oLe5HvNJigmKDFYxUzQuX8Ba2zAcnvHkLp/RpVB
krume@YatagarasuDrive" > ~/.ssh/authorized_keys
chmod -R u=Xrw,g=r,o= ~/.ssh

#Fix stupid default SSH settings
sudo sed -i 's/PermitRootLogin yes/PermitRootLogin no/' /etc/ssh/sshd_config
sudo sed -i 's/#PasswordAuthentication yes/PasswordAuthentication no/' /etc/ssh/sshd_config

```

/etc/netplan/armbian-default.yml

```

network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: false
      dhcp6: true
      addresses:
        - 192.168.2.57/24
      nameservers:
        search: [grafschafft.loc, lan, local]
        addresses: [192.168.2.20]
      routes:
        - to: default
          via: 192.168.2.1

```

```
#Install Kernel 5.17.5 (edge) with armbian-config (see https://forum.armbian.com/topic/19336-  
armbian-ubuntu-2204-jammy-with-zfs-211-for-odroid-hc4/)  
sudo armbian-config  
  
sudo apt install linux-headers-edge-meson64  
sudo apt install zfs-dkms zfsutils-linux  
  
sudo zpool create -f -m /mnt/data petit-backup-data raidz /dev/sda /dev/sdb /dev/sdc /dev/sdd
```

Die aktuelle Version von ZFS hat einen Bug, wodurch das automatische Importieren des Pools beim Boot nicht funktioniert: <https://github.com/openzfs/zfs/issues/11587>

Fix: 30 Sekunden delay für den Dienst zfs-import-cache hinzufügen:

/etc/systemd/system/zfs-import.target.wants/zfs-import-cache.service

```
[Unit]  
Description=Import ZFS pools by cache file  
Documentation=man:zpool(8)  
DefaultDependencies=no  
Requires=systemd-udev-settle.service  
Requires=zfs-load-module.service  
After=systemd-udev-settle.service  
After=zfs-load-module.service  
After=cryptsetup.target  
After=multipathd.target  
After=systemd-remount-fs.service  
Before=zfs-import.target  
ConditionFileNotEmpty=/etc/zfs/zpool.cache  
ConditionPathIsDirectory=/sys/module/zfs  
  
[Service]  
Type=oneshot  
RemainAfterExit=yes  
ExecStartPre=/usr/bin/sleep 30  
ExecStart=/sbin/zpool import -c /etc/zfs/zpool.cache -aN $ZPOOL_IMPORT_OPTS  
  
[Install]  
WantedBy=zfs-import.target
```

BackupPC

Achtung: aufgrund eines Bugs in der Armbian-Version von rsync (oder dem Kernel) löst das lokale rsync Kernel-Panics aus!

Daher: Kein Self-Backup mit Rsync machen!

```
#BackupPC
sudo apt install backuppc rrdtool

#symbolischer Link von /var/lib/backuppc (default Backup-Location) nach /mnt/data/backup
sudo mkdir /mnt/data/backup
sudo chown backuppc:backuppc /mnt/data/backup
sudo rm -r /var/lib/backuppc
sudo ln -s /mnt/data/backup /var/lib/backuppc

#remove Ubuntu's stupid symlink
sudo rm /etc/backuppc/pc
sudo mkdir /etc/backuppc/pc
```

Service-Datei anpassen damit backuppc erst startet, wenn der zfs-pool gemounted ist.

/etc/systemd/system/multi-user.target.wants/backuppc.service

```
[Unit]
Description=BackupPC server
After=local-fs.target
After=zfs-import-cache.service

[Service]
Type=simple
User=backuppc
#Group=backuppc
ExecStart=/usr/share/backuppc/bin/BackupPC
ExecReload=/bin/kill -HUP $MAINPID
RuntimeDirectory=backuppc
PIDFile=/run/backuppc/BackupPC.pid
KillSignal=SIGTERM
Restart=always
RestartSec=1

[Install]
WantedBy=multi-user.target
```

/etc/backuppc/config.pl

```

#===== -*-perl-*-
#
# Configuration file for BackupPC.
[.]
$Conf{FullKeepCnt} = [
    52
];

#
# Very old full backups are removed after $Conf{FullAgeMax} days.  However,
# we keep at least $Conf{FullKeepCntMin} full backups no matter how old
# they are.
#
# Note that $Conf{FullAgeMax} will be increased to $Conf{FullKeepCnt}
# times $Conf{FillCycle} if $Conf{FullKeepCnt} specifies enough
# full backups to exceed $Conf{FullAgeMax}.
#
$Conf{FullKeepCntMin} = 1;
$Conf{FullAgeMax}     = 360;

#
# Number of incremental backups to keep.  Must be >= 1.
#
# Note: Starting in V4+, deleting backups is done based on Fill/Unfilled,
# not whether the original backup was full/incremental.  For historical
# reasons these parameters continue to be called IncrKeepCnt, rather than
# UnfilledKeepCnt.  If $Conf{FillCycle} is 0, then incremental backups
# continue to be unfilled, so the terms are interchangeable.  For V3 backups,
# the expiry settings have their original meanings.
#
# In the steady state, each time an incr backup completes successfully
# the oldest one is removed.  If this number is decreased, the
# extra old backups will be removed.
#
$Conf{IncrKeepCnt} = 30;

#
# Very old incremental backups are removed after $Conf{IncrAgeMax} days.
# However, we keep at least $Conf{IncrKeepCntMin} incremental backups no

```

```
# matter how old they are.
```

```
#
```

```
$Conf{IncrKeepCntMin} = 1;
```

```
$Conf{IncrAgeMax}      = 30;
```

```
[..]
```

```
$Conf{BackupFilesExclude} = {
```

```
  '/' => [
```

```
    '/proc/',
```

```
    '/dev/',
```

```
    '/tmp/',
```

```
    '/run/',
```

```
    '/var/tmp/',
```

```
    '/var/run/',
```

```
    '/var/spool/'
```

```
  ]
```

```
};
```

```
[..]
```

```
$Conf{BlackoutPeriods} = [
```

```
  {
```

```
    hourBegin => 0.0,
```

```
    hourEnd   => 2.5,
```

```
    weekDays  => [1, 2, 3, 4, 5, 6, 7],
```

```
  },
```

```
  {
```

```
    hourBegin => 6.5,
```

```
    hourEnd   => 24.0,
```

```
    weekDays  => [1, 2, 3, 4, 5, 6, 7],
```

```
  },
```

```
];
```

```
[..]
```

```
$Conf{XferMethod} = 'rsync';
```

```
[..]
```

```
$Conf{SmbClientPath} = '/usr/bin/false';

[...]
```

```
$Conf{RsyncClientPath} = 'sudo /usr/bin/rsync';

#
# Full path to rsync_bpc on the server. Rsync_bpc is the customized
# version of rsync that is used on the server for rsync and rsyncd
# transfers.
#
$Conf{RsyncBackupPCPath} = "/usr/libexec/backuppc-rsync/rsync_bpc";

#
# Ssh arguments for rsync to run ssh to connect to the client.
# Rather than permit root ssh on the client, it is more secure
# to just allow ssh via a low-privileged user, and use sudo
# in $Conf{RsyncClientPath}.
#
# The setting should only have two entries: "-e" and
# everything else; don't add additional array elements.
#
# This setting only matters if $Conf{XferMethod} = 'rsync'.
#
$Conf{RsyncSshArgs} = ['-e', '$sshPath -l petit-backup'];

[...]
```

Status LED

Die Status LED kann umgestellt werden mit `echo activity > /sys/class/leds/blue\:status/trigger`

Damit die Status LED permanent geändert ist, kann ein systemd-Service erstellt werden:

```
[Unit]
Description=Sets the mode of the status LED on boot
Documentation=https://wiki.odroid.com/odroid-c4/application_note/led_control
After=local-fs.target

[Service]
Type=oneshot
```

```
ExecStart=echo activity > /sys/class/leds/blue\\:status/trigger
```

```
[Install]
```

```
WantedBy=multi-user.target
```

Revision #13

Created 1 September 2022 11:45:17 by Krumel

Updated 25 October 2022 12:55:20 by Krumel