V1.0.19

THIRTYSOMETHING

DIY-NAS

About my DIY-NAS

ThirtySomething

26.03.2023

Contents

Contents . . . 1

Change history . . . 3

1 Initial situation . . . 4

2 Search for alternatives . . . 5
- 2.1 Use a mini PC in front of the NAS . . . 5
- 2.2 Hardware alternatives . . . 5
- 2.3 The planned hardware . . . 5
- 2.4 The bought hardware . . . 6
- 2.5 The difference . . . 7
- 2.6 The consequences . . . 7

3 The hardware build . . . 8

4 The software installation . . . 9
- 4.1 The operating system . . . 9
- 4.2 The disks . . . 9
- 4.3 EXT4 storage . . . 10
- 4.4 ZFS storage . . . 11
- 4.5 The filesystem . . . 12
- 4.6 Shared folders . . . 12
- 4.7 Users . . . 13
- 4.8 CIFS shares . . . 14
- 4.9 Tuning . . . 15
  - 4.9.1 SMB/CIFS . . . 15
- 4.10 Performance . . . 15

5 OMV plugins . . . 19
- 5.1 OMV extras . . . 19
- 5.2 ClamAV . . . 19
- 5.3 MiniDLNA . . . 22
- 5.4 rsync . . . 23

6 Docker . . . 26
- 6.1 Docker installation . . . 26
- 6.2 Portainer . . . 26
- 6.3 Container list . . . 28

6.4 SCM-Manager 28
6.5 Syncthing 30
6.6 Pi-hole 31

7 Ports 32
7.1 Ports by application 32
7.2 Ports by port number 32

A svnExport.sh 34

List of Figures I

List of Tables II

Glossary III

Change history

Version	Date	Description	Name
1.0.0	02.04.2022	• Start with description	ThirtySomething
1.0.1	16.04.2022	• Rename section WOL to Autoshutdown • Fill section Autoshutdown with content	ThirtySomething
1.0.2	16.04.2022	• Split file DIY-NAS-Content into separate files	ThirtySomething
1.0.3	21.04.2022	• Update to latest template version • Add section about used ports • Add svnExport.sh as appendix	ThirtySomething
1.0.4	23.04.2022	• Add documentation about Syncting docker container	ThirtySomething
1.0.5	25.04.2022	• Split section OMV plugins into files • Add description of PhotoPrism	ThirtySomething
1.0.6	28.04.2022	• Add list of containers from old document	ThirtySomething
1.0.7	30.04.2022	• Use latest template version • Add section about Pi-hole	ThirtySomething
1.0.8	11.05.2022	• Use latest template version • Add copyright hint to the images	ThirtySomething
1.0.9	19.05.2022	• Add description of Watchtower	ThirtySomething
1.0.10	21.05.2022	• Add description of MariaDB and phpMyAdmin • Switch SCM-Manager from manual to docker-compose.yaml • Switch syncthing from manual to docker-compose.yaml • Add description of Mosquitto	ThirtySomething
1.0.11	26.05.2022	• Add description of AIO docker compose file • Add description of Gerbera	ThirtySomething
1.0.12	27.05.2022	• Remove description of Gerbera • Add basic description of Nextcloud	ThirtySomething
1.0.13	28.05.2022	• Fix some minor errors • Add links to components	ThirtySomething
1.0.14	03.06.2022	• Add section monitoring	ThirtySomething
1.0.15	14.07.2022	• Used correct image for SCM-Manager	ThirtySomething
1.0.16	10.09.2022	• Add monitorning using cAdvisor	ThirtySomething
1.0.17	07.02.2022	• Replace monitoring with dockprom • Added chapter about ZFS • Removed docker AIO	ThirtySomething
1.0.18	07.02.2022	• Change order of OMV plugins • Add chapter about rsync	ThirtySomething
1.0.19	25.03.2023	• Re-installation	ThirtySomething

Table 1: Change history

1 Initial situation

In 2010 I bought my first NAS. It was a Synology DS411slim. The device is running up to now. For a few years now, it is only supplied with security updates. That is quite remarkable for the fact that it has already 12 years on the hump.

Now I've run out of space - I have 4*2TB running there in RAID 10, which results in about 3.7TB. In addition, the performance is, well, not quite up to date.

2 Search for alternatives

2.1 Use a mini PC in front of the NAS

Realizing this situation I was searching for alternatives. The first idea was to use an existing mini PC in front of the NAS. This failed for some reasons.

- Using the NAS directly for Docker volumes only possible using iSCSI – everything else I was not successful.
- The CPU of the mini PC does not support Intel VT for virtualization in case of Proxmox for example.
- The network itself as bottleneck for Docker containers.

2.2 Hardware alternatives

Then I decided to by a new NAS system. But which kind of NAS will it be? I've spent a long time to search the internet for possible soultions. I didn't know about the possibility to pimp a TerraMaster NAS with a different OS. This is also possible for Western Digital NAS as described here and here. Also the variant with a mini PC and separated storage case was interesting. The idea is to get the most out of money, so the ranking here is done by price/performance ratio and the DIY NAS wins the comparison.

1. DIY NAS system on PC base
2. DIY NAS system with separated storage case and mini PC
3. Commercial NAS system
4. Commercial NAS with custom OS

2.3 The planned hardware

When comparing the different hardware solutions, I also made a comparison between an Intel and an AMD based NAS. The AMD variant won the price/power chapter, so I decided to buy:

- 32GB G.Skill RipJaws V black DDR4-3200 DIMM Dual Kit
- 250GB Samsung 970 Evo Plus M.2 2280

- 400 Watt be quiet! Pure Power 11 CM Modular 80+ Gold
- Black Fractal Design Node 304 cube without power supply
- 4x 4000GB WD Red Plus WD40EFZX 128MB 3.5"
- ASRock Fatal1ty B450 Gaming-ITX/AC AMD B450
- AMD Athlon 3000G with Radeon Vega Graphics 3.5GHz
- Noctua NH-L9a-AM4 topblow cooler

2.4 The bought hardware

„Life is what happens to you while you're busy making other plans."

Figure 1: A quote from John Lennon

This means that the availability of chips and other events affect the market and make it difficult to realize these plans. For example, motherboards in mini-ITX format with AM4 socket have been really hard to get in the first half of 2022. That's why I revised my decision on the components:

- 32GB G.Skill RipJaws V black DDR4-3200 DIMM Dual Kit
- 250GB Samsung 970 Evo Plus M.2 2280
- 400 Watt be quiet! Pure Power 11 CM Modular 80+ Gold
- Black Fractal Design Node 304 cube without power supply
- 4x 4000GB WD Red Plus WD40EFZX 128MB 3.5"
- ASRock H610M-ITX/AC mITX Intel H610 DDR4 S1700
- Intel Core I3-12100 tray
- Noctua NH-L9i-17xx topblow cooler

2.5 The difference

The difference is that the CPU is a latest generation Intel I3 processor. The corresponding motherboard is also brand new. Both were only launched in Q1/2022.

2.6 The consequences

The fresh introduction to the market has some consequences: OMV 5 does not support the network chip of the motherboard. So I decided to install the OMV 6 beta on my DIY NAS. I also tried TrueNAS – it looks much more professional than OMV, but it is also a bit more complicated to understand and configure. It also doesn't have native support for Docker.

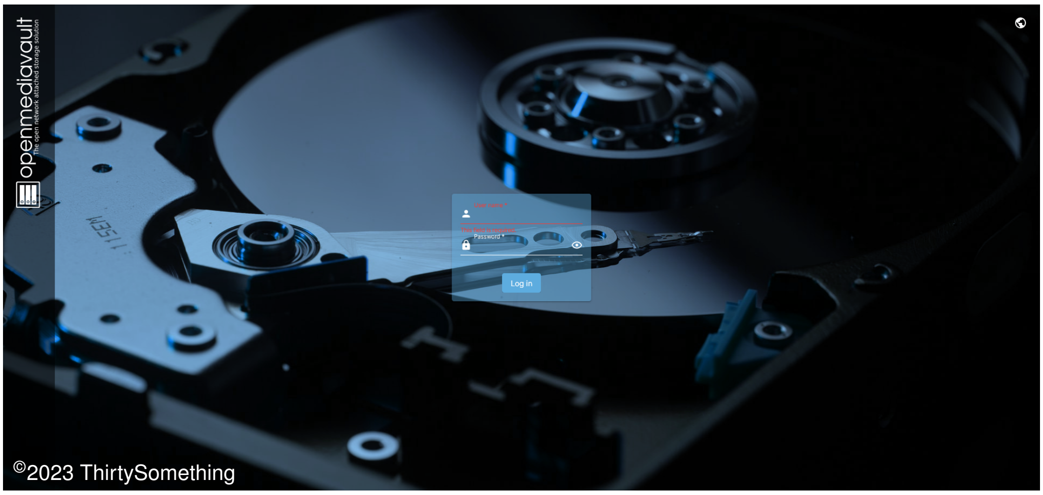
3 The hardware build

Although I build up my last PC more than 25 years ago assembling the hardware was a breeze. The only surprise was the mounting position of the power supply. As I understand how to do this everything was fine.

4 The software installation

4.1 The operating system

The basic installation of OMV works without any problems. This is known from debian. Additional Volker Theile, the founder of the OMV project, and his team have done a very good job. Thank you guys!



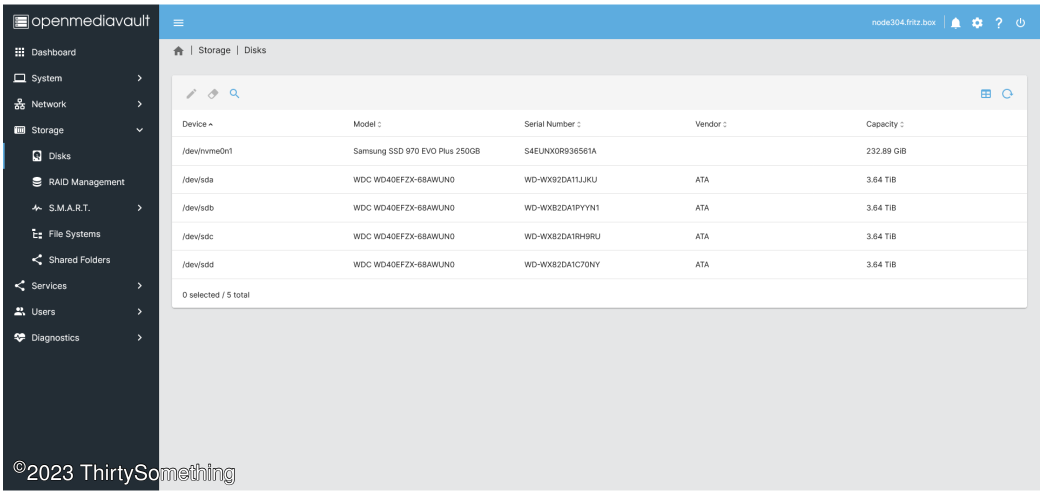
This field is required.

©2023 ThirtySomething

Figure 2: The OMV login page

4.2 The disks

All attached disks are working. The system disk is a NVME SSD with a 250 GB capacity. All others are 4 TB WD RED disks.



Device	Model	Serial Number	Vendor	Capacity
/dev/nvme0n1	Samsung SSD 970 EVO Plus 250GB	S4EUNX0R936561A		232.89 GiB
/dev/sda	WDC WD40EFZX-68AWUN0	WD-WX92DA11JJKU	ATA	3.64 TiB
/dev/sdb	WDC WD40EFZX-68AWUN0	WD-WXB2DA1PYYN1	ATA	3.64 TiB
/dev/sdc	WDC WD40EFZX-68AWUN0	WD-WX82DA1RH9RU	ATA	3.64 TiB
/dev/sdd	WDC WD40EFZX-68AWUN0	WD-WX82DA1C70NY	ATA	3.64 TiB

0 selected / 5 total

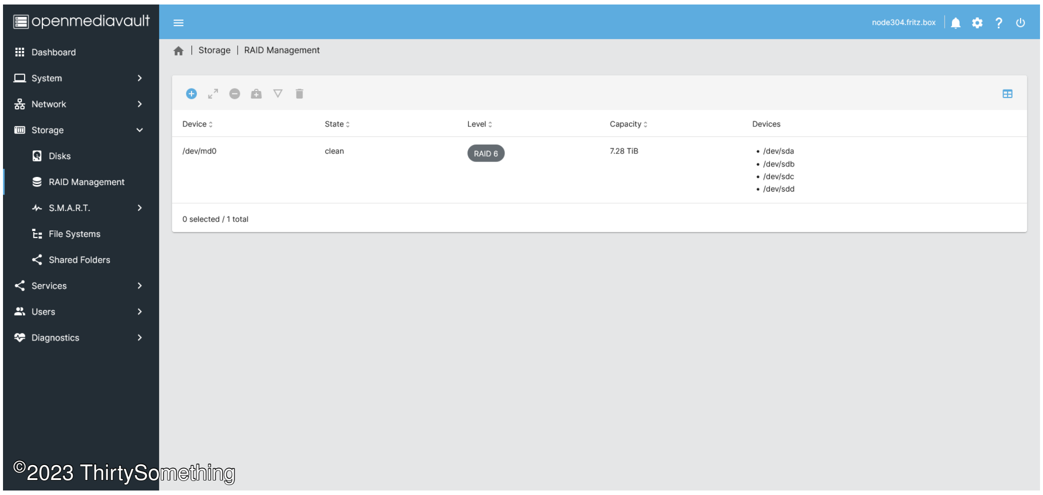
Figure 3: The OMV disks

Before you can create a filesystem on the disks, you have to erase them first.

4.3 EXT4 storage

Configuring the RAID was also less problematic. But there is an important point: As long as the RAID system is created, do not use the RAID! During the RAID build I was playing around with OMV plugins to see the diskstats – and damaged the RAID build. After a reboot the system hangs on the filesystem check. I was shocked - also about the large amount of blocks the fsck found and tried to repair. It took a while to understand what happens. Then I've startet from scratch and everything was okay. The creation of the filesystem at the end went without problems. A filesystem check found nothing to do.

V1.0.19



Device	State	Level	Capacity	Devices
/dev/md0	clean	RAID 6	7.28 TiB	• /dev/sda • /dev/sdb • /dev/sdc • /dev/sdd

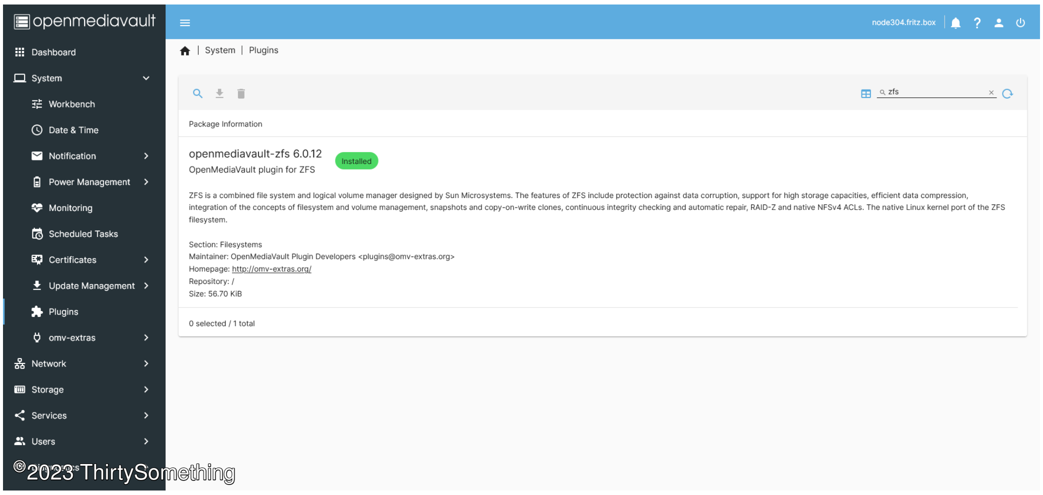
0 selected / 1 total

Figure 4: The OMV RAID

Setting up a RAID 6 on the four disks using EXT4 took aproximately 50 hours.

4.4 ZFS storage

Using the ZFS plugin offers an enterprise level filesystem compared to EXT4. The creation of this filesystem is done in a few seconds compared to EXT4. Transmitting data through the network to the NAS seems to be much faster than doing this on the same device using EXT4.



Package Information

openmediavault-zfs 6.0.12 Installed
OpenMediaVault plugin for ZFS

ZFS is a combined file system and logical volume manager designed by Sun Microsystems. The features of ZFS include protection against data corruption, support for high storage capacities, efficient data compression, integration of the concepts of filesystem and volume management, snapshots and copy-on-write clones, continuous integrity checking and automatic repair, RAID-Z and native NFSv4 ACLs. The native Linux kernel port of the ZFS filesystem.

Section: Filesystems
Maintainer: OpenMediaVault Plugin Developers <plugins@omv-extras.org>
Homepage: http://omv-extras.org/
Repository: /
Size: 56.70 KiB

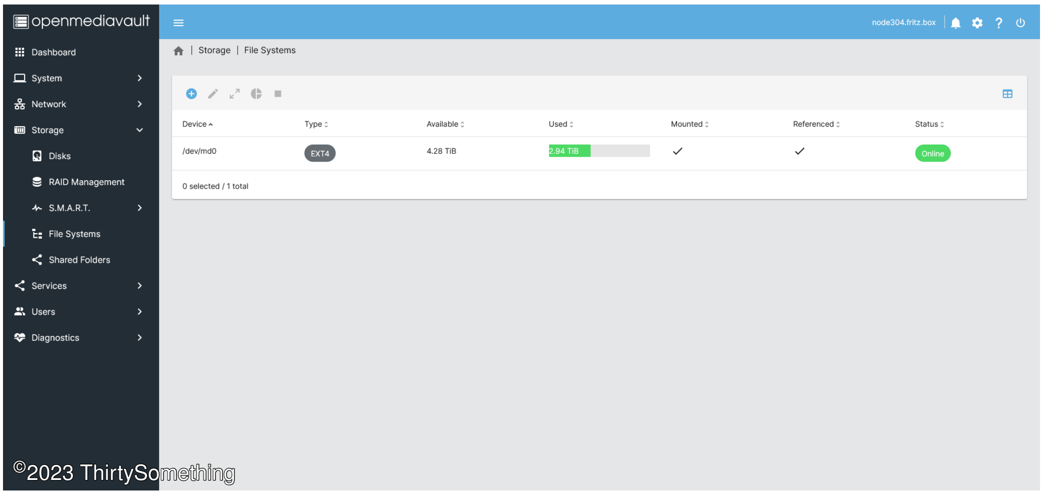
0 selected / 1 total

Figure 5: The OMV ZFS plugin

Testing and playing around with the ZFS showed that there is a lot of knowledge about this filesystem required – and I'm lacking this knowledge. So I decided not to use the ZFS storage.

4.5 The filesystem

This was a simple step – just mount the previously created RAID volume.



Device	Type	Available	Used	Mounted	Referenced	Status
/dev/md0	EXT4	4.28 TiB	2.94 TiB	✓	✓	Online

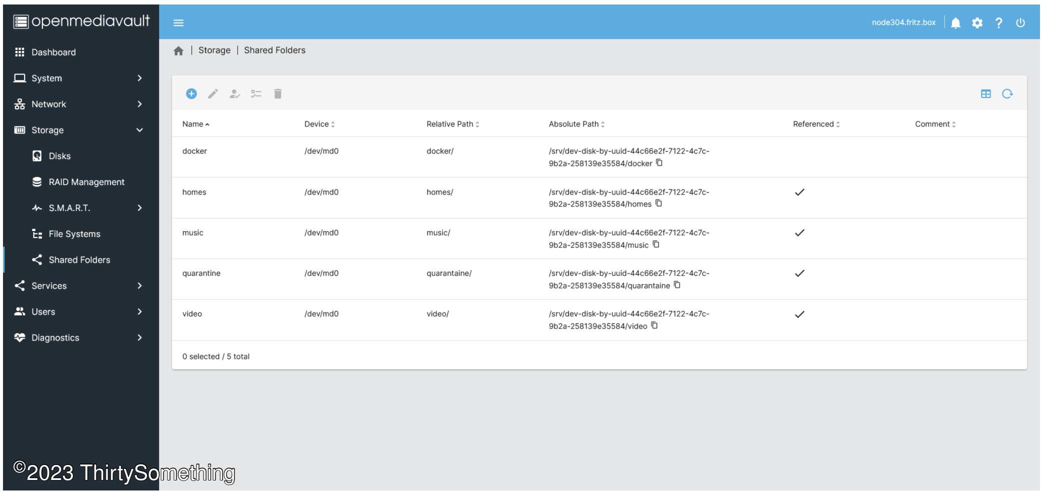
0 selected / 1 total

Figure 6: The OMV filesystem on the RAID

4.6 Shared folders

Some common folders should be defined before anything else is done:

- The folder `docker` for the use of Docker.
- The folder `homes` as base folder for the users home directories.
- The folder `music` for the miniDLNA plugin.
- The folder `quarantine` for the use of ClamAV.
- The folder `video` also for the miniDLNA plugin.



Name	Device	Relative Path	Absolute Path	Referenced	Comment
docker	/dev/md0	docker/	/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/docker		
homes	/dev/md0	homes/	/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/homes	✓	
music	/dev/md0	music/	/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/music	✓	
quarantine	/dev/md0	quarantaine/	/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/quarantaine	✓	
video	/dev/md0	video/	/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/video	✓	

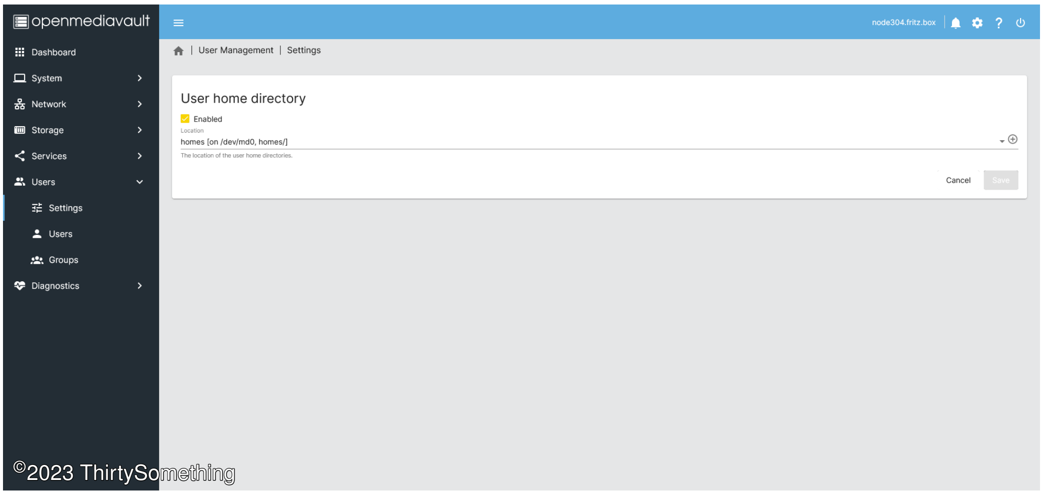
0 selected / 5 total

Figure 7: The OMV shared folders

This „shared folders“ are defined as container to be used inside of OMV. To make them accessible from the network you have to enable services for them.

4.7 Users

In the settings the option `User home directory` is enabled and points to the previously created `homes` folder. Then I created the users.



User home directory

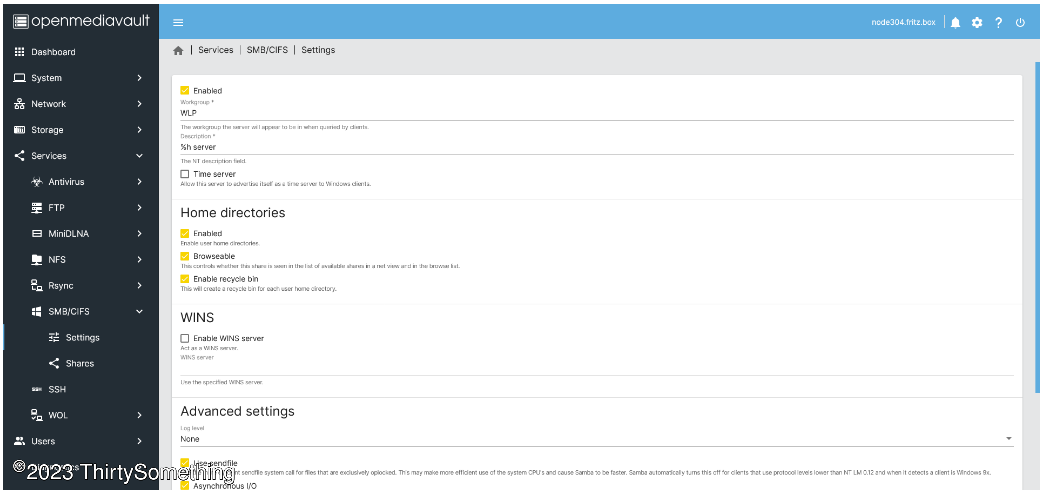
☑ Enabled

Location
homes [on /dev/md0, homes/]
The location of the user home directories.

Figure 8: The OMV users home directory

4.8 CIFS shares

Simple - enable the `SMB/CIFS` service, enable the `home directories` and then create shares for the previously defined `shared folders`. Allow read/write access for the administrator of the shares (me), the others got read access to them.



Services | SMB/CIFS | Settings

☑ Enabled

Workgroup *
WLP
The workgroup the server will appear to be in when queried by clients.

Description *
%h server
The NT description field.

□ Time server
Allow this server to advertise itself as a time server to Windows clients.

Home directories

☑ Enabled
Enable user home directories.

☑ Browseable
This controls whether this share is seen in the list of available shares in a net view and in the browse list.

☑ Enable recycle bin
This will create a recycle bin for each user home directory.

WINS

□ Enable WINS server
Act as a WINS server.

WINS server

Use the specified WINS server.

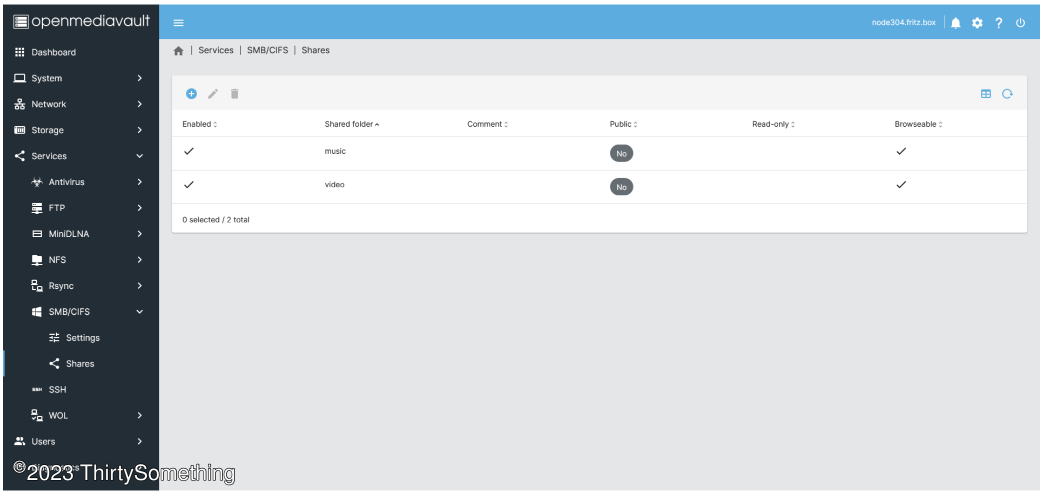
Advanced settings

Log level
None

☑ Use sendfile
[illegible] sendfile system call for files that are exclusively oplocked. This may make more efficient use of the system CPU's and cause Samba to be faster. Samba automatically turns this off for clients that use protocol levels lower than NT LM 0.12 and when it detects a client is Windows 9x.

☑ Asynchronous I/O

Figure 9: The OMV CIFS settings



Services | SMB/CIFS | Shares

Enabled	Shared folder	Comment	Public	Read-only	Browseable
✓	music		No		✓
✓	video		No		✓

0 selected / 2 total

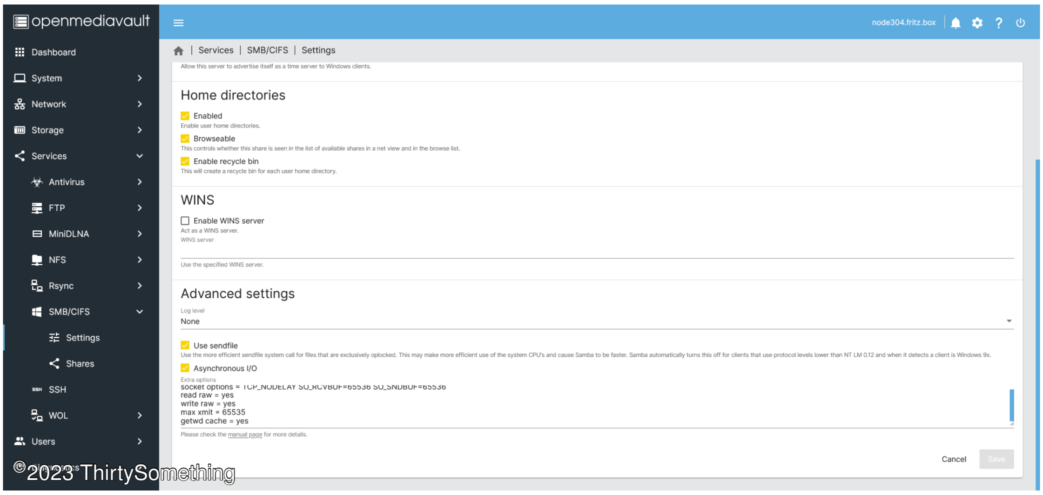
Figure 10: The OMV CIFS shares

As you can see there is actualy no public access to this shares. This means that only priviledged users can access them.

4.9 Tuning

4.9.1 SMB/CIFS

Transfering data from the old NAS to the new one takes a lot of time. To speed up the process I searched for some tuning and found this here:



Services | SMB/CIFS | Settings

Allow this server to advertise itself as a time server to Windows clients.

Home directories

☑ Enabled
Enable user home directories.
☑ Browseable
This controls whether this share is seen in the list of available shares in a net view and in the browse list.
☑ Enable recycle bin
This will create a recycle bin for each user home directory.

WINS

☐ Enable WINS server
Act as a WINS server.
WINS server
Use the specified WINS server.

Advanced settings

Log level
None

☑ Use sendfile
Use the more efficient sendfile system call for files that are exclusively oplocked. This may make more efficient use of the system CPU's and cause Samba to be faster. Samba automatically turns this off for clients that use protocol levels lower than NT LM 0.12 and when it detects a client is Windows 9x.
☑ Asynchronous I/O

Extra options
```
socket options = TCP_NODELAY SO_RCVBUF=65536 SO_SNDBUF=65536
read raw = yes
write raw = yes
max xmit = 65535
getwd cache = yes
```
Please check the manual page for more details.

Figure 11: The OMV advanced CIFS settings

Just add the following tuning parameters as extra options on the SMB/CIFS settings:

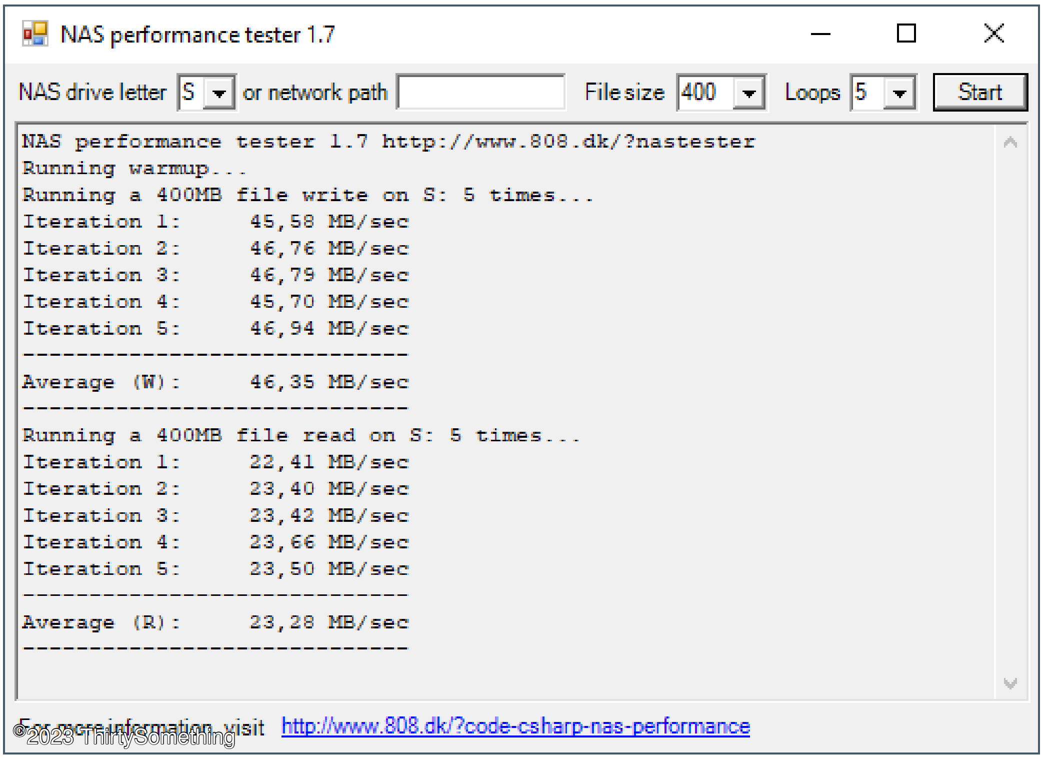
```
socket options = TCP_NODELAY SO_RCVBUF=65536 SO_SNDBUF=65536
read raw = yes
write raw = yes
max xmit = 65535
getwd cache = yes
```

Figure 12: SMB/CIFS tuning options

4.10 Performance

For analyzing the speed of the NAS I've been using the NAS performance tester.

First the old Synology DS411Slim is measured. There is a manufacturer specific OS called DiskStation Manager DSM running using EXT4 as filesystem.

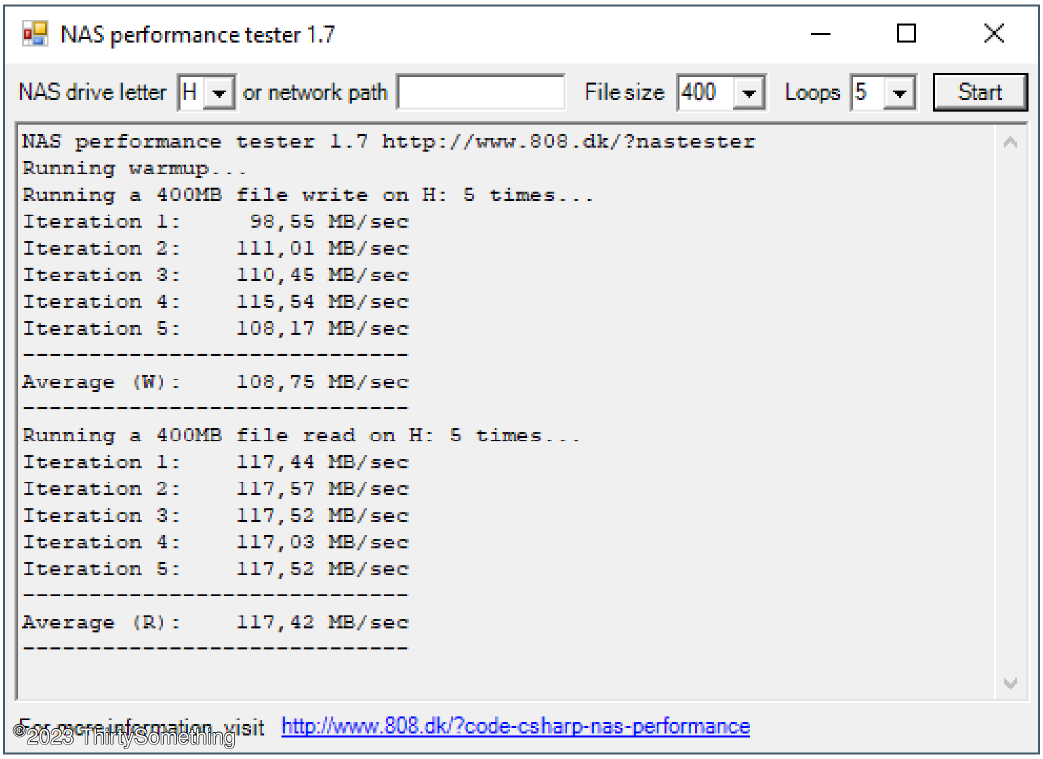


```
NAS performance tester 1.7 http://www.808.dk/?nastester
Running warmup...
Running a 400MB file write on S: 5 times...
Iteration 1:     45,58 MB/sec
Iteration 2:     46,76 MB/sec
Iteration 3:     46,79 MB/sec
Iteration 4:     45,70 MB/sec
Iteration 5:     46,94 MB/sec
-----------------------------
Average (W):     46,35 MB/sec
-----------------------------
Running a 400MB file read on S: 5 times...
Iteration 1:     22,41 MB/sec
Iteration 2:     23,40 MB/sec
Iteration 3:     23,42 MB/sec
Iteration 4:     23,66 MB/sec
Iteration 5:     23,50 MB/sec
-----------------------------
Average (R):     23,28 MB/sec
-----------------------------
```

For more information, visit http://www.808.dk/?code-csharp-nas-performance

Figure 13: The speed of the old DS411Slim

Then the new Node304 is measured second. This is running OMV with an EXT4 filesystem and untuned SMB/CIFS settings.



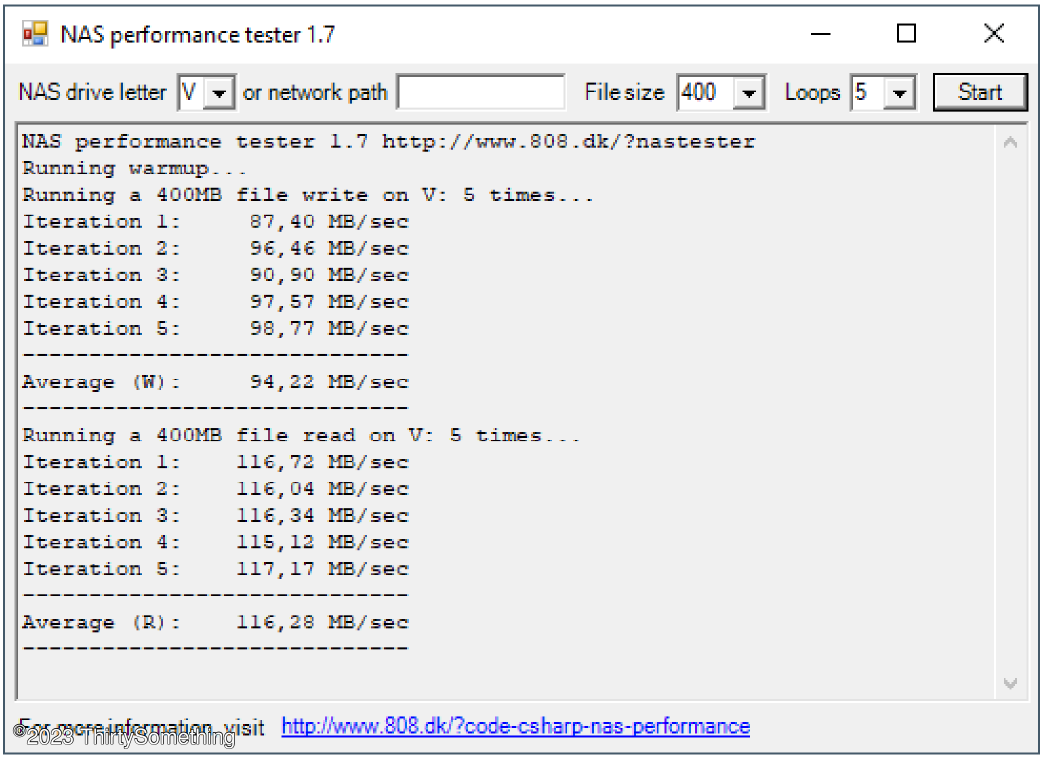
```
NAS performance tester 1.7 http://www.808.dk/?nastester
Running warmup...
Running a 400MB file write on H: 5 times...
Iteration 1:     98,55 MB/sec
Iteration 2:    111,01 MB/sec
Iteration 3:    110,45 MB/sec
Iteration 4:    115,54 MB/sec
Iteration 5:    108,17 MB/sec
-----------------------------
Average (W):    108,75 MB/sec
-----------------------------
Running a 400MB file read on H: 5 times...
Iteration 1:    117,44 MB/sec
Iteration 2:    117,57 MB/sec
Iteration 3:    117,52 MB/sec
Iteration 4:    117,03 MB/sec
Iteration 5:    117,52 MB/sec
-----------------------------
Average (R):    117,42 MB/sec
-----------------------------
```

For more information visit http://www.808.dk/?code-csharp-nas-performance

Figure 14: The speed of the OMV NAS

Reading is more than twice faster than before. And writing even more than five times faster. This is great!

Applying the settings mentioned in the Tuning chapter surprisingly result is a slower performance:



NAS performance tester 1.7
NAS drive letter V or network path
File size 400
Loops 5
Start

```
NAS performance tester 1.7 http://www.808.dk/?nastester
Running warmup...
Running a 400MB file write on V: 5 times...
Iteration 1:      87,40 MB/sec
Iteration 2:      96,46 MB/sec
Iteration 3:      90,90 MB/sec
Iteration 4:      97,57 MB/sec
Iteration 5:      98,77 MB/sec
-----------------------------
Average (W):      94,22 MB/sec
-----------------------------
Running a 400MB file read on V: 5 times...
Iteration 1:     116,72 MB/sec
Iteration 2:     116,04 MB/sec
Iteration 3:     116,34 MB/sec
Iteration 4:     115,12 MB/sec
Iteration 5:     117,17 MB/sec
-----------------------------
Average (R):     116,28 MB/sec
-----------------------------
```

For more information, visit http://www.808.dk/?code-csharp-nas-performance

Figure 15: OMV NAS with tuned SMB/CIFS

So the final settings will be the default without the tuning settings.

5 OMV plugins

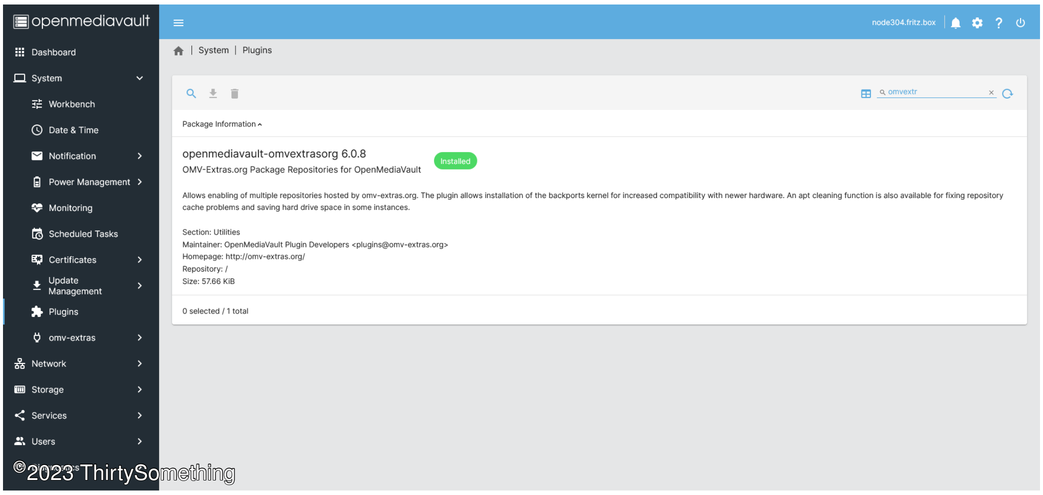
5.1 OMV extras

The OMV extras are not available in the default plugin list. The way to go to install them is described here. Login as user `root` using SSH and enter the following command:

```
wget -O - https://github.com/OpenMediaVault-Plugin-Developers/packages/raw/master/install | bash
```

Figure 16: The OMV extras installation

Using this plugin enables at least the Docker and Portainer installation to enhance the capabilities of the NAS. Also ReadyMedia is now available in the OMV extras.



Package Information

openmediavault-omvextrasorg 6.0.8 Installed
OMV-Extras.org Package Repositories for OpenMediaVault

Allows enabling of multiple repositories hosted by omv-extras.org. The plugin allows installation of the backports kernel for increased compatibility with newer hardware. An apt cleaning function is also available for fixing repository cache problems and saving hard drive space in some instances.

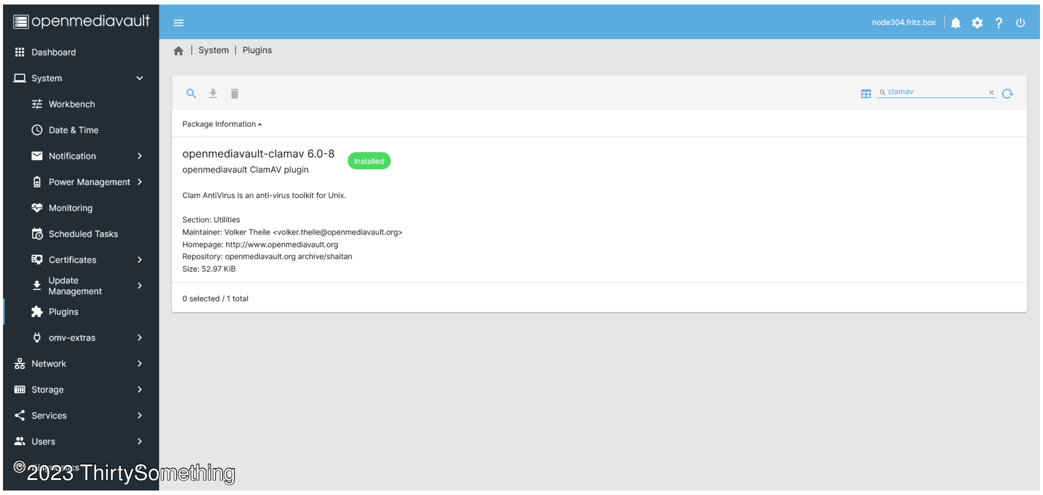
Section: Utilities
Maintainer: OpenMediaVault Plugin Developers <plugins@omv-extras.org>
Homepage: http://omv-extras.org/
Repository: /
Size: 57.66 KiB

0 selected / 1 total

Figure 17: The OMV extras plugin

5.2 ClamAV

To protect the data I want to use an antivirus program. As open source solution there is ClamAV available – and also as plugin for OMV.



openmediavault-clamav 6.0-8
openmediavault ClamAV plugin

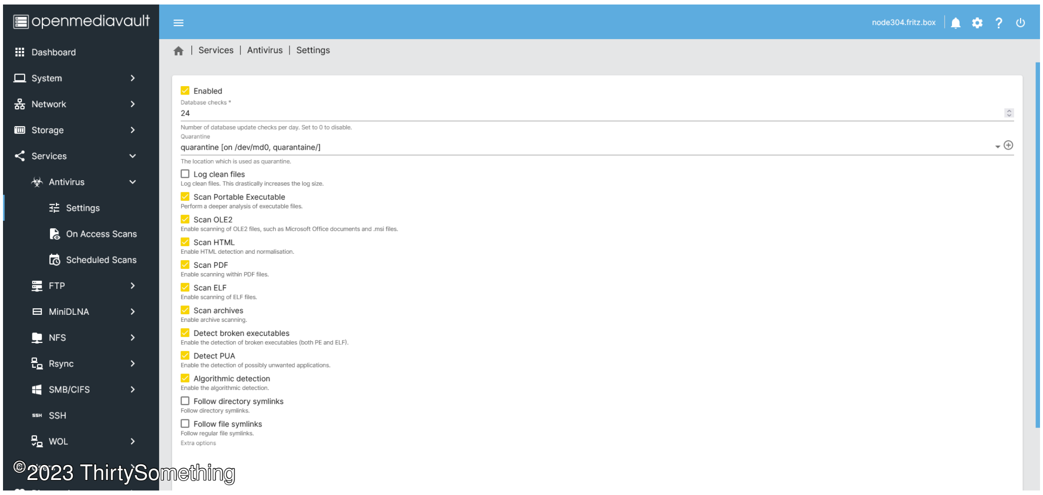
Clam AntiVirus is an anti-virus toolkit for Unix.

Section: Utilities
Maintainer: Volker Theile <volker.theile@openmediavault.org>
Homepage: http://www.openmediavault.org
Repository: openmediavault.org archive/shaitan
Size: 52.97 KiB

0 selected / 1 total

Figure 18: The OMV ClamAV plugin

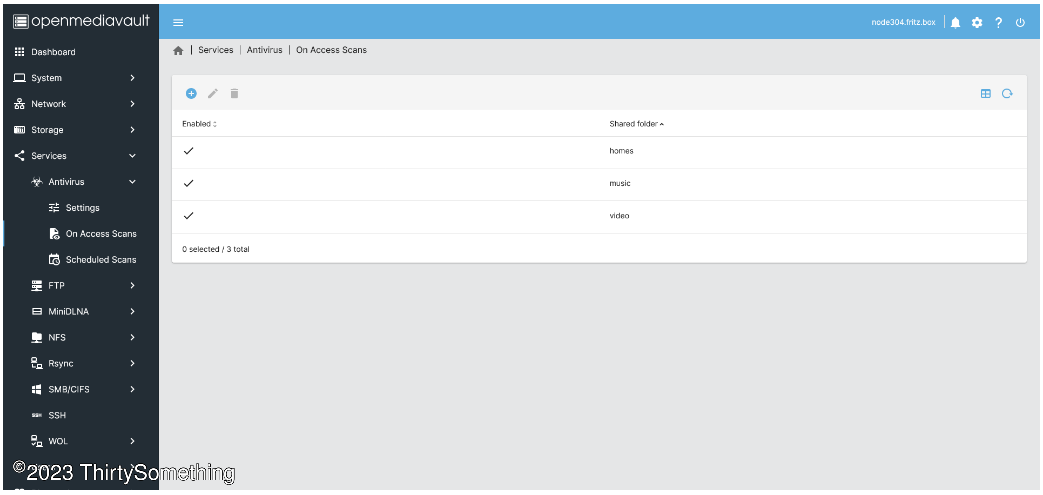
In the setup we use the previously defined quarantine folder.



☑ Enabled
Database checks *
24
Number of database update checks per day. Set to 0 to disable.
Quarantine
quarantine [on /dev/md0, quarantaine/]
The location which is used as quarantine.
☐ Log clean files
Log clean files. This drastically increases the log size.
☑ Scan Portable Executable
Perform a deeper analysis of executable files.
☑ Scan OLE2
Enable scanning of OLE2 files, such as Microsoft Office documents and .msi files.
☑ Scan HTML
Enable HTML detection and normalisation.
☑ Scan PDF
Enable scanning within PDF files.
☑ Scan ELF
Enable scanning of ELF files.
☑ Scan archives
Enable archive scanning.
☑ Detect broken executables
Enable the detection of broken executables (both PE and ELF).
☑ Detect PUA
Enable the detection of possibly unwanted applications.
☑ Algorithmic detection
Enable the algorithmic detection.
☐ Follow directory symlinks
Follow directory symlinks.
☐ Follow file symlinks
Follow regular file symlinks.
Extra options

Figure 19: The antivirus settings

I enabled a scan on access for specific folders.

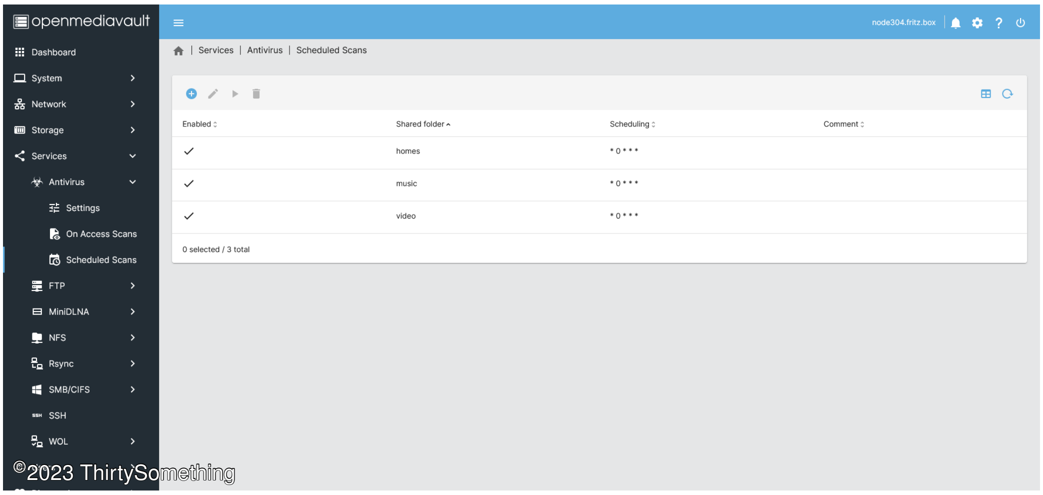


Enabled	Shared folder
✓	homes
✓	music
✓	video

0 selected / 3 total

Figure 20: The antivirus on access scan

Also I've enabled a scheduled scan for these folders, too.



Enabled	Shared folder	Scheduling	Comment
✓	homes	* 0 * * *	
✓	music	* 0 * * *	
✓	video	* 0 * * *	

0 selected / 3 total

Figure 21: The antivirus scheduled scan

Maybe somebody will claim that all scheduled scans run on the same time. Yes – this was setup to check the power of the CPU. The system load increases to a load of about 3 – that's fantastic from my point of view! This means that there are more than enough reserves. When I'm spreading the schedule I can lower the load.

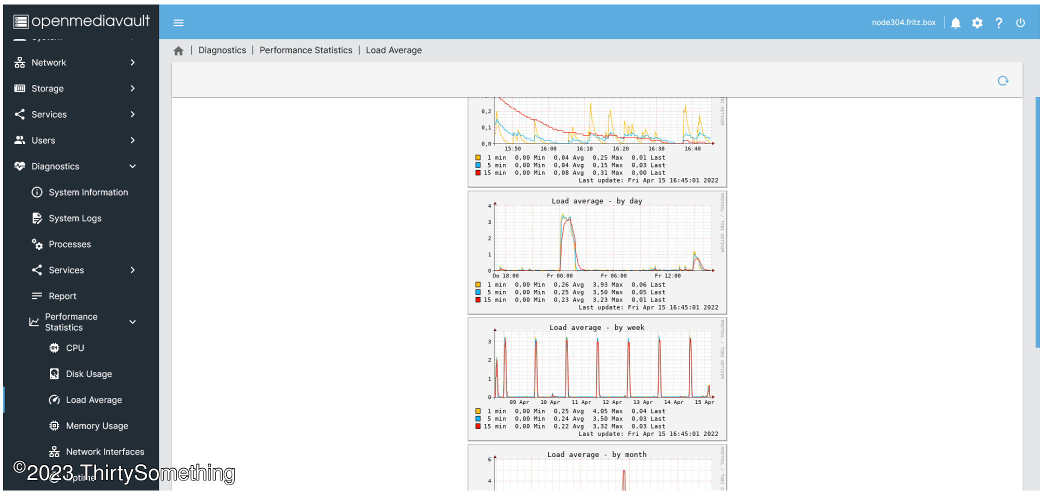
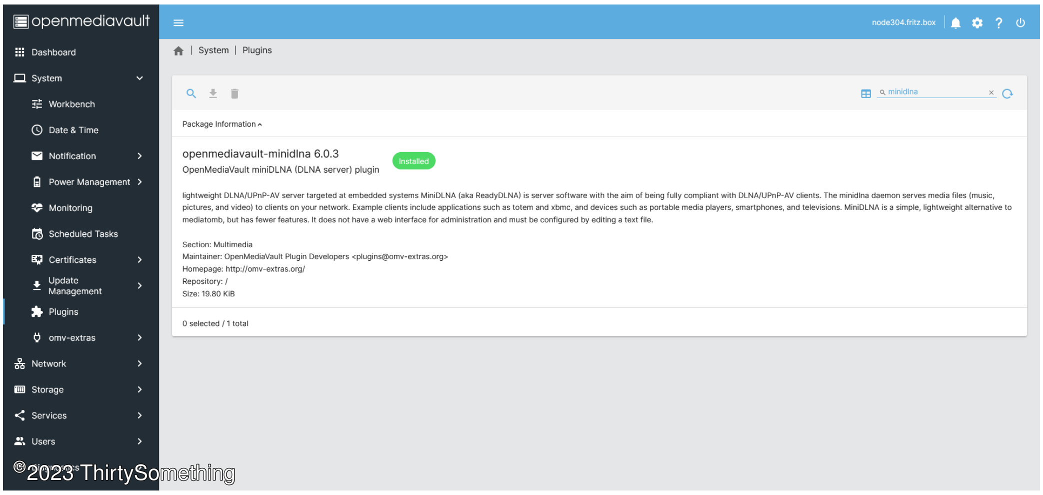

openmediavault
Network
Storage
Services
Users
Diagnostics
System Information
System Logs
Processes
Services
Report
Performance Statistics
CPU
Disk Usage
Load Average
Memory Usage
Network Interfaces
node304.fritz.box
Diagnostics | Performance Statistics | Load Average
0,2
0,1
0,0
15:50 16:00 16:10 16:20 16:30 16:40
1 min 0,00 Min 0,04 Avg 0,25 Max 0,01 Last
5 min 0,00 Min 0,04 Avg 0,15 Max 0,03 Last
15 min 0,00 Min 0,08 Avg 0,31 Max 0,00 Last
Last update: Fri Apr 15 16:45:01 2022
Load average - by day
4 3 2 1 0
Do 18:00 Fr 00:00 Fr 06:00 Fr 12:00
1 min 0,00 Min 0,26 Avg 3,93 Max 0,06 Last
5 min 0,00 Min 0,25 Avg 3,50 Max 0,05 Last
15 min 0,00 Min 0,23 Avg 3,23 Max 0,01 Last
Last update: Fri Apr 15 16:45:01 2022
Load average - by week
3 2 1 0
09 Apr 10 Apr 11 Apr 12 Apr 13 Apr 14 Apr 15 Apr
1 min 0,00 Min 0,25 Avg 4,05 Max 0,04 Last
5 min 0,00 Min 0,24 Avg 3,50 Max 0,03 Last
15 min 0,00 Min 0,22 Avg 3,32 Max 0,03 Last
Last update: Fri Apr 15 16:45:01 2022
Load average - by month
6 4
RRDTOOL / TOBI OETIKER

Figure 22: The antivirus system load on scan

5.3 MiniDLNA

To stream music and videos from the NAS to the network I'm using the ReadyMedia media server software. In OMV this is used with the old name MiniDLNA.



openmediavault-minidlna 6.0.3
OpenMediaVault miniDLNA (DLNA server) plugin

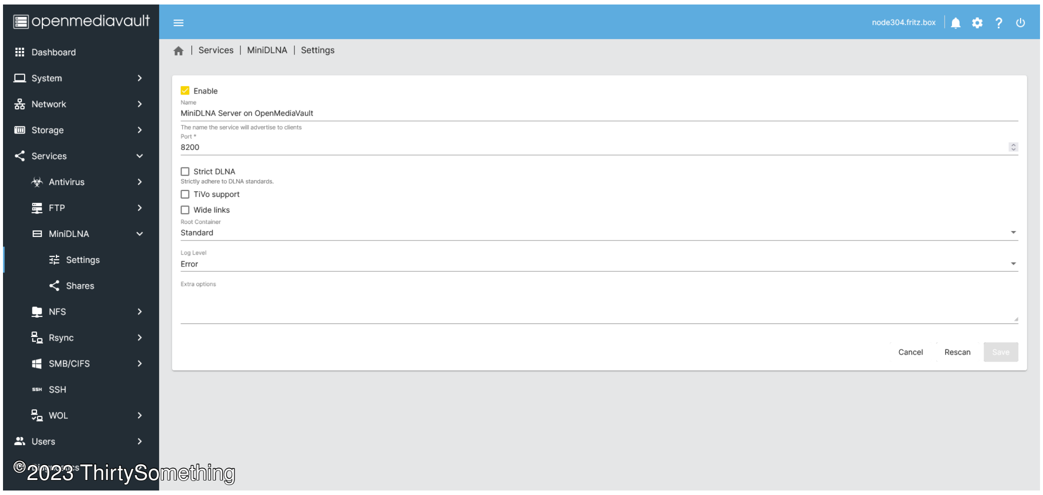
lightweight DLNA/UPnP-AV server targeted at embedded systems MiniDLNA (aka ReadyDLNA) is server software with the aim of being fully compliant with DLNA/UPnP-AV clients. The minidlna daemon serves media files (music, pictures, and video) to clients on your network. Example clients include applications such as totem and xbmc, and devices such as portable media players, smartphones, and televisions. MiniDLNA is a simple, lightweight alternative to mediatomb, but has fewer features. It does not have a web interface for administration and must be configured by editing a text file.

Section: Multimedia
Maintainer: OpenMediaVault Plugin Developers <plugins@omv-extras.org>
Homepage: http://omv-extras.org/
Repository: /
Size: 19.80 KiB

0 selected / 1 total

Figure 23: The MiniDLNA plugin

The basic setup is simple - I've checked the `Enable` box and that's it.



Services | MiniDLNA | Settings

☑ Enable

Name
MiniDLNA Server on OpenMediaVault
The name the service will advertise to clients

Port *
8200

☐ Strict DLNA
Strictly adhere to DLNA standards.

☐ TiVo support

☐ Wide links

Root Container
Standard

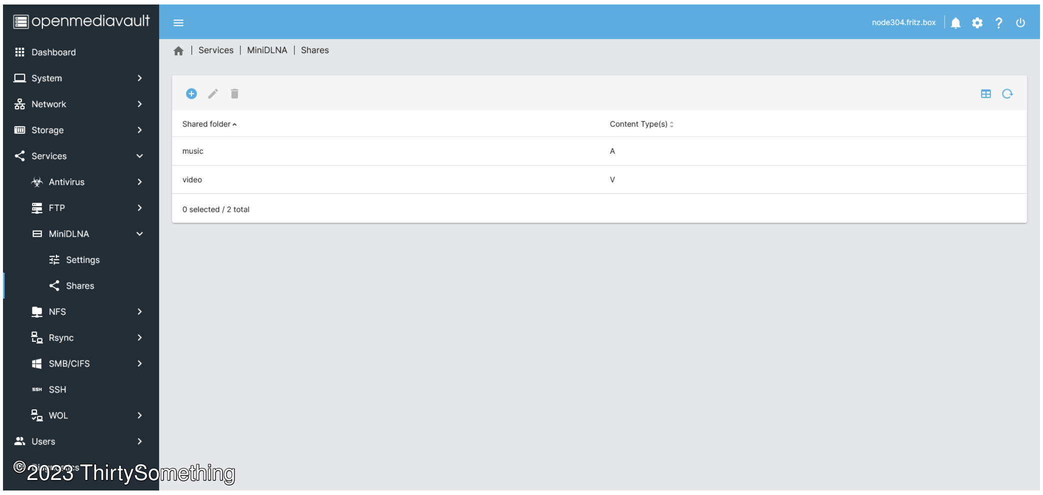
Log Level
Error

Extra options

Cancel Rescan Save

Figure 24: The MiniDLNA settings

Then I have to define the shares and the kind of content of the shares.



Services | MiniDLNA | Shares

Shared folder ^	Content Type(s)
music	A
video	V

0 selected / 2 total

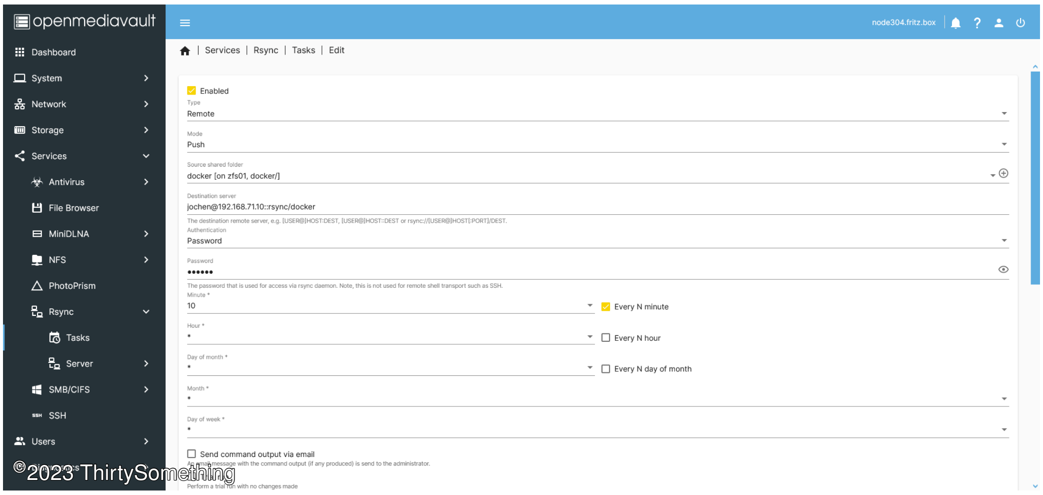
Figure 25: The MiniDLNA shares

Note to myself: The logfile of this software bloated twice all available space on the OS drive. Search for an alternative for this!

5.4 rsync

On Linux it's usual to use rsync to synchronize data between different locations. OMV offers a rsync service – it's not a plugin in the usual meaning. There are many different scenarios possible – I'll describe two of them.

The first one is the sync in the direction of the Node304 to the Synology DS411Slim, also known as rsync push:



☑ Enabled

Type
Remote

Mode
Push

Source shared folder
docker [on zfs01, docker/]

Destination server
jochen@192.168.71.10::rsync/docker
The destination remote server, e.g. [USER@]HOST:DEST, [USER@]HOST::DEST or rsync://[USER@]HOST[:PORT]/DEST.

Authentication
Password

Password
••••••
The password that is used for access via rsync daemon. Note, this is not used for remote shell transport such as SSH.

Minute *
10 ☑ Every N minute

Hour *
* ☐ Every N hour

Day of month *
* ☐ Every N day of month

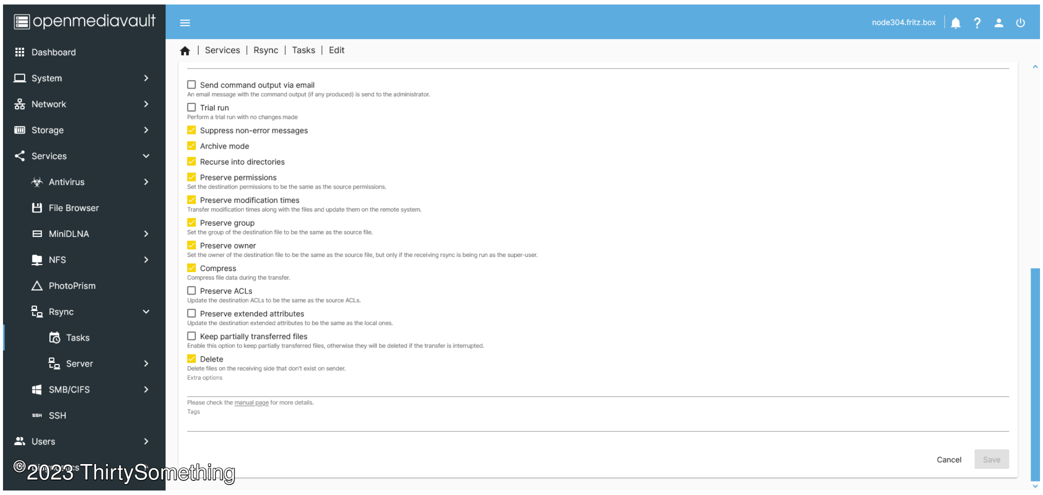
Month *
*

Day of week *
*

☐ Send command output via email
An email message with the command output (if any produced) is send to the administrator.

Perform a trial run with no changes made

Figure 26: rsync push 1



☐ Send command output via email
An email message with the command output (if any produced) is send to the administrator.

☐ Trial run
Perform a trial run with no changes made

☑ Suppress non-error messages

☑ Archive mode

☑ Recurse into directories

☑ Preserve permissions
Set the destination permissions to be the same as the source permissions.

☑ Preserve modification times
Transfer modification times along with the files and update them on the remote system.

☑ Preserve group
Set the group of the destination file to be the same as the source file.

☑ Preserve owner
Set the owner of the destination file to be the same as the source file, but only if the receiving rsync is being run as the super-user.

☑ Compress
Compress file data during the transfer.

☐ Preserve ACLs
Update the destination ACLs to be the same as the source ACLs.

☐ Preserve extended attributes
Update the destination extended attributes to be the same as the local ones.

☐ Keep partially transferred files
Enable this option to keep partially transferred files, otherwise they will be deleted if the transfer is interrupted.

☑ Delete
Delete files on the receiving side that don't exist on sender.

Extra options

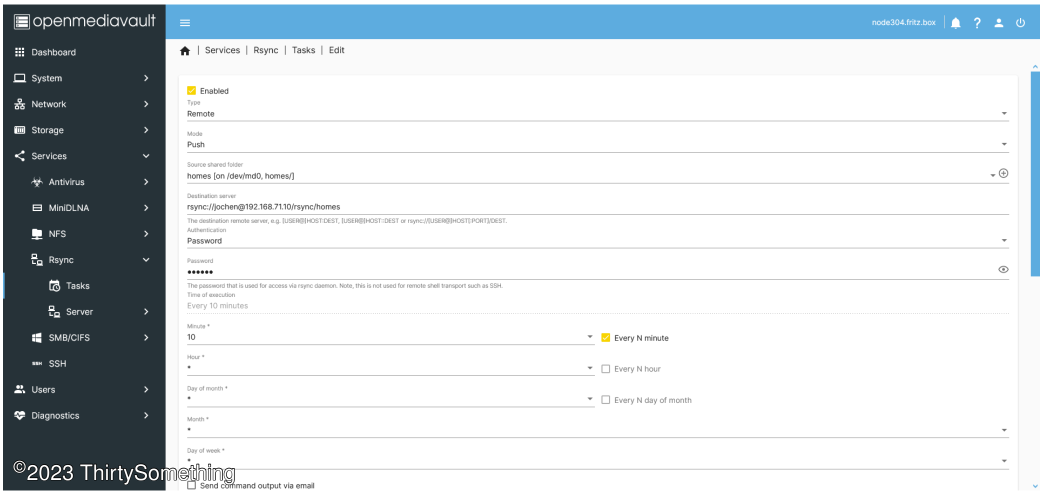
Please check the manual page for more details.

Tags

Cancel Save

Figure 27: rsync push 2

The second one is the sync in the direction of the Synology DS411Slim to the Node304, also known as rsync pull:



Services | Rsync | Tasks | Edit

☑ Enabled

Type
Remote

Mode
Push

Source shared folder
homes [on /dev/md0, homes/]

Destination server
rsync://jochen@192.168.71.10/rsync/homes
The destination remote server, e.g. [USER@]HOST:DEST, [USER@]HOST::DEST or rsync://[USER@]HOST[:PORT]/DEST.

Authentication
Password

Password
••••••
The password that is used for access via rsync daemon. Note, this is not used for remote shell transport such as SSH.

Time of execution
Every 10 minutes

Minute *
10 ☑ Every N minute

Hour *
* □ Every N hour

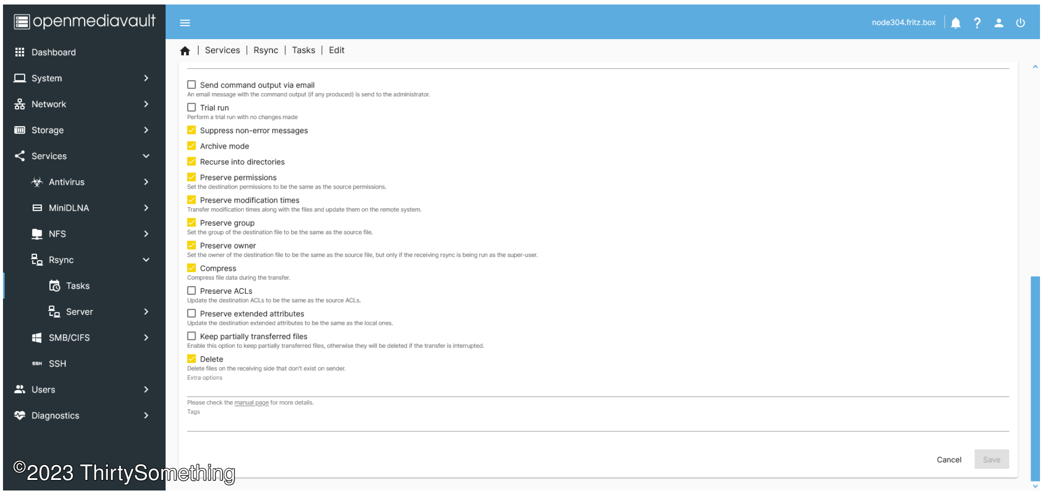
Day of month *
* □ Every N day of month

Month *
*

Day of week *
*

□ Send command output via email

Figure 28: rsync pull 1



Services | Rsync | Tasks | Edit

□ Send command output via email
An email message with the command output (if any produced) is send to the administrator.

□ Trial run
Perform a trial run with no changes made

☑ Suppress non-error messages

☑ Archive mode

☑ Recurse into directories

☑ Preserve permissions
Set the destination permissions to be the same as the source permissions.

☑ Preserve modification times
Transfer modification times along with the files and update them on the remote system.

☑ Preserve group
Set the group of the destination file to be the same as the source file.

☑ Preserve owner
Set the owner of the destination file to be the same as the source file, but only if the receiving rsync is being run as the super-user.

☑ Compress
Compress file data during the transfer.

□ Preserve ACLs
Update the destination ACLs to be the same as the source ACLs.

□ Preserve extended attributes
Update the destination extended attributes to be the same as the local ones.

□ Keep partially transferred files
Enable this option to keep partially transferred files, otherwise they will be deleted if the transfer is interrupted.

☑ Delete
Delete files on the receiving side that don't exist on sender.

Extra options

Please check the manual page for more details.

Tags

Figure 29: rsync pull 2

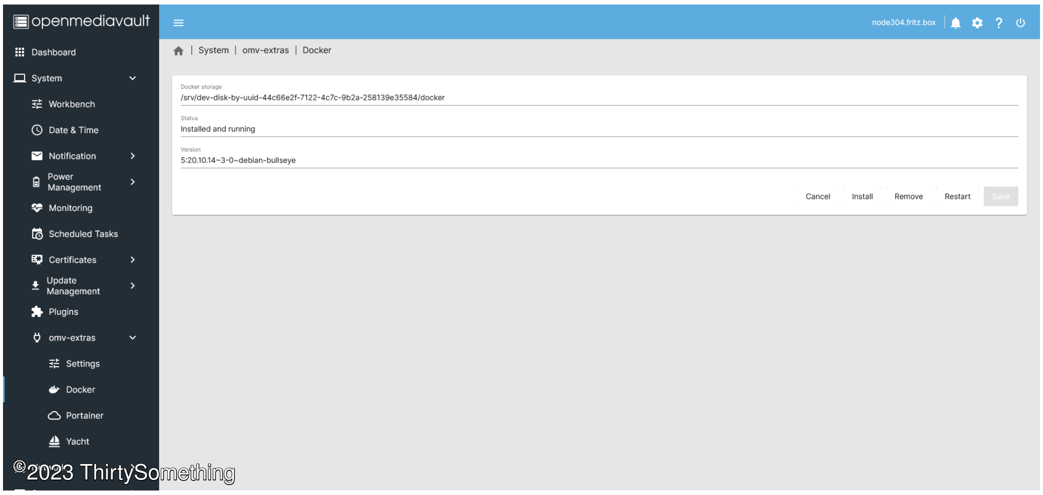
I've both methods to backup/restore my data using the Synology DS411Slim while setting up the Node304.

6 Docker

By using Docker I want to enhance the NAS with features which are not available out-of-the-box. Especially with services my previous NAS offers and which are not native available for OMV.

6.1 Docker installation

The default location of Docker is `/var/lib/docker` on the system disk. Allthough the system disk is a fast SSD, I want to install Docker on the slower RAID storage. So I have to change the path of Docker storage.



Docker storage
/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/docker

Status
Installed and running

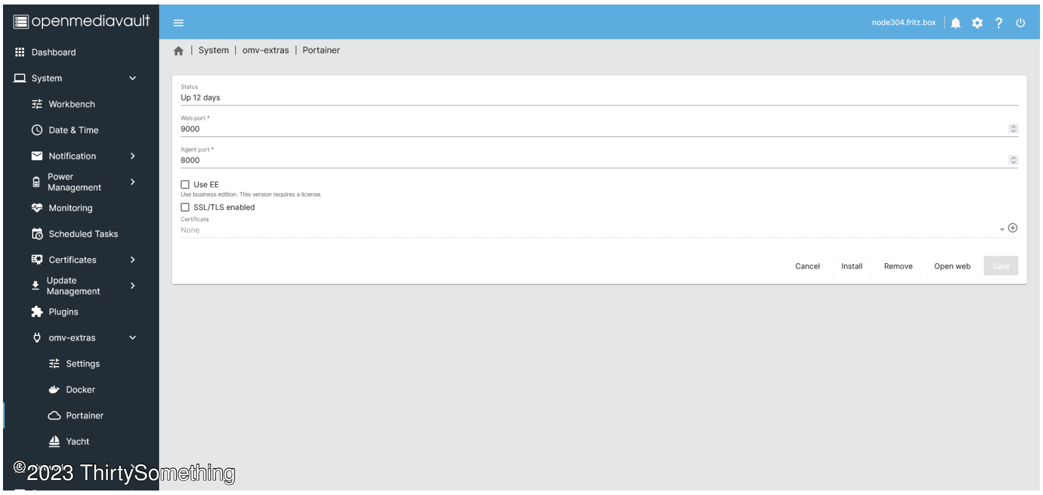
Version
5:20.10.14~3-0~debian-bullseye

Figure 30: The Docker setup

In case Docker is already installed, see here how to move the Docker storage to another location.

6.2 Portainer

To have more comfort in dealing with Docker we install also Portainer from the OMV extras.



System | omv-extras | Portainer

Status
Up 12 days

Web port *
9000

Agent port *
8000

□ Use EE
Use business edition. This version requires a license.

□ SSL/TLS enabled

Certificate
None

Figure 31: The Portainer setup

After installation Portainer is up and running.

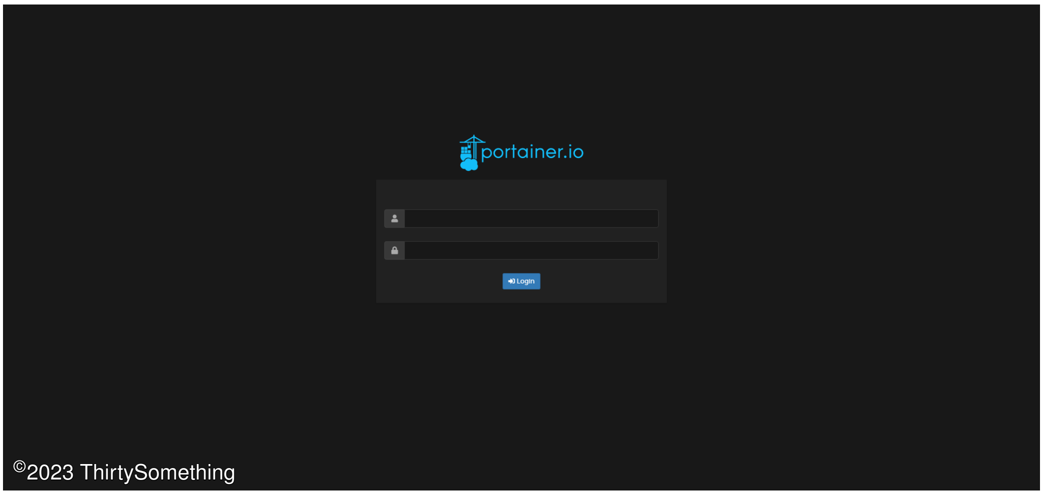


Figure 32: The Portainer login

6.3 Container list

- ~~Portainer – UI for docker~~ – done with docker installation.
- ~~SCM-Manager – SCM with git, svn and hg~~ – done.
- MariaDB – Database for various projects
- phpMyAdmin – UI for the database
- Mosquitto – A MQTT broker
- ~~Syncthing – File synchronization~~
- Nextcloud – Private cloud service
- Backup server?
  - restic
  - Duplicati
- MinIO – As data sink for Nextcloud, restic or Duplicati
- ~~Pi-hole – Adblocker for the network~~ – won't do that, see section 6.6 for explanation
- gotify – Notification server

Figure 33: Containerlist

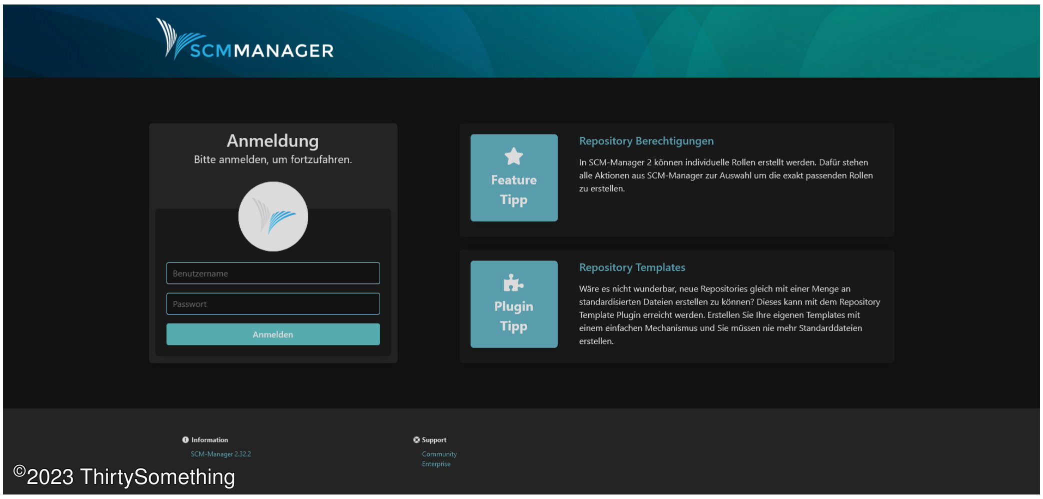
6.4 SCM-Manager

SCM-Manager provides a comfortable user interface for git, Mercurial and Subversion. Up to now I use Subversion for version control. Some of my repositories are private and I will never publish them to a public hoster like GitHub allthough they offer private repositories. All my other repositories are hosted on my NAS – the plan is to move them to git if possible – just to have them on a more modern version control system.

The `docker-compose.yaml` for SCM-Manager looks like:

```
version: "3.8"
services:
  scmmanager:
    container_name: scmmanager
    environment:
      - PUID=1000
      - GUID=100
    image: scmmanager/scm-manager:latest
    ports:
      - 2222:2222
      - 8080:8080
    restart: unless-stopped
    volumes:
      - scm-config:/var/cache/scm/work
      - scm-data:/var/lib/scm
volumes:
  scm-config:
  scm-data:
```

The startup of the container should work without any problems. For the setup of SCM-Manager see the documentation.



Anmeldung

Bitte anmelden, um fortzufahren.

Repository Berechtigungen

In SCM-Manager 2 können individuelle Rollen erstellt werden. Dafür stehen alle Aktionen aus SCM-Manager zur Auswahl um die exakt passenden Rollen zu erstellen.

Repository Templates

Wäre es nicht wunderbar, neue Repositories gleich mit einer Menge an standardisierten Dateien erstellen zu können? Dieses kann mit dem Repository Template Plugin erreicht werden. Erstellen Sie Ihre eigenen Templates mit einem einfachen Mechanismus und Sie müssen nie mehr Standarddateien erstellen.

©2023 ThirtySomething

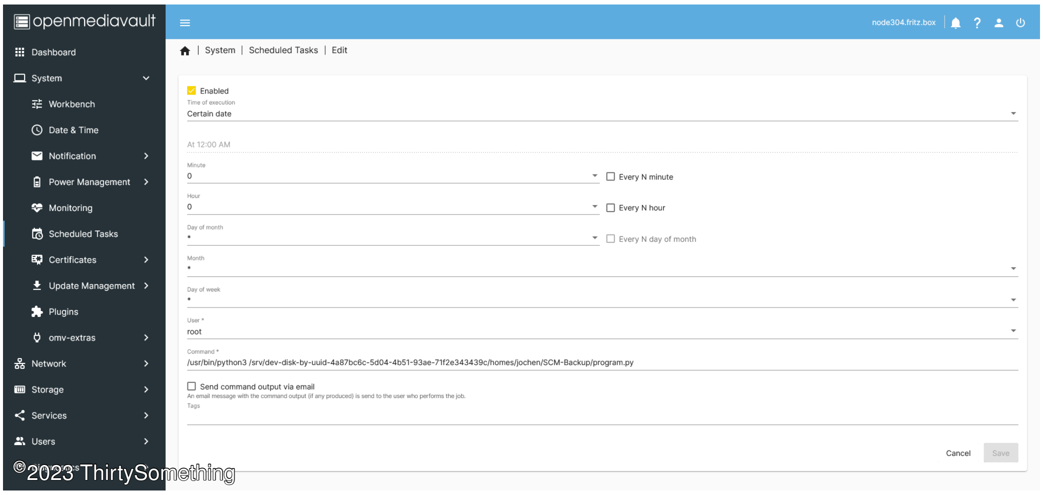
Figure 34: The SCM-Manager login

```
/bin/bash /srv/<omv-raid>/<path-to-script>/svnExport.sh >>
/srv/<omv-raid>/<path-to-script>/svnExport.log 2>&1 &
```

Figure 35: Cron command for svnExport

To setup a backup of the repositories I'm using a self written Python script.

The job is then setup in OMV scheduled tasks accessing the script.



☑ Enabled

Time of execution
Certain date

At 12:00 AM

Minute
0 ☐ Every N minute

Hour
0 ☐ Every N hour

Day of month
* ☐ Every N day of month

Month
*

Day of week
*

User *
root

Command *
/usr/bin/python3 /srv/dev-disk-by-uuid-4a87bc6c-5d04-4b51-93ae-71f2e343439c/homes/jochen/SCM-Backup/program.py

☐ Send command output via email
An email message with the command output (if any produced) is send to the user who performs the job.

Tags

Figure 36: The SCM-Manager backup task

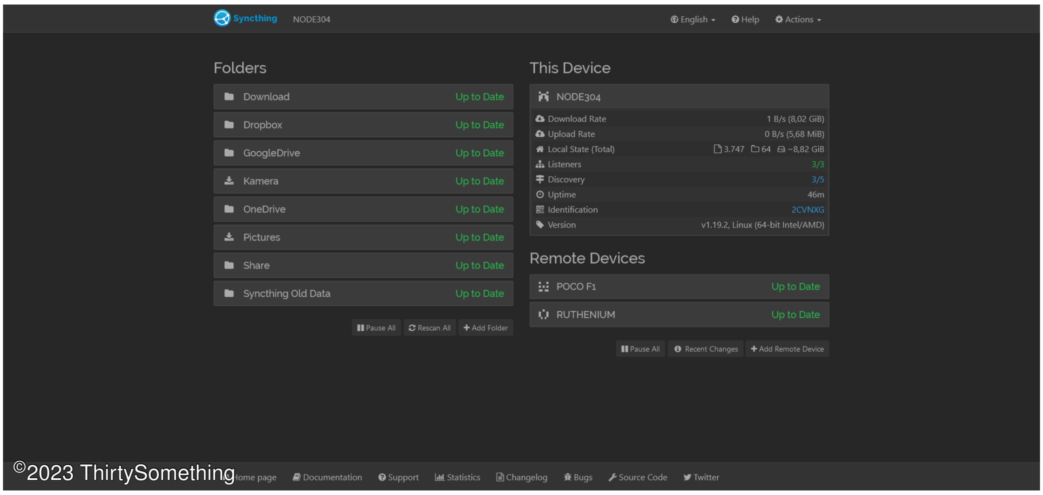
NOTE: The script needs some configuration settings. For more details have a look here.

6.5 Syncthing

Syncthing is a file synchronization service. I use this to synchronize data between my Node304, my PC and my smartphone. The setup is easy regarding the Docker README of the project. The `docker-compose.yaml` for Syncthing looks like:

```
version: "3.8"
services:
  syncthing:
    container_name: syncthing
    environment:
      - PUID=1000
      - GUID=100
    image: syncthing/syncthing:latest
    ports:
      - 8384:8384
      - 21027:21027/udp
      - 22000:22000/tcp
      - 22000:22000/udp
    restart: unless-stopped
    volumes:
      - syncthing-data:/var/syncthing
volumes:
  syncthing-data:
```

Setting up the synchronized folders is as usual with Syncthing. The result may look like this one.



Folders

Download	Up to Date
Dropbox	Up to Date
GoogleDrive	Up to Date
Kamera	Up to Date
OneDrive	Up to Date
Pictures	Up to Date
Share	Up to Date
Syncthing Old Data	Up to Date

This Device

NODE304

Download Rate	1 B/s (8,02 GiB)
Upload Rate	0 B/s (5,68 MiB)
Local State (Total)	3.747 64 ~8,82 GiB
Listeners	3/3
Discovery	3/5
Uptime	46m
Identification	2CVNXG
Version	v1.19.2, Linux (64-bit Intel/AMD)

Remote Devices

POCO F1	Up to Date
RUTHENIUM	Up to Date

Figure 37: Syncthing

6.6 Pi-hole

I'm already using Pi-hole on a old Raspberry Pi 2B and I'm very pleased with that software. Because of it's nature working as a DNS resolver I won't move this to the NAS. The reason is quite simple – in case the NAS crashes or I want to re-install it, the Pi-hole will run and the internet is still available.

7 Ports

It's always a good idea to remember the used ports. This section will give you an overview about the used ports and the name of the application using this port.

7.1 Ports by application

Application	Port number	Comment
OMV	80	UI
Portainer	9000	UI
ReadyMedia	8200	UPnP
SCM-Manager	8123	UI
SCM-Manager	2222	ssh
Syncthing	8384	UI
Syncthing	TCP:22000	Filetransfer
Syncthing	UDP:22000	Filetransfer
Syncthing	UPD:21027	Receive local discovery broadcasts

Table 2: Ports by application

7.2 Ports by port number

Application	Port number	Comment
OMV	80	UI
SCM-Manager	2222	ssh
SCM-Manager	8123	UI
ReadyMedia	8200	UPnP
Syncthing	8384	UI

Syncthing	UPD:21027	Receive local discovery broadcasts
Syncthing	TCP:22000	Filetransfer
Syncthing	UDP:22000	Filetransfer

Table 3: Ports by port number

A svnExport.sh

```
#!/bin/bash
#+-----------------------------------------------------------------------------+
#| Script to export configured svn repositories                                |
#+-----------------------------------------------------------------------------+
# Comment line out for debugging purposes
# set -x
#+-----------------------------------------------------------------------------+
#| Variable definitions                                                        |
#+-----------------------------------------------------------------------------+
# location of all repositories
VAR_PATH_SVN="/srv/dev-disk-by-uuid-44c66e2f-7122-4c7c-9b2a-258139e35584/docker/volumes/scmmanager_scm-data/_data/repositories/*"
# suffix for SCM organized repositories
SUFFIX_SVN=data
# filename containing real repository name
META_SVN=metadata.xml
# maximum days to keep a backup
INT_AGE=5
# file extension
STR_EXT=gz
# current date
STR_DATE=$(date +%Y-%m-%d)
# get current name of backup folder
DIR_EXPORT=$(dirname "${0}")
DIR_EXPORT=$(realpath "${DIR_EXPORT}")

#+-----------------------------------------------------------------------------+
#| Check for valid SVN repository                                              |
#+-----------------------------------------------------------------------------+
function is_svn_repository {
    svnlook info "${1}" >/dev/null 2>&1
    echo $?
}

#+-----------------------------------------------------------------------------+
#| Get base name for SVN repository                                            |
#+-----------------------------------------------------------------------------+
function get_svn_base_name {
    REPOBASE="$(dirname "${1}")/${META_SVN}"
    REPONAME=$(xmlstarlet sel -T -t -m "/repositories/name" -v "/repositories/name" < "${REPOBASE}")
    echo "${REPONAME}"
}

#+-----------------------------------------------------------------------------+
#| Create name for SVN respoistory for export                                  |
#+-----------------------------------------------------------------------------+
function get_svn_destination_name {
    REPOBASE=$(get_svn_base_name "${1}")
    echo "${DIR_EXPORT}/${REPOBASE}-${STR_DATE}.${STR_EXT}"
}

#+-----------------------------------------------------------------------------+
#| Delete backups older than specified age                                     |
#+-----------------------------------------------------------------------------+
function drop_old_exports {
    PATTERN=$(get_svn_base_name "${1}")
    FTK=$((0 + "${INT_AGE}"))
    echo "$(date +’%Y%m%d-%H:%M:%S’): Keep the last ${INT_AGE} backups"
    COUNTER=0
    for CURRENT_DUMP in $(find "${DIR_EXPORT}" -name "${PATTERN}-*.${STR_EXT}" | sort -nr); do
        if [[ "${COUNTER}" -lt "${FTK}" ]]; then
            echo "Keep dump [${CURRENT_DUMP}]"
        else
            echo "Delete dump [${CURRENT_DUMP}]"
            rm "${CURRENT_DUMP}"
        fi
        COUNTER=$((COUNTER + 1))
    done
    echo ""
}

#+-----------------------------------------------------------------------------+
#| Export SVN repository                                                       |
#+-----------------------------------------------------------------------------+
function export_svn_repository {
    VAR_DEST_NAME=$(get_svn_destination_name "${1}")
    echo "$(date +’%Y%m%d-%H:%M:%S’): Dumping repo [${1}] to [${VAR_DEST_NAME}]"
    svnadmin dump "${1}" | gzip > "${VAR_DEST_NAME}"
}

#+-----------------------------------------------------------------------------+
#| Loop over all repositories                                                  |
#+-----------------------------------------------------------------------------+
echo "================================================================================"
echo "***** Start export of SVN repositories for date [${STR_DATE}]"
for VAR_CURRENT_DIR in ${VAR_PATH_SVN}; do
    SVN_REPO="${VAR_CURRENT_DIR}/${SUFFIX_SVN}"
    if [[ $(is_svn_repository "${SVN_REPO}") -eq 1 ]]; then
        echo "$(date +’%Y%m%d-%H:%M:%S’): Skip non SVN repository [${SVN_REPO}]."
        echo ""
    else
        export_svn_repository "${SVN_REPO}"
        drop_old_exports "${SVN_REPO}"
    fi
    echo "--------------------------------------------------------------------------------"
done
echo ""
```

List of Figures

1 A quote from John Lennon . . . 6
2 The OMV login page . . . 9
3 The OMV disks . . . 10
4 The OMV RAID . . . 11
5 The OMV ZFS plugin . . . 11
6 The OMV filesystem on the RAID . . . 12
7 The OMV shared folders . . . 13
8 The OMV users home directory . . . 13
9 The OMV CIFS settings . . . 14
10 The OMV CIFS shares . . . 14
11 The OMV advanced CIFS settings . . . 15
12 SMB/CIFS tuning options . . . 15
13 The speed of the old DS411Slim . . . 16
14 The speed of the OMV NAS . . . 17
15 OMV NAS with tuned SMB/CIFS . . . 18
16 The OMV extras installation . . . 19
17 The OMV extras plugin . . . 19
18 The OMV ClamAV plugin . . . 20
19 The antivirus settings . . . 20
20 The antivirus on access scan . . . 21
21 The antivirus scheduled scan . . . 21
22 The antivirus system load on scan . . . 22
23 The MiniDLNA plugin . . . 22
24 The MiniDLNA settings . . . 23
25 The MiniDLNA shares . . . 23
26 rsync push 1 . . . 24
27 rsync push 2 . . . 24
28 rsync pull 1 . . . 25
29 rsync pull 2 . . . 25
30 The Docker setup . . . 26
31 The Portainer setup . . . 27
32 The Portainer login . . . 27
33 Containerlist . . . 28
34 The SCM-Manager login . . . 29
35 Cron command for svnExport . . . 29
36 The SCM-Manager backup task . . . 30
37 Syncthing . . . 31

List of Tables

1 Change history 3
2 Ports by application 32
3 Ports by port number 33

Glossary

ClamAV ClamAV, an open source antivirus software 19

Docker Container based virtualization 5, 7, 12, 19, 26

Duplicati Duplicati, free backup software 28

EXT4 EXT4, the default Linux filesystem 11, 15, 16

git git, a version control system 28

MariaDB MariaDB, a RDBMS 28, III

MinIO MinIO, multi-cloud object storage 28

Mosquitto Mosquitto, an open source MQTT broker 28

NAS Network Attached Storage 4, 5, 7, 11, 15, 19, 22, 26, 28, III, IV

Nextcloud Nextcloud, the self-hosted productivity platform that keeps you in control 28

Node304 The new NAS system 16, 24, 25, 30

OMV Open Media Vault, a NAS operating system 7, 9, 13, 16, 19, 22, 23, 26, 29, 32, III

OMV extras Open Media Vault Extras, value added plugins to OMV 19, 26

phpMyAdmin phpMyAdmin, an UI for MariaDB 28

Pi-hole Pi-hole, network-wide ad blocking 31

Portainer Portainer, a container management tool 19, 26–28, 32

RAID RAID, redundant array of independent disks 10, 12

ReadyMedia ReadyMedia, a simple media server software 19, 22, 32

restic restic, backups done right 28

rsync rsync, fast incremental file transfer 23

SCM-Manager SCM-Manager, a version control system server 28, 29, 32

Subversion Subversion, a version control system 28

Syncthing Syncthing, file synchronization 28, 30, 32, 33

Synology Manufacturer of NAS systems 4

Synology DS411Slim The old NAS system 15, 24, 25

ZFS ZFS, an enterprise level filesystem 11, 12